Laboratório de Modelagem, Análise e Controle de Sistemas Não-Lineares

Departamento de Engenharia Eletrônica

Universidade Federal de Minas Gerais

Av. Antônio Carlos 6627, 31270-901 Belo Horizonte, MG Brasil

Fone: +55 3499-4866 - Fax: +55 3499-4850

Sensor Fusion for Irregularly Sampled Systems

Taiguara Melo Tupinambás

Dissertação submetida à banca examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais, como parte dos requisitos necessários à obtenção do grau de Mestre em Engenharia Elétrica.

Orientadores: Prof. Bruno Otávio Soares Teixeira, Dr.
Prof. Leonardo Antônio Borges Tôrres, Dr.

Belo Horizonte, Maio de 2018

Contents

List of Figures x

List of Tables xi

List of Symbols and Acronyms xiii

1 Introduction 1

1.1 Motivation 1

1.2 Problem Formulation 4

1.3 Objectives 6

1.4 Text Outline 6

2 Sensor Fusion 9

2.1 Introduction 9

2.2 Motivation and Advantages 10

2.3 Taxonomy and Classification 12

2.4 Fusion of Imperfect Data 17

2.4.1 Probabilistic 18

2.4.2 Evidential Belief 19

2.4.3 Fuzzy Logic 20

2.4.4 Possibilistic 21

2.4.5 Random Set 22

2.4.6 Rough Set 23

2.5 Chapter Summary and Final Remarks 25

3 Irregular Sampling **27**

3.1 Introduction . . . 27

3.2 Sampling Irregularity Types . . . 30

3.2.1 Time Delay . . . 30

3.2.2 Packet Loss . . . 31

3.2.3 Uncertain Observation . . . 34

3.2.4 Aperiodic Sampling . . . 34

3.2.5 Multi-Rate Sampling . . . 38

3.3 Time Synchronization . . . 41

3.4 Chapter Summary and Final Remarks . . . 43

4 Methodology **47**

4.1 Bayesian Estimation . . . 47

4.1.1 Kalman Filter . . . 52

4.1.2 Unscented Kalman Filter . . . 56

4.2 State Estimation with Aperiodic Sampling . . . 58

4.2.1 With Timestamp . . . 58

4.2.2 Without Timestamp . . . 60

4.3 Performance Metrics . . . 62

5 Results **63**

5.1 Linear System . . . 63

5.1.1 System Description . . . 63

5.1.2 Measurement Signal-to-Noise Ratio Variation . . . 69

5.1.3 Average Sampling Rate Variation . . . 70

5.1.4 Regular and Average Irregular Time Interval Relation Variation . . . 70

5.2 Unicycle Position Estimation . . . 71

5.2.1 System Description . . . 71

5.2.2 Measurement Signal-to-Noise Ratio Variation . . . 74

5.2.3 Average Sampling Rate Variation . . . 75

5.2.4 Regular and Average Irregular Time Interval Relation Variation . . . 76

6 Conclusions **79**

6.1 Project Overview 79

6.2 Main Results 79

6.3 Future Work 79

References **90**

Todo list

Falta incluir time-delay e simulação com filtro de partículas 63

List of Figures

1.1 Irregular sampling process modeled by a Poisson random process 5
1.2 Randomly delayed measurements, modeled as exponential random variables 5

2.1 Probability density functions for different amount of fused measurements 11
2.2 Expected advantages in sensor fusion 12
2.3 Classification of data fusion based on sensor interaction. 13
2.4 Input Output model and the three fusion levels. 15
2.5 Sensor network architectures for data fusion. 16
2.6 Data fusion categorization based on data challenges hierarchy 17
2.7 Best approximation of the rough set X by lower and upper crisp sets. . . 24

3.1 Irregular sampling diagram 28
3.2 Different classes of out-of-sequence measurements irregularities 31
3.3 Time delay diagram 32
3.4 Packet loss diagram 33
3.5 Uncertain observations diagram 35
3.6 Time-driven and event-driven sampling schemes 36
3.7 Aperiodic sampling categories 37
3.8 Aperiodic sampling diagram 39
3.9 Periodic and aperiodic multi-rate sampling scheme 40
3.10 Multi-rate sampling diagram 41

4.1 *Posterior* density obtained by the fusion *prior* and *likelihood* 52
4.2 Exemplos de instantes de amostragem da entrada e das estimações . . . 59
4.3 Diagrama ilustrativo da versão *online* do estimador 61

5.1 Bode diagram of both modes 64
5.2 PRBS input and system outputs 66
5.3 Output of the fourth order system to the PRBS input signal 66
5.4 Estimated and true states 67
5.5 Performance index temporal cut for one realization 68
5.6 First group of subfigures. 68
5.7 Performance, as a function of input and observation SNR 69
5.8 Performance, as a function of observation SNR 69
5.9 Performance, as a function of λ 70
5.10 Performance, as a function of α 70
5.11 Nonholomonic robot system representation 71
5.12 Simulation input signals 73
5.13 True position, noisy measurements and UKF estimates 74
5.14 Performance index temporal cut for one realization 75
5.15 Performance index J variation, as a function of measurement noise . . . 76
5.16 Performance index J variation, as a function of λ 77
5.17 Performance index J variation, as a function of α 78

List of Tables

2.1 Possibility and Probability functions associated with X 21
2.2 Mathematical parallels between single and multi sensor statistics 22
2.3 Data fusion methods for imperfect data 25

3.1 Comparison of time synchronization methods 44

List of Symbols and Acronyms

Symbols

Chapter 1

$\mathbb{N}^+$	positive integers;
$\mathbb{R}$	real numbers;
$\mathbb{R}^n$	$\mathbb{R}^{nx1}$ n-dimensional Euclidean space;
$\forall$	for all;
$\in$	belongs to;
$>$	greater than;
$<$	less than;
$\geq$	greater than or equal to;
$\leq$	less than or equal to;
$\triangleq$	equals by definition;
$\approx$	approximately equal to;
$\sim$	is distributed as;
$\mathcal{N}$	Gaussian distribution;
$\mathcal{E}$	Exponential distribution;
x_i	ith entry of x;
f	process model;
g	observation model;
t	continuous-time index;
k	discrete-time index of measurements;
i	discrete-time index of input;
t_k	continuous-time sampled instants;

T	input sampling interval;
$x(t)$	state vector;
$x(t_k)$	state vector at continuous-time sampled instants;
$u(t)$	input vector;
$u(iT)$	input vector at continuous-time sampled at time $t = iT$;
$w(t)$	process noise vector;
y_{t_k}	output vector at continuous-time sampled instants;
δ_k	measurements time-delay;
$v(t_k)$	measurement noise vector;
h_k	time intervals between two continuous-time sampled instants;
λ	parameter of the exponential distribution;
λ_h	parameter of the exponentially distributed random variable h_k;
λ_{δ_k}	parameter of the exponentially distributed random variable δ_k;
N	amount of identical sensors in the sampling model;
L	sampling period of the identical sensors in the sampling model;
P	covariance matrix;
α	relation between output (expected) and input sampling time intervals .

Chapter 2

$\mathcal{N}$	Gaussian distribution;
$\mathbb{P}$	power set;
μ	mean of a random variable;
σ	standard deviation of a random variable;
n	number of homogeneous sensors;
E	evidence;
H	hypothesis;
R_r	r^{th} fuzzy rule;
x_k	k^{th} input fuzzy variable;
A_i	i^{th} antecedent linguistic value;
y	output variable;
C_j	j^{th} consequent class;
A	fuzzy set;
U	universe of discourse;

B	subset of features;
$B_*(X)$	*B-lower* approximation of X;
$B^*(X)$	*B-upper* approximation of X;
$BN_B(X)$	*B-boundary* region of X;
$\rho(x)$	probability distribution of x;
$\rho(A\|B)$	conditional probability distribution of A given B;
$\pi_x(u)$	possibility distribution of x associated with u;
$\mu_A(x)$	membership function of x of a fuzzy se A;
$\subseteq$	is a subset of;
$\cap$	set intersection;
$\emptyset$	empty set;

Chapter 3

$\mathbb{N}$	natural numbers;
$\mathbb{R}$	real numbers;
$\mathbb{R}^n$	$\mathbb{R}^{nx1}$ n-dimensional Euclidean space;
$Ber(p)$	Bernoulli distribution;
$\mathcal{N}$	Gaussian distribution;
Γ	gamma distribution;
$\forall$	for all;
$>$	greater than;
$\geq$	greater than or equal to;
$\sim$	is distributed as;
$\pm$	plus or minus;
y_t, yk	output vector at continuous-time sampled instants;
$z(t)$	measurement output;
$H(k)$, $L(k)$, C, D	matrices of the observation model;
$v(t)$, $v(k)$	measurement noise vector;
t_i	continuous-time sampled index;
k	discrete-time index of measurements;
T	expected time interval;
$d(k)$	observation sampling time delays;
l	amount of different known delays;

$\xi(t)$	random variable that models packet dropout;
$\gamma(t)$	random variable that models uncertain observation;
$C_i(t)$	real time approximation from the i^{th} computer clock;
a_i	i^{th} clock drift;
b_i	i^{th} clock offset;
$a_i j$	relative drift between i^{th} and j^{th} node;
$b_i j$	relative offset between i^{th} and j^{th} node;
Δ	measured value interval between observations;
δ_k	random time interval;
$p(k)$	Bernoulli distribution parameter;
ϵ_k	deviation from expected time instants;
σ	standard deviation;
κ	shape parameter;
θ	scale parameter;
μs	micro seconds;
$E.E$	energy efficiency;
$Comp.$	complexity;

Chapter 4

$\triangleq$	equals by definition;
$\arg\max_x f(x)$	argument x that maximizes function $f(x)$;
$\rho(x\|y)$	conditional probability density function of X given Y;
x_k	state vector;
$\hat{x_k}$	estimated state vector;
y	observation vector;

Chapter 5

Acronyms

CDF	Cumulative Distribution Function;
CU	Covariance Union;
DAI	Data In;
DAO	Data Out;
DEI	Decision In;
DEO	Decision Out;
DMTS	Delay Measurement Time Synchronization;
DSET	Dempster-Shafer Evidence Theory;
EKF	Extended Kalman Filter;
FEI	Feature In;
FEO	Feature Out;
FISST	Finite-Set Statistics;
FTSP	Flooding Time Synchronization Protocol;
GPS	Global Positioning System;
HMM	Hidden Markov Model;
KF	Kalman Filter;
UKF	Unscented Kalman Filter;
LEETS	Lightweight and Energy Efficient Time Synchronization;
LTS	Lightweight Tree-based Synchronization;
LS	Least Squares;
MAP	Maximum A Posteriori;
ML	Maximum Likelihood;
MMSE	Minimum Mean Square Error;
MOP	Measures Of Performance;
NMR	Nuclear Magnetic Resonance;
NTP	Network Time Protocol;
NUS	Non-Uniform Sampling;
OOSM	Out-Of-Sequence Measurement;
PDF	Probability Density Function;
PDAF	Probabilistic Data Association Filter;
PF	Particle Filter;
RBS	Reference Broadcast Synchronization;

RMSE	Root Mean Square Error;
RTPF	Real-Time Particle Filter;
RV	Random Variable;
SOD	Send-On-Delta;
SNR	Signal-to-Noise Ratio;
TDS	Time-Delay System;
TDP	Time-Diffusion Protocol;
TPSN	Timing-sync Protocol for Sensor Networks;
TSST	Time Synchronization based on Spanning Tree;
TTF	Track-to-Track Fusion;
UKF	Unscented Kalman Filter;
WSN	Wireless Sensor Networks;

Chapter 1

Introduction

1.1 Motivation

In nature it is possible to observe data fusion in a variety of phenomena. Animals combine signals from different senses, such as sight, hearing, smell, taste and touch, to recognize the surroundings. Plants have analogous mechanisms, which are used to modulate water consumption, to change the color of its leaves or to bend its structure towards the light, for instance. Throughout history, the sensory systems in living beings have evolved to assimilate multiple information coming from numerous sources in a highly complex and efficient way, in order to have a better perception of the environment.

Nowadays information fusion is studied in many fields of science, as a way of exploiting data from multiple sources to achieve better outcomes in comparison to those obtained if any of the sources were used separately (Dasarathy, 2001). Other terms have been used to denote the synthesis of information in technical literature, for instance, data fusion, sensor fusion, combination of evidence and synthesis of observations (Goodman et al., 1997). To avoid confusion, the terminology used by (Elmenreich, 2002) will be adopted, whereby information fusion is understood as the overall term and sensor fusion is used in cases for which the sources of information are sensors.

Some research fields have been increasingly exploiting the advantages of sensor fusion techniques, such as robotics, military, biometrics and image processing. The main benefits expected are related to accuracy, due to the use of redundant or complimentary data; to dimensionality, that is additional information being created by a group of data; and to robustness against failures and interference. Consequently much effort has been put into the development and investigation of data fusion techniques. The work of (Khaleghi et al., 2013) presents an extensive review of different approaches available, categorizing them by the way sensor data imperfection is represented, namely, proba-

bilistic fusion, evidential belief reasoning, fuzzy reasoning, possibilistic fusion, rough set-based fusion, random set-based fusion and hybrid fusion.

Data fusion techniques based on probability theory are the earliest available and perhaps the most popular until now. They are concerned with estimating the probability density functions (PDFs) of the system states by means of the Bayesian approach. If the system is linear and Gaussian, then the Kalman filter (KF) guarantees optimal estimation. For nonlinear processes, KF generalizations were proposed, such as the extended Kalman filter (EKF) or the unscented Kalman filter (UKF) (Julier and Uhlmann, 2004). On the other hand, particle filters (PF) can be used to deal with both nonlinearities in the dynamics and non-Gaussian distributions (Arulampalam et al., 2002).

The most common class of systems studied in state estimation is the class of sampled-data systems, due to the wide use of digital devices. Although often described by continuous time differential equations, they can be modeled using discrete state equations, using approximation techniques (Phillips and Nagle, 1995). Usually the sampling period of such systems are constant and known. In other words, the sensors are considered to transmit data at regular intervals. However, for many applications, such assumption is not valid. The use of several redundant sensors, for example, with different sampling rates or unsynchronized with one another, leads to data being received at irregular instants. Additionally, when data from multiple sensors are transmitted through several subsystems in a network, there might be loss of packets and delays (Schenato et al., 2007) or even multiple information arriving simultaneously (Moayedi et al., 2011). In networked control systems, event-triggered sampling schemes have been proposed to optimize the access to communication channels (Hu et al., 2017), which will also generate time-varying sampling intervals. Nowadays, because of the ever-growing scientific advances, the technology of microprocessors, sensors and communication has become increasingly accessible, which continues to ensure that multiple sensor networks are more and more common.

Thus, despite improving accuracy and robustness of the estimation process, the fusion of data from multiple sensors might introduce challenges to the state estimation algorithms, due to sampling irregularities. Depending on how they take place, modifications to the KF and its generalizations can be carried out to tackle these abnormalities. In the work of (Fatehi and Huang, 2017), a fusion KF is proposed to estimate the states of a system with multi-rate measurements, whereby one of them is fast, regular and delay-free and the other is slow, irregular and randomly delayed. One application

of such system is for industrial process control, where there is online instrumentation characterized by the former and data from laboratory analysis, which are much more accurate. For a more general case, when the random delays are unknown, the work of (Gopalakrishnan et al., 2010) presents a critical analysis of the available methods for data fusion. They are separated into two categories: those that incorporate the delayed measurements upon arrival, and methods that rely in state augmentation, in order to assimilate the delayed information between estimation steps.

In general the proposed methods and their performance will depend on the characteristics of the sampling irregularities and how they are modeled. Time delays can be multiples of a base sampling period, for instance. In those cases, delays can happen at single or multiple lags (Peñarrocha et al., 2012), can lead to out-of-sequence measurements (Anxi et al., 2005; Westenberger et al., 2013) or there can also be data dropout (Zhu et al., 2013). Nevertheless, the system can be described by a time-invariant discrete state equation, but with a particular representation of the measurement model. When the instants take place after random time intervals, the discrete-state representation leads to a time-varying system, since the sampling period changes over time. Some researches treat the variable measurement instants as stochastic processes (Micheli and Jordan, 2002) or as a periodic sampling interval subject to noisy perturbations (Shen et al., 2016). Generally, the instant is considered to be known and the methods assimilate such information in the algorithm.

To the best of the author's knowledge, no method was proposed so far to take the irregularity into account and improve the estimation efficiency, for the cases in which the random time instant the signal was measured and its statistics are not known or not reliable. If the sampling irregularity is caused due to the lack of sensor synchronization in the network, several algorithms can ensure a common timescale (Sivrikaya and Yener, 2004), at the expense of additional investments or energy use. Another approach, believed to be largely used on practice, is to disregard the irregularities, assimilating the measurements as soon as possible (Kwok et al., 2004; Huck et al., 2011). In this case, additional noises are tuned in the estimation process, but it might be irrelevant to the overall performance.

Knowing to what extent the estimation accuracy is deteriorated by ignoring the additional uncertainty caused by the sampling irregularity is important to decide whether or not to invest in synchronization. In addition, the investment in more sensors to the network in order to improve accuracy might not pay off, if it increases the occurrence

of irregularities. However there are no detailed studies on the behavior of the degradation in accuracy due to neglecting the irregularity in the sampling process. Therefore, this work assesses the differences in state estimation performance for systems with random sampling intervals with and without timestamp for different scenarios. The purpose is to shed some light on the trade-off for investments in sensor networks and synchronization.

1.2 Problem Formulation

Consider the stochastic nonlinear sampled system

$$\dot{x}(t) = f(x(t),u(t),w(t),t), \tag{1.1}$$

$$y(t_k) = g(x(t_k - \delta_k),v(t_k),t_k, \tag{1.2}$$

where f: $\mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \times \mathbb{R}^+ \rightarrow \mathbb{R}^n$ and g: $\mathbb{R}^n \times \mathbb{R}^r \times \mathbb{R}^+ \rightarrow \mathbb{R}^m$ are, respectively, the process and observation models, considered as known. We assume that for all $k \geq 1$, the observations $y(t_k) \in \mathbb{R}^m$ are available. Process and observation noises, $w(t)$ and $v(t_k)$ respectively, are white, Gaussian, zero-mean and mutually independent, with known covariance matrices. The first two moments of the initial random state vector $x(0)$ are also known. Input data $u(t)$ are available at regularly spaced intervals T, that is $u(iT) \in \mathbb{R}^p$, $\forall i \geq 1$ are known.

Observations are taken at random time instants t_k and are considered to be sorted ($t_{k+1} > t_k, \forall k \in \mathbb{N}^+$) and defined by the time intervals $h_0 \triangleq t_1$, $h_k \triangleq t_k - t_{k-1}$, $\forall k \geq 1$. In this work, we assume that the observation time instants t_k are given by a Poisson random process. That is, the time intervals h_k are independent and identically distributed (i.i.d.) exponential random variables with a known parameter λ_h, that is $h_k \sim \mathcal{E}(\lambda_h)$, where $\mathcal{E}(\lambda)$ defines an exponential PDF, with parameter λ, which is also its expected value. An example of time intervals produced by such a random process is illustrated in Figure 1.1. This sampling model characterizes a common application for an event-based sampling scheme or for a networked control system with unsynchronized sensors. (Micheli and Jordan, 2002), for instance, considers a set of N identical sensors measuring the state variables of a physical process every L seconds. They prove that, if the sensors are independent and unsynchronized and N is big enough, the waiting time between the

realization of two consecutive measurements can be approximated by an exponential random variable $\mathcal{E}(\lambda)$, where the parameter is given by $\lambda = N/L$.

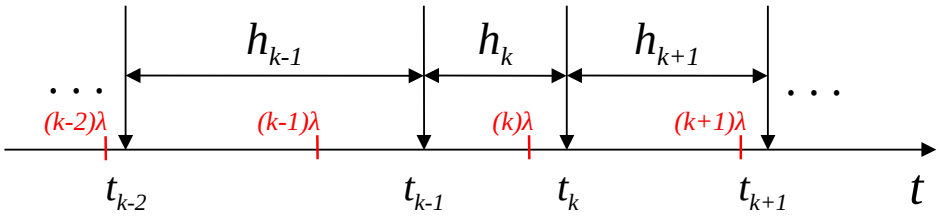


Figure 1.1: Irregular sampling process modeled by a Poisson random process. Regularly spaced time intervals λ are shown in red. An example of time instants t_k realization is also shown, with the respective random time intervals h_k. The expected value of time interval is given by $E(h_k) = \lambda$.

Arrival times to the estimator may be delayed during transmission by a random time amount δ_k, also given by exponential random variables, with parameter λ_{δ_k}, according to Figure 1.2. Out-of-sequence measurements (OOSM) are not considered in this work. We assume that, in case a delayed measurement is to arrive later than future measurements, it gets lost in transmission.

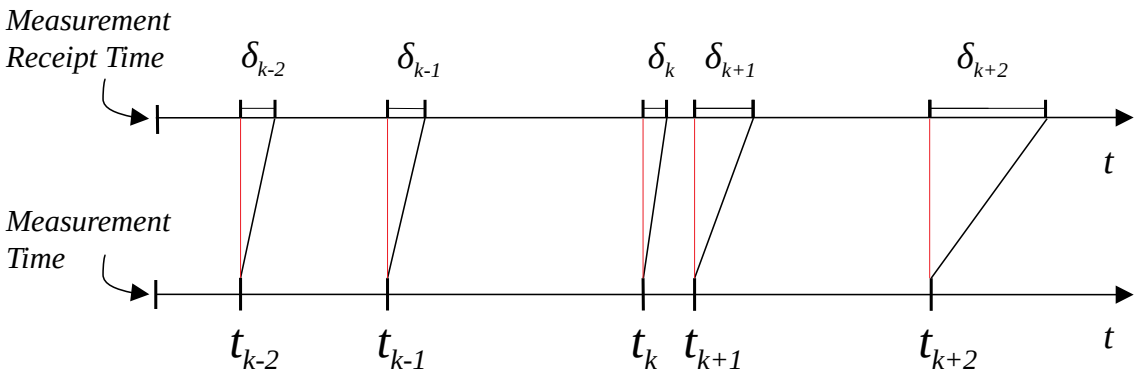


Figure 1.2: Randomly delayed measurements, modeled as exponential random variables. Random measurement times t_k are received by the estimator after a random delay δ_k

When time-stamp information is available, data assimilation can be performed at the correct measurement instants t_k. When they are not, the assimilation moment is

considered to be the random receipt time instant or the next estimation moment, when there are no transmission time delays.

We wish to estimate the state vector $x(t)$ and its covariance recursively, at regularly spaced time intervals T, given their initial values (x_0, P_0), the process model (**??**), the input or control signals, $u(t) : t \leq kT$, and the set of past observations, $y(t_k) : t_k - \delta_k \leq kT$. The knowledge of the time intervals $h_{k-1} : t_k - \delta_k \leq kT$ is also taken into consideration when time-stamp information is available. We assume that the average time interval of observations λ_h is greater than or equal to T by a factor $\alpha \geq 1$, i.e. $\lambda_h = \alpha T$.

1.3 Objectives

1.4 Text Outline

This text is organized in six chapters, including this one. Chapter 1 presents an introduction to the object of study, with an overview of the motivations and historical perspective of the theme, objectives and problem formulation.

Chapter 2 presents a review of the sensor fusion field of science, addressing not only the definitions and taxonomy, but also the advantages of combining information from multiple sources. From the four categorization models of the data fusion problem, the one based on data challenges is further explored. We then introduce and discuss the methods proposed in literature that handles imperfect data.

Chapter 3 discusses the data-related challenges that arise for sampled-data systems, regarding sampling irregularities. Diagrams are built to organize the types, effects and causes of irregularities. We describe the necessary modifications to state estimation observation models, in order to handle such abnormalities in sampling schemes. In order to perform effective state estimation in the presence of sampling irregularities, methods depend on the knowledge of the exact time measurements were taken. Therefore, we explore time synchronization methods suited for sensor networks.

After the literature review, we focus on the probabilistic sensor fusion approach to sampled-data systems with sampling irregularities without time-stamp. The study of the impact of neglecting time-stamp information is performed.

Chapter 4 describes the methods used for simulation, considering the adaptations for the scenarios with and without time-stamp. We describe the filtering algorithm

and the assumptions used for each scenario. Furthermore, we define the performance metrics used for the results assessment.

In Chapter 5 we present results from the simulation of two systems: an arbitrary linear system and a unicycle position estimation system. Signal parameters are varied to assess the impact in performance of both considering and not considering time-stamp in estimation algorithms. Performances are evaluated using estimated state errors and estimation consistency.

Finally, we conclude the study in Chapter 6, highlighting the study limitations and suggestions of future work, apart from evaluating if the proposed objectives were achieved.

Chapter 2

Sensor Fusion

In this chapter, we review sensor fusion techniques, advantages and terminologies. We start with a brief explanation on the motivations of this field of science, grouping them in two main categories: data authenticity and data availability. We continue with a definition of sensor fusion and an exploration of the available fusion methods classification models. The categorization based on data challenges, that is uncertainty, imperfection, inconsistency and disparateness, is further discussed, specially the data fusion methods that handle imperfect data, the most fundamental challenge present in data. We end the chapter narrowing the focus on probabilistic methods used to fuse information from sampled-data systems.

2.1 Introduction

The idea that combining information from multiple sensors to improve overall system performances has been in discussion for several decades. In the early days, there were those who argued against the synergism hype that was being spread in military systems, using the multi-sensor concept (Fowler, 1979). In his work, Fowler created what he called his seventh law:

> "Be wary of proposals for synergistic systems. Most of the time when you try to make 2 + 2 = 5, you end up with 3... and sometimes 1.9"

Although he was right to affirm that the added complexity and high costs are not always worth it - specially back then, when devices were more expensive - many posterior studies advocated that the fusion of sensor data will always be better, in the sense that the probability of correctly classifying a target increases. A direct answer to Fowler's work came in the year after (Nahin and Pokoski, 1980), when Nahin and Pokoski used strict concepts and definitions to prove that the addition of sensors improve network performance, but also acknowledged their assumptions of discarding

complexity and costs for the sake of their arguments. This topic continued to draw scientific attention throughout the years, like the work of (Rao, 1998) and (Dasarathy, 2000). Rao focused on fusion methods and its comparison to classifiers' performance. He also established conditions that guaranteed that the fused system will at least perform as good as the best classifier. Dasarathy's work extended that of Rao's, but was able to show a scenario at which a two-sensor suite outperforms a three sensor suite from a parametric fusion benefits domain perspective. In order to compare performances of fusion levels or algorithms and assess possible benefits of sensor fusion, (Theil et al., 2000) discussed three measures of performance (MOP), one for each sensor management process, which are Detection, Tracking and Classification.

