UCLouvain
Institute for Information
and Communication Technologies,
Electronics and Applied Mathematics

# Optimal Sampling for State Estimation of Stochastic Dynamical Systems

Antoine Aspeel

Thesis submitted in partial fulfillment
of the requirements for the degree of
*Doctor in Applied Sciences*

Dissertation committee:

Prof. Raphaël Jungers (UCLouvain, Advisor)
Prof. Benoit Macq (UCLouvain, Advisor)
Prof. Jean-Charles Delvenne (UCLouvain, Reader)
Prof. Vincent François-Lavet (VU Amsterdam, Reader)
Prof. Necmiye Özay (Univ. of Michigan, Ann Arbor, Reader)
Prof. David Bol (UCLouvain, Chair)

Version of April 14, 2022.

# Abstract

The problem of estimating a quantity that evolves with time from noisy measurements can be found in many fields, such as engineering (e.g., the position of a drone), epidemiology (e.g., the number of infected people), or medicine (e.g., the position of a lung tumor moving due to breathing). In some cases, the number of measurements must be reduced because the acquisition of measurements is expensive, energy-consuming, time-consuming or dangerous.

In this thesis, we assume that the amount of measurements is limited, and we study the problem of the optimal choice of measurement times to estimate the state of a stochastic dynamical system over a finite time horizon.

We first focus on the case of linear and Gaussian dynamical systems. We propose the optimal intermittent Kalman filter for which the measurement times are obtained by solving a combinatorial optimization problem. Then, we generalize this approach to non-linear and non-Gaussian dynamical systems. This leads to the formulation of three variants of optimal intermittent particle filters. The first one, for which the measurements are time-triggered, is based on combinatorial optimization. The other two are self-triggered. One requires a combinatorial program to be solved for each measurement, and the other requires to solve a stochastic program to choose when to perform a measurement. Robust estimation of the state of a linear dynamical system subject to bounded disturbances is also studied. Using the polytope containment method and Youla-parameterization, we show that the measurement times that minimize the worst-case estimation error are obtained by solving a mixed integer linear program.

Numerical and empirical experiments are presented, with a particular focus on the problem of tumor tracking and its use in radiation therapy.

# Acknowledgements

The exchange and criticism of ideas are at the foundation of scientific knowledge, and the people influencing a scientific publication always exceed the list of its authors. Therefore, I would like to thank all the people I met and with whom I exchanged and discussed during my PhD thesis.

First of all, I would like to sincerely thank my advisors, Benoit Macq and Raphaël Jungers, for their trust and support during my thesis. Their advice and feedback during the writing of this manuscript was invaluable. Benoit's optimism has always made the work stimulating. He manages to create fruitful connections within his team thanks to a warm atmosphere (among other things, via regular lunches at the Louvain House), and is always attentive to the well-being of everyone. I also thank him for the opportunities he offered me, among many others, to join him at McGill University when he was on sabbatical. Working with Raphaël is a guarantee of sound advice and insightful discussions in a relaxed and productive work environment. Between research retreats, barbecues, Tuesday group meetings and afterworks, team cohesion is guaranteed. I thank him for his availability, his precious advice and his constant encouragement. He has always pushed me to give the best of myself and is a great source of inspiration for my future.

I am honored and grateful to have on my thesis jury bright and prolific researchers who have inspired me throughout our collaborations: Prof. David Bol, Prof. Jean-Charles Delvenne, Prof. Vincent François-Lavet, and Prof. Necmiye Özay. My thanks to David for our stimulating discussions during the preparation of the courses on sustainable development. I also thank Jean-Charles who confirmed my passion for research during my master thesis, and who accepted to be part of my PhD thesis accompanying committee. My thanks also go to Vincent for the stimulating discussions on

reinforcement learning since our meeting at McGill University. His advice has enlightened me many times. My sincere thanks also go to Necmiye for her guidance during our collaboration on networked control systems. Her remarks were always on point. I hope to have the opportunity to continue working with them in the future.

I am also grateful to Dr. Rudi Labarbe who accepted to be a member of my PhD thesis accompanying committee alongside Jean-Charles. His expertise on proton therapy and his pedagogy will have been a great help. Our discussions had a decisive impact on the direction of this thesis.

I would like to thank Kwesi Rutledge for our collaboration on networked control systems. His insight and broad knowledge of the control theory literature was invaluable. I hope we will continue to work together in the future and have fond memories of our weekly discussions.

Thanks also to Amaury Gouverneur for our collaboration on intermittent particle filtering. Working with him is as pleasant as it is productive.

During my thesis, I had the chance to work with a lot of great people. Thanks to my colleagues Alexis, Alice, Anne-Sophie, Arnould, Benoit L., Boris, Damien, Dani, Eliott, Estelle, François W., Gaetan, Gauthier, Guillaume B., Guillaume O., Jaroslav, Jean, Jonathan, Julien, Karim, Lucas, Mahsa, Mariana, Matteo, Maxime, Merence, Michaël F., Nicolas, Paul, Sébastien L., Sébastien M., Simon C., Sylvain, Thiago, Umair, Valentin, Victor, Victorien, Virginie, Wei, and Zheming for these amazing and stimulating years. A special thanks to Alexis, Lucas, and Matteo who kindly accepted to proofread this thesis. Their comments undoubtedly improved the manuscript.

Many thanks also to the academic, technical and cleaning staffs of the ICTEAM institute for making this place a first-class research environment where it is so pleasant to work. In particular, I would like to thank Patricia for her unfailing daily support during these years. Her organizational skills, her rigor and her benevolence have been invaluable. I also thank Brigitte and François for their computer assistance, and Christine and Patrick from the cleaning staff for their great work and their kindness.

My deepest thanks go to my family -and especially to my mom-, for their love, and for having always believed in me. Also, I am infinitely grateful to my lover, Anne-Sophie, for her unfailing support. Her daily presence is simply wonderful. Finally, I thank my friends for the great times we had together. They always gave me the relaxation and fun I needed.

# Contents

# 1

# Introduction

DYNAMICAL systems have demonstrated their ability to model real-world phenomena [AL12, Ado88, FP00]. They arise whenever a quantity evolving in time is studied. For example, the evolution of the voting intentions of a population, or the movements of a drone can be seen as dynamical systems.

Some dynamical systems are described as deterministic, for example, the motion of planets in Newtonian physics. Other real systems will be modeled with a stochastic component, such as Brownian motion, or the trajectory of a drone subjected to wind. Even when a deterministic model is available, adding a stochastic component will often result in a more robust modeling because such a stochastic component can take into account the modeling error between the real system and the model that is made.

In these situations, it is interesting to track the evolution of the system state in real time, e.g., to know the voting intentions of a population or the position of a drone. To do this, we often have access to measurements of the system state, e.g., through surveys or thanks to GPS localization. However, these measurements are often indirect and noisy.

Estimating the state of a stochastic system from noisy measurements is a classical problem in dynamical systems theory. In 1960, Kalman proposed the eponymous filter, which solves this problem in the case of linear and Gaussian systems [Kal60]. The more general case of nonlinear and non-Gaussian systems has received a lot of attention since the end of

the 20th century, with the work on the particle filter (also called sequential Monte Carlo method) of Gordon, Salmond, and Smith [GSS93]; Kitagawa [Kit93]; Del Moral [DM97]; Doucet [Dou98]; and Liu and Chen [LC98], to name a few. See [God19] for a historical perspective on particle filtering.

Depending on the application, taking measurements can be an issue. For example, measuring the voting intentions of a population requires a survey, which can be time consuming and expensive. Another concern may be the energy consumption that is linked to measurement acquisition. This is essential for embedded control systems. For example, the use of GPS in a drone can discharge its battery and reduce its autonomy. In light of current events, the example of a pandemic comes to mind. In such a situation, policy makers need a daily estimate of the number of infected people to make well-informed decisions. To do this, a portion of the population must be tested, which requires tests, trained and available health care personnel, and people who are willing to be tested. In practice, all of these quantities are limited.

Another example of a real-world application where one wants to track a quantity and for which it is problematic to take measurements is radiation therapy. This case study has a particular role in this thesis because Chapter 6 is dedicated to it and it will be our typical use case. Radiation therapy is a method of cancer treatment whose objective is to irradiate the tumor with an ionizing beam in order to eliminate it. In the case of tumors located in the lung or liver, the tumor moves continuously, mainly due to respiratory movements and, to a lesser extent, due to the heartbeat. This movement makes it particularly challenging to treat liver or lung cancers with radiation therapy. Indeed, if the position of the tumor is not well known, there is a risk of irradiating the healthy tissues around the tumor, which can lead to complications, including a second cancer. One way to best account for tumor movement is to acquire X-ray images of the patient, from which information about the tumor's position can be extracted [SJSS04]. Indeed, having an estimate of the tumor's position as accurate as possible makes it possible to direct the ionizing beam to irradiate the tumor in the best possible way, or to interrupt the irradiation if the tumor leaves the area irradiated by the beam. An important disadvantage of this X-ray tracking method is that each X-ray image acquisition irradiates the patient, including her/his healthy tissues, which again can cause complications such as a second cancer. This illustrates that another reason why it may be difficult to measure the state of a system is that measuring may be hazardous to health if the number of measurements exceeds a safety threshold.

The previous examples show that the acquisition of measurements can be time-consuming, expensive, energy-consuming or hazardous to health. In such situations, it is necessary to reduce the number of measurements acquired to estimate the state of the system. In most cases, the measurements are acquired in a regular way, i.e., at a constant rate. In this thesis, we propose to relax this assumption. We assume that we have a measurement budget, i.e., a maximum number of measurements allowed, and the question we address is *"When should we measure"* If the measurements are expensive, time-consuming or energy-consuming, then the measurement budget will be set by the amount of money, time or energy available, respectively. In the case of tumor tracking for radiation therapy, the number of X-ray measurements can be set by a physician based on a maximum allowed imaging irradiation.

The choice of measurement times depends on the information available on the system. Indeed, without *a priori* information, there is no reason for irregular sampling. When the dynamics of the system are known, there are various reasons why regular measurement times may be suboptimal. On the one hand, the sooner a measurement is taken, the sooner (and therefore the longer) the information it contains can be used for estimation. This observation pushes to take measurements as soon as possible. On the other hand, if the signal amplitude varies greatly over time, it may be worth taking more measurements when the signal-to-noise ratio is larger.

When all the measurement times are chosen in advance, i.e., offline, we have a *time-triggered* mechanism. More precisely, the measurements are time-triggered when the time at which a measurement is taken does not depend on the previous measurements, i.e., the time of the $(j+1)$-th measurement does not depend on the first $j$ measurements. On the contrary, if the time of the $(j+1)$-th measurement depends on the first $j$ measurements, then the measurements are called *self-triggered*. In this case, the choice of the measurement times must necessarily be made online. Self-triggered mechanisms have the potential to give better performance because they use all available information to choose when to measure. However, they require online computation, which can be problematic in the case of embedded devices with reduced computing power. Time-triggered mechanisms do not suffer from this drawback.

In this thesis, we focus on the estimation of the state of a dynamic system over a finite time horizon. Moreover, we assume that the (finite) duration of the time horizon is known in advance, which is not always the case in practice.

To summarize, we are interested in the optimal allocation of a measurement budget for the estimation of the state of a stochastic dynamical system on a finite horizon. In Chapters 3 and 4, the optimality criterion will be the expected mean squared estimation error over the whole horizon. Then, in Chapter 5, we will study the problem of the choice of measurement times to minimize the worst-case estimation error. Chapters 3, 5 and 6 will focus on linear systems while Chapter 4 will deal with nonlinear systems.

## 1.1 Related work

This section is devoted to a review of the literature concerning the works related to this thesis. In addition, each chapter will discuss the works that are specifically related to it.