Despite all philosophical discussions, many real applications have been taking advantage of sensor fusion benefits since its advent, like remote sensing (Foster and Hall, 1981), robotics (Richardson and Marsh, 1988) and intelligent systems (Luo and Kay, 1989). Recently, with the modernization and popularization of sensors, its use has grown significantly, with hot topics emerging in the area, such as body sensor networks for health-care applications (Gravina et al., 2017), artificial intelligence (Safari et al., 2014; Jordao et al., 2018) and smart grids (Liu and Wang, 2012; Kordestani and Saif, 2017). Recent reviews of the state of the art (Khaleghi et al., 2013; Jing et al., 2013) provide a very broad understanding of the field and its advances.

We continue this chapter advocating for the benefits of sensor fusion in Section 2.2, describing its motivations and advantages. In Section 2.3, most of the taxonomies used in the area are discussed. We end the chapter with a deeper discussion of the aspects of data fusion related to imperfect data.

2.2 Motivation and Advantages

Whether or not all sensor fusion approaches outperform the use of less sensors in every aspect for any given condition, the fact is that many fields of science and engineering have been benefiting from choosing the fusion approach.

The reasons why one chooses to fuse information from different sources are various. The works of (Hall and Llinas, 1997; Elmenreich, 2002; Andler and Brohede, 2009; Khaleghi et al., 2013) provide a detailed study on the motivations and advantages of multi-sensor data fusion techniques. A common benefit from the use of multiple

redundant sensors, for instance, is the increase in accuracy. By averaging all the measurements, the expected error decreases by the rate of $\frac{1}{\sqrt{n}}$, where n is the number of homogeneous sensors, in case of the presence of i.i.d Gaussian noises. In Figure 2.1 different probability density functions (PDFs) with different standard deviations - thus different expected errors - are shown for various quantities of random variables being averaged.

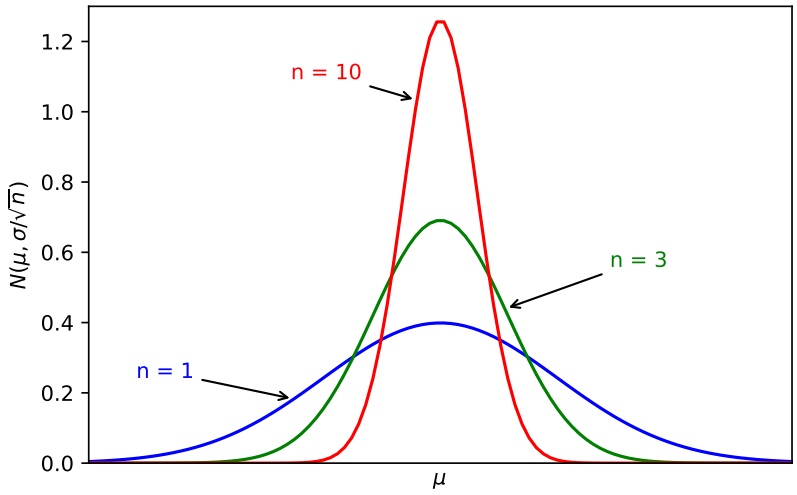


Figure 2.1: Probability distribution functions for different amount of fused measurements, obtained by 1 (blue), 3 (green) and 10 (red) redundant sensors, considering i.i.d additive Gaussian noise. The higher the value of n, the lower is the standard deviation, thus the lower is the expected measurement error.

There are many other reasons to fuse information from multiple sensors, however. Elmenreich gives an interesting example to illustrate some of them. Imagine a car parking assist system with only one distance sensor mounted at its rear, with limited precision and considerable update time (sampling interval). Now picture how would be the system's performance, considering that the sensor may suffer from: deprivation or occlusion, in the sense that it could be blocked by some physical barrier; coverage, both in spatial and temporal contexts, since it can only sense objects in front of him (spatial) and can only provide information periodically (temporal); precision, with measurement errors being liable to be the cause of unexpected bumps; and uncertainty, when an object, such as a small bicycle, might be missed by the sensor even in the presence of valid signal. The system would definitely perform at insufficient levels.

All these issues could be solved by adding multiple sensors to the architecture.

Based on the studies from Hall, Elmenreich and Andler, we can group most of the advantages in two categories: authenticity and availability improvements. The first group refers to those benefits that improve the quality of the measurement, whereas the second one encompasses those benefits regarding ranges or data dimensions. The addition of sensors can also be of different types (heterogeneous) or of the same type (homogeneous), with redundant measurements. There may be benefits that are exclusive to the addition of different sensors, others exclusive to the addition of redundant sensors and those that can happen both ways. Figure 2.2 presents a schematic with the different advantages expected in sensor fusion.

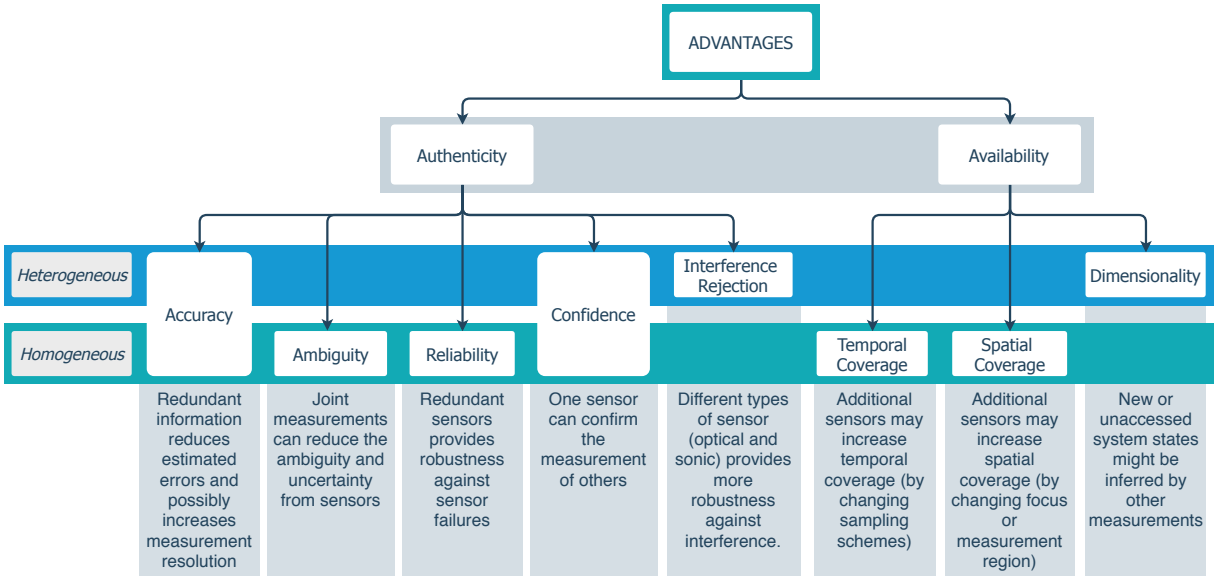


Figure 2.2: Sensor fusion categorization hierarchy based on expected advantages.

2.3 Taxonomy and Classification

Sensor fusion definitions have evolved throughout the years. The work of (Boström et al., 2007) analyses more than 30 papers on this matter, to propose a more comprehensible and precise definition to the broad area of information, data and sensor fusion, which we reproduce here:

> "Information fusion is the study of efficient methods for automatically or semi-automatically transforming information from different sources and

different points in time into a representation that provides effective support for human or automated decision making."

Researchers in the field made additional efforts to categorize the fusion techniques, using different approaches. One of the earliest attempts comes from (Durrant-Whyte, 1988), where he considered the dynamic use of information in the fusion processes, creating the so called dependence model, which grouped sensor fusion in three categories: competitive, complementary and cooperative. Competitive type occurs when multiple sensors measure the same properties, usually referred to as redundant architecture. Complementary fusion describes the scheme of different types of sensors measuring different information about the same global object or feature, enabling a more complete fused information, like multiple cameras covering a large area. And the last category, cooperative fusion, happens when more complex data are combined to provide information that would not be available (or hard to obtain) otherwise. An example would be multiple measurements being processed to create softs sensors. A illustrative schematic is presented in Figure 2.3. An abstraction from the human sensory system to Whyte's model can be done by understanding the way flavors are perceived by our taste and smell sensors, tongue and nose, respectively, in a cooperative fashion.

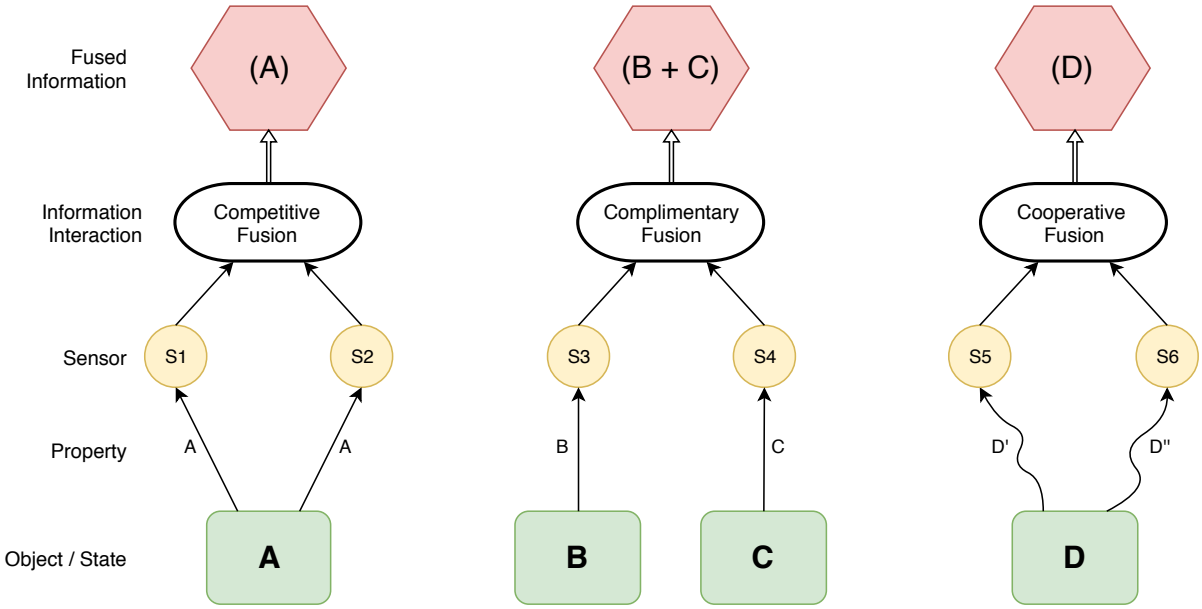


Figure 2.3: Classification of data fusion based on sensor interaction.

Another common way to categorize sensor fusion is by the three-level hierarchy based on input / output characteristics, which depends on the processing stage that

information is fused. The lower-level is related to raw-data fusion, where signals from sensors are combined. The mid-level is usually related to feature fusion, where information about characteristics of the object are used in the process. The higher level involves decision-fusion, that can be understood as a reasoning process, like the methods of evidential belief or fuzzy logic. (Dasarathy, 1997) extended this terminology, proposing five fusion modes, according to Figure 2.4. *Data in - data out fusion* (DAI-DAO), the lowest level fusion, processes raw data and outputs raw data, but with some improvements, such as increased accuracy. *Data in - feature out fusion* (DAI-FEO) extracts features from raw data to describe characteristics of the measured environment. *Feature in - feature out* (FEI-FEO) aims at the refinement of the features entering the process, similarly to what DAI - DAO does to raw data. *Feature in - decison out* (FEI-DEO) usually performs classification based on a set of features received as inputs. *Decision in - decision out* (DEI-DEO) outputs a better global decision based on local, restricted decisions. Using our human brain data fusion analogy, many examples can be framed into Dasarathy's terminology. The processing of raw signals, such as letters symbols, into features such as words and texts can be interpreted as DAI-FEO fusion. On the other hand, the process of assimilating features of objects from our eyes and ears, and fusing them into a decision about what they are, for instance, is an example of FEI-DEO fusion.

The work of (Castanedo, 2013) provides a comprehensive review of these and other classifications of data fusion techniques. His efforts went beyond as he proposed an interesting new approach, based on the type of architecture, summarized in Figure 2.5. In his model, fusion techniques that collects all measurements in a central processor lie on the Centralized Architecture category. Assuming perfect data alignment and association, such scheme should be optimal. However it is keen to many sampling irregularities issues and might provoke network congestion. When a network of nodes is used, each with its own processing capability, the architecture becomes decentralized. Such modular strategy ensures scalability, since there are no limits to centralized bottlenecks, and survivability to the loss of a particular sensing node. However it can greatly increase communication costs. The third and last configuration is the distributed architecture, where each data association is performed by local nodes. The separate outputs are then transmitted to a fusion node, that processes these locally obtained estimates to produce a fused global estimate. This scheme will reduce both the communication costs from decentralized architecture and computational costs from

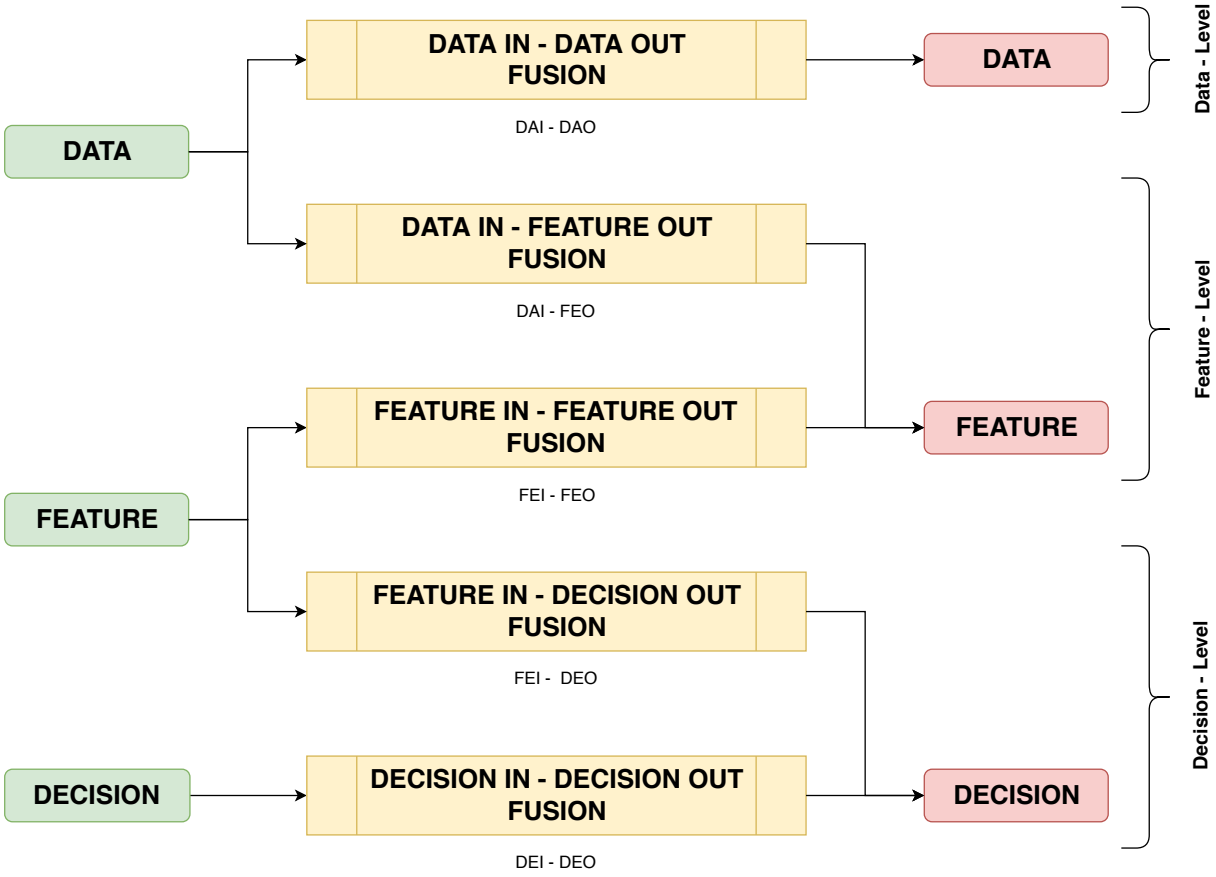


Figure 2.4: Input Output model and the three fusion levels.

the centralized one, while lacking some of their benefits. A fourth architecture could be defined as hierarchical, which combines decentralized and distributed schemes, performing fusion at multiple levels. Getting back to our human sensory capacity, a very complex hierarchical architecture would best describe our brain fusion scheme in Castanedo's classification.

A final note on the taxonomy of data fusion methodologies will be given to the work of (Khaleghi et al., 2013), due to its connections to sensor fusion in the presence of irregularities, like the ones that will be discussed in Chapter 3. The idea was to study the methods based on data-related challenges they address. An overview of the challenge hierarchy proposed by the authors is presented in Figure 2.6, with four main categories of how challenging input data can be: imperfect, correlated, inconsistent and disparate. The most fundamental problem present in data is imperfection. Indeed, most of the algorithms framed in the other three categories are basically methods that

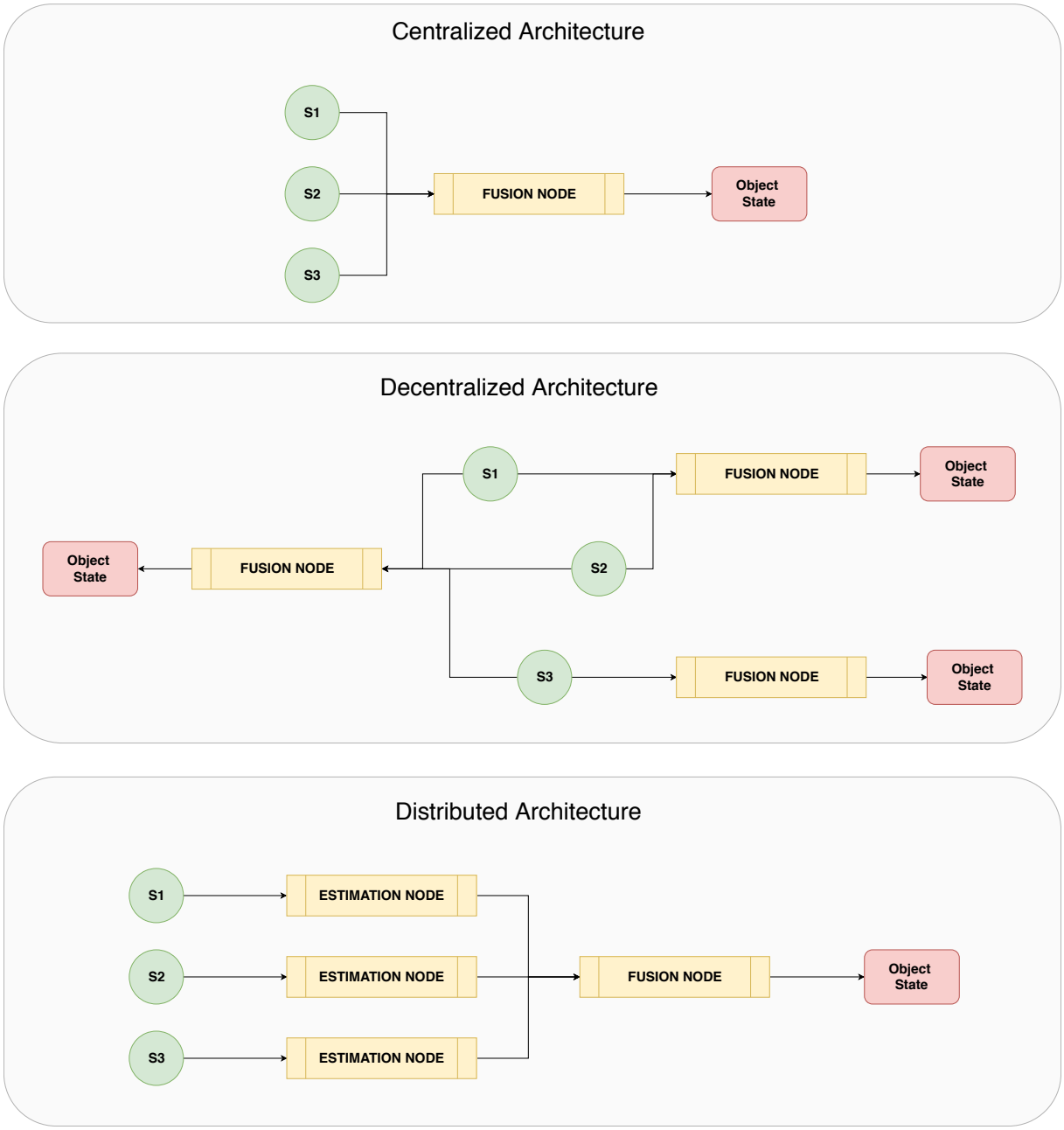


Figure 2.5: Sensor network architectures for data fusion.

try to neutralize, avoid or minimize their aspects, so that imperfection is the only thing left on data. When correlation is present on data, for instance, the requirements for typical fusion algorithms, such as the Kalman Filter, are broken, so there are methods to eliminate correlation or to minimize its effects, given certain assumptions. In case of data inconsistency, due to outliers, one can act on the sensor outputs directly, to validate information or to detect and remove them automatically. If there is out-of-sequence

measurements (OOSM) in data, usual frameworks would be to ignore, reprocess or use backward/forward prediction, or augment the state matrices to incorporate delayed measurements. Conflicted and disparate data are more specific and are beyond the scope of this study.

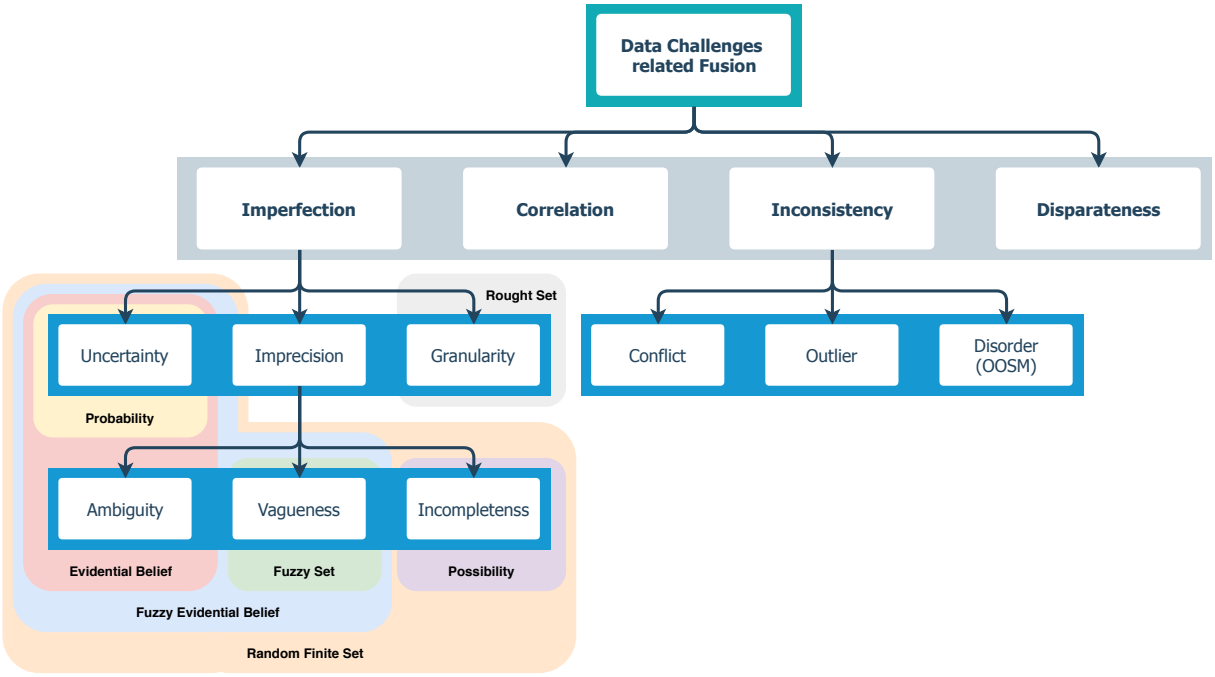


Figure 2.6: Categorization based on data challenges hierarchy. For imperfect data, fusion methods and the issues they deal with are also presented.

2.4 Fusion of Imperfect Data

Being the most fundamental and common challenge present on data, imperfection is also the main focus for research in the area. Based on Khaleghi's classification, imperfection on data can be manifested as uncertainty, imprecision and granularity. We can distinguish uncertainty and imprecision with two information examples: *I believe Mary is one point eight meter tall*; *I am sure that Mary is tall*. In the first sentence, the information is precise, but uncertain. In the second, the information about Mary's height is certain, but imprecise, with some vagueness (fuzziness) attached to it. Usually the amount of precision in data is inversely proportional to the level of certainty. The source of imprecision can also be ambiguity, as in the phrase: *Mary gave birth to her daughter yesterday at 5*. We don't know if it was 5 PM or 5 AM, though the information

was certain. The last type of imprecision would be for incomplete data, when we have missing information. The sentence "*Mary's height is above one point five meter*", is incomplete, for example, meaning that only one bound was given. Any height above one hundred and fifty centimeters is possible and any height less than or equal to it is impossible, defining the so called possibility measures. Finally, granularity is an imprecision related to the internal structure of data, referring to the capacity of distinction among states. Different attributes on the data or a different set of possible states will generate different levels of imprecise information.

Given the amount of potential problems in data and their particularities, it is only natural that no data fusion approach alone could tackle all of them. Researchers have proposed various approaches that focus on one or a few of these issues and Khaleghi's paper explores methods for all the challenges in his categorization model. In this study we will limit ourselves to imperfect data, for which, in most cases, the mathematical background of the algorithms relies on the representation of imperfection. These methods are highlighted in Figure 2.6 in different colors and covering one or multiple aspects. In the next subsections we will explore them. Table 2.3 presents a summary of all methods with their main advantages and limitations, adapted from Khaleghi's study.

2.4.1 Probabilistic

Uncertainty is the most natural source of data imperfection and it is usually expressed by probability density functions (PDFs). Thus probabilistic fusion methods are the most adequate to handle it. The most classical approach to fuse data based on uncertain measurements is using Bayes theorem. The idea is to update the probabilities of an hypothesis H given some evidence E by using his famous equation (Stone, 2013)

$$\rho(H|E) = \frac{\rho(E|H)\rho(H)}{\rho(E)} \tag{2.1}$$

where $\rho(A|B)$ represents a conditional probability, that is the probability of A being true, given B. We can interpret Equation 2.1 as a fusion of *a-priori* belief of an hypothesis and some current evidence to come up with a better current estimate of the hypothesis.