### 1.1.1 Networked control systems

In control theory, we consider networked control system when the sensors, the controller, and/or the actuators communicate through a network. In practice, these communication networks are imperfect and make the control task challenging because of packet losses (also called dropout), transmission delays, quantization, etc. [SSF+07]. Limited bandwidth is also a problem when considering a networked control system. Indeed, a high frequency of controller and communication updates across the network may overburden it. So, in works like [BPW02, TB10, XH04], adaptive communication protocols are developed to minimize the amount of information that must be sent across the network. For example, in [XH04], a measurement is sent to the controller only if it is sufficiently unlikely to occur. More precisely, the measurement is sent if it deviates too much from its expected value. In this case, the decision to send a measurement to the controller depends on the measurement itself. In this case, the transmission of measurements to the controller is event-triggered, but the measurements themselves are not. For a survey on networked control systems, see e.g., [HNX07, ZHG+19].

Just as networked control problems have received a lot of attention, so has the problem of estimation via remote sensors. For this reason, estimation problems subject to measurement failures have been studied intensively in the recent decades. In most of these works, dropouts are assumed to occur randomly, generally, according to a Bernoulli distribution [SXXS08, LCP10, LWYG16, CLM10, YFS19, SSF+04, KDMS20, CRZL20,

YD16, YJCW20, KN19]. Under these assumptions, different problems have been studied. In [KDMS20] it is assumed that the times at which dropouts occur are unknown (when there is one dropout, the received measurement is pure noise), and a method to detect dropouts is proposed. In [KCS19], a system with multiple sensors is subject to denial-of-service and false data injection attacks. The proposed method first detects the attack and isolates the concerned subsystems, then identifies the kind of attack, and uses a robust observer.

In other works, the dropouts can only occur according to certain patterns (described, e.g., by a finite-length language [RYO18, RYO20], or by an automaton [YO21]), and the objective is to design an estimator or a controller that is robust to all these patterns.

### 1.1.2 The sensor scheduling problem

In the above cited papers, measurement times (or equivalently, the dropout times) are *not* design variables: they are endured and not chosen. On the contrary, the sensor scheduling problem (or sensor management problem) consists in designing the behavior of the sensors. More precisely, a set of sensors is available but only one (or a few) of them can be used at the same time. In most of the works, the active sensor is chosen in order to minimize the variance of the estimation error.

This problem has been studied in the case of an infinite horizon in both the discrete [GCHM06, LNFD10, MGS14, SCC11, ZZH$^+$14, LFMV14] and the continuous-time settings [HC19]. In the continuous-time case, Ha *et al.* show in [HC19] that the infinite horizon optimal cost and optimal scheduling are independent of the initial error covariance. In addition, they prove that a periodic scheduling can approximate arbitrarily the cost of an optimal scheduling. Similar results are obtained in the discrete-time case in [MGS14, ZZH$^+$14]. Some works consider a cost associated to the use of the sensors [LNFD10, MPD67]. Other works dealing with periodic scheduling associate a usage budget per period to each sensor [SCC11, LFMV14].

The finite horizon problem has also received attention in the past decades. The linear and Gaussian case has been studied in both discrete [MPD67, VZA$^+$12] and continuous-time [Ath72, LTL01, WRL10] settings. In 1972, Athans [Ath72] studied the best prediction at the end of a horizon (and not on average over the horizon). In that framework, different sensors are available, each of which is associated with a cost. The goal is to select sensors (one at a time) in order to minimize a trade-off between

the sensing cost and the prediction error. In [LTL01], Lee *et al.* consider that different sensors are available for measurements and that the user has to choose only a few of them at each time. In addition, a (possibly zero) cost can be associated with each sensor switch. In the work described in Woon *et al.*'s [WRL10] published in 2010, different sensors are available but only one can be activated at a time. There is no budget constraint on the use of each measurement process. In 2012, Vitus *et al.* [VZA+12] proposed two algorithms to tackle the sensor scheduling problem in the linear and Gaussian case. The first one provides an optimal solution but can be computationally expensive. The second one provides a suboptimal solution and depends on a tuning parameter to make a trade-off between the optimality gap and the algorithmic complexity. For this suboptimal algorithm, a bound on the performance gap is given. However, no measurement cost or budget constraint on the number of uses of each sensor is considered. See references in [VZA+12] for a literature review on the sensor scheduling problem in the linear and Gaussian case. In 2016, Tzoumas *et al.* [TJP16a, TJP16b] focused on a modified version of the sensor scheduling problem. The objective function to be minimized is not the averaged expected mean squared error (which is the trace of the error covariance matrix), but the log of the determinant of the error covariance matrix. Under specific assumptions, it is shown that the problem is NP-hard. Then, using the fact that the problem has a matroid structure, it is proven that a polynomial-time greedy algorithm reaches a solution that is up to a multiplicative factor $1/2$ from the optimal solution. These results are generalized to the nonlinear and non-Gaussian cases in [TAJP17].

Still when the time horizon is finite, the sensor scheduling problem has also been studied for nonlinear and non-Gaussian systems. In [GOR10], a Gaussian process-based global optimization method is used as a black-box optimizer to dynamically select sensors in order to minimize the error at the end of a horizon (and not on average over the horizon). In [HC06, LKCG09], the authors formalize the sensor scheduling problem with a sensing cost as a partially observable Markov decision process and use a Monte Carlo Q-value approximation method to solve it approximately. Doucet *et al.* propose in [DVAD02] an optimal one-step ahead policy to maximize the expected Kullback-Leibler divergence between the filtering and the prediction density in order to maximize the information obtained from the current measurement. Similarly, in [YN18], a set of active sensors is selected at each time step in order to trade off between the number of active sensors and the posterior Cramér-Rao lower bound,

which is a lower bound on the expected mean squared estimation error. The sensor scheduling problem has also been studied when the set of sensors is continuous (e.g., to optimize the spatial position of a mobile sensor). For this problem, [SVDE03, SKV+07] propose an algorithm using a simulation-based gradient approximation. When the state space is a finite set (the dynamics is a finite hidden Markov model), [Kri02] proposes a dynamic programming algorithm to find an optimal solution at the cost of an exponential complexity. Then, an algorithm with a better complexity but providing a suboptimal solution is proposed.

Table 1.1 summarizes the main differences between these articles on the sensor scheduling problem. Even if this table is not exhaustive, it allows some observations. First, one can see that in the nonlinear and non-Gaussian case, most (if not all) works assume discrete time and a finite horizon (except for approaches based on a one step look-ahead policy, which can be used on a finite or infinite horizon). Second, the most common optimality criterion is the average mean squared error. This is the criterion that will be used in Chapters 3 and 4.

### 1.1.3 Time-/self-/event-triggered mechanisms

In the same way that measurements can be time- or self-triggered for estimation, in control theory, control actions can be too. In fact, the problem of minimizing the number of control actions has been studied mainly in three different cases: time-triggered, self-triggered and event-triggered control [HJT12].

Similarly to time-triggered measurements, in time-triggered control actions, the control instants are chosen in advance and are, therefore, independent of the measurements.

In self-triggered control, the controller determines when the next action will take place at the moment that the current action is taken. These controllers rely on the fact that the self-triggered control action always occurred simultaneously with a self-triggered measurement.

On the contrary, an event-triggered controller decides to take an action only when the measurement of the state satisfies a certain condition (e.g., state leaves a set). Note that in most event-triggered controllers the state of the system or a measurement of it is available to the controller at all times [ZHZ17].

With this terminology, we can talk about time-, self-, or event-triggered control actions. On the other hand, we only speak of time- and self-

**Table 1.1** Summary of the state of the art of the sensor scheduling problem. "(non)linear and (non-)Gaussian" indicates that the linear and Gaussian cases, and the nonlinear and non-Gaussian case have both been studied separately in that paper. When the sensors are chosen by a one-step ahead policy, the mention "(one step)" is indicated in the column "Objective function". In this case, the method can be applied on a finite or infinite horizon, which is denoted "(in)finite". MSE stands for mean squared error.

| Dynamics | Horizon | Time | Sensing cost | Objective function | References |
|---|---|---|---|---|---|
| linear and Gaussian | infinite | discrete | no | steady state error covariance | [GCHM06] |
| | | | | average MSE | [MGS14] [SCC11] [ZZH+14] [LFMV14] |
| | | continuous | yes | average MSE | [LNFD10] |
| | | | no | average MSE | [HC19] |
| | finite | discrete | no | average MSE | [VZA+12] |
| | | | | initial state and process noise covariance log determinant | [TJP16b] |
| | | continuous | yes | terminal MSE | [Ath72] |
| | | | no | average + terminal MSE | [LTL01] [WRL10] |
| | | | | average error covariance log determinant | [TJP16a] |
| (non)linear and (non-)Gaussian | finite | discrete | yes | (non-)quadratic control cost | [MPD67] |
| nonlinear and non-Gaussian | finite | discrete | yes | average MSE | [HC06] [LKCG09] |
| | | | no | average MSE | [SVDE03] [SKV+07] |
| | | | | conditional entropy of state knowing measurements | [TAJP17] |
| | (in)finite | discrete | no | Kullback-Leibler divergence between prediction and filtering densities (one-step) | [DVAD02] |
| nonlinear and Gaussian | (in)finite | discrete | yes | posterior Cramér-Rao lower bound (one step) | [YN18] |
| hidden Markov model | finite | discrete | yes | average MSE | [Kri02] |
| model-free | finite | discrete | no | average root MSE | [GOR10] |

triggered measurements, but not of event-triggered measurements. Indeed, an event-triggered mechanism requires measuring the state all the time. Consequently, doing event-triggered measurements would correspond to choosing when to measure while measuring all the time. In most cases, this does not make sense and it is not considered in this thesis. Note however that if there are two sensors, a cheap one that can be used continuously, and an expensive one whose use we want to limit, we could activate the expensive sensor based on the measurements of the cheap one. This could be considered as an event-triggered measurement mechanism. Finally, note that, depending on the case, time- or self-triggered measurements can be used for estimation (as in this thesis) or for control. For example, in [KF15], measurements are self-triggered in order to control a linear system.

Time-triggered measurements have been considered for estimation. In the continuous-time one-dimensional, linear and Gaussian case, Sano *et al.* [ST70] studied in 1970 the problem of selecting measurement times to minimize the error variance at the end of a horizon. They found an explicit formula when there was only one measurement and proposed a numerical method when the number of measurements was greater than one. In the same setting, Tanaka *et al.* proposed in 1985 to select the measurement times to minimize the maximum over time of the error variance, see [TO85]. A closed form expression is provided for the optimal measurement times. The time interval between two consecutive measurements is constant, except for the first measurement, so that, at each measurement time, the error variance is the same. More recently, Aksenov *et al.* studied the Brownian motion case over similar assumptions [AAMJ17, AAMJ19] and obtained the same results.

A few authors have considered the case of self-triggered measurements for the estimation of the state of a system. For linear systems, an optimal stationary sampling policy is proposed in [SHB16] where the cost function is related to the Fisher information matrix. In [ANSV13], the authors use self-triggered measurements for estimating the state of a continuous time deterministic nonlinear system with noiseless sampling. Their approach relies on an incremental bound related to the global Lipschitz constant of the system. Also for continuous-time nonlinear systems with discrete measurements, [MP14] proposes a self-triggered measurement mechanism for set-membership state estimation when the deterministic dynamics are partially unknown.

## 1.2 Contribution and thesis organization

The rest of this thesis is organized as follows.

Chapter 2 provides the necessary background for the other chapters. First, we define the mean squared error estimate in the context of stochastic dynamical systems (Section 2.2), then we present the Kalman filter (Section 2.3) and the particle filter (Section 2.4). Finally, we describe the linear quadratic regulator and show that it is the dual of the Kalman filter (Section 2.5).