In addition to Bayes' contribution, Gauss' least squares methods and Fisher's maximum likelihood estimation are the foundations of all probabilistic approaches to data fusion, recursive filtering and state estimation. (Kolmogorov, 1962), in 1941 and

(Wiener, 1949) in 1942[1] independently designed a linear minimum mean-square estimation technique, that is considered to be the first probabilistic designed filter, the so called Wiener-Kolmogorov filter. A few years later, perfecting the work of its predecessors, (Kalman, 1960) developed the recursive mean-square filter, a groundbreaking method known as Kalman Filter (KF). Its impact was so big, that only one year after its publication, Kalman's algorithm was used in Apollo Project (Mohinder and Angus, 2010), to solve its guidance and navigation problem. A review on the evolution of least-squares estimation theory is provided by (Sorenson, 1970).

When it comes to a multi-sensor system, (Willner et al., 1976) introduced three approaches for the discrete Kalman Filter: parallel, sequential and data compression filters. In the parallel filter, all measurements are processed by a KF in parallel, producing a multi-output estimation. For the sequential filter design, multiple KFs are used, where the estimates of a predecessor KF are used as input for the successor KF. Data compression or output fusion filter design compresses similar data using their noise covariance matrix beforehand and the fused output is used as the measurement for a single KF. A fourth method was proposed by (Singer and Kanyuck, 1971), referred to as track-to-track fusion (TTF), that employs single-output KFs and fuses their outputs considering the correlation between them.

2.4.2 Evidential Belief

A different framework for managing imperfection is based on Dempster-Shafer evidence theory (DSET) (Shafer, 1976). Shafer argued that his theory, extended from Dempster findings, includes the Bayesian approach as a special case of evidence combination (or information fusion). The difference between both relies on the assignment of uncertainty. Whereas Bayesian framework considers each state to be uncertain with its own PDF, evidential belief theory assigns uncertainties not just to each state, but also to all its possible subsets, using probability mass functions. Imagine the example of Mary giving birth and the information about the time it happened. We can can assign uncertainties to the ambiguous possibilities "5 PM" and "5 AM", in order to use the data. If the uncertainty about the information was also an aspect of the data, evidential belief approach could handle it likewise.

[1] Their discoveries went public a few years later due to secrecy during war times.

Therefore, DSET can be more adequate when fusion takes place at a higher-level estimation, that is the decision-level, referring back to the input / output classification model from Figure 2.4. Situations related to risk assessment configure classical applications (Srivastava, 2011), where we have features or local decisions as inputs, usually with ambiguous and conflicting information and we need to take a global decision out of it.

2.4.3 Fuzzy Logic

Fuzzy logic, first proposed by (Zadeh, 1965) is ideal to handle vagueness in information. Unlike classical crisp definitions, where an element x belongs or not to some definition A, fuzzy sets are characterized by a *membership function* $\mu_A(x)$ which associates a degree of pertinence of x in the interval $[0, 1]$ in A. That is, the closer $\mu_A(x)$ is to the unity, the higher is the grade of membership of x in A. Instead of defining tall people as those with height above certain lower bound, we can define a membership function that assigns continuous degrees of "tallness" to different people. Vague data can be then fused using fuzzy rules to produce outputs. An example of fusion method would be the Mamdani-type fuzzy inference (Mamdani and Assilian, 1975), that classifies an output y from a set of inputs x based on rules R, as below:

Rule R_r: If x_k is A_j and ... x_n is A_m, Then y is C_l

where

R_r: fuzzy rules, with $r = 1,\ldots,r$
x_k: input fuzzy variables, with $k = 1,\ldots,n$
A_i: antecedent linguistic value, with $i = 1,\ldots,m$
y: output variable
C_j : consequent class, with $j = 1,\ldots,c$

Fuzzy set theory differences itself from probabilistic and evidential reasoning theories by modeling the fuzzy membership of a state whose class is ill-defined, whereas the other methods model uncertainties in well-defined state classes.

It is possible to combine fuzzy theory with DSET to handle the imperfections that both approaches can deal with, that is uncertainty, ambiguity and vagueness all together, using the fuzzy evidential belief framework.

2.4.4 Possibilistic

13 years after developing the mathematical background of fuzzy information, Zadeh introduced the concept of possibility theory (Zadeh, 1978), using fuzzy sets as its basis. According to him, fuzzy sets are to possibility theory what measures are to probability.

A membership function $\mu_A(x)$ of a fuzzy set A of a universe of discourse X can be interpreted as the compatibility of x with the concept labeled as A. Letting U be a variable that takes values in X, we can define the *possibility function* π_x of x associated with U, to be equal to the membership function of A, that is $\pi_X(u) \triangleq \mu_F(x)$. The interpretation, however, is that the closer $\pi_X(u)$ is to the unity, the more plausible that value is to be true.

Another way to understand possibility functions is in its comparison to density functions in probability. In the information "Mary has failed a few times in Digital Control course", let us consider X as the number of times in the universe U = 1,2,3.... The membership function $\mu_F(x)$ can model how close to "a few times" is the fuzzy variable x. From such model, we also define the possibility function $\pi_X(u)$ that can be interpreted as how possible it is that Mary has failed u times in that course. We could also model such lack of information in data by a probability function $\rho_X(u)$, as how likely it is for Mary to fail u times. Let us consider that an educated set of criteria was employed to define the discrete values of both functions as shown in Table 2.1

Table 2.1: Possibility and probability functions associated with how many times Mary has failed Digital Control

u	1	2	3	4	5	6	7	8	9
$\pi_X(u)$	1	1	1	1	0.8	0.6	0.4	0.2	0.1
$\rho_X(u)$	0.3	0.4	0.2	0.05	0.03	0.02	0	0	0

Note that according to Table 2.1 it is perfectly possible for Mary to have failed 1, 2, 3 of 4 times. The degree of possibility decreases for higher values and there is an upper bound, from which Mary would have been expelled, and could not have failed that many times. On the other hand, based on recent flunk history, the most probable number of times for Mary to have failed is 2, whereas since no student failed more than 6 times, the probability that Mary will fail that many is 0. Thus, possibilistic approach can be more appropriate to cope with incomplete data, with missing information about

a lower or an upper bound, for instance, in which case it is clear that some values are impossible instead of unlikely.

The fusion approach to possibilistic data models was extensively studied by (Dubois and Prade, 2000) and it is similar to the rules employed in fuzzy fusion. The design of the rules set is based on how plausible the sources of data are.

2.4.5 Random Set

So far, the methods presented are able to cover all aspects of uncertainty and imprecision. Although some of them can tackle multiple aspects, like DSET and Fuzzy DSET, none of them can handle all the sources of imprecision and uncertainty altogether. The random set approach to data fusion, proposed by (Goodman et al., 1997), offers such potential to integrate all these aspects in one unifying structure. The idea was to generalize the single-sensor, single-target statistics (random variables) to a broader multi-sensor, multi-target statistics (random sets), also known as finite-set statistics (FISST). The direct mathematical parallels between them are presented in Table 2.2.

Table 2.2: Direct mathematical parallels between single-sensor, single-target and multi-sensor, multi-target. Adapted from (Goodman et al., 1997; Mahler, 2004)

Single-sensor / target	**Multi-sensor / target**
random vector, Z	finite random set, Σ
sensor	global sensor
target	global target
observation, z	global observation-set, Z
parameter, θ	global parameter-set, Θ
derivative, $\frac{dp_z}{dz}$	set derivative, $\frac{\delta\beta_\Sigma}{\delta Z}$
integral, $\int_S f_Z(z\|x)d\lambda(z)$	set integral, $\int_S f_\Sigma(Z\|X)\delta Z$
probability measure, $p_Z(S\|x)$	belief measure, $\beta_\Sigma(S\|X)$
pior PDF, $f_X(X)$	global PDF, $f_\Gamma(X)$
likelihood, $L_z(X)$	global likelihood, $L_Z(X)$
information theory	multi-target information theory
filtering theory	multi-target filtering theory

By modeling the system states and measurements as random sets of finite size in-

stead of vectors of random variables, a variety of different phenomena can be described, such as target disappearance/appearance, extended/unresolved targets, missing measurements and false alarms (Khaleghi et al., 2013). As described by (Goodman et al., 1997), random set approach can model systems that are comprised of randomly varying numbers of randomly varying objects of various kinds.

Mathematically speaking, random sets are random elements whose values are sets. A random set U is a finite set, whose power set $\mathcal{P}(U)$ is composed of elements described by some specific probability law. In other words, it is defined by

$$f : \mathcal{P}(U) \rightarrow [0,1] \quad \text{with} \quad \sum_{A \in \mathcal{P}(U)} f(A) = 1 \tag{2.2}$$

where f can be interpreted as a PDF defined on sets rather than on points of U. That is, the probability that the subset A of U is selected is $f(A)$.

Efficient applications of random set theory have been studied in tasks such as system identification and time-series forecasting (Nuñez-Garcia and Wolkenhauer, 2002), target tracking (Maehlisch et al., 2006) and econometrics (Molchanov and Molinari, 2014).

2.4.6 Rough Set

Granularity is the only type of imperfection left from the categorization presented in Figure 2.6. It refers to the extent to which objects can be distinguished by data, considering the features or attributes that define them. Additionally, the way a set of features is designed will depend on a given knowledge base. Some objects might be definable considering one set, but undefinable in another set. Data can be collected in a very refined universe of discourse, while the universe of concepts in which we transform data into knowledge is coarser, and thus some objects in the data might be indiscernible.

Based on these concepts, (Pawlak, 1991) developed the rough set theory, which enables dealing with different data granularities by means of *approximation spaces*. The idea is to provide crisp lower and upper bounds to undefined rough set objects in a given knowledge base. Let B be a subset of features chosen to describe objects in the universe of A, that is $B \subseteq A$. If there is a target X that is undefined in B, its definition can be approximated by two sets (Pawlak and Skowron, 2007)

$$
\begin{aligned}
B_*(X) &= \{x \in A : B(x) \subseteq X\}, \\
B^*(X) &= \{x \in A : B(x) \cap X \neq \emptyset\},
\end{aligned}
\tag{2.3}
$$

and

$$
BN_B(X) = B^*(X) - B_*(X), \tag{2.4}
$$

where $B_*(X)$ and $B^*(X)$ are the *B-lower* and *B-upper* approximation of X, respectively and $BN_B(X)$ is the approximated X, also called as *B-boundary* region. If the boundary region is empty, set X is crisp or exact with respect to B, whereas if it is not empty, set X is rough with respect to B. Figure 2.7 illustrates an approximate space in which the the universe is partitioned into elementary squares for which subset X is undefined, but can be approximated is by upper and lower bounds.

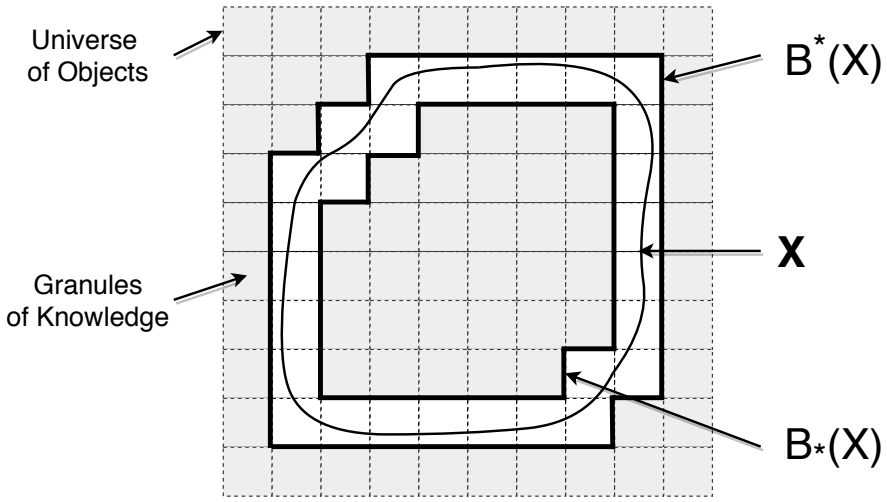


Figure 2.7: Best approximation of the rough set X by lower and upper crisp sets.

In Pawlak's book and in the references therein, many real life applications of rough set theory are explored, such as civil engineering, medical data analysis, aircraft pilot performance evaluation, vibration analysis and image processing.

Table 2.3: Data fusion methods for imperfect data. Adapted from (Khaleghi et al., 2013), Table 1, page 35.

Algorithm	Approach	Advantages	Limitations
Probabilistic	Bayesian framework to fuse uncertain data represented by probability density functions	Well-established and optimal for certain conditions	Might be unsuited for other data imperfections
Evidential Belief	Data fusion based on probability mass function, using Dempster-Shafer theory and combination rules	Enables fusion of both uncertain and ambiguous information	Incapable of dealing with other aspects of imperfection
Fuzzy Reasoning	Vague data represented by fuzzy set theory and fusion based on fuzzy rules	Intuitive and interpretable approach for vague data, such as human generated	Only applicable to vague data
Possibilistic	Data fusion based on fuzzy theory, with data representation similar to probabilistic and evidential belief	Indicated for poorly informed environment with incomplete data	Not very common and well-established
Rough Set	Imprecise data is approximated based on granularity and manipulated via classical set theory	Dispenses preliminary or additional information	Data granularity must be adequate
Random Set	Extension of Bayesian filter, representing the state space as a random set to capture many aspects of imperfection	Can potentially provide a unified framework for fusion of imperfect data	Not very appreciated by the fusion community
Hybridization	Combination of different fusion methods and data representation	More comprehensive treatment of data imperfection and benefits from complementary fusion	Computational expensive and very problem specific

2.5 Chapter Summary and Final Remarks

In this chapter, sensor fusion literature is reviewed. We explore the discussion about performance improvement from the combination of information from multiple

sources, that have been drawing attention for many decades. The main reasons for the evolution of sensor fusion as a field of science are presented in a comprehensive way, divided by the expected advantages in data authenticity and data availability. We continue with the definition and classification of sensor fusion approaches. Four main taxonomies are reviewed: classification based on sensor interaction; the input/output model based on the three fusion levels approach; sensor network architecture designs; and the categorization based on data challenge hierarchy. The last approach is specially interesting, since it relates to the core of this study: sensor fusion in the presence of challenges presented by data. Algorithms that deal with the many aspects of data imperfection are summarized in Table 2.3, such as: probabilistic; evidential belief; fuzzy logic; possibilistic; random set; and rough set theory. Hybrid methods have also being studied in the field, with interesting results.

Being the most popular method in the fusion community, the remaining of our study will focus on probabilistic methods for data fusion and its application on state estimation for sampled-data systems. A new data challenge is often present when complex sensor architectures are used to observe system states, which is sampling irregularities.

Chapter 3

Irregular Sampling

In the last chapter, we reviewed sensor fusion motivations, advantages and techniques. Despite all the growth and benefits from fusing data form multiple sensors, some challenges naturally appear. For the state estimation problem in sampled-data systems, a common challenge is related to sampling irregularities introduced in the network.

In this chapter, we review the irregular sampling problem. First, in Section 3.1 we categorize the different types of irregularities that may occur in sampling and discuss their main causes and particularities. Then, in Secton 3.2, each irregularity is further discussed, with examples, mathematical models and their subdivisions, where applicable. We end this chapter with a discussion of time synchronization in Section 3.3, which is needed to guarantee a common time scale for all nodes in network, enabling the irregularities to be dealt with appropriately.

3.1 Introduction

Sampling irregularities may occur due to a variety of issues. Sometimes as undesired side effects of using large sensor networks architectures and others due to deliberate non-uniform sampling schemes. In this section we try to categorize and review the main irregularities observed in practice. The diagram in Figure 3.1 provides a simplified overview of them, separated by their sources.

Networked system monitoring and control appears to be the main cause of irregular sampling. Unreliable communication channels may lead to random time delays and loss of information, specially if data are transmitted using a common media (Sahebsara et al., 2007; Moayedi et al., 2011). In case they get randomly interrupted during transmission or if a sensor fails at some point, the signal received may predominantly contain noise, causing uncertain observation or packet dropouts (Hadidi and Schwartz,

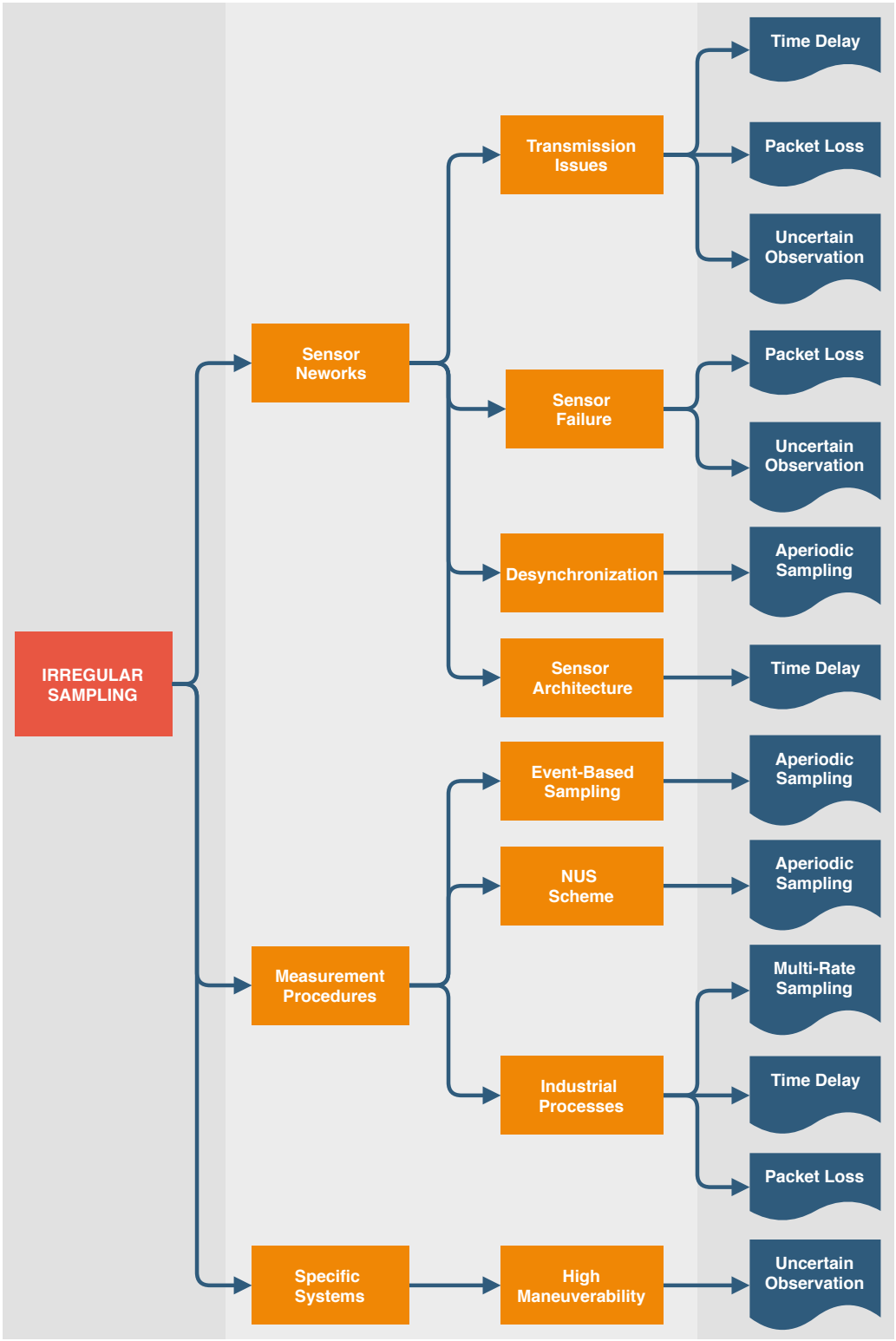


Figure 3.1: Irregular sampling diagram, showing the main causes (in orange) and effects (in blue) of irregularities

1979; Wang et al., 2009). Systems that are observed by a large number of desynchronized sensors will provide observations at random time intervals (Micheli and Jordan, 2002). If they are synchronized but designed to operate in a centralized fashion, there is a chance that different time delays are produced due to distinct transmission routes for each sensor (Bar-Shalom, 2000; Challa et al., 2003; Anxi et al., 2005).

However the communication networks shall not always be held responsible. Some applications are designed to be measured in an irregular way. In event-based schemes, for example, the measurements are transmitted only when certain conditions are met (Liu et al., 2014; Zou et al., 2017). Such approach can reduce communication resource consumption substantially (Hu et al., 2017), but will cause aperiodic sampling. Non-Uniform Sampling (NUS) is also intentionally used as an alias detection method (Kunoh, 2015) or to enhance the spectral resolution of signals, largely used in Nuclear Magnetic Resonance (NMR) spectroscopy analysis (Hyberts et al., 2013). In other situations, due to the nature of the process being observed, the measurement strategy relies on different procedures. A lot of chemical processes, for instance, can be measured in an online, fast rate and delay free fashion, but provides inaccurate data. Therefore, lab analyses are used to improve estimation quality, but they are usually gathered at slower rates, sometimes irregularly and with possible time delays (Fatehi and Huang, 2017). Other industrial applications suffer from the same dilemma, and the sampling scheme ends up with a multi-rate data transmission, with random time delays and possibly measurement scarcity (Peñarrocha et al., 2012).

Finally, sampling irregularities might also appear due to a specific nature of a system. In some high maneuverable target-tracking applications, for example, there is a chance that the sensor misses the target, transmitting only noise, leading to the so called uncertain observation issue (Wang et al., 2009; Chen et al., 2013).

Whatever the reason for the irregularities, data need be associated in order to be fused into knowledge. A crucial part of association is temporal synchronization of observations, so that the exact time at which measurements are taken are available (Ping, 2003). Most sensor fusion methods for irregularly sampled systems rely on the correct time-stamps to develop modifications in classical algorithms. If observations are imprecisely time stamped, some alternatives have been proposed (Julier and Uhlmann, 2005; Huck et al., 2011) that incorporate some aspects of that imprecision in the estimation method. Still, some knowledge about the irregularity is assumed to be known. Alternatively, many techniques for time synchronization can be performed, to ensure

a common time scale for all the sensors.

On the next sections, we review the main irregular sampling effects and the main approaches to deal with time synchronization in sensor networks.

3.2 Sampling Irregularity Types

3.2.1 Time Delay

Time-delay systems (TDS) are probably the most common mathematical representation to time delays in practice. The works of (Richard, 2003; Fridman, 2014) and the references therein provide a good coverage of the subject. In TDSs, there might be delays in the input or in the output signals, introduced by communication networks, or even in the states themselves. The latter phenomenon is called system with aftereffect or dead-time. Since we are studying the irregular sampling issue, only signal delays are relevant to us.

Considering delays in the measurement model only, (Lu et al., 2005) studied the estimation problem when they are constant and known. They describe a linear measurement model as

$$y_i(t) = H_i(t)x(t_i) + v_i(t) \tag{3.1}$$

where $i = 0, 1, \ldots, l$ and l is the number of different known delays. $y_i(t) \in \mathbb{R}^{p_i}$ are delayed measurements and $v_i(t) \in \mathbb{R}^{p_i}$ the measurement noises. The known delayed time instants are given by $t_i = t_{i-1} - d_i$, with $d_0 = 0$, $d_i > 0$ for $i > 0$ and $t_0 = t$.

For some systems, delays might not be known and constant, but still multiple of a fixed value. In such cases, observations might be received in a burst, when more than one packet arrive between two consecutive sampling instants. When that happens, the estimator might use only the latest measurement and discard all others, or implement a buffer to iterate over all received packets (Moayedi et al., 2011).

However, in many applications the measurements are received by the estimator with irregular and unknown delays, although taken at regular time intervals. In such cases, time delays can be interpreted as a stochastic process $d(k)$, varying randomly throughout time. (Han and Zhang, 2009) describes a discrete-time measurement model for random delayed observations as

$$y(k) = H(k)x(k - d(k)) + L(k)v(k) \tag{3.2}$$

where $d(k)$ is a random but bounded time delay, assumed to be a discrete-time Markov Chain observable at each sampling time k.

Multiple of a known lag or not, delayed measurements from a multisensor system are subject to arrive disordely, which leads to the sampling irregularity commonly known as out-of-sequence-measurements (OOSM). It can be classified in three ways, depending on the number of lags, according to Figure 3.2.

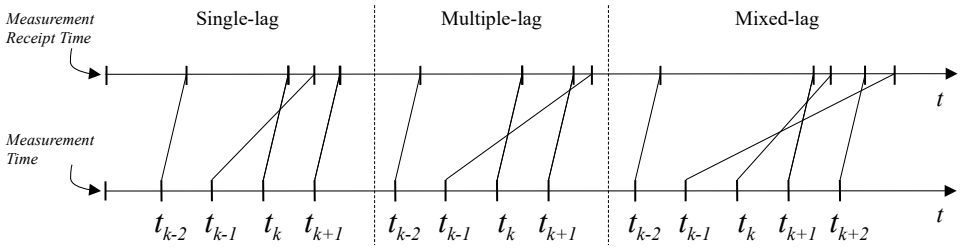


Figure 3.2: Different classes of out-of-sequence measurements irregularities

(Anxi et al., 2005) describes four different filtering approaches to deal with OOSM: reprocessing, that stores filter results to rollback with the time-delayed measurement; data buffering, that holds a set of measurements, greater than the maximum expected lag, to be sorted before filtering; discarding data, that neglects time-delayed measurements; and directly updating, that uses the delayed information to update current state estimate. (Bar-Shalom, 2000) used the last approach to describe an optimal filter for the single-lag case.

A summary of causes and effects that time delay causes in an estimator are illustrated in Figure 3.3.