In Chapter 3, we focus on the choice of measurement times that minimize the expected mean squared estimation error in the case of a linear and Gaussian system. We prove that in this case, there is an equivalence between the optimal time-triggered and self-triggered strategies. This leads us to the *optimal intermittent Kalman filter* for which the optimal measurement times are obtained by solving a combinatorial optimization problem.

Chapter 4 generalizes the results of Chapter 3 to the case of nonlinear and non-Gaussian systems. Contrary to the linear and Gaussian case, we have here a difference between the time-triggered and self-triggered strategies. First, we present the *time-triggered particle filter*, for which the measurement times are obtained by solving a single combinatorial optimization problem. Then, two self-triggered strategies are presented. The first one, the *ex post particle filter*, is a recursive version of the time-triggered particle filter and requires to solve a combinatorial optimization problem after each measurement acquisition. The second self-triggered particle filter, called the *self-triggered particle filter* is formalized as a stochastic program. Moreover, we prove that, in terms of expected mean squared error, the time-triggered particle filter is outperformed by the ex post particle filter, which is, in turn, outperformed by the self-triggered particle filter.

While the previous chapters focus on minimizing the expected mean squared error, Chapter 5 focuses on robust estimation for linear systems. The process and measurement noises are assumed to be bounded and the goal is to co-design the measurement times and the estimator to minimize the worst-case estimation error. In fact, we solve a more general problem of control theory: we co-design a controller, the measurement times and the control times in order to keep the state of the system in a safety set. We prove that this problem reduces to a mixed integer linear program, which can, in practice, be readily solved efficiently by off-the-shelf software.

In Chapter 6, the optimal intermittent Kalman filter developed in Chapter 3 is used in radiation therapy to track mobile tumors from X-ray images.

We demonstrate on real patient data that optimizing X-ray measurement times improves tracking performances, allowing a better treatment of the tumor.

Finally, we conclude by discussing the results obtained in this thesis and open perspectives for future work.

# List of publications

HERE is the list of articles written during my PhD thesis, whose content is included in this manuscript. It also includes attendance at conferences.

- Antoine Aspeel, Axel Legay, Raphaël M Jungers, and Benoit Macq. Optimal measurement budget allocation for Kalman prediction over a finite time horizon by genetic algorithms. *EURASIP Journal on Advances in Signal Processing*, 2021(1):1–24, 2021.

- Antoine Aspeel, Damien Dasnoy, Raphaël M Jungers, and Benoit Macq. Optimal intermittent measurements for tumor tracking in X-ray guided radiotherapy. In *Medical Imaging 2019: Image-Guided Procedures, Robotic Interventions, and Modeling*, volume 10951, page 109510C. International Society for Optics and Photonics, 2019.

- Antoine Aspeel, Amaury Gouverneur, Raphaël M Jungers, and Benoit Macq. Optimal measurement budget allocation for particle filtering. In *27th IEEE International Conference on Image Processing*, 2020.

- Antoine Aspeel, Kwesi Rutledge, Raphaël M Jungers, Benoit Macq, and Necmiye Özay. Optimal control for linear networked control systems with information transmission constraints. In *The 60th IEEE International Conference on Decision and Control*, 2021.

- Antoine Aspeel, Axel Legay, and Benoit Macq. Genetic algorithms for optimal intermittent measurements for tumor tracking. In *The International Conference on the Use of Computers in Radiation Therapy*, 2019. (No proceedings).

- Antoine Aspeel, Amaury Gouverneur, Raphaël M Jungers, and Benoit Macq. Optimal intermittent particle filter. To appear.

The following articles were also written during my PhD thesis, but their content is not covered by this manuscript.

- Damien Dasnoy, Antoine Aspeel, Kevin Souris, and Benoit Macq. Locally tuned deformation fields combination for 2D cine-MRI-based driving of 3D motion models. In *Physica Medica*, 94:8–16, 2022.
- Michaël Fanuel, Antoine Aspeel, Jean-Charles Delvenne, and Johan AK Suykens. Positive semi-definite embedding for dimensionality reduction and out-of-sample extensions. In *SIAM Journal on Mathematics of Data Science*, 4(1):153–178, 2022
- Tom Duterme, Quentin Thomas, and Antoine Aspeel. The development of model risk management: a Luhmannian perspective. In *1st MUSEES Conference Modeling Uncertainty in Social, Economic, and Environmental Sciences*, 2022. (No proceedings).
- Antoine Aspeel, Vincent François-Lavet, Raphaël M Jungers, and Benoit Macq. Self-triggered measurements for estimation with deep reinforcement learning. To appear.
- Antoine Aspeel, Jean-Charles Delvenne, Michaël Fanuel, and Michael T. Schaub. Ellipsoidal embedding of graphs. To appear.[1]
- Antoine Germain, Tom Duterme, and Antoine Aspeel. Poverty measurement and Adaptive preferences: a synthesis. To appear.

[1] The order of the authors is not determined but alphabetical here.

# List of symbols

BELOW is a list of notations used in this thesis. In each chapter, a section will specify the relevant notations.

**Sets**

| | |
|---|---|
| $\emptyset$, $\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}_+$ | The empty set, the set of nonnegative integers, the set of real numbers, the set of nonnegative real numbers. |
| $\mathbb{S}_+^n$ | The set of $n \times n$ positive semi-definite matrices. |
| $\lvert\mathcal{A}\rvert$ | The cardinality of the set $\mathcal{A}$. |
| $\mathcal{A} \times \mathcal{B}$ | The Cartesian product of sets $\mathcal{A}$ and $\mathcal{B}$. |
| $\mathcal{A}^n$ | The $n$-th Cartesian power of the set $\mathcal{A}$. |
| $\partial\mathcal{A}$ | The boundary of the set $\mathcal{A}$. |

**Vectors and matrices**

| | |
|---|---|
| $I_n$ | The $n \times n$ identity matrix. |
| $0_{m\times n}$ | The $m \times n$ zero matrix. |
| $\mathbb{1}_n$ | The $n$ dimensional vector of ones. |
| $x_i$, $X_{i,j}$ | The $i$-th entry of the vector $x$, the $(i,j)$-th entry of matrix $X$. |
| $X_{i,:}$, $X_{:,j}$ | The $i$-th row of matrix $X$, the $j$-th column of matrix $X$. |

| | |
|---|---|
| $X_{\mathcal{I},:}$, $X_{:,\mathcal{J}}$ | The matrix whose rows are the rows of $X$ of indices in the set $\mathcal{I}$, the matrix whose columns are the columns of $X$ of indices in the set $\mathcal{J}$. |
| $x^\top$, $X^\top$ | The transpose of the vector $x$, the transpose of the matrix $X$. |
| $X^{-1}$ | The inverse of the invertible matrix $X$. |
| $\|x\|$, $\|a\|$ | The Euclidean norm of the vector $x$, the absolute value of the real number $a$. |
| $\mathrm{Tr}[X]$ | The trace of the square matrix $X$. |
| $X \otimes Y$ | The Kronecker product between matrices $X$ and $Y$. |

**Random variables**

| | |
|---|---|
| $x \sim \mathcal{X}$ | The random variable $x$ is distributed according to the distribution $\mathcal{X}$. |
| $\mathbb{E}_x[z\|y]$ | The expectation of the random variable $z$ knowing $y$, according to the distribution of the random variable $x$. |

# 2

# Background

THIS chapter presents the background on which the other chapters are based. After a formalization of stochastic dynamical systems and a definition of the mean squared error estimate (Section 2.2), the topics covered are the Kalman filter (Section 2.3), the particle filter (Section 2.4), and the duality between Kalman filter and linear quadratic regulator (Section 2.5).

## 2.1 Notation

In this chapter, the following notations will be used. The sets of natural and real numbers are denoted $\mathbb{N}$ and $\mathbb{R}$, respectively. $I_n$ and $0_{m\times n}$ denote the identity matrix of $\mathbb{R}^{n\times n}$ and the null matrix of $\mathbb{R}^{m\times n}$, respectively. The dimensions will sometimes be omitted when they are non-ambiguous. For a matrix or vector $x$, $x^\top$ is its transpose, for a square matrix $X$, its inverse is $X^{-1}$. Matrices are noted in upper case, e.g., $X$, and vectors in lower case, e.g., $x$. The notation $a \propto b$ indicates a proportional relationship between $a$ and $b$.

The notation $x \sim \mathcal{X}$ indicates that the random variable $x$ follows a distribution $\mathcal{X}$. The Gaussian distribution of mean $\mu$ and covariance matrix $\Sigma$ is written $\mathcal{N}(\mu, \Sigma)$, and the expectation operator is written $\mathbb{E}[\cdot]$, or $\mathbb{E}_{x\sim p(x)}[\cdot]$ to emphasize the probabilty density function $p(x)$ of the random variable $x$ according to which the expectation is computed.

Finally, the following notation is used: $y(0{:}t) = \big[y(0), \dots, y(t)\big]$.

## 2.2 Stochastic dynamical systems

We consider discrete-time nonlinear stochastic dynamical systems, for which the noises follow arbitrary distributions. For $t \in \mathbb{N}$, it is

$$x(t+1) = f(x(t), w(t)), \tag{2.1a}$$
$$y(t) = g(x(t), v(t)), \tag{2.1b}$$
$$z(t) = h(x(t)), \tag{2.1c}$$
$$x(0) \sim \mathcal{X}_0, \tag{2.1d}$$

where $x(t) \in \mathbb{R}^{n_x}$ is the state of the system, $w(t) \in \mathbb{R}^{n_w}$ is the process noise and follows a known distribution $w(t) \sim \mathcal{W}$. $y(t) \in \mathbb{R}^{n_y}$ is the measurement, and $v(t) \in \mathbb{R}^{n_v}$ is the measurement noise and follows a known distribution $v(t) \sim \mathcal{V}$. The output $z(t) \in \mathbb{R}^{n_z}$ is the quantity we want to estimate from the measurements. The initial state follows a known distribution, i.e., $x(0) \sim \mathcal{X}_0$. In addition, functions $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_w} \to \mathbb{R}^{n_x}$, $g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_v} \to \mathbb{R}^{n_y}$, and $h : \mathbb{R}^{n_x} \to \mathbb{R}^{n_z}$ are known. Equations (2.1a), (2.1b) and (2.1c), are referred to as transition equation, measurement equation, and output equation, respectively. For example, in the case of the lung tumor tracking problem, $x(t)$ can be a description of the state of the lung at time $t$, $y(t)$ can be a 2D projection of the tumor position obtained from X-ray images, and $z(t)$ can be the 3D position of the tumor center.

The optimal *mean squared error* (MSE) estimate of $z(t)$ knowing the measurements $y(0{:}\tau)$ is defined by

$$\hat{z}(t|\tau) = \arg\min_{\hat{z}} \mathbb{E}\left[\|z(t) - \hat{z}\|^2 \Big| y(0{:}\tau)\right], \tag{2.2}$$

where $\|\cdot\|$ is the Euclidean norm. We can define the MSE estimate of $x(t)$ knowing $y(0{:}\tau)$, denoted $\hat{x}(t|\tau)$, in the same way. $\hat{z}(t|\tau)$ will sometimes be noted $\hat{z}(t|y(0{:}\tau))$ to make explicit the dependence in $y(0{:}\tau)$. When $t > \tau$, we talk about prediction, when $t = \tau$, we talk about filtering, and when $t < \tau$, we talk about smoothing. The prediction and filtering are said to be causal because the future is not used to estimate the present. For this reason, they can be used online, while this is not the case for smoothing. In this thesis, we will focus on filtering and prediction. A classical result of estimation theory is that the MSE estimate is given by the expectation

knowing the measurements [LXP17, Chapter 1],

$$\hat{z}(t|\tau) = \mathbb{E}\left[z(t)|y(0{:}\tau)\right].$$

In addition, this estimate is unbiased.