3.2.2 Packet Loss

When data are being transmitted by a large network of sensors, there is a probability that they get lost in the way or they might arrive after a significant delay, which is equivalent to a loss for practical applications (Sinopoli et al., 2004). Usually referred to as packet dropout/loss, missing/intermittent observations or scarse measurements (Al-

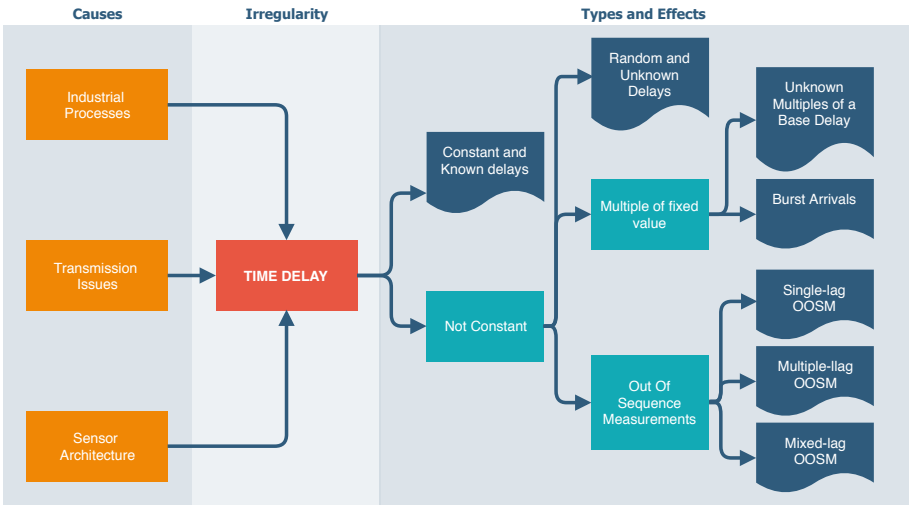


Figure 3.3: Time delay diagram, showing the main causes (in orange) and effects (in dark blue) of time delay irregularity. The light blue boxes indicate different types that lead to different effects.

bertos et al., 2004) they may happen due to node failures, network congestion, limited bandwidth or temporal failure.

Mathematical description of packet dropouts can be carried out recursively, as described in (Ma and Sun, 2011), by

$$\begin{aligned} z(t) &= H(t)x(t) + v(t), \\ y(t) &= \xi(t)z(t) + (1 - \xi(t))y(t-1), \end{aligned} \tag{3.3}$$

where $z(t) \in \mathbf{R}^m$ is the measured output transmitted to the estimator, $v(t) \in (R)^m$ is white noise, $y(t) \in \mathbf{R}^m$ is the measurement received by the estimator and $\xi(t) \sim Ber(p)$ is a Bernoulli random variable that takes the value 1 with probability p and 0 with probability $1-p$. That is, when $\xi(t)$ is 1, there is no packet dropout. If $\xi(t)$ is 0, however, the latest output is used at current time, in a recursive fashion.

Another way of describing multiple packet dropouts is by limiting the amount of consecutive dropouts (Shuli Sun et al., 2008), where the received measurements are defined by

$$\begin{aligned} y(t) =& \xi(t)z(t) + (1-\xi(t))\xi(t-1)z(t-1)+\dots \\ &+ (1-\xi(t))(1-\xi(t-1))\dots(1-\xi(t-N+1))z(t-N), N \geq 1, \end{aligned} \tag{3.4}$$

Such a model dictates that the measurement used by the estimator will be only the most recent available, and the amount of missing observations is limited to N. This conclusion can be drawn by the fact that

$$\xi(t) + (1-\xi(t))\xi(t-1)+\dots+(1-\xi(t))(1-\xi(t-1))\dots(1-\xi(t-N+1)) = 1. \tag{3.5}$$

A summary of causes and effects that time delay causes in an estimator are illustrated in Figure 3.4.

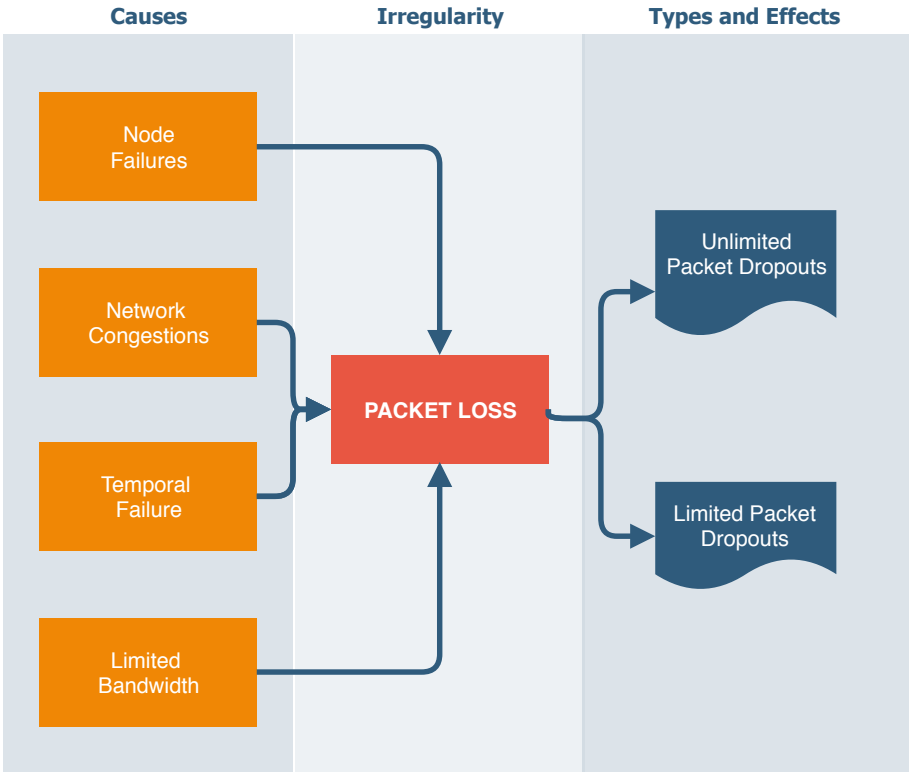


Figure 3.4: Packet loss diagram, showing the main causes (in orange) and effects (in dark blue) of time delay irregularity.

3.2.3 Uncertain Observation

For some applications, there is a chance that the observation signal sent to the estimator contains only noise. According to (Jaffer and Gupta, 1971), it happens as a consequence of two situations: the observation was taken, but was lost during transmission, due to communication failures; or it was not transmitted at all, as it may happen for target tracking systems, for example, when the object being observed is not tracked at a sample time. An observation model for a sampled-data system with uncertain observations can be described as

$$y(k) = \gamma(k)Cx(k) + Dv(k) \tag{3.6}$$

where $\gamma(k) \sim Ber(p(k))$ is a Bernoulli random variables, taking values of 0 or 1, with probabilities $p(k)$ and $1 - p(k)$, respectively.

Unlike the packet dropout problem, when the missing data are associated with the total absence of signal, the issue of uncertain observation has to be dealt with differently. A common approach is to detect the existence of signal prior to the assimilation, using a likelihood ratio test. (Middleton and Esposito, 1968) proposes a joint approach to systematically detect and extract information from observation signals. If the estimator and detector are developed separately, the probability of false alarms is not used in the estimator, making it sub-optimal. (Nahi, 1969) developed an optimal recursive estimator, that uses the information of the random variable γ in the algorithm, assuming its sequence is independent and identically distributed. (Hadidi and Schwartz, 1979) generalized the work of Nahi, for the case when the uncertainty of the signals presence is described by a Markovian sequence of binary random variables.

A summary of causes and effects that time delay causes in an estimator are illustrated in Figure 3.5.

3.2.4 Aperiodic Sampling

All irregularities discussed so far may be present even in a periodic sampling scheme. However, for some applications, the sampling intervals are time-varying due to a variety of phenomena, causing the so called aperiodic or asynchronous sampling. It can be the case of networked and embedded control systems, with unpredictable networked-induced issues, like irregular faults on samplers, oscillated loads, intermit-

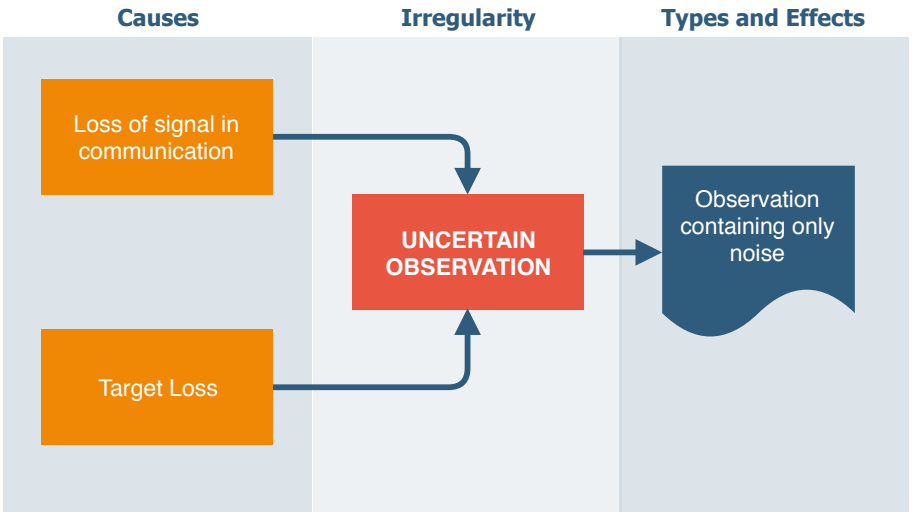


Figure 3.5: Uncertain observations diagram, showing the main causes (in orange) and effects (in dark blue) of time delay irregularity.

tent saturation or even variations in system components or parameters (Shen et al., 2016). Some imperfections may cause what is known as sampling jitter noise, which leads to time intervals being almost uniform. Automotive applications, radar imaging or event controlled systems are a few examples. In them, jitter noise happens due to a sampling frequency similar to the clock frequency; to sampling requests delayed by the network; or to imperfect synchronization (Eng and Gustafsson, 2005). For networks with a large amount of unsynchronized sensors, measurement arrival time instants are randomly spaced and can be modeled as a stochastic process (Micheli and Jordan, 2002). Sometimes, the system being observed has particularities that causes the aperiodic sampling. One example is seismology, where the spatial coordinates are irregularly sampled, because of natural obstacles (Marvasti, 2001). Other large scale systems, such as power grids, have sensors with huge geographical separations and different communication links to the estimation hub, which causes multiple and random inter-observation intervals (Yan et al., 2017).

Whereas for most cases the deformities in sampling time intervals appear as unwanted effects, there are cases when the sampling rule is designed to work aperiodically. If there are limitations of communication resources (limited bandwidth or computation capacity) or a need for a reduced energy consumption, for example, time-driven sampling might be neglected in favor of an event-based scheme. In such strategy, an event-triggering mechanism is responsible for determining the sampling instants,

according to Figure 3.6. For time-driven schemes, a clock triggers the transmission instants, while event-driven sampling instants depends on the sensor output itself with an optional feedback loop from the estimator, to assess estimation performance. Therefore, the trigger mechanism design provides a trade-off between performance and resource consumption efficiency, attracting a lot of research interest (Liu et al., 2014). The most common strategy for event-driven state estimation is the send-on-delta (SOD) (Miskowicz, 2006), which triggers the transmission when the value of the measured state deviates from the previous assimilated observation by an interval $\pm\Delta$, with $\Delta > 0$. Other strategies were studied in (Zou et al., 2017). To avoid the risk of unexpected high amount of triggered measurements in a short period of time, which can lead to the dreaded Zeno behavior (Tabuada, 2007), lower-bounds can be defined both for the Δ value or for some explicit minimum inter-event time.

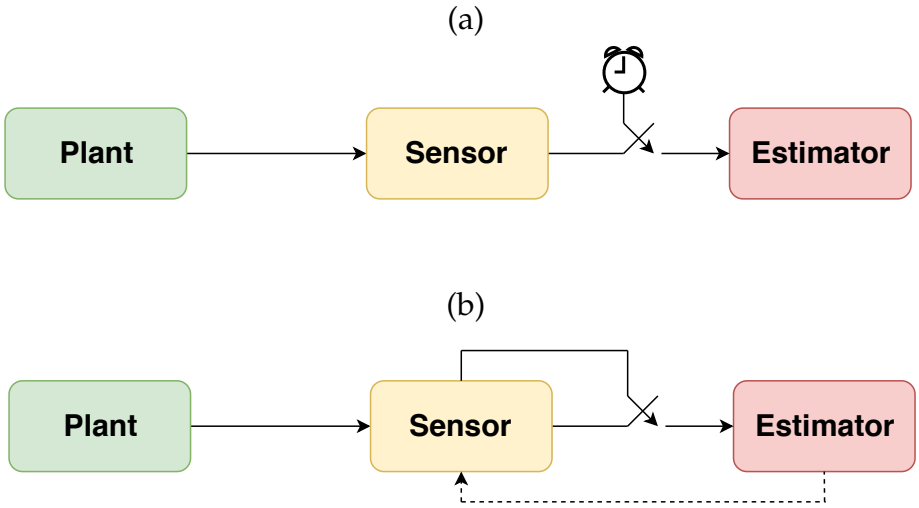


Figure 3.6: Time-driven (a) and event-driven (b) sampling schemes. The connection between sensor and estimator is triggered by different mechanisms.

The measurement model of a linear system with aperiodic sampling can be defined as

$$y(t_k) = H(t_k)x(t_k) + v(t_k) \tag{3.7}$$

where t_k is the random sampling time instant and the observation model matrix $H(t_k)$ is time-varying, if derived from the discretization of a continuous time system.

Generalizations of aperiodic sampling can be divided in two categories, based on how the estimator perceives the irregularity: as time noise added to a periodic pattern;

or as a stochastic process, according to Figure 3.7.

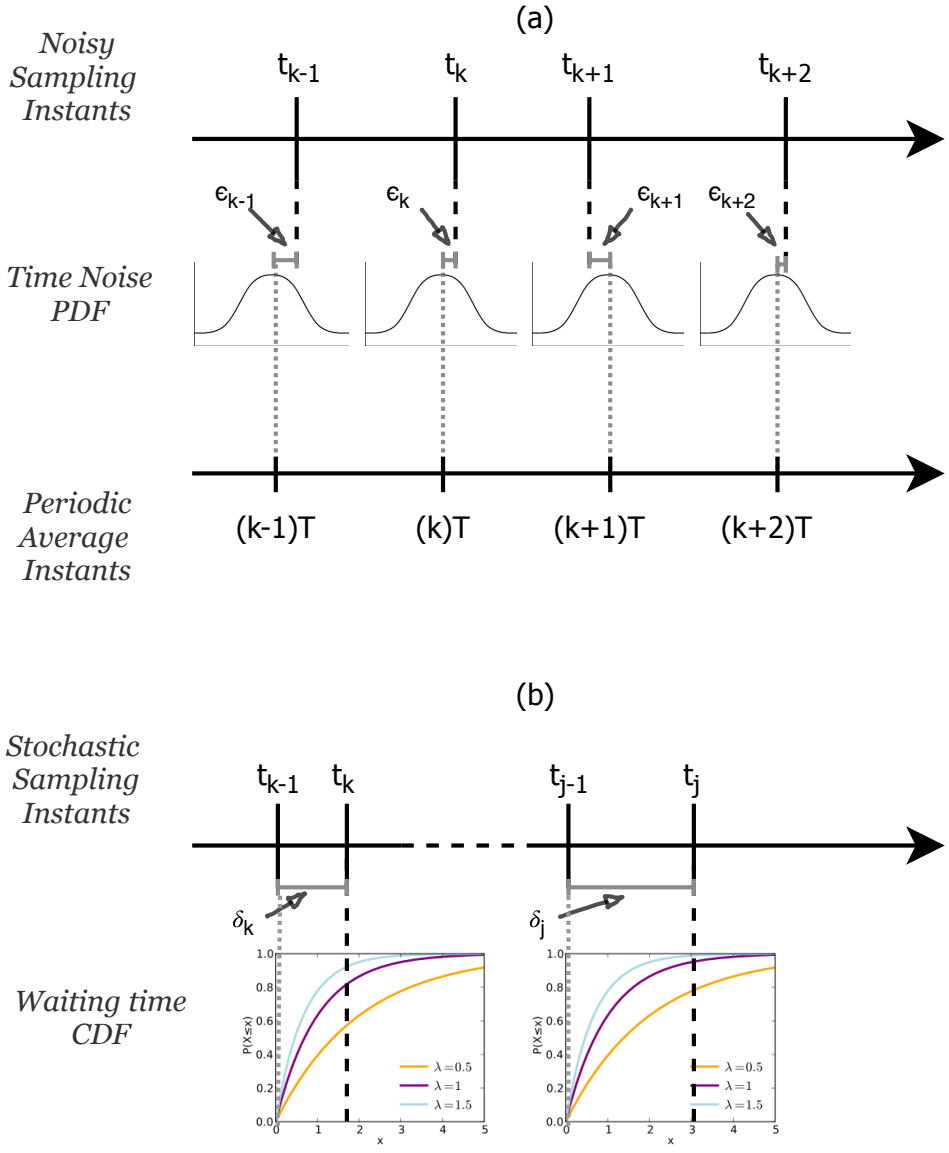


Figure 3.7: Aperiodic sampling categories: (a) noisy sampling over periodic intervals, with a Gaussian random variable added to expected time instants kT; and (b) sampling instants modeled as a stochastic process, with time intervals characterized by an exponential random variable (cumulative distribution functions are shown, for different λ parameter values.)

For the first case, random time instants t_k and the random time intervals δ_k can be defined as:

$$\begin{aligned} t_k &\triangleq kT + \epsilon_k, \\ \delta_k &\triangleq t_k - t_{k-1} \end{aligned} \tag{3.8}$$

where t_k is the kth sampling instant, T is the periodic time interval and ϵ_k is the deviation from the expected value kT. Note that, if the sampling time instants are a sequence of i.i.d Gaussian random variables, with variance σ^2, that is $t_k \sim \mathcal{N}(kT,\sigma^2)$, $\forall k \sim \mathbb{N}$, then the time interval random variable is Gaussian, with expected value T and variance $2\sigma^2$, that is $\delta_k \sim \mathcal{N}(T,2\sigma^2)$.

For the stochastic process generalization, sampling time instants t_k can be defined by the random time intervals δ_k, such as:

$$\begin{aligned} \delta_k &\triangleq t_k - t_{k-1}, \\ \delta_0 &\triangleq t_1 \end{aligned} \tag{3.9}$$

where random time intervals δ_k can be modeled, in the most flexible way, as a gamma probability density function, that is $\delta_k \sim \Gamma(\kappa,\theta)$. If the shape parameter k is a positive integer, it becomes an Erlang distribution, as used in (Kanchanaharuthai and Wongsaisuwan, 2002). For the most common case, where κ is held constant, the random time interval δ_k follows an exponential PDF and the time sequence t_k is represented by a Poisson stochastic process (Micheli and Jordan, 2002). In fact, Micheli and Jordan provided mathematical proof that for a network with N unsynchronized sensors with sampling period T, the waiting time between two arrivals tends, in distribution, to the exponential random variable, that is $\delta_k \sim \mathcal{E}(\lambda)$, where $\lambda = N/T$, as N tends to infinity.

A summary of causes and types of aperodic sampling is presented Figure 3.8.

3.2.5 Multi-Rate Sampling

The last irregularity discussed is the multi-rate sampling. Generally, it refers to multiple sensors measuring the same system at different sampling rates. Many industrial processes need to control challenging variables that can be measured by online instruments that provide regular, fast rate and delay free information, but with low precision. Therefore, more accurate data are needed and usually available after slow,

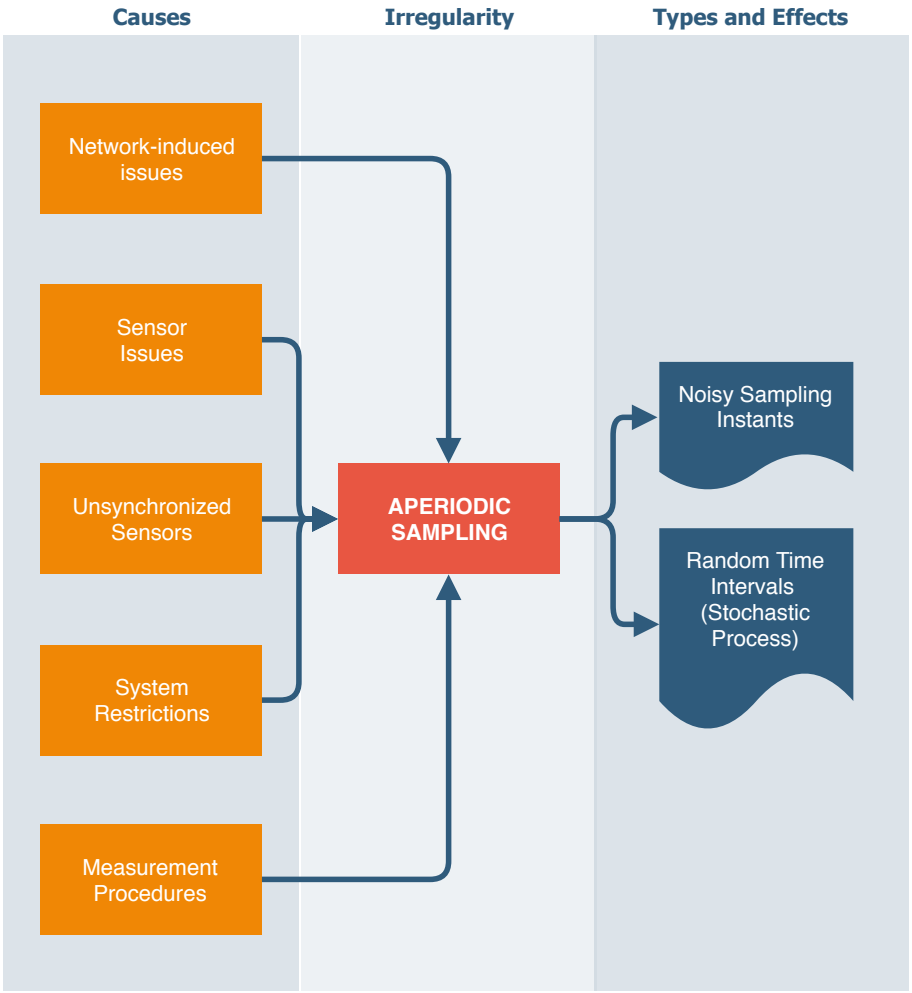


Figure 3.8: Aperiodic sampling diagram, showing the main causes (in orange) and types (in dark blue) of irregularity.

irregular and human-dependent laboratory analysis (Peñarrocha et al., 2012; Fatehi and Huang, 2017). The combination of both sources of measurements leads to a multi-rate sampling scenario.

A more common approach is the use of various sensors measuring the same physical information, to obtain better estimates, which has been drawing attention from real world applications, such as target tracking, robotics, surveillance and military. For such strategy, the sampling rates perceived by the estimator are often different from one another. The work of (Lin and Sun, 2016) and the references therein provide a wide coverage of scenarios derived from multi-sensor multi-rate systems.

Figure 3.9 illustrates the ways multi-rate sampling can be manifested in a system. The different rates from the various sensor devices can be periodic (a), aperiodic (b) or

even a mixture of both, as it is the case for most industrial applications with laboratory analysis.

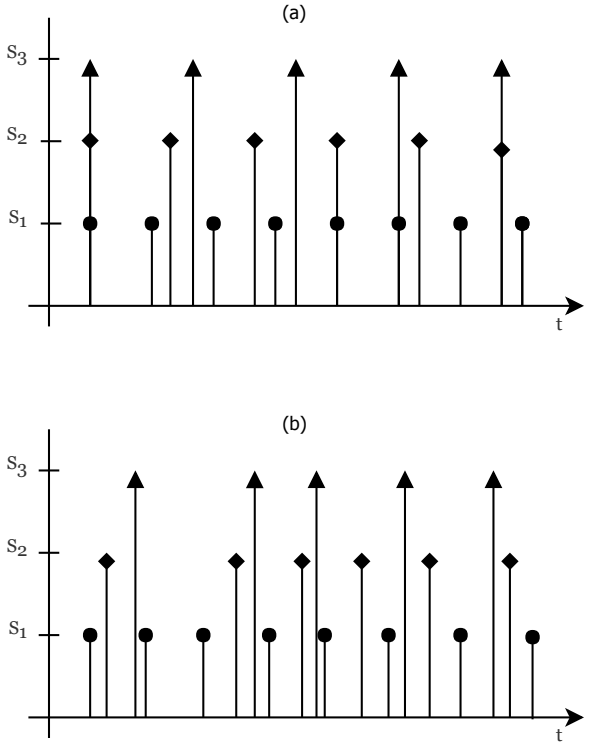


Figure 3.9: (a) Periodic and (b) aperiodic multi-rate sampling scheme. Labels S_1, S_2, S_3 refers to the sampling instants of three different sensors, from the highest rate to the lowest rate.

Aperiodic sampling rates can be described the same way as in Section 3.2.4, by equation 3.7. Periodic multi-rate measurements can be modeled as

$$y_i(k_i) = H_i(x(k_i)) + v_i(k_1) \tag{3.10}$$

where $y_i(k_i)$ represents the k_i^{th} observation from sensor i and H_i is the discrete measurement model matrix related to that sensor.

Figure 3.10 shows a schematic of causes and types of the multi-rate sampling irregularity.

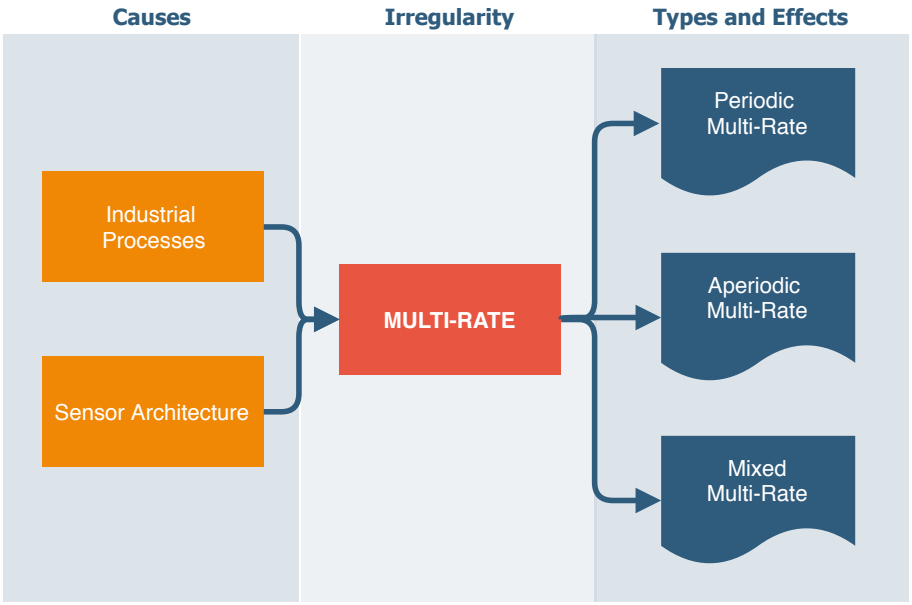


Figure 3.10: Multi-rate sampling diagram, showing the main causes (in orange) and types (in dark blue) of multi-rate sampling.

3.3 Time Synchronization

There are techniques to handle all the irregularities discussed in this chapter and still achieve efficient state estimation performance. Provided that the estimator acknowledges the time that measurements were taken, there are many studies that propose different algorithms with interesting results. For some examples, we encourage the reader to check the references from each subsection in this chapter.

In general, multi-sensor data fusion techniques for state estimation require that exact measurement time-stamps are available in order to assimilate data properly (Ping, 2003; Brahmi et al., 2013). However, that is not always the case and situations may arise in which time-stamps were not collected or their values are unreliable. Examples of the latter may occur when measurements are time stamped when they are received by the estimator instead of the moment it was taken, or when they are time stamped at local clocks without centralized synchronization (Julier and Uhlmann, 2005).

In many practical applications, if sampling irregularity cannot be accounted for accordingly, data are fused using the time of arrival as time-stamp (Huck et al., 2011), or irregularities such as OOSM are just disregarded completely (Kwok et al., 2004). (Julier and Uhlmann, 2005) proved, however, that neglecting the uncertainty in time-

stamp information may degrade estimation performance significantly in certain cases. They compared the results from traditional approaches that disregard or underestimate uncertain time stamps to algorithms that take the uncertainty statistics into account, knowingly: using fixed value for time instants derived from a calculated mean delay; a maximum likelihood algorithm for determining the predicted time measurement was taken; probabilistic data association filter (PDAF), proposed by (Bar-shalom et al., 2009); their approach, the Covariance Union (CU) algorithm; and the algorithm that uses the perfect time-stamp, to use as reference for the comparison. When irregularities happens in a situation where sensor information is sampled at much faster rates than filter update rates, the real-time particle filter (RTPF), proposed by (Kwok et al., 2004), makes efficient use of all sensor information, instead of discarding sensor readings. That is achieved by dividing the received measurements among sample sets and then representing the states as a mixture of those sets.

Alternatively, in order to avoid performance degradation, one can make use of time synchronization schemes, widely used in communication networks, to ensure global time stamps. Wireless sensor networks (WSN) are particularly dependent on such techniques, due to limited computation, energy and communication resources of the sensing devices used. The work of (Sivrikaya and Yener, 2004) provide detailed reviews of the time synchronization problem in sensor networks. They explain the problem through computer clock mechanism.

With the aid of a hardware oscillator, local clocks from a sensing device node i approximates real time t as $C_i(t)$ by

$$C_i(t) = a_i t + b_i \tag{3.11}$$

where $a_i(t)$ is the clock *drift*, that is the clock frequency, and $b_i(t)$ represents an *offset* value, or the difference from real value t.

Clock approximations from two nodes in a network are compared by

$$C_1(t) = a_{12} C_2(t) + b_{12} \tag{3.12}$$

where a_{12} is the relative *drift* and b_{12} is the relative offset between nodes.

Under Equations 3.11 and 3.12, the synchronization problem becomes the equalization of the computer clocks from all different n devices, in its most strict form. Thus, synchronization strategies can either match all clock frequencies and offsets once or

perform repeated offset corrections over time. There are more relaxed versions of synchronizations, such as the one proposed by Römer (2003), that aims at maintaining the order of events only.

Probably the most popular time synchronization method is the one being used in the internet environment for years, the Network Time Protocol (NTP), designed by (Mills, 1991). For most control and WNS applications, however, it is not suitable, due to very different requirements, such as energy consumption, precision and scalability (Ganeriwal et al., 2003). An easy solution would be to equip all sensing devices in the network with a global positioning system (GPS) for a global time synchronization, but such solution is very expensive, not energy efficient and its signal might not work properly in every environment.

Therefore, many alternative methods have been proposed, and the work of (Kaur and Abhilasha, 2015) updates Sivrikaya and Yener's studies with an exploration of the most recent synchronization protocols for sensor networks, that is: reference broadcast synchronization (RBS) (Elson et al., 2002); timing-sync protocol for sensor networks (TPSN) (Ganeriwal et al., 2003); delay measurement time synchronization (DMTS) (Ping, 2003); lightweight tree-based synchronization (LTS) (van Greunen and Rabaey, 2003); tiny-sync mini-sync (Sichitiu and Veerarittiphan, 2003); flooding time synchronization protocol (FTSP) (Maróti et al., 2004); lightweight and energy efficient time synchronization (LEETS) (Mingxia Xu et al., 2005); time diffusion protocol (TDP) (Su and Akyildiz, 2005); and time synchronization based on spanning tree (TSST) (He, 2008). A comparison adapted from Kaur ad Abhilasha is presented in Table 3.1.

3.4 Chapter Summary and Final Remarks

In this chapter, the main sampling irregularities are reviewed: time delay, packet loss, uncertain observation, aperiodic sampling and multi-rate sampling. Diagrams describing their causes, types and effects are shown for each of them. We also describe the necessary modifications to the observation models of state estimation algorithms. Most of the methods proposed in the literature to handle sampling irregularities rely on correct time stamping of observations. Thus, time synchronization in sensor networks becomes crucial and its further explored. The most recent protocols developed to ensure a global time scale from sensing devices in large sensor networks are shown

Table 3.1: Comparison of time synchronization methods. Parameters are *Precision*, *Energy Efficiency (E. E.) and Complexity (Comp.)*. Adapted from (Kaur and Abhilasha, 2015)

Protocol	**Advantages**	**Limitations**	**Parameters**
RBS	Eliminates random delays on the sender side	High amount of message exchanges and low transmission range	29.1 μs High *E. E.* High *Comp.*
TPSN	Eliminates the access, byte alignment and propagation times	Does not estimate the clock drift; does not handle dynamic topology changes and demands high communication load	16.9 μs High *E. E.* Low *Comp.*
DMTS	Reduces the number of message exchanges	Restricted to low resolution and low frequency external clocks	32 μs V. High *E. E.* Low *Comp.*
LTS	Robust and works well in the presence of dynamic links and fading.	The accuracy of synchronization decreases linearly in the depth of the synchronization tree	Unknown Low *E. E.* Low *Comp.*
Tiny-Sync Mini-Sync	Tolerant to message losses and adequate for networks with limited bandwidth and computational power	Unsuited for mobile sensor networks, high convergence time, not scalable and little robustness	945 μs High *E. E.* Low *Comp.*
FTSP	Robust, handles dynamic topology changes well and eliminates maximum delay components	Does not eliminate propagation delay and is not scalable	1.48 μs High *E. E.* High *Comp.*
LEETS	Low power consumption and low amount of message exchanges	Requires a GPS receiver in the root node	30 μs High *E. E.* Low *Comp.*
TDP	Tolerant to message losses, high mobility and performs well even without external servers	Very high convergence time	100 μs High *E. E.* High *Comp.*
TSST	Low synchronization error	Not scalable	Unknown Low *E. E.* Low *Comp.*

and compared.

However, the use of any time synchronization method will require computational, energy and resource consumption to some extent, apart from complex algorithms implementations. For sensor fusion performance applications in state estimation of sampled-data systems with irregular sampling, the investment might not be worth it. Thus, the next chapters try to shed some light in the evaluation of performance degradation in state estimation in the presence of irregular sampling, if time-stamps are not available.

Chapter 4

Methodology

This chapter narrows the field of sensor fusion under irregular sampling down to the problem of estimating the states of a system with a known process model, that is observed by aperiodically sampled measurements. We begin with a review of the adopted algorithm, that is the Kalman Filter, which is the most common approach to probabilistic data fusion. The nonlinear extension based on the unscented transform is also explored. In Section 4.2 we describe the particularities of the filtering algorithms for when the correct time-stamp is available to the estimator and when it is not. We end with a description of the performance criteria used for results assessment, designed to quantify estimation accuracy and consistency.

4.1 Bayesian Estimation

The Bayesian approach to state estimation can be interpreted as a data fusion algorithm in which the inferred knowledge about the system's states is updated continuously as new information arrives, using not just the new data, but also the prior information. For that, both the desired knowledge of the states and the continuous information are modeled as random variables (RV), hence its classification falls under the probabilistic fusion framework.

The goal of the estimator is to statistically and recursively infer the values of the system's states, the random vector $x_k \in \mathbb{R}^n$, from noisy data, the observation vector sequence $y_1,\ldots,y_k \in \mathbb{R}^m$. The respective conditional PDF, $\rho(x_k|(y_1,\ldots,y_k))$, is called the *posterior* density function, describing the statistics of the random vector $x_k \in \mathbb{R}^n$, after the present and past experimental observations $\rho(x_k|(y_1,\ldots,y_k))$ have been taken. Thus we can find the estimation of $x_k \in \mathbb{R}^n$ using methods such as the *maximum a posteriori* (MAP) or *minimum mean square error* (MMSE), given by (Bar-Shalom et al., 2001)

$$\hat{x}_k^{MAP} \triangleq \arg\max_{x_k} \hat{\rho}(x_k|(y_1,\ldots,y_k)), \tag{4.1}$$

$$\hat{x}_k^{MMSE} \triangleq \arg\min_{x_k} E[(\hat{x}_k - x_k)^T(\hat{x}_k - x_k)|(y_1,\ldots,y_k)], \tag{4.2}$$

where $\hat{x_k}$ is the estimated value of $x_k \in \mathbb{R}^n$, $\hat{\rho}(x_k|(y_1,\ldots,y_k))$ is the estimated posterior density of x_k given the observation sequence, and $E[(\hat{x}_k - x_k)^T(\hat{x}_k - x_k)|(y_1,\ldots,y_k)]$ it the variance of the random vector x_k, given the observation sequence. MAP can be interpreted as the bayesian approach to *maximum likelihood* (ML) estimation, whereas MMSE is the counterpart of the *least squares* (LS) estimator.

Finding the *posterior* $\rho(x_k|(y_1,\ldots,y_k))$ defines the complete state estimation problem, whereas the estimates $\hat{x_k}^{MAP} \in \mathbb{R}^n$ and $\hat{x_k}^{MMSE} \in \mathbb{R}^n$ are the optimal state estimates, under their optimality criteria. A recursive Bayesian solution to the state estimation problem, that is finding the *posterior* PDF, considering that the system evolves according a Markov process is presented in Proposition 4.1.1. But first, we need to define two lemmas.

Lemma 4.1.1. For a Markov system with initial state $x_0 \sim \rho(x_0)$, the transition density of the future state x_{k+1} given the present state x_k is independent of past states, that is

$$\rho(x_{k+1}|(x_0,\ldots,x_k)) = \rho(x_{k+1}|x_k), \tag{4.3}$$

and observation vector y_k is independent of past observations and past states, if present state x_k is given, that is

$$\rho(y_k|(x_0,\ldots,x_k,y_0,\ldots,y_{k-1}) = \rho(y_k|x_k). \tag{4.4}$$

Lemma 4.1.2. For a Markov system, we can find the transition density from step n to step s as a function of the transition densities between them and an intermediate step s r, as long as $n > r > s$, by the Chapman-Kolmogorov equation (Papoulis, 1984)

$$\rho(x_n|x_s) = \int_{-\infty}^{\infty} \rho(x_n|x_r)\rho(x_r|x_s)dx_r. \tag{4.5}$$

Note that the system described in Section 1.2 is Markovian, since the transition or process model given by (1.1) and the observation model given by (1.2) follow the definitions (4.3) and (4.4), respectively.

Proposition 4.1.1. The posterior density of system states $x_k \in \mathbb{R}^n$ conditioned on the observation vector sequence $y_1,\ldots,y_k \in \mathbb{R}^m$ is recursively given by

$$\rho(x_k|(y_1,\ldots,y_{k-1})) = \int_{\mathbb{R}^n} \rho(x_k|x_{k-1})\rho(x_{k-1}|y_1,\ldots,y_{k-1})dx_{k-1}, \tag{4.6}$$

$$\rho(x_k|(y_1,\ldots,y_k)) = \frac{\rho(y_k|x_k)\rho(x_k|(y_1,\ldots,y_{k-1}))}{\rho(y_k|(y_1,\ldots,y_{k-1}))} \tag{4.7}$$

where $k \in \mathbb{N}$, $\rho(y_k|x_k)$ is the *likelihood density*, $\rho(x_k|(y_1,\ldots,y_{k-1})$ is the *prior density*, defined before the latest measurement, $\rho(x_k|x_{k-1})$ is the *transition density* that models the evolution of x_k and $\rho(y_k|(y_1,\ldots,y_{k-1}))$ is the *evidence*, also referred to as normalizing factor, or marginalization. The evidence term is commonly presented as a constant (Särkkä, 2013), since it does not depend on the state vector x_k, and it can be computed by the Chapman-Kolmogorov equation from Lemma 4.1.2

$$\begin{aligned}\rho(y_k|(y_1,\ldots,y_{k-1}) &= \int_{\mathbb{R}^n} \rho(y_k|x_k)\rho(x_k|(y_1,\ldots,y_{k-1}))dx_k, \\ &= z_k.\end{aligned} \tag{4.8}$$

The algorithm is initialized by a known prior $\rho(x_0)$ and recursion is achieved by introducing the PDF calculated in the *forecast* step, given by (4.6), in the *data assimilation* step, given by (4.7).

Proof. The posterior PDF can be computed by the *Bayes' rule* (Stone, 2013)

$$\rho(x_k|(y_1,\ldots,y_k)) = \frac{\rho((y_1,\ldots,y_k)|x_k)\rho(x_k)}{\rho(y_1,\ldots,y_k)}. \tag{4.9}$$

Using the definition of the conditional probability, given by (Papoulis, 1984)

$$\rho((a_1,\ldots,a_k)|(a_{k+1},\ldots,a_n)) = \frac{\rho(a_1,\ldots,a_k,\ldots,a_n)}{\rho(a_{k+1},\ldots,a_n)}, \tag{4.10}$$

and the chain rule of probability, that is (Papoulis, 1984)

$$\rho(a_1,\ldots,a_n) = \rho(a_n|a_1,\ldots a_{n-1})\rho(a_1,\ldots a_{n-1}), \tag{4.11}$$

it is possible to rewrite (4.9) as

$$\rho(x_k|(y_1,\ldots,y_k)) = \frac{\rho(y_k|(y_1,\ldots,y_{k-1},x_k))\rho(y_1,\ldots,y_{k-1}|x_k)\rho(x_k)}{\rho(y_k|(y_1,\ldots,y_{k-1}))\rho(y_1,\ldots,y_{k-1})}. \tag{4.12}$$

From Lemma 4.1.1, given the current state x_k, the present y_k is independent of past observations, thus the first term of the dividend becomes $\rho(y_k|x_k)$. Additionally, Bayes' rule in the second term yields

$$\rho((y_1,\ldots,y_{k-1})|x_k)) = \frac{\rho(x_k|(y_1,\ldots,y_{k-1}))\rho((y_1,\ldots,y_{k-1}))}{\rho(x_k)}. \tag{4.13}$$

Finally, by combining all together, we have

$$\rho(x_k|(y_1,\ldots,y_k)) = \frac{\rho(y_k|x_k)\rho(x_k|(y_1,\ldots,y_{k-1}))\rho((y_1,\ldots,y_{k-1}))\rho(x_k)}{\rho(y_k|(y_1,\ldots,y_{k-1}))\rho(y_1,\ldots,y_{k-1})\rho(x_k)}, \tag{4.14}$$

which, after canceling the equal terms (in gray), proves (4.7).

To prove (4.6), we introduce a predicted state x_{k+1} in the posterior PDF, that is $\rho(x_{k+1},x_k|(y_1,\ldots,y_k))$ (Bergman, 1999). Rewriting the new conditional density with the aid of (4.10), (4.11) and Lemma 4.1.1, we have

$$\begin{aligned}\rho(x_{k+1},x_k|(y_1,\ldots,y_k)) &= \rho(x_{k+1}|(x_k,y_1,\ldots,y_k))\rho(x_k|(y_1,\ldots,y_k)) \\ &= \rho(x_{k+1}|(x_k)\rho(x_k|(y_1,\ldots,y_k)).\end{aligned} \tag{4.15}$$

The integration of both sides of (4.15) with respect to x_k yelds (4.6), which is also a Chapman-Kolmogorov equation. This last piece of proof is also an application of Lemma 4.1.2.

□

Thus, (4.7) can be interpreted as the fusion of the *prior* density or knowledge of the state with the *likelihood* density or evidence. For a better understanding, we can illustrate this process for a one-dimensional Gaussian case (Faragher, 2012), for which the densities are given by

$$\begin{aligned}\rho(x_k|(y_1,\ldots,y_{k-1})) &= \mathcal{N}(x_k|\hat{x}_{prior},\hat{\sigma}_{prior}) \\ &= \frac{1}{\sqrt{2\pi}\hat{\sigma}_{prior}}\exp\left(-\frac{(x_k-\hat{x}_{prior})^2}{2\hat{\sigma}^2_{prior}}\right),\end{aligned} \tag{4.16}$$

$$\begin{aligned}\rho(y_k|x_k) &= \mathcal{N}(y_k|x_k,\hat{\sigma}_{evidence}) \\ &= \frac{1}{\sqrt{2\pi}\hat{\sigma}_{evidence}}\exp\left(-\frac{(y_k-x_k)^2}{2\hat{\sigma}^2_{evidence}}\right),\end{aligned} \tag{4.17}$$

where the *prior* defined in (4.16) is obtained by forecasting and the *likelihood* given y (4.17) is obtained by a linear observation model, say $y_k = x_k + w_k$, with $w_k \sim \mathcal{N}(0,\sigma_{evidence})$. Under the MAP estimation method, we can combine (4.1), (4.7) (4.16) and (4.17), discarding the normalization factor, which is independent of x_k, yielding

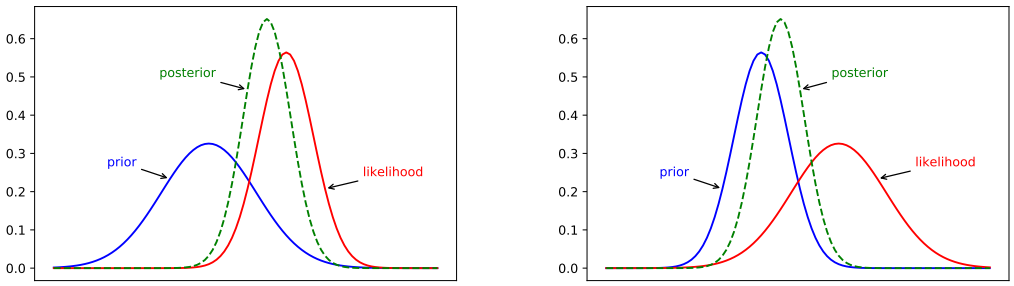
$$\begin{aligned}\hat{x}_k^{MAP} &= \arg\max_{x_k} \rho(y_k|x_k)\rho(x_k|(y_1,\ldots,y_{k-1})) \\ &= \arg\max_{x_k} \mathcal{N}(y_k|x_k,\hat{\sigma}_{evidence})\mathcal{N}(x_k|\hat{x}_{prior},\hat{\sigma}_{prior}) \\ &= \arg\max_{x_k} \mathcal{N}(x_k|\hat{x}_{posterior}(y_k),\hat{\sigma}_{posterior}) \qquad (4.18) \\ &= \hat{x}_{posterior}(y_k), \qquad (4.19)\end{aligned}$$

where

$$\begin{aligned}\hat{x}_{posterior}(y_k) &= \frac{\hat{\sigma}^2_{evidence}}{\hat{\sigma}^2_{prior}+\hat{\sigma}^2_{evidence}}\hat{x}_{prior} + \frac{\hat{\sigma}^2_{prior}}{\hat{\sigma}^2_{evidence}+\hat{\sigma}^2_{prior}}y_k \\ &= \hat{x}_{prior} + \frac{\hat{\sigma}^2_{prior}}{\hat{\sigma}^2_{evidence}+\hat{\sigma}^2_{prior}}(y_k-\hat{x}_{prior}),\end{aligned} \tag{4.20}$$

$$\begin{aligned}\hat{\sigma}_{posterior} &= \frac{\hat{\sigma}^2_{prior}\hat{\sigma}^2_{evidence}}{\hat{\sigma}^2_{prior}+\hat{\sigma}^2_{evidence}} \\ &= \hat{\sigma}^2_{prior} - \frac{\hat{\sigma}^2_{prior}}{\hat{\sigma}^2_{evidence}+\hat{\sigma}^2_{prior}}\hat{\sigma}^2_{prior}.\end{aligned} \tag{4.21}$$

The derivation from (4.20) and (4.21) comes from thes multiplication of two Gaussian functions. Hence the *posterior* density will have a Gaussian distribution, with its new parameters determined by a weighted combination of the variances of both *prior* and *likelihood* densities. The information that holds the lowest uncertainty will be favored. Figure 4.1 presents the result of two fusions, each with a different density being the most certain one. Since the densities are gaussian,



(a) (b)

Figure 4.1: *Posterior* density obtained by the fusion *prior* and *likelihood*. In (a) the variance of the *likelihood* is smaller than the variance of the *prior*, hence the *posterior* is closer to the *likelihood*. In (b) is the other way around.

4.1.1 Kalman Filter

The Bayesian recursive solution described by proposition 4.1.1 enables the computation of the optimal estimation of state vector x_k. However, its implementation is impossible, since it relies on mathematical integrations and as time evolves, the observation sequence grow indefinitely. The sequential algorithm proposed by (Kalman, 1960) solves that problem adding a restriction in the system assumptions, that is linearity and Gaussianity.

Consider the Gauss-Markov discrete-time linear system

$$x_k = A_{k-1}x_{k-1} + B_{k-1}u_{k-1} + G_{k-1}w_{k-1}, \tag{4.22}$$

$$y_k = C_k x_k + v_k, \tag{4.23}$$

where, $\forall k \geq 1$, time varying matrices $A_{k-1} \in \mathbb{R}^{n\times n}$, $B_{k-1} \in \mathbb{R}^{n\times p}$, $G_{k-1} \in \mathbb{R}^{n\times q}$ and $C_{k-1} \in \mathbb{R}^{m\times n}$ are known, as well as the input $u_{k-1} \in \mathbb{R}^p$ and output $y_k \in \mathbb{R}^m$ vectors. Process and observation noise vectors, $w_{k-1} \in \mathbb{R}^q$ and $v_{k-1} \in \mathbb{R}^m$ are white, zero-mean and mutually independent, apart from being Gaussian, with known covariance matrices Q_{k-1} and R_k, respectively.

Define $\mathcal{N}(x|\mu,\Sigma)$ as a multivariate Gaussian probability density function on x, with mean μ and covariance Σ, defined by

$$\mathcal{N}(x|\mu,\Sigma) = \frac{1}{(2\pi)^{n/2}\,|\Sigma|^{1/2}} exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right) \tag{4.24}$$

Some identities of multivariate Gaussian probability densities are needed for the next steps and are described in properties 4.1.1 and 4.1.2.

Property 4.1.1. If the random variables x and y have the joint Gaussian probability density

$$x,y \sim \mathcal{N}\left(\begin{pmatrix} a \\ b \end{pmatrix}, \begin{pmatrix} A & C \\ C^T & B \end{pmatrix}\right) \tag{4.25}$$

then, the marginal and conditional densities of x and y are given by

$$x \sim \mathcal{N}(a,A) \tag{4.26}$$
$$y \sim \mathcal{N}(b,B) \tag{4.27}$$
$$x|y \sim \mathcal{N}(a + CB^{-1}(y-b), A - CB^{-1}C^T) \tag{4.28}$$
$$y|x \sim \mathcal{N}(b + C^TA^{-1}(x-a), B - C^TA^{-1}C) \tag{4.29}$$

Property 4.1.2. If the random variables x and y have the Gaussian densities

$$\rho(x) = \mathcal{N}(x|a,A), \tag{4.30}$$
$$\rho(y|x) = \mathcal{N}(y|Hx,B), \tag{4.31}$$

Then, the joint and marginal densities are given by

$$x,y \sim \mathcal{N}\left(\begin{pmatrix} a \\ Ha \end{pmatrix}, \begin{pmatrix} A & AH^T \\ HA & HAH^T + B \end{pmatrix}\right), \tag{4.32}$$

$$y \sim \mathcal{N}(Ha, HAH^T + B) \tag{4.33}$$

From the Gauss-Markov assumptions, we can rewrite (4.22) and (4.23) from a probabilistic perspective, as

$$p(x_k|x_{k-1}) = \mathcal{N}(x_k|A_{k-1}x_{k-1} + B_{k-1}u_{k-1}, Q_k), \tag{4.34}$$

$$p(y_k|x_k) = \mathcal{N}(y_k|C_k x_k, R_k), \tag{4.35}$$

where (4.34) is the *transition* density representing the system dynamics and (4.35) is the *likelihood* density, given by the observation model.

The Bayesian recursive solution to such system is defined by *forecast* and *data assimilation* steps according to

$$p(x_k|(y_1,\ldots,y_{k-1}) = \mathcal{N}(\hat{x}_{k|k-1}, P^{xx}_{k|k-1}), \tag{4.36}$$

$$p(x_k|(y_1,\ldots,y_k) = \mathcal{N}(\hat{x}_{k|k}, P^{xx}_{k|k}), \tag{4.37}$$

with the *posterior* density function from a previous step given by

$$p(x_{k-1}|(y_1,\ldots,y_{k-1}) = \mathcal{N}(\hat{x}_{k-1|k-1}, P^{xx}_{k-1|k-1}), \tag{4.38}$$

where $\hat{x}_{k|k-1}$ and $P^{xx}_{k|k-1}$ are the *forecast* state and covariance estimates, whereas $\hat{x}_{k|k}$ and $P^{xx}_{k|k}$ are the *data assimilation* state and covariance estimates.