Before showing how to compute an approximation of $\hat{z}(t|\tau)$ in the case of nonlinear and non-Gaussian dynamics such as (2.1), let us first present the case of linear and Gaussian systems.

## 2.3 The Kalman filter

In this section, we present the Kalman filter. It is a method to compute efficiently the MSE estimate for linear and Gaussian dynamical systems. We will not present a complete derivation of the Kalman filter equations (which can be found in many places, e.g., in [Kal60, LXP17]), but we will rather give an intuitive interpretation and emphasize its important properties.

The general form for linear and Gaussian systems is

$$x(t+1) = Ax(t) + b + Gw(t), \tag{2.3a}$$
$$y(t) = Cx(t) + d + v(t), \tag{2.3b}$$
$$z(t) = Dx(t), \tag{2.3c}$$
$$x(0) \sim \mathcal{N}(\bar{x}_0, \bar{P}_0), \tag{2.3d}$$

where $w(t) \sim \mathcal{N}(0, Q)$, $v(t) \sim \mathcal{N}(0, R)$ and $x(0)$ follow independent Gaussian distributions. Matrices $A$, $G$, $C$, $D$, $\bar{P}_0$, $Q$, $R$ and vectors $b$, $d$ and $\bar{x}_0$ can be time-dependent and have compatible dimensions (note that $n_y = n_v$). In addition, $Q$ and $\bar{P}_0$ are symmetric and positive semi-definite, while $R$ is symmetric and positive definite.

The Kalman filter keeps track of the distribution of the state $x(t)|y(0{:}\tau)$ and the estimation error $(x(t) - \hat{x}(t|\tau))\,|y(0{:}\tau)$, for $\tau \in \{t-1, t\}$. In fact, since an affine combination of Gaussian random variables also follows a Gaussian distribution, these quantities follow Gaussian distributions for all $t$. Therefore, they are completely characterized by their mean and covariance matrices. Moreover, they both have the same covariance matrix,

noted $P(t|\tau)$. The MSE estimate beeing unbiased, one can write

$$\begin{aligned} x(t)|y(0{:}\tau) &\sim \mathcal{N}(\hat{x}(t|\tau), P(t|\tau)), \\ (x(t) - \hat{x}(t|\tau))|y(0{:}\tau) &\sim \mathcal{N}(0, P(t|\tau)), \end{aligned}$$

for $\tau \in \{t-1, t\}$.

The Kalman filter alternates between a prediction step computing $\hat{x}(t+1|t)$ from $\hat{x}(t|t)$, and a correction step computing $\hat{x}(t+1|t+1)$ from $\hat{x}(t+1|t)$. Then, $\hat{z}(t+1|t)$ or $\hat{z}(t+1|t+1)$ can be obtained from $\hat{x}(t+1|t)$ or $\hat{x}(t+1|t+1)$, respectively. First, the initialization of the Kalman filter is done as follows,

$$P(0|-1) = \bar{P}_0, \tag{2.4a}$$
$$\hat{x}(0|-1) = \bar{x}_0, \tag{2.4b}$$
$$\hat{z}(0|-1) = D\bar{x}_0. \tag{2.4c}$$

Then, the correction, or measurement update equations to incorporate the information from the last measurement $y(t)$ are as follows

$$K(t) = P(t|t-1)C^\top [CP(t|t-1)C^\top + R]^{-1} \tag{2.4d}$$
$$P(t|t) = (I - K(t)C)P(t|t-1), \tag{2.4e}$$
$$\hat{x}(t|t) = \hat{x}(t|t-1) + K(t)[y(t) - C\hat{x}(t|t-1) - d], \tag{2.4f}$$
$$\hat{z}(t|t) = D\hat{x}(t|t), \tag{2.4g}$$

where the quantity $K(t)$ is called the Kalman gain.

Finally, the prediction, or time update equations are

$$P(t+1|t) = AP(t|t)A^\top + GQG^\top, \tag{2.4h}$$
$$\hat{x}(t+1|t) = A\hat{x}(t|t) + b, \tag{2.4i}$$
$$\hat{z}(t+1|t) = D\hat{x}(t+1|t). \tag{2.4j}$$

*Remark* 2.1. Some important remarks must be made concerning these equations. First, the Kalman filter requires the inversion of only one matrix (in equation (2.4d)) of size $n_y \times n_y$ and most of the time $n_y < n_x$. Overall, the number of operations required at each time step is on the order of $n_x^3$. Secondly, the quantities $P(t|t)$, $K(t)$ and $P(t+1|t)$ do not depend on $y(t)$. Therefore, they can be computed offline. The computations that must be performed online are vector and matrix additions and multiplica-

tions. Third, it is not necessary to keep track of previous measurements. In fact, the memory needed to apply the Kalman filter is constant in time. All these features have contributed to the popularity and wide use of the Kalman filter. ◇

### 2.3.1 Summary

Let us summarize the Kalman filter in a theorem.

**Theorem 2.2.** *The optimal MSE estimates $\hat{z}(t|t)$ and $\hat{z}(t|t-1)$ (see definition (2.2)) for the linear and Gaussian dynamics (2.3) are given by equations (2.4).*

*Proof.* See [Kal60] for the original proof, or [LXP17, Chapter 2] for a more recent presentation. □

Finally, to use the Kalman filter, one can apply the following steps:

1. **Initialization:** Define $P(0|-1)$, $\hat{x}(0|-1)$ and $\hat{z}(0|-1)$ according to equations (2.4a), (2.4b) and (2.4c), respectively.
2. **Correction:** After a measurement $y(t)$ is received at time $t \in \mathbb{N}$, compute $K(t)$, $P(t|t)$, $\hat{x}(t|t)$ and $\hat{z}(t|t)$ according to equations (2.4d), (2.4e), (2.4f) and (2.4g), respectively.
3. **Prediction:** Compute $P(t+1|t)$, $\hat{x}(t+1|t)$ and $\hat{z}(t+1|t)$ according to equations (2.4h), (2.4i) and (2.4j), respectively. Back to step 2.