Now we combine the *forecast steps* from (4.6) and from (4.36), using the process model from (4.34), the previous estimate from (4.38) and the identities from property 4.1.2, yielding

$$\begin{aligned}\rho(x_k|(y_1,\ldots,y_{k-1})) &= \int_{\mathbb{R}^n} \rho(x_k|x_{k-1})\rho(x_{k-1}|y_1,\ldots,y_{k-1})dx_{k-1},\\ &= \mathcal{N}(x_k|A_{k-1}\hat{x}_{k-1|k-1} + B_{k-1}u_{k-1}, A_{k-1}P^{xx}_{k-1|k-1}Ak-1^T + G_{k-1}QG^T_{k-1}, \quad (4.39)\end{aligned}$$

that is, the *forecast* state and covariance estimates are computed by

$$\hat{x}_{k|k-1} = A_{k-1}\hat{x}_{k-1|k-1} + B_{k-1}u_{k-1}, \quad (4.40)$$

$$P^{xx}_{k|k-1} = A_{k-1}P^{xx}_{k-1|k-1}Ak-1^T + G_{k-1}QG^T_{k-1}. \quad (4.41)$$

And for the *data assimilation step*, we find the joint density of y_k with the *forecast* estimate from (4.39), using property 4.1.1

$$\rho(x_k,y_k|(y_1,\ldots,y_{k-1})) = \mathcal{N}\left(\begin{pmatrix}\hat{x}_{k|k-1}\\ C_k\hat{x}_{k|k-1}\end{pmatrix}, \begin{pmatrix}P^{xx}_{k|k-1} & P^{xx}_{k|k-1}C^T_k\\ C_kP^{xx}_{k|k-1} & C_kP^{xx}_{k|k-1}C^T_k + R_k\end{pmatrix}\right), \quad (4.42)$$

and the marginal density for x_k is

$$\rho(x_k|(y_1,\ldots,y_k)) = \mathcal{N}(\hat{x}_{k|k-1} + K_k(y_k - C_k\hat{x}_{k|k-1}), P^{xx}_{k|k-1} - P^{xx}_{k|k-1}C^T_kK^{-1}_kP^{xx}_{k|k-1}C^T_k), \quad (4.43)$$

that is, the *data assimilation* sate and covariance estimates are calculated by

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - C_k\hat{x}_{k|k-1}), \quad (4.44)$$

$$P^{xx}_{k|k} = P^{xx}_{k|k-1} - P^{xx}_{k|k-1}C^T_kK^{-1}_kP^{xx}_{k|k-1}C^T_k. \quad (4.45)$$

where $K_k \in \mathbb{R}^{n\times m}$ is defined as the Kalman gain and is given by

$$K_k = P^{xx}_{k|k-1}C^T_k(C_kP^{xx}_{k|k-1}C^T_k + R_k)^{-1}. \quad (4.46)$$

For more details on this derivation, refer to (Särkkä, 2013). We summarize the Kalman filter algorithm as below

Algorithm 4.1.1. Kalman filter (KF) algorithm.

Forecast step: Using the linear model, calculate the estimated state vector $\hat{x}_{k|k-1}$ and the estimated state covariance $P^{xx}_{k|k-1}$, using (4.40) and (4.41). Calculate Kalman gain K_k by (4.46).

Data assimilation step: Using the measurement vector y_k, update estimations from previous step with (4.44) and (4.45), obtaining estimates $\hat{x}_{k|k}$ and $P^{xx}_{k|k}$.

The linear KF is the optimal estimator under both MAP and MMSE criteria, it is unbiased and its cross-covariance will asymptotically achieve the lower bound of the Cramér-Rao inequality (Teixeira, 2008).

4.1.2 Unscented Kalman Filter

For non-linear systems, the Gaussianity requirement does not hold, even if the uncertainty and initial conditions are Gaussian. Therefore, we can not characterize the posterior PDF only by its first two moments, mean and covariance, and the solution given by proposition 4.1.1 are not suited for the estimation problem. In fact, optimal solutions are generally not possible to be obtained by a recursive algorithm.

Adaptations to the Kalman filter have been proposed in the literature. The extended Kalman filter (EKF), for example, linearizes the system around the state estimates, to achieve approximations through algorithm 4.1.1. A different approach is to approximate the statistics of the posterior PDF, instead of the model. The unscented Kalman filter (UKF) (Julier and Uhlmann, 2004) performs statistical approximations via the unscented transformation (UT), addressing many of the issues that appear in the EKF implementation, such as the necessity of differentiability, the performance loss due to systems that are poorly approximated by linearization and computational efficiency. Therefore the UKF is the chosen method for the non-linear system state estimation simulated in Chapter 5 and will be further discussed in this section.

Based on the Monte Carlo approach to statistical approximation, the UT defines a set of samples $\chi \triangleq [\chi_0, \ldots, \chi_{2n}] \in \mathbb{R}^{n\times(2n+1)}$, also referred to as sigma points, to approximate the mean $\bar{x}$ and covariance $\bar{P}^{xx}$ of the random variable being estimated, by

ALTERAR AQUI!!!!!!

$$\chi \triangleq \left[\bar{x}1_{1\times(2n+1)} + \sqrt{n}(\bar{P}^{xx})^{1/2} \quad \bar{x}1_{1\times(2n+1)} - \sqrt{n}(\bar{P}^{xx})^{1/2}\right], \tag{4.47}$$

For this study we will adopt a simplified version of the UKF, with fixed parameters. Assuming that the non-Gaussian distribution is symmetric, the choice of the sigma points is done deterministically by

$$\chi \triangleq \left[\bar{x}1_{1\times(2n+1)} + \sqrt{n}(\bar{P}^{xx})^{1/2} \quad \bar{x}1_{1\times(2n+1)} - \sqrt{n}(\bar{P}^{xx})^{1/2}\right], \tag{4.48}$$

with weights given by

$$\gamma = \frac{1}{2n}, \tag{4.49}$$

Based on this approximations, and considering the process and observation models given by (1.1) and (1.2), the UKF algorithm is defined by

Algorithm 4.1.2. Unscented Kalman filter (UKF) algorithm.

Forecast step: Calculate sigma points based on last estimation and use them to forecast state estimate via the observation model and the respective covariance.

$$\chi_{k-1|k-1} = \left[\hat{x}_{k-1|k-1}1_{1\times(2n+1)} + \sqrt{n}(P^{xx}_{k-1|k-1})^{1/2} \quad \hat{x}_{k-1|k-1}1_{1\times(2n+1)} - \sqrt{n}(P^{xx}_{k-1|k-1})^{1/2}\right], \tag{4.50}$$

$$\chi_{j,k|k-1} = f(\chi_{j,k-1|k-1}, u_{k-1}, k-1), \quad j = 1,\ldots,2n+1, \tag{4.51}$$

$$\hat{x}_{k|k-1} = \sum_{j=1}^{2n+1} \gamma \chi_{j,k|k-1}, \tag{4.52}$$

$$P^{xx}_{k|k-1} = \sum_{j=1}^{2n+1} \gamma \left[(\chi_{j,k|k-1} - \hat{x}_{k|k-1})(\chi_{j,k|k-1} - \hat{x}_{k|k-1})^T\right] + G_{k-1} Q G^T_{k-1}. \tag{4.53}$$

Calculate new sigma points based on state estimate forecast and propagate them to an estimated observation vector using the observation model. Calculate observation and state vectors covariance and the cross-covariance.

$$\chi_{k|k-1} = \left[\hat{x}_{k|k-1}1_{1\times(2n+1)} + \sqrt{n}(P^{xx}_{k|k-1})^{1/2} \quad \hat{x}_{k|k-1}1_{1\times(2n+1)} - \sqrt{n}(P^{xx}_{k|k-1})^{1/2}\right], \tag{4.54}$$

$$\Upsilon_{j,k|k-1} = g(\chi_{j,k|k-1},k), \quad j = 1,\ldots,2n+1, \tag{4.55}$$

$$\hat{y}_{k|k-1} = \sum_{j=1}^{2n+1} \gamma\Upsilon_{j,k|k-1}, \tag{4.56}$$

$$P^{yy}_{k|k-1} = \sum_{j=1}^{2n+1} \gamma\left[(\Upsilon_{j,k|k-1} - \hat{y}_{k|k-1})(\Upsilon_{j,k|k-1} - \hat{y}_{k|k-1})^T\right] + R_k. \tag{4.57}$$

$$P^{xy}_{k|k-1} = \sum_{j=1}^{2n+1} \gamma\left[(\chi_{j,k|k-1} - \hat{x}_{k|k-1})(\Upsilon_{j,k|k-1} - \hat{y}_{k|k-1})^T\right]. \tag{4.58}$$

Data assimilation step: Calculate Kalman gain and finish the estimation of state estimate and covariance at time k.

$$K_k = P^{xy}_{k|k-1}(P^{yy}_{k|k-1})^{-1}, \tag{4.59}$$

$$\hat{x}_{k-1|k-1} = \hat{x}_{k|k-1} + K_k(y_k - \hat{y}_{k|k-1}), \tag{4.60}$$

$$P^{xx}_{k|k} = P^{xx}_{k|k-1} - K_k(P^{yy}_{k|k-1})K_k^T. \tag{4.61}$$

4.2 State Estimation with Aperiodic Sampling

Um esquemático ilustrativo é apresentado na Fig., onde $\alpha = 5$, i.e. a taxa média de amostragem das observações $1/\lambda$ é cinco vezes mais lenta que a frequência de amostragem $1/T$ dos sensores que disponibilizam informações de entrada. A cada T segundos, ou seja no intervalo de tempo entre dois instantes de amostragem das entradas pode haver ou não informações de observação. No cenário ilustrado pela letra **A** na Fig. 4.2, não há medições disponíveis. Durante o intervalo **B** há apenas uma medição, ao passo que em **C** há mais de uma observação disponível.

4.2.1 With Timestamp

Sabendo o momento exato t_k em que as observações são medidas, é possível assimilá-las no instante correto, considerando intervalos de tempo variáveis para o algoritmo

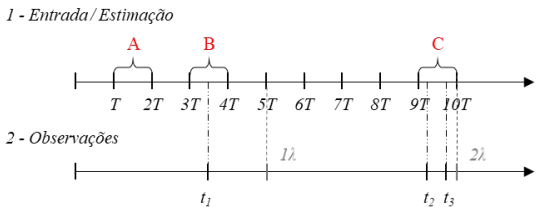


Figure 4.2: Exemplos de instantes de amostragem da entrada e das estimações (1) e das observações (2). Os instantes múltiplos de λ, igual ao intervalo médio de amostragem das observações são apresentados em cinza escuro, para referência.

de filtragem. Para isso, o modelo matemático em () é discretizado a uma taxa δt_j^* variável. Para a simulação deste artigo, os valores de δt_j^* são calculados a partir da união de todos os instantes de tempo correspondentes às chegadas de sinais de entrada ou de medição em um único vetor, de forma ordenada. Por meio da subtração de dois instantes consecutivos, é possível obter os intervalos de tempo de integração δt_j^* correspondente. No caso de uma versão *online*, a integração das equações diferenciais é executada a medida que um sinal de entrada ou um sinal de medição de saída é recebido. Nesses momentos, são utilizados o intervalo de tempo entre o último sinal recebido e o instante atual. Além disso, quando o sinal recebido é um sinal de entrada, é executada apenas a etapa de predição. E, quando o sinal for de medição, acontecem as duas etapas de predição e assimilação de dados, considerado um segurador de ordem zero para a entrada. O fluxograma da Fig. 4.3 representa o passo a passo desse algoritmo, apresentando as etapas executadas quando o sinal é de entrada ou de observação.

Dessa forma, no cenário **B** da Fig. 4.2, o algoritmo executa uma etapa completa de predição e assimilação de dados dos instantes $3T$ até t_1, considerando o intervalo $\delta t_4^* = t_1 - 3T$ (note que há 3 intervalos de tempo δt_j^*, antes o instante t_1). Em seguida é feita uma etapa de predição entre t_1 e $4T$. Durante o intervalo de tempo de integração entre t_1 e $4T$, considerou-se que a entrada permaneceu constante desde a sua última atualização em 3T. Ou seja, assume-se que não houve variação na entrada para essa etapa de predição. Caso mais de uma observação seja medida entre duas entradas (cenário **C**), são executadas etapas completas de filtragem para cada observação disponível e, ao final, uma etapa de predição entre a última observação e a próxima entrada.

4.2.2 Without Timestamp

Quando o carimbo de tempo está disponível, a assimilação de dados do algoritmo pode ser feita nos momentos exatos da medição. Para isso, a integração das equações diferenciais via discretização deve acontecer com intervalos de tempo variáveis, no caso deste trabalho utilizando o método de Runge-Kutta de 4ª ordem. Dessa forma, (**??**) pode ser reescrita como

$$x(t_j^*) = f_\mathrm{d}(x(t_{j-1}^*),u(t_{j-1}^*),w(t_{j-1}^*),t_{j-1}^*), \tag{4.62}$$

em que $t_j^* = t_{j-1}^* + \delta t_j^*$ e $t_0^* = 0$. Cada valor δt_j^* corresponde ao intervalo de tempo entre o último instante t_{j-1}^* em que se registrou a chegada de um sinal, seja ele de entrada ou de observação, e o próximo instante de tempo t_j^* em que um valor de entrada ou de observação é obtido, conforme será detalhado no Capítulo 3. Entre instantes de observação e de entrada é utilizado um segurador de ordem zero, considerando a ultima informação disponível.

Por outro lado, se o carimbo de tempo não é levado em conta, (**??**) pode ser reescrito como

$$x_n = f_\mathrm{d}^*(x_{n-1},u_{n-1},w_{n-1},n), \tag{4.63}$$

em que $t = nT$.

Quando não há carimbo de tempo nas medições, o algoritmo de filtragem não sabe o momento exato da medição t_k. Assim, o instante de tempo considerado para a etapa de assimilação de dados é sempre o próximo instante múltiplo de T, i.e. quando a próxima informação de entrada está disponível. Existem apenas dois cenários de estimação nesse caso. Primeiro, quando não há medições disponíveis entre duas entradas, o estimador executa apenas a etapa de predição entre os intervalos de tempo iT e $(i+1)T$, conforme cenário **A** da Fig. 4.2. Segundo, nos casos representados pela letra **B**, o estimador considera que a observação foi feita no próximo instante multiplo de T em que há informações de entrada. No exemplo da Fig. 4.2, a medição feita no instante t_1 é assimilada no instante $4T$. Caso haja mais de uma observação entre duas entradas (cenário **C**), a mais antiga é descartada. Os passos de discretização do modelo utilizados pelo UKF são sempre $\delta t = T$, havendo ou não observações disponíveis.

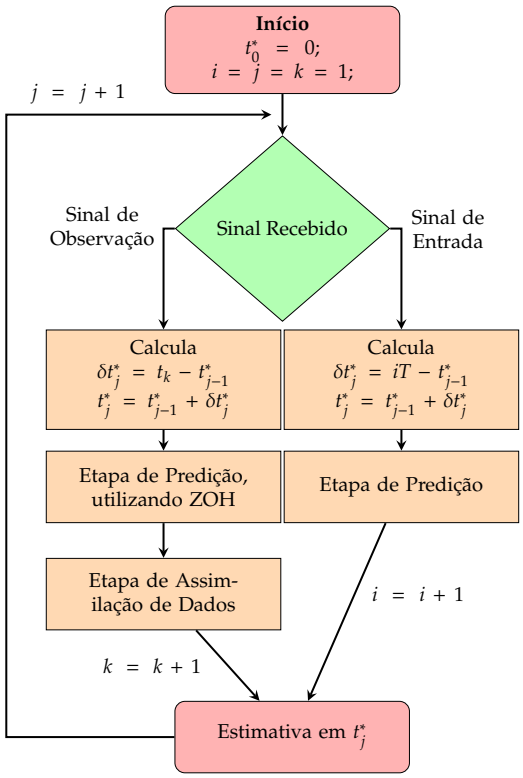


Figure 4.3: Diagrama ilustrativo da versão *online* do estimador que considera o carimbo de tempo. Os índices i, j e k representam, respectivamente os contadores dos sinais de entrada, estimação e da saída.

4.3 Performance Metrics

Chapter 5

Results

In this chapter, we present simulation results for two different systems: a linear system with two under damped modes and a nonholomonic unicycle position system.

REVISAR

We begin by describing the simulated systems as a state-space representation. First system is discretized and a discrete Kalman filter is used for the estimation of the unicycle position. The second one is represented as a continuous-time process model with digital measurements of one stby aate. We wish to estimated a all states sampled-data Kalman filter.

For both cases, we start evaluating the effect of time-stamp information in the estimation algorithm for three different irregular sampling scenarios, without time-delay in the observations: variation of the observation signal-to-noise ratio, variation of the observation average sampling rate, and variation of the relation between estimation regular sampling rate and output irregular average sampling rate.

Then we add random time-delay to the observation model and evaluate the effect for the same three scenarios plus a fourth one, where we vary the average time-delay.

Falta incluir time-delay e simulação com filtro de partículas

5.1 Linear System

Descrever as motivações por trás do estudo do sistema linear: variáveis controladas, resultado esperado conhecido, comparação mais clara.

5.1.1 System Description

The linear system designed for simulation is the serial combination of two under damped modes, with distinct band pass behaviors.

One of the modes, henceforth termed as low-pass (lp) system, has time constant $\tau_{lp} = 1$ s, natural frequency $\omega_{n,lp} = 10$ Hz and a damping constant of $\zeta_{lp} = 0.1$. The frequency response is given by:

$$G_{lp}(s) = \frac{100}{s^2 + 2s + 100} \tag{5.1}$$

The second mode is a high-pass (hp) system, with a much lower time constant $\tau_{hp} = 0.01$ s, natural frequency $\omega_{n,hp} = 1000$ Hz, same damping constant of $\zeta_{hp} = 0.1$ and two zeros, one at the origin and another one at 0.001. The high-pass frequency response is given by:

$$G_{hp}(s) = \frac{s^2 - 0.001s}{s^2 + 200s + 10^6} \tag{5.2}$$

Figure 5.1 shows the bode diagrams of both systems separately.

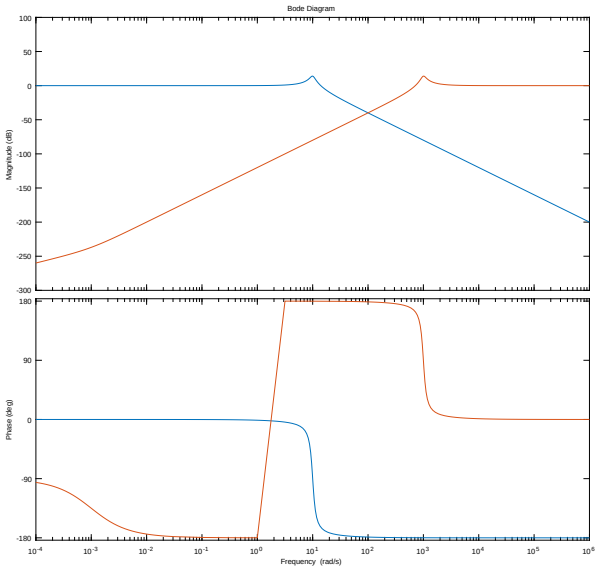


Figure 5.1: Bode diagram of both modes.

The resulting fourth order system is described as a state space representation, in a modal canonical form given by:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{5.3}$$

where $x(t) \in \mathbf{R}^4$ is the state vector, $u(t) \in \mathbf{R}^1$ is the single input vector and $y(t) \in \mathbf{R}^1$ is the single output vector, and

$$A = \begin{bmatrix} -100 & 994.99 & 0 & 0 \\ -994.99 & -100 & 0 & 0 \\ 0 & 0 & -1 & 9.949 \\ 0 & 0 & -9.949 & -1 \end{bmatrix}$$

$$B = \begin{bmatrix} -24.6435 \\ -18.8943 \\ -4.1746 \\ -0.2675 \end{bmatrix} \tag{5.4}$$

$$C = \begin{bmatrix} 24.41 & -21.2522 & -0.1537 & 2.3977 \end{bmatrix}$$

$$D = 1$$

We simulate a pseudo-random binary sequence (PRBS) as input, with 200 samples and simulate the high-pass and low-pass systems separately according to Figure 5.2. As expected, the low-pass response is much slower. Just for comparison, if the same 200 samples of PRBS was generated over 20 seconds, the low-pass system response would account for more variations in the input. Figure 5.3 show the response of the final fourth order system.

The linear system is discretized using MATLAB's 'c2d'fic

The sampled-data observations are available according to:

$$y(t_k) = \mathbf{B}_d x(t) + \mathbf{D}_d u(t) + v(t_k) \tag{5.5}$$

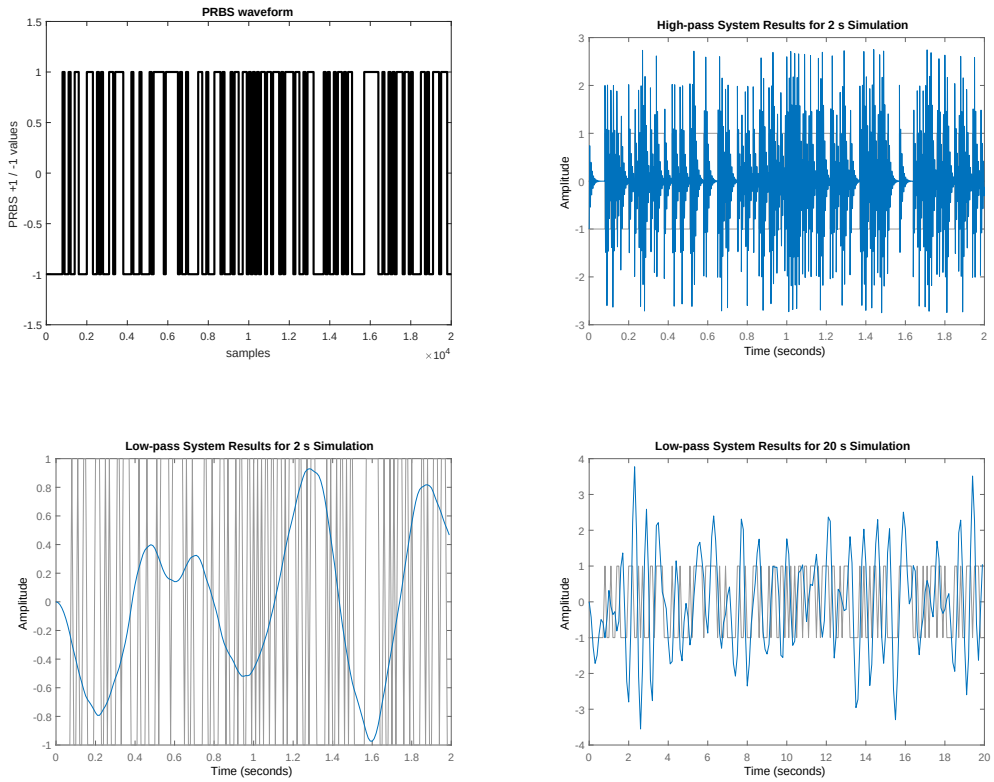


Figure 5.2: PRBS samples used as input; output of high-pass system, considering 2 seconds of simulation and low-pass system results for 2 and 20 seconds simulation

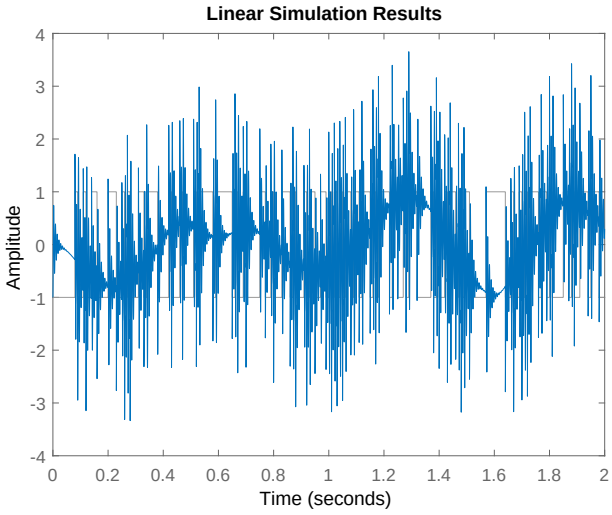


Figure 5.3: Output of the fourth order system to the PRBS input signal.

where $v(t_k) \sim \mathcal{N}(0,R_{t_k})$ is the observation noise, with zero mean and covariance R_{t_k}. When time-stamp is not available, the observation vector is approximated by $\tilde{y}_i \approx y(t_k)$, where i is the index of the next time instant, multiple of T.

Discrete-time Kalman filter is used for estimation. Realizations of the estimated states and true values for both scenarios, with and without time-stamp, for a signal-to-noise ration in the observations of 40 dB are shown in Figures 5.4

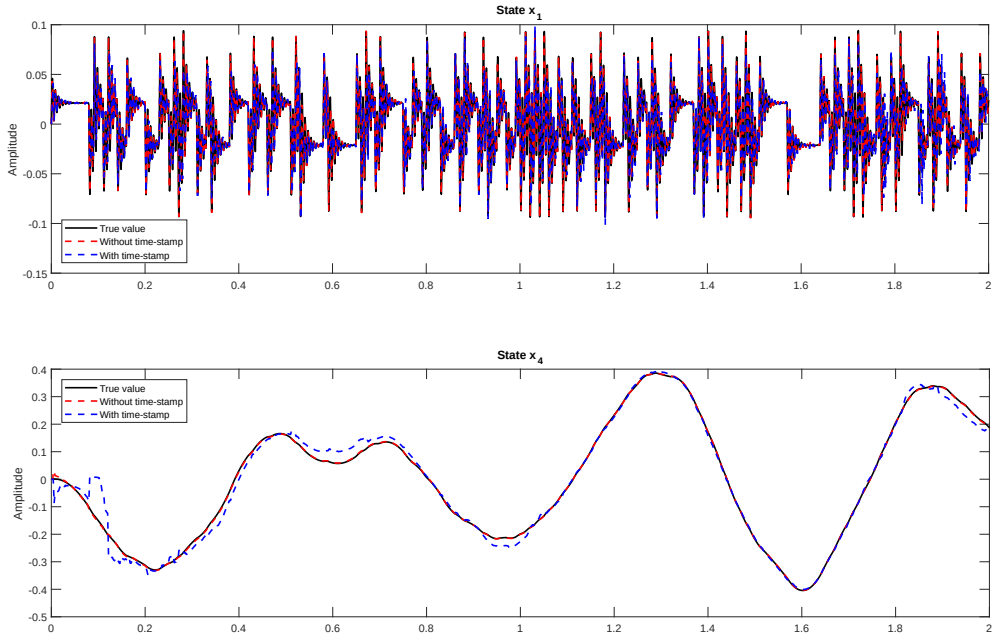


Figure 5.4: States x_1 and x_4 estimates with time-stamp (red), without time-stamp (blue) and true values (black).

Figure 5.5 presents a window data from 0 to 0.045 seconds of the first state RMSE for the algorithms with and without time-stamp. As expected, we observe a distancing from the RMSE at the instant the first observation was taken t_1, favoring the algorithm that considers time-stamp.

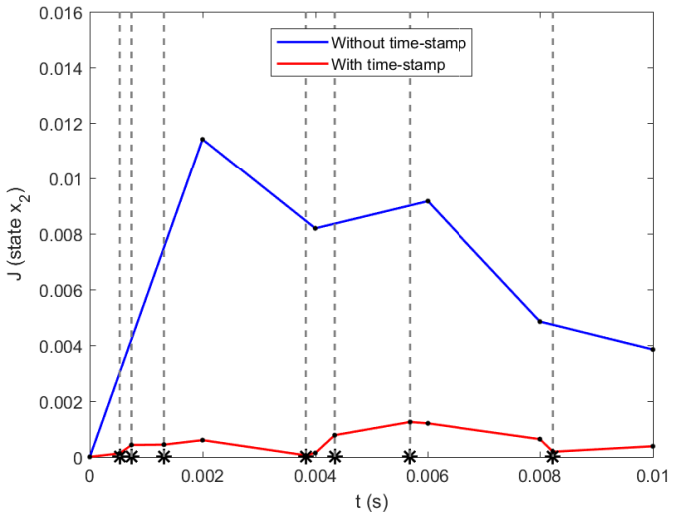


Figure 5.5: Temporal cut from 0.02 to 0.045 seconds, for a realization of pitch rate about the center of mass RMSE of both estimators, with (—) and without (—) time-stamp. Vertical dashed lines match the measurement sampling instants t_k. Black dots represent the regular estimation instants.

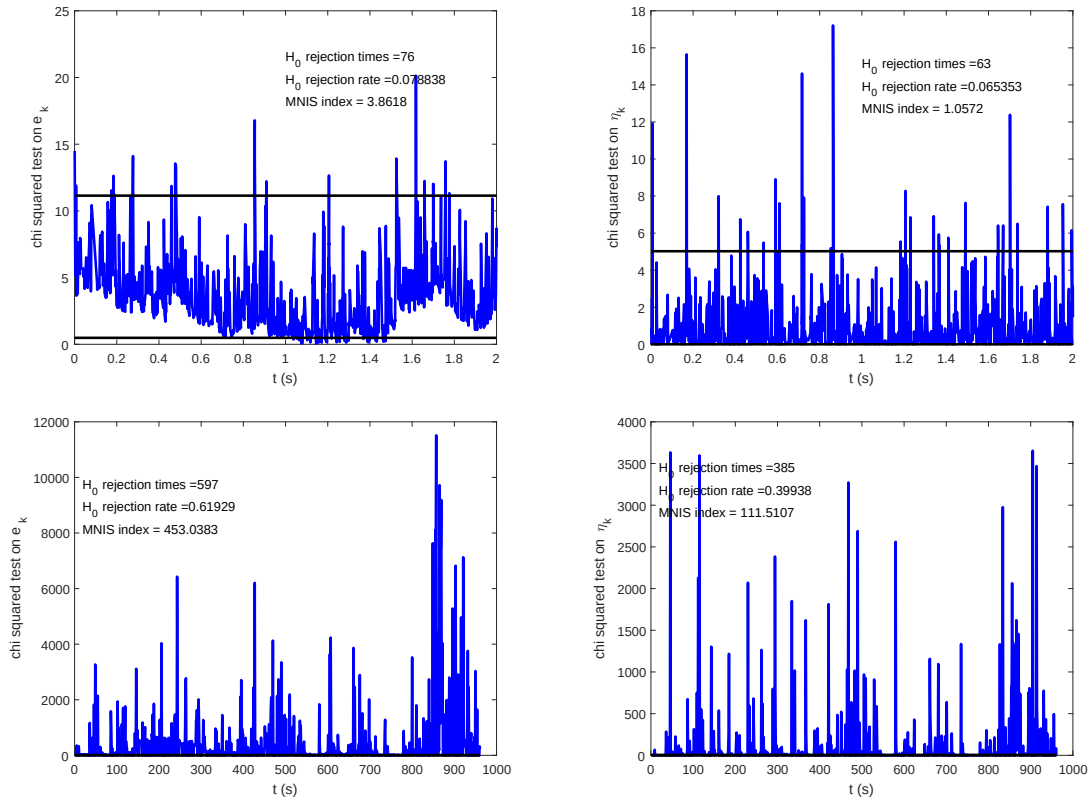


Figure 5.6: First group of subfigures.

5.1.2 Measurement Signal-to-Noise Ratio Variation

Changing input and measurement noise jointly

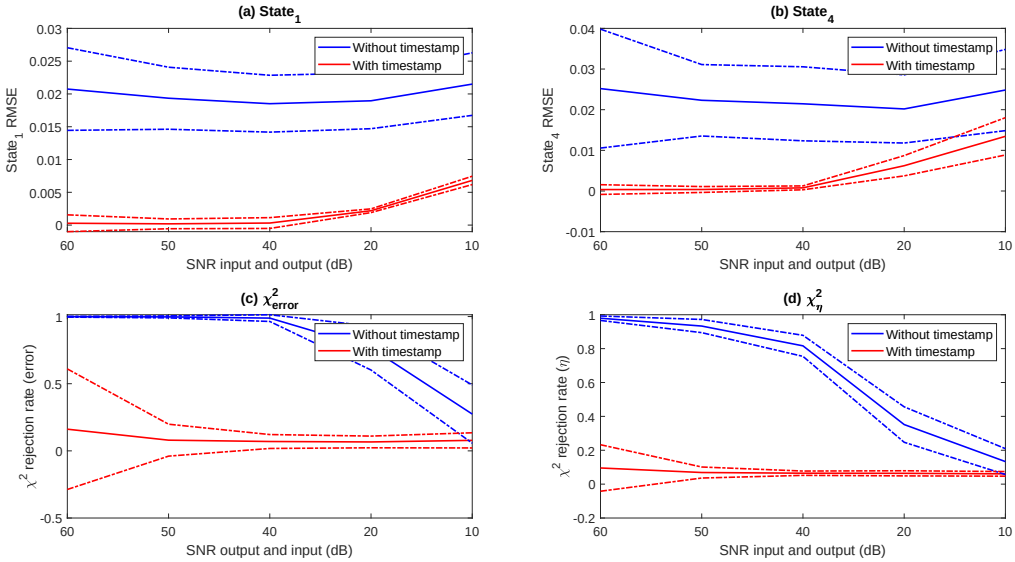


Figure 5.7: Performance, as a function of input and observation SNR.

Changing only measurement noise

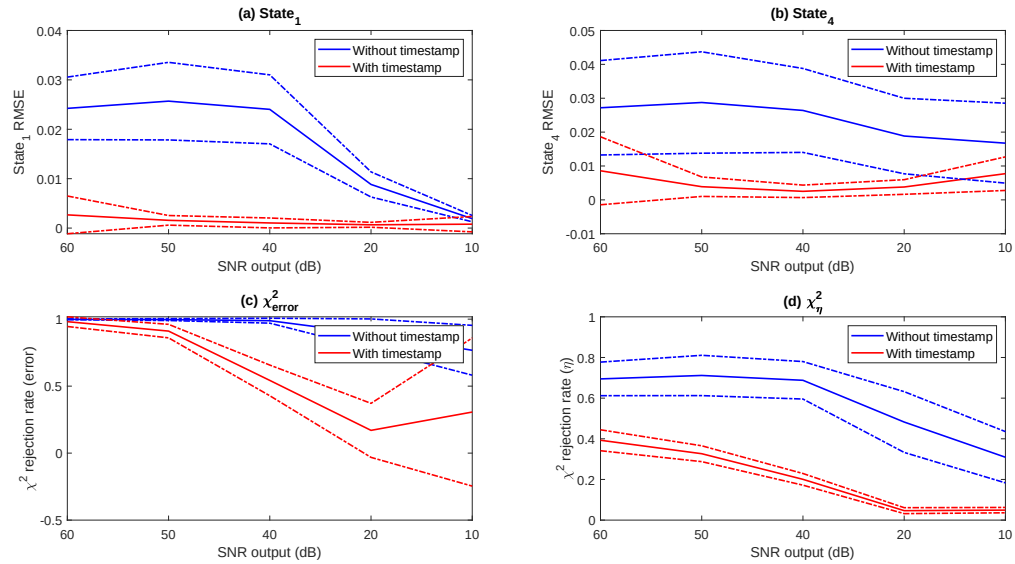


Figure 5.8: Performance, as a function of observation SNR.

5.1.3 Average Sampling Rate Variation

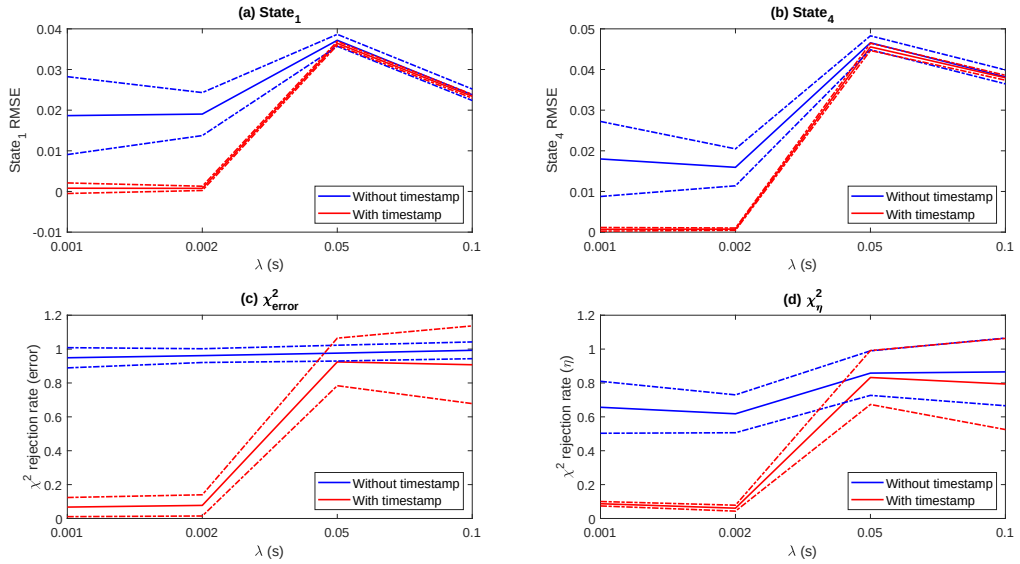


Figure 5.9: Performance, as a function of λ.

5.1.4 Regular and Average Irregular Time Interval Relation Variation

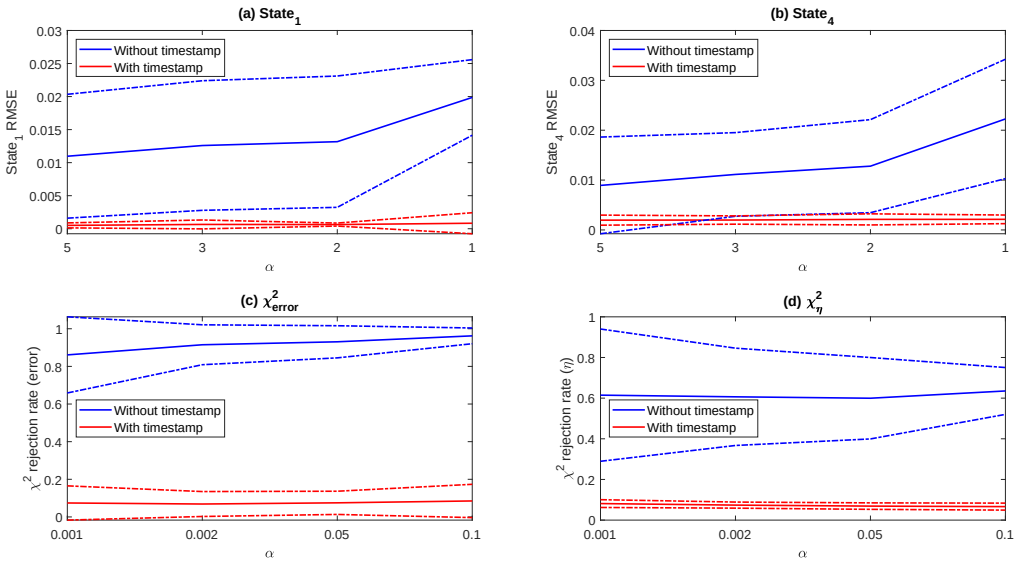


Figure 5.10: Performance, as a function of α.

5.2 Unicycle Position Estimation

Descrever as motivações por trás do estudo da amostragem irregular para problemas de rastreamento de robôs. Exemplos: medições feitas por câmeras não sincronizadas e utilizando redes de internet (com time delay).

5.2.1 System Description

Consider a nonholomonic moving robot, with the cinematic process model given by

$$\begin{aligned} \dot{p}_{\mathrm{x}} &= v\cos(\theta), \\ \dot{p}_{\mathrm{y}} &= v\sin(\theta), \\ \dot{\theta} &= u_1(t), \\ \dot{v} &= u_2(t), \end{aligned} \tag{5.6}$$

where p_{x} and p_{y} are the position coordinates, θ the angular orientation, v the linear velocity and inputs u_1 and u_2 are the angular velocity ω and the linear acceleration a, respectively. Figure 5.11 shows a schematic of the robot and its states,

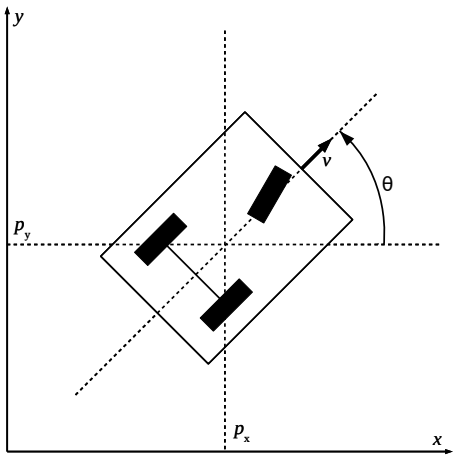


Figure 5.11: Nonholomonic robot system representation. The system states p_{x}, p_{y}, v and θ are highlighted.

The system described by 5.6 is discretized by a 4^{th} order Runge-Kutta method and the state vector x_i is given by $x_i \triangleq [p_{\mathrm{x},i}\ p_{\mathrm{y},i}\ \theta_i\ v_i]^T$.

The observation model $y(t_k) \in \mathbb{R}^2$

$$y(t_k) = \begin{bmatrix} p_x(t_k) \\ p_y(t_k) \end{bmatrix} + v(t_k), \tag{5.7}$$

is given by the position coordinates and $v(t_k) \sim \mathcal{N}(0, R_{t_k})$ is the observation noise, with zero mean and covariance R_{t_k}. When time-stamp is not available, the observation vector is approximated by $\tilde{y}_i \approx y(t_k)$, where i is the index of the next time instant, multiple of T.

Input vector $u_i = [\omega_i \, a_i]^T$ is measured by a girometer and accelerometer, respectively. We assume that

$$u_i = \tilde{u}_i - w_i, \tag{5.8}$$

where $\tilde{u}$ is the sensors measured valueand $w \sim \mathcal{N}(0, Q_i)$ represents the corresponded noise, of zero mean and covariance Q_i.

We simulate 60 seconds of robot movement, considering a step size of $\delta t_{\text{sim}} = 10^{-4}$. Irregular sampling time intervals h_k are simulated by the exponential probability distribution function from MatLab™ and approximated to the nearest discrete time instant, from the 600,000 available samples. Input signals were generated arbitrarily according to Figure 5.12. Figure 5.13 shows robot trajectory on xy-plane for the given input signals, leaving from point (0,0), as well as a realization of noisy and aperiodic measurements with signal-to-noise ratio of $\text{SNR}_y = 30$ dB and $\lambda = 0.3$ s as red dots and the hatched blue line represents UKF estimation, considering time-stamp, $\alpha = 5$ and $\text{SNR}_u = 10$ dB.

To analyze the effect of considering time-stamp in the estimation algorithm, we use the performance index J

$$J = \frac{\sum_{i=1}^{N} \sqrt{(\hat{p}_{x,i} - p_{x,i})^2 + (\hat{p}_{y,i} - p_{y,i})^2}}{N} \tag{5.9}$$

where $\hat{p}_{x,i}$ and $\hat{p}_{y,i}$ are filter position estimates produced at a regular interval T, $\hat{p}_{x,i}$ and $\hat{p}_{y,i}$ the true coordinates of the robot at the same time instants and N the total number of estimates. This index represents the average estimator position error in xy-plane.

Figure 5.14 shows a timespan from 0 to 1.3 seconds of a J realization, considering $\lambda = 0.5$, $\alpha = 5$, $\text{SNR}_y = 60dB$ dB and $\text{SNR}_u = 20$ dB, for the UKF considering and not considering time-stamp. Black dots represent the regular time instants kT whereas the

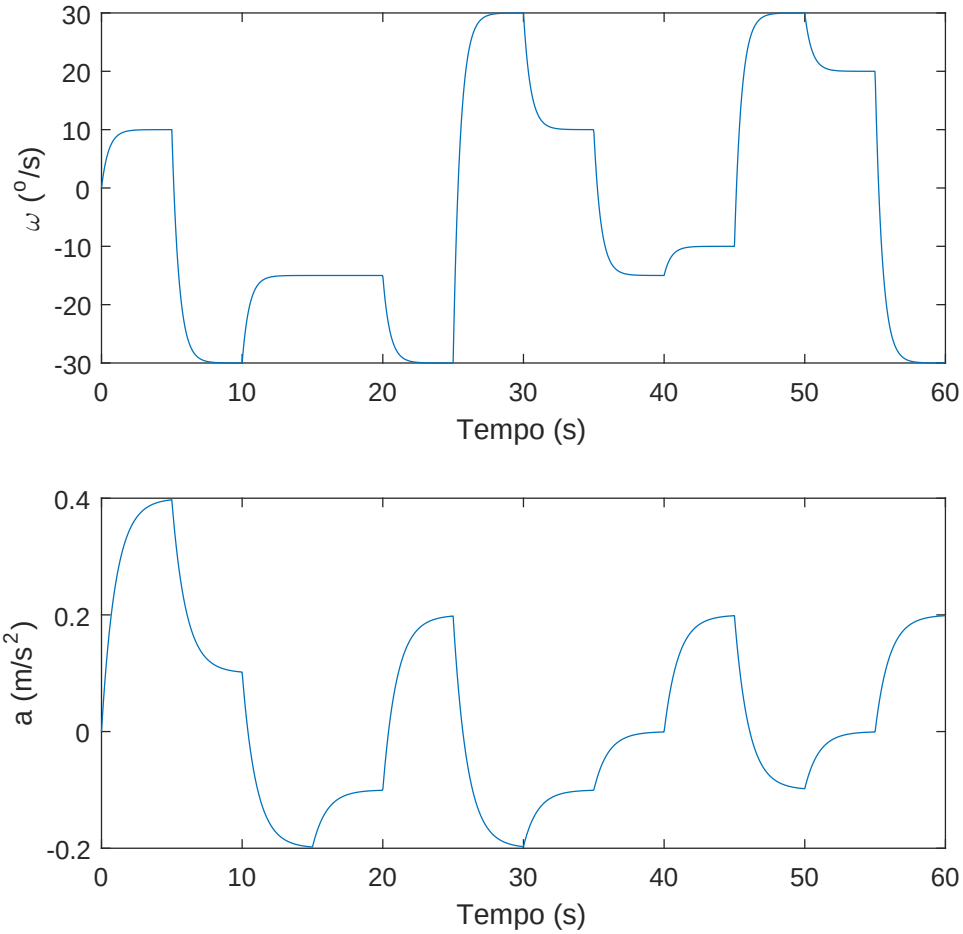


Figure 5.12: Simulation input signals. (a) shows temporal sequence of angular velocity ω and (b) shows linear acceleration a.

asterisks on x-axis match the exact measurement instants t_k. As expected, before the first data assimilation steps, both performances are identical. Since the first measurement t_1 was taken almost at the same time as regular estimation time instant $4T$, both estimation performances maintain close to each other. That is because the approximation error due to $\tilde{y}_i \approx y(t_k)$ is irrelevant. At t_2 we can observe significant deviation from index J in benefit of the algorithm considering time-stamp, since it performs data assimilation at the exact time measurement was taken. The same effect can be noted at t_4, after 1 second of simulation, when there is a significant different between time the measurement was taken and the next regular estimation instant.

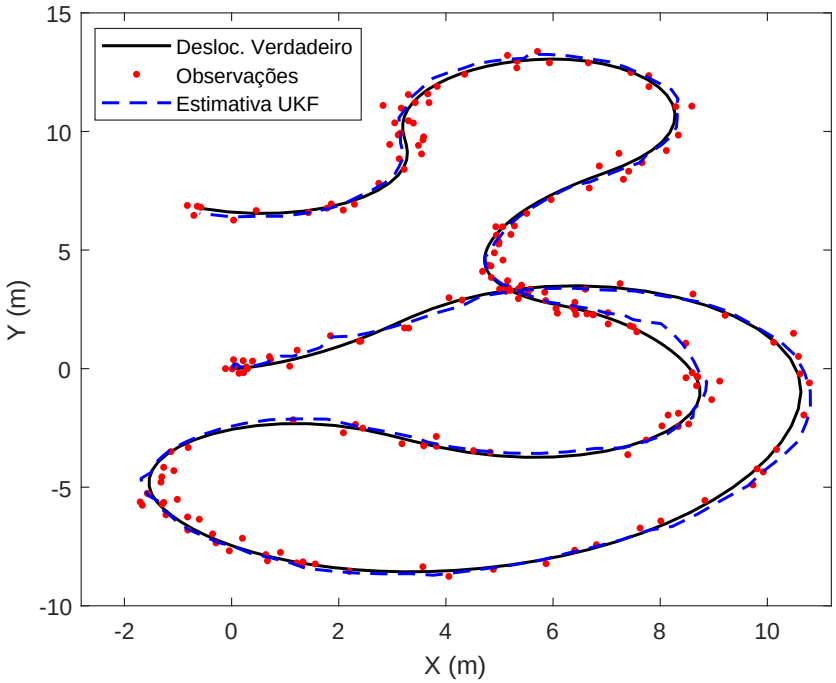


Figure 5.13: True position, noisy measurements and UKF estimates considering time-stamp, for a measurement noise of $\text{SNR}_y = 30$ dB, $\lambda = 0.3$ s and $\alpha = 5$.

5.2.2 Measurement Signal-to-Noise Ratio Variation

In this subsection we compare the estimation performance impact of considering time-stamp in the algorithm, by varying the measurement noise level. The signal-to-noise ratio for the input sensors and the measurement average time interval are held constant, $\text{SNR}_u = 20$ dB and $\lambda = 0.1$ s.

We considered an observation signal-to-noise variation of $\text{SNR}_y = \infty$, 80, 60, 40, 20, 10 dB. That is, initially the system was simulated considering no measurement noise and then it was gradually increased. For each noise scenario, we ran 100 realizations of the random variables for each algorithm and Figure 5.15 shows the results. Solid lines represent the average values for J and the dashed lines are the 95% confidence interval.

É possível observar que o estimador que considera o carimbo de tempo possui desempenho estatisticamente superior apenas para baixos níveis de ruído da observação. Quando SNR_y é igual a ∞ e 80 dB, o índice de desempenho J do filtro com carimbo de tempo é aproximadamente 1.25 e 1.26 cm, respectivamente, mais preciso do que sem carimbo de tempo e com variância pequena. Quando a relação sinal ruído se aproxima de 40 dB, no entanto, não é possível

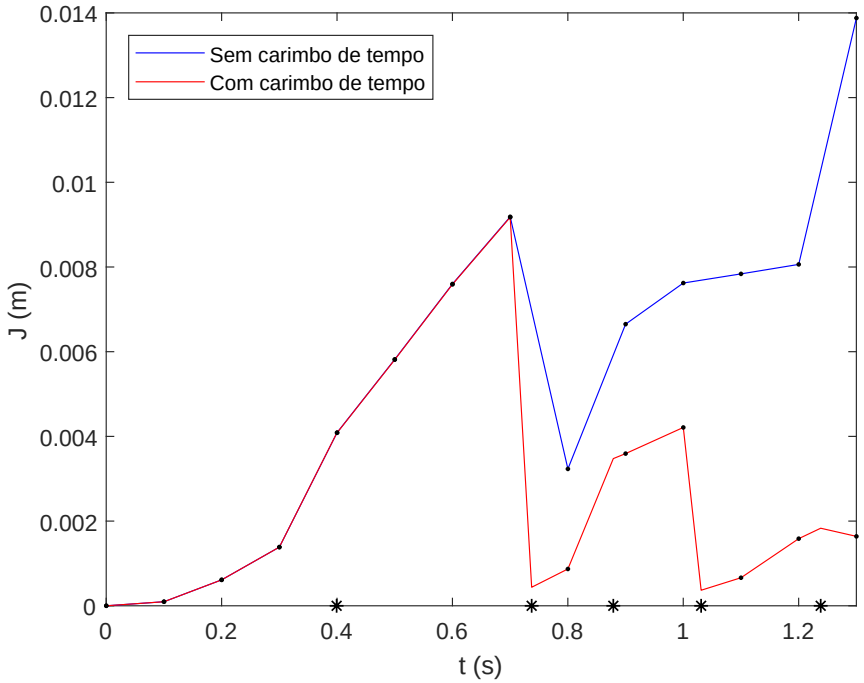


Figure 5.14: Temporal cut from 0 to 1.3 seconds, for a realization of J of both estimators, with and without time-stamp. Asterisks on x axis match the measurement sampling instants t_k. Black dots represent the regular estimation instants, same as input regular sampling instants.

distinguir estatisticamente o efeito de se considerar ou não carimbo de tempo.

5.2.3 Average Sampling Rate Variation

We now consider the variation of the average time interval that observations are taken, according to $\lambda = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6$ s, maintaining the other parameters constant, namely the noise level of inputs and observations, $\text{SNR}_\text{u} = 20$ dB and $\text{SNR}_\text{y} = 40$ dB, respectively. We carried out 100 realizations for each λ and for each algorithm and Figure 5.16 presents the results. Again, the solid lines are the average values of J and the 95% confidence interval is between the dashed lines.

Os resultados obtidos demonstram que a diferença no desempenho dos filtros é mais significativa para intervalos de tempo mais espaçados, mantendo-se a dinâmica do sistema e os outros parâmetros fixos. Inicialmente, para valores pequenos de λ , não há diferença estatística entre o desempenho dos estimadores, mas a medida que o intervalo cresce, ela aparece, chegando a aproximadamente 3.1 cm para um $\lambda = 0.6$ s. Uma interpretação possível é que, se o intervalo

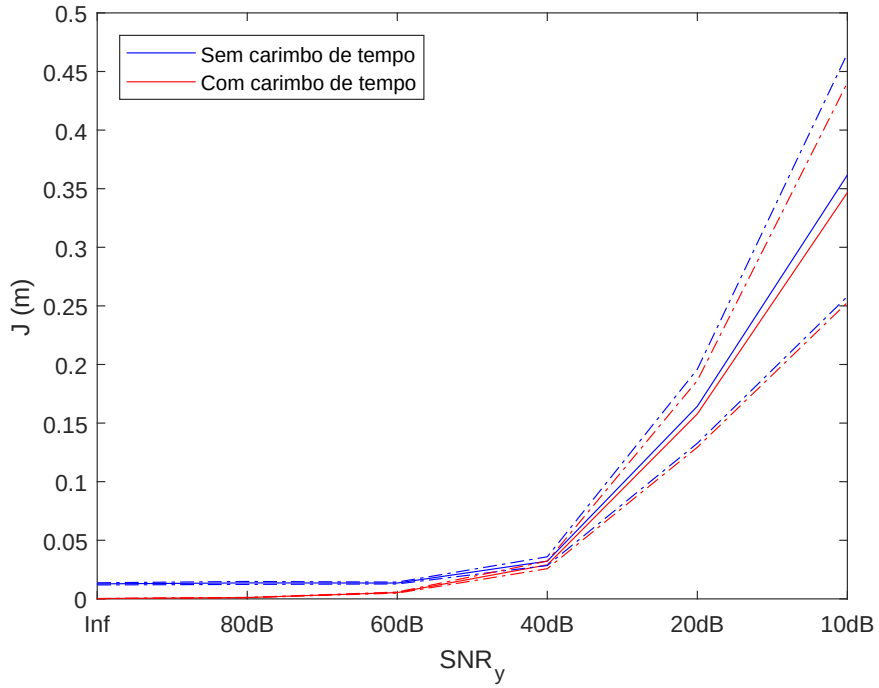


Figure 5.15: Performance index J variation, as a function of measurement noise for both UKF algorithms considering and not considering time-stamp

de tempo médio das observações for muito pequeno em comparação com a velocidade em que a dinâmica do sistema varia, o erro de aproximação do instante de amostragem das observações t_k é reduzido. O ruído do sensor pode acabar sendo mais relevante do que o erro devido ao tempo incorreto de assimilação.

5.2.4 Regular and Average Irregular Time Interval Relation Variation

We also analyze the performance impact of varying the parameter α, which is the relation between regular input time interval T and average measurements time interval λ. The simulated values were $\alpha = 10, 5, 2, 1$. That means we start with higher frequency values for input sampling in comparison to measurement sampling and this values is gradually decreased. Other parameters were maintained constant $\text{SNR}_\text{u} = 20$ dB and $\text{SNR}_\text{y} = 40$ dB.

The same way we did for the other scenarios, 100 realizations were simulated and Figure 5.17 presents the results. Solid lines are the average values for J and dashed lines represent the 95% confidence interval.

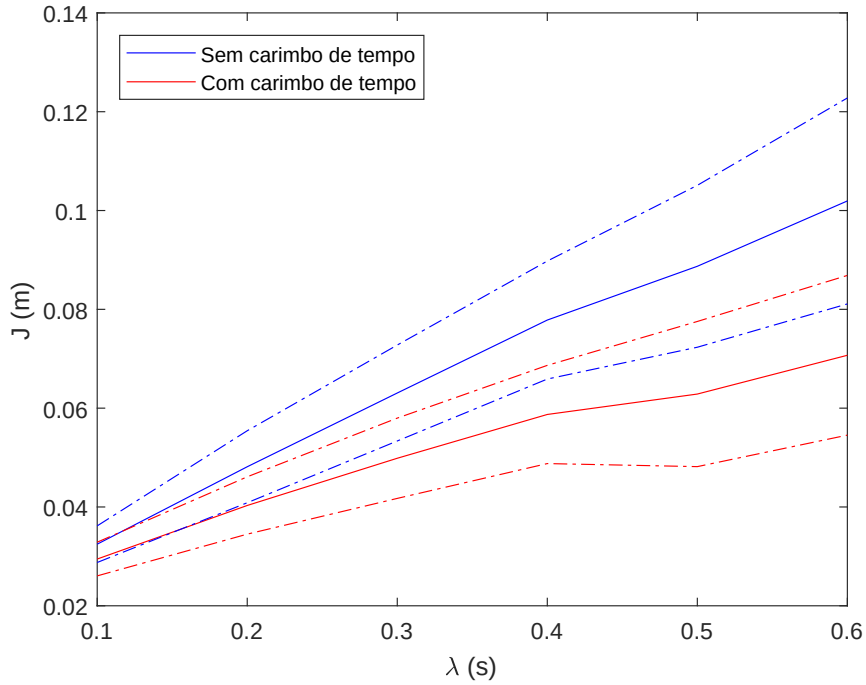


Figure 5.16: Performance index J variation, as a function of measurement noise both UKF algorithms considering and not considering time-stamp

Nota-se que, quando o carimbo de tempo é considerado, não há diferença significativa em se variar o α no desempenho do filtro, com o índice de desempenho J se mantendo pouco abaixo dos 3 cm. Ou seja, não importa a relação entre as frequências de amostragem da observação e das entradas. Por outro lado, quando não se considera o carimbo de tempo, essa relação se torna relevante para o índice de desempenho do estimador. Quanto mais lento a frequência da entrada em comparação com a frequência da saída, maior o erro obtido. Para o caso mais extremo utilizado, $\alpha = 1$, a diferença no índice J foi mais do que o dobro. Esse resultado era esperado, uma vez que quanto maior o valor de α, mais rápida é a taxa de discretização do modelo de processo em relação à frequência dos sensores de observação. Consequentemente, o erro obtido na aproximação $\tilde{y}(i) \approx y(t_k)$ diminui.

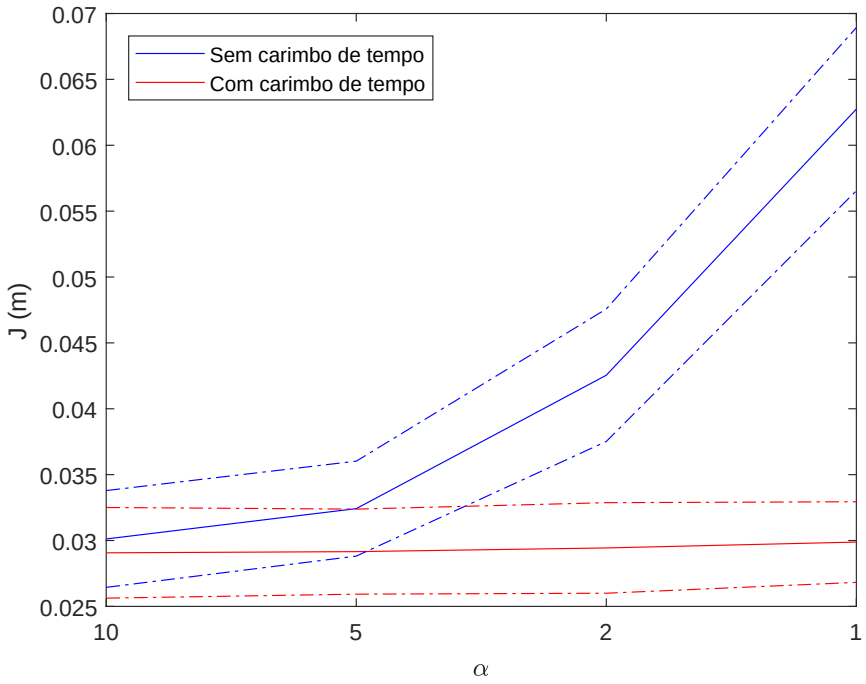


Figure 5.17: Performance index J variation, as a function of λ for both UKF algorithms considering and not considering time-stamp

Chapter 6

Conclusions

6.1 Project Overview

Investigou-se isso e aquilo... (caso algum leitor pule direto para a conclusão)

6.2 Main Results

Como os resultados corroboram os objetivos perseguidos. O que é relevante dizer sobre cada objetivo descrito no capítulo 1, com o que foi feito no trabalho.

Autocrítica sobre o trabalho: alguma hipótese foi pouco realista? o que não foi feito tão bem? o que foi feito bem?

6.3 Future Work

Talvez uma lista itemizada Trabalhos futuros podem vir diretamente da autocrítica.

Bibliography

Albertos, P., Peñarrocha, I., and Garcia, P. (2004). Virtual sensors under delayed scarce measurements. *7th IFAC Symposium on Cost Oriented Automation*, 37(5):85–90.

Andler, S. F. and Brohede, M. (2009). Information Fusion from Databases, Sensors and Simulations: Annual Report 2008. *Skövde: Institutionen för kommunikation och information*, page 65.

Anxi, Y., Diannong, L., Weidong, H., and Zhen, D. (2005). A unified out-of-sequence measurements filter. In *IEEE National Radar Conference - Proceedings*, pages 453–458.

Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T. (2002). A Tutorial on Particle Filters for Online Nonlinear / Non-Gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188.

Bar-Shalom, Y. (2000). Update with out-of-sequence measurements in tracking: Exact solution. *Proceedings of SPIE*, 4048(3):769–778.

Bar-shalom, Y., Daum, F., and Huang, J. I. M. (2009). The Probabilistic Data Association Filter. *IEEE Control Systems Magazine*, 29(6):82–100.

Bar-Shalom, Y., Rong Li, X., and Kirubarajan, T. (2001). *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. John Wiley & Sons, Inc.

Bergman, N. (1999). *Recursive Bayesian Estimation - Navigation and Tracking Applicationss*. PhD thesis, Linköping University.

Boström, H., Andler, S. F., Brohede, M., Johansson, R., Karlsson, E., Laere, J. V., Niklasson, L., Nilsson, M., Persson, A., and Ziemke, T. (2007). On the definition of information fusion as a field of research. Technical report, University of Skösvde.

Brahmi, M., Schueler, K., Bouzouraa, S., Maurer, M., Siedersberger, K. H., and Hofmann, U. (2013). Timestamping and latency analysis for multi-sensor perception systems. *Proceedings of IEEE Sensors*.

Castanedo, F. (2013). A review of data fusion techniques. *The Scientific World Journal*, 2013.

Challa, S., Evans, R. J., and Wang, X. (2003). A Bayesian solution and its approximations to out-of-sequence measurement problems. *Information Fusion*, 4:185–199.

Chen, B., Yu, L., Zhang, W.-A., and Liu, A. (2013). Robust Information Fusion Estimator for Multiple Delay-Tolerant Sensors With Different Failure Rates. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 60(2):401–414.

Dasarathy, B. (2001). Information Fusion - what, where, why, when, and how? *Information Fusion*, 2(2):75–76.

Dasarathy, B. V. (1997). Sensor fusion potential exploitation-innovative architectures and illustrative applications. *Proceedings of the IEEE*, 85(1):24–38.

Dasarathy, B. V. (2000). More the merrier...or is it? - Sensor suite augmentation benefits assessment. *Proceedings of the 3rd International Conference on Information Fusion, FUSION 2000*, 2:20–25.

Dubois, D. and Prade, H. (2000). Possibility theory in information fusion. *Proceedings of the 3rd International Conference on Information Fusion, FUSION 2000*, 1.

Durrant-Whyte, H. F. (1988). Sensor Models and Multisensor Integration. *The International Journal of Robotics Research*, 7(6):97–113.

Elmenreich, W. (2002). An Introduction to Sensor Fusion. Research report, Vienna University of Technology.

Elson, J., Girod, L., and Estrin, D. (2002). Fine-grained network time synchronization using reference broadcasts. *Proceedings of the 5th Symposium on Operating Systems Design and Implementation (OSDI)*, 36(SI):147.

Eng, F. and Gustafsson, F. (2005). SYSTEM IDENTIFICATION USING MEASUREMENTS SUBJECT TO STOCHASTIC TIME JITTER. *IFAC Proceedings Volumes*, 38:1179–1184.

Faragher, R. (2012). Understanding the basis of the kalman filter via a simple and intuitive derivation [lecture notes]. *IEEE Signal Processing Magazine*, 29(5):128–132.

Fatehi, A. and Huang, B. (2017). Kalman filtering approach to multi-rate information fusion in the presence of irregular sampling rate and variable measurement delay. *Journal of Process Control*, 53:15–25.

Foster, J. L. and Hall, D. K. (1981). Muitisensor Analysis of Hydrologic Features with Emphasis on the. *Photogrammetric Engineering and Remote Sensing*, 47(5):655–664.

Fowler, C. A. (1979). Comments on the Cost and Performance of Military Systems. *IEEE Transactions on Aerospace and Electronic Systems*, AES-15(1):2–10.

Fridman, E. (2014). Introduction to time-delay and sampled-data systems. In *2014 European Control Conference (ECC)*, pages 1428–1433. IEEE.

Ganeriwal, S., Kumar, R., and Srivastava, M. B. (2003). Timing-sync protocol for sensor networks. In *Proceedings of the first international conference on Embedded networked sensor systems - SenSys '03*, page 138.

Goodman, I. R., Mahler, R. P., and Nguyen, H. T. (1997). *Mathematics of Data Fusion*. Kluwer Academic Publishers.

Gopalakrishnan, A., Kaisare, N. S., and Narasimhan, S. (2010). Incorporating delayed and infrequent measurements in Extended Kalman Filter based nonlinear state estimation. *Journal of Process Control*, 21:119–129.

Gravina, R., Alinia, P., Ghasemzadeh, H., and Fortino, G. (2017). Multi-sensor fusion in body sensor networks: State-of-the-art and research challenges. *Information Fusion*, 35:68–80.

Hadidi, M. and Schwartz, S. (1979). Linear recursive state estimators under uncertain observations. *IEEE Transactions on Automatic Control*, 24(6):944–948.

Hall, D. L. and Llinas, J. (1997). An introduction to multisensor data fusion. *Proceedings of the IEEE*, 85(1):6–23.

Han, C. and Zhang, H. (2009). Linear optimal filtering for discrete-time systems with random jump delays. *Signal Processing*, 89(6):1121–1128.

He, L. M. (2008). Time synchronization based on spanning tree for wireless sensor networks. *2008 International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM 2008*, pages 4–7.

Hu, R. J., Wang, Z., Alsaadi, F. E., and Hayat, T. (2017). Event-based filtering for time-varying nonlinear systems subject to multiple missing measurements with uncertain missing probabilities. *Information Fusion*, 38:74–83.

Huck, T., Westenberger, A., Fritzsche, M., Schwarz, T., and Dietmayer, K. (2011). Precise timestamping and temporal synchronization in multi-sensor fusion. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 242–247. IEEE.

Hyberts, S. G., Robson, S. A., and Wagner, G. (2013). Exploring signal-to-noise ratio and sensitivity in non-uniformly sampled multi-dimensional NMR spectra. *Journal of Biomolecular NMR*, pages 167–178.

Jaffer, A. G. and Gupta, S. C. (1971). Recursive Bayesian Estimation With Uncertain Observation. *IEEE Transactions on Information Theory*, I(September):614–616.

Jing, Z., Pan, H., and Qin, Y. (2013). Current progress of information fusion in China. *Chinese Science Bulletin*, 58(36):4533–4540.

Jordao, A., Torres, L. A. B., and Schwartz, W. R. (2018). Novel approaches to human activity recognition based on accelerometer data. *Signal, Image and Video Processing*, 12(7):1387–1394.

Julier, S. J. and Uhlmann, J. K. (2004). Unscented Filtering and Nonlinear Estimation. *Proceedings of the IEEE*, 92(3):401–422.

Julier, S. J. and Uhlmann, J. K. (2005). Fusion of time delayed measurements with uncertain time delays. In *Proceedings of the 2005 American Control Conference*, pages 4028–4033.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of ASME – Journal of Basic Engineering*, 82:35–45.

Kanchanaharuthai, A. and Wongsaisuwan, M. (2002). Stochastic H/sub 2/-optimal controller design for sampled-data systems with random sampled measurement. In

Proceedings of the 41st SICE Annual Conference. SICE 2002., volume 3, pages 2042–2047. Soc. Instrument & Control Eng. (SICE).

Kaur, P. and Abhilasha (2015). Time synchronization in wireless sensor networks: a review. *International Journal of Computer Engineering and Applications*, IX(VI):403–414.

Khaleghi, B., Khamis, A., Karray, F. O., and Razavi, S. N. (2013). Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, 14(1):28–44.

Kolmogorov, A. N. (1962). *Interpolation and extrapolation of stationary random sequences*. Translated by W. L. Doyle, and Ivan Selin, RAND Corporation.

Kordestani, M. and Saif, M. (2017). Data Fusion for Fault Diagnosis in Smart Grid Power Systems. In *IEEE 30th Canadian Conference on Electrical and Computer Engineering*, pages 516–521.

Kunoh, F. M. (2015). *Procedimento para Detecção de Falseamento via Amostragem Não Uniforme*. PhD thesis, Universidade Federal de Minas Gerais.

Kwok, C., Fox, D., and A, M. M. (2004). Real-Time Particle Filters. *Proceedings of the IEEE*, 92(3):469 – 484.

Lin, H. and Sun, S. (2016). Distributed fusion estimator for multi-sensor asynchronous sampling systems with missing measurements. *IET Signal Processing*, 10(7):724–731.

Liu, Q., Wang, Z., He, X., and Zhou, D. H. (2014). A survey of event-based strategies on control and estimation. *Systems Science and Control Engineering*, 2(1):90–97.

Liu, Z. and Wang, J. (2012). Multi-scale data fusion for smart grids. In *Proceedings of the 2012 7th IEEE Conference on Industrial Electronics and Applications, ICIEA 2012*, pages 2059–2062.

Lu, X., Zhang, H., Wang, W., and Teo, K. L. (2005). Kalman filtering for multiple time-delay systems. *Automatica*, 41(8):1455–1461.

Luo, R. C. and Kay, M. G. (1989). Multisensor integration and fusion in intelligent systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(5):901–931.

Ma, J. and Sun, S. (2011). Optimal Linear Estimators for Systems With Random Sensor Delays , Multiple Packet Dropouts and Uncertain Observations. *IEE TRANSACTIONS ON SIGNAL PROCESSING*, 59(11):5181–5192.

Maehlisch, M., Schweiger, R., Ritter, W., and Dietmayer, K. (2006). Multisensor vehicle tracking with the probability hypothesis density filter. In *Proceedings of the International Conference on Information Fusion*, pages 1–8.

Mahler, R. P. (2004). "Statistics 101" for multisensor, multitarget data fusion. *IEEE Aerospace and Electronic Systems Magazine*, 19(1 II):53–64.

Mamdani, E. H. and Assilian, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1–13.

Maróti, M., Kusy, B., Simon, G., and Lédeczi, Á. (2004). The Flooding Time Synchronization Protocol. In *Proceedings of International Conference on Embedded Networked Sensor Systems (SenSys)*, pages 39–49.

Marvasti, F. (2001). *Nonuniform Sampling Theory and Practice*. Springer US, 1 edition.

Micheli, M. and Jordan, M. I. (2002). Random sampling of a continuous-time stochastic dynamical system. In *in Proc. of the 15th International Symposium on the Mathematical Theory of Networks and Systems (MTNS 2002), (University of Notre Dame, South)*.

Middleton, D. and Esposito, R. (1968). Simultaneous Optimum Detection and Estimation of Signals in Noise. *IEEE Transactions on Information Theory*, 14(3):434–444.

Mills, D. L. (1991). Internet Time Synchronization: The Network Time Protocol. *IEEE Transactions on Communications*, 39(10):1482–1493.

Mingxia Xu, Minjian Zhao, and Shiju Li (2005). Lightweight and energy efficient time synchronization for sensor network. *Proceedings. 2005 International Conference on Wireless Communications, Networking and Mobile Computing, 2005.*, 2:947–950.

Miskowicz, M. (2006). Send-on-delta concept: An event-based data reporting strategy. *Sensors*, 6(1):49–63.

Moayedi, M., Foo, Y. K., and Soh, Y. C. (2011). Filtering for networked control systems with single/multiple measurement packets subject to multiple-step measurement delays and multiple packet dropouts. *International Journal of Systems Science*, 42(3):335–348.

Mohinder, S. G. and Angus, P. A. (2010). Applications of Kalman Filtering in Aerospace 1960 to the Present. *IEEE Control Systems Magazine*, pages 69–78.

Molchanov, I. and Molinari, F. (2014). Applications of Random Set Theory in Econometrics. *Annual Review of Economics*, 6:229–251.

Nahi, N. (1969). Optimal recursive estimation with uncertain observation. *IEEE Transactions on Information Theory*, 15(4):457–462.

Nahin, P. J. and Pokoski, J. L. (1980). NCTR Plus Sensor Fusion Equals IFFN or Can Two Plus Two Equal Five? *IEEE Transactions on Aerospace and Electronic Systems*, AES-16(3):320–337.

Nuñez-Garcia, J. and Wolkenhauer, O. (2002). Random set system identification. *IEEE Transactions on Fuzzy Systems*, 10(3):287–296.

Papoulis, A. (1984). *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, Inc.

Pawlak, Z. (1991). *Rough Sets: Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, 1991.

Pawlak, Z. and Skowron, A. (2007). Rudiments of rough sets. *Information Sciences*, 177(1):3–27.

Peñarrocha, I., Sanchis, R., and Romero, J. A. (2012). State estimator for multisensor systems with irregular sampling and time-varying delays. *International Journal of Systems Science*, 43(8):1441–1453.

Phillips, C. L. and Nagle, H. T. (1995). *Digital control system analysis and design*. Prentice-Hall, Inc.

Ping, S. (2003). Delay Measurement Time Synchronization for Wireless Sensor Networks. *Intel Research Berkeley Lab*, pages 1–10.

Rao, N. (1998). To fuse or not to fuse: Fuser versus best classifier. *Proceedings of SPIE - The International Society for Optical Engineering*, 3376:25–34.

Richard, J.-P. (2003). Time-delay systems: an overview of some recent advances and open problems. *Automatica*, 39(10):1667–1694.

Richardson, J. M. and Marsh, K. A. (1988). Fusion of Multisensor Data. *The International Journal of Robotics Research*, 7(6):78–96.

Römer, K. (2003). Temporal Message Ordering in Wireless Sensor Networks. In *IFIP Mediterranean Workshop on Ad-Hoc Networks*, pages 131–142.

Safari, S., Shabani, F., and Simon, D. (2014). Multirate multisensor data fusion for linear systems using Kalman filters and a neural network. *Aerospace Science and Technology*, 39:465–471.

Sahebsara, M., Chen, T., and Shah, S. L. (2007). Optimal filtering with random sensor delay, multiple packet dropout and uncertain observations. *International Journal of Control*, 80(2):292–301.

Särkkä, S. (2013). *Bayesian Filtering and Smoothing*. Cambridge University Press.

Schenato, L., Sinopoli, B., Franceschetti, M., Poolla, K., and Sastry, S. S. (2007). Foundations of Control and Estimation Over Lossy Networks. *Proceedings of the IEEE*, 95(1):163–187.

Shafer, G. (1976). *A Mathematical Theory of Evidence*. Princeton University Press.

Shen, B., Wang, Z., and Huang, T. (2016). Stabilization for sampled-data systems under noisy sampling interval. *Automatica*, 63:162–166.

Shuli Sun, Lihua Xie, Wendong Xiao, and Nan Xiao (2008). Optimal Filtering for Systems With Multiple Packet Dropouts. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 55(7):695–699.

Sichitiu, M. L. and Veerarittiphan, C. (2003). Simple, accurate time synchronization for wireless sensor networks. *IEEE Wireless Communications and Networking Conference, WCNC*, 2(C):1266–1273.

Singer, R. A. and Kanyuck, A. J. (1971). Computer control of multiple site track correlation. *Automatica*, 7(4):455–463.

Sinopoli, B., Schenato, L., Franceschetti, M., Poolla, K., Jordan, M. I., and Sastry, S. S. (2004). Kalman filtering with intermittent observations. *IEEE Transactions on Automatic Control*, 49(9):1453–1464.

Sivrikaya, F. and Yener, B. (2004). Time Synchronization in Sensor Networks: A Survey. *IEEE Network*, 18(4):45 – 50.

Sorenson, H. W. (1970). Least-squares estimation: from Gauss to Kalman. *IEEE Spectrum*, 7:63–68.

Srivastava, R. P. (2011). An introduction to evidential reasoning for decision making under uncertainty: Bayesian and belief function perspectives. *International Journal of Accounting Information Systems*, 12(2):126–135.

Stone, J. V. (2013). *Bayes' Rule: A Tutorial Introduction to Bayesian Analysis*. Sebtel Press.

Su, W. and Akyildiz, I. F. (2005). Time-diffusion synchronization protocols in wireless sensor networks. *IEEE/ACM TRANSACTIONS ON NETWORKING*, 13(2):384–397.

Tabuada, P. (2007). Event-Triggered Real-Time Scheduling of Stabilizing Control Tasks. *IEEE Transactions on Automatic Control*, 52(9):1680–1685.

Teixeira, B. O. S. (2008). *Constrained state estimation for linear and nonlinear dynamic systems*. PhD thesis, Universidade Federal de Minas Gerais.

Theil, A., Kester, L. J., and Bossé, É. (2000). On measures of performance to assess sensor fusion effectiveness. *Proceedings of the 3rd International Conference on Information Fusion, FUSION 2000*, 2:3–7.

van Greunen, J. and Rabaey, J. (2003). Lightweight Time Synchronization for Sensor Networks. In *Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications(WSNA)*, pages 11–19.

Wang, Z., Ho, D. W., Liu, Y., and Liu, X. (2009). Robust H_∞ control for a class of nonlinear discrete time-delay stochastic systems with missing measurements. *Automatica*, 45(3):684–691.

Westenberger, A., Waldele, S., Dora, B., Duraisamy, B., Muntzinger, M., and Dietmayer, K. (2013). Multi-sensor fusion with out-of-sequence measurements for vehicle environment perception. In *Proceedings - IEEE International Conference on Robotics and Automation*.

Wiener, N. (1949). *The Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. Wiley.

Willner, D., Chang, C. B., and Dunn, K. P. (1976). Kalman filter algorithms for a multi-sensor system. In *IEEE Conference on Decision and Control including the 15th Symposium on Adaptive Processes*, volume 15, pages 570–574.

Yan, B., Lev-Ari, H., and Stankovic, A. M. (2017). Networked State Estimation with Delayed and Irregularly-Spaced Time-Stamped Observations. *IEEE Transactions on Control of Network Systems*, 5870(c):1–1.

Zadeh, L. A. (1965). Fuzzy Sets. *Information and Control*, 8:338–353.

Zadeh, L. A. (1978). Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 1(1):3–28.

Zhu, C., Xia, Y., Xie, L., and Yan, L. (2013). Optimal linear estimation for systems with transmission delays and packet dropouts. *IET Signal Processing*, 7(9):814–823.

Zou, L., Wang, Z.-D., and Zhou, D.-H. (2017). Event-based Control and Filtering of Networked Systems: A Survey. *International Journal of Automation and Computing*, 14(3):239–253.