### 2.3.2 Example

We now illustrate the Kalman filter on an example that will also be studied in Chapter 3. It is a mass-spring system subjected to a random force, discretized with a discretization step $\delta = 0.1[s]$ (a complete derivation of these equations will be described in Section 3.3.4). The discrete system is given by

$$A = \begin{bmatrix} 0.995 & 0.100 \\ -0.100 & 0.995 \end{bmatrix}, \quad C = D = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad d = 0, \tag{2.5a}$$

$$G = 1, \quad Q = 10^{-4} \begin{bmatrix} 0.002 & 0.031 \\ 0.031 & 0.623 \end{bmatrix}, \quad R = 1, \quad \bar{P}_0 = I_2, \quad \bar{x}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{2.5b}$$

Figure 2.1 shows a trajectory $z(t)$ over 100 time steps, along with its MSE estimate $\hat{z}(t|t)$ obtained using the Kalman filter.

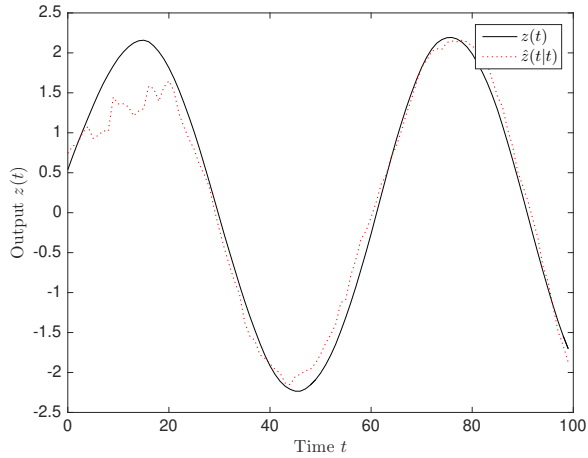



**Fig. 2.1** Trajectory $z(t)$ for the linear dynamical system described by (2.5), and its MSE estimate $\hat{z}(t|t)$ given by the Kalman filter.

## 2.4 Particle filter

Let us now consider the general case of a nonlinear and non-Gaussian system described by the equations (2.1). An approximation of $\hat{z}(t|\tau)$ (for $\tau \in \{t-1, t\}$) can be obtained using a particle filter. The main idea is to recursively approximate the probability density function of $x(t)|y(0{:}t)$, that we denote $p(x|y(0{:}t))$. We use similar notations for other probability density functions. The presentation of this section is inspired by [AMGC02].

### 2.4.1 Importance sampling

Suppose we want to construct a discrete weighted approximation of a probability density function $p(x)$ from which sampling is difficult. Let us also assume that we have a function $\pi(\cdot) \propto p(\cdot)$ that is easy to evaluate. Finally, suppose we have $N_s$ samples $\{x^i\}_{i=1}^{N_s}$ distributed according to a distribution with density $q(\cdot)$ such that $p(x) > 0 \Rightarrow q(x) > 0$. We are looking for weights $\{w^i\}_{i=1}^{N_s}$ such that

$$p(x) \approx \hat{p}(x) = \sum_{i=1}^{N_s} w^i \delta(x - x^i),$$

where $\delta(\cdot)$ is the Dirac delta measure and where the weights are normalized, i.e., $\sum_{i=1}^{N_s} w^i = 1$.

The importance sampling algorithm consists in defining the weights as

$$w^i \propto \frac{\pi(x^i)}{q(x^i)}. \tag{2.6}$$

Thus, for a measurable function $h(\cdot)$, we have

$$\begin{aligned}
\mathbb{E}_{x\sim p(\cdot)}[h(x)] &= \int p(x)h(x)\,dx \\
&\approx \int \hat{p}(x)h(x)\,dx \\
&= \int \sum_{i=1}^{N_s} w^i \delta(x - x^i)h(x)\,dx \\
&= \sum_{i=1}^{N_s} w^i h(x^i).
\end{aligned}$$

In particular, it is possible to show that this estimate is asymptotically unbiased. Importantly, only the knowledge of $h(\cdot)$, $\{x^i\}_{i=1}^{N_s}$ and $\{w^i\}_{i=1}^{N_s}$ is required for this computation.

### 2.4.2 Sequential importance sampling algorithm

In this section, we will show how importance sampling can be used to recursively construct an approximation of $p(x(t)|y(0{:}t))$ using samples $x^i(t)$ drawn from a distribution $q(x(t)|y(0{:}t))$. In that case, (2.6) becomes

$$w^i(t) \propto \frac{p(x^i(t)|y(0{:}t))}{q(x^i(t)|y(0{:}t))}. \tag{2.7}$$

Using the Bayes' formula and the Markov property, one can write

$$\begin{aligned}
p(x(t)|y(0{:}t)) &= \frac{p(y(t)|x(t), y(0{:}t-1))p(x(t)|y(0{:}t-1))}{p(y(t)|y(0{:}t-1))} \\
&= \frac{p(y(t)|x(t))p(x(t)|x(t-1))p(x(t-1)|y(0{:}t-1))}{p(y(t)|y(0{:}t-1))} \\
&\propto p(y(t)|x(t))p(x(t)|x(t-1))p(x(t-1)|y(0{:}t-1)).
\end{aligned}$$

Then, by choosing a distribution $q(x(t)|y(0{:}t))$ that can be written recursively as

$$q(x(t)|y(0{:}t)) = q(x(t)|x(t-1), y(t))q(x(t-1)|y(0{:}t-1)),$$

relation (2.7) can be rewritten as

$$\begin{aligned} w^i(t) &\propto \frac{p(y(t)|x^i(t))p(x^i(t)|x^i(t-1))p(x^i(t-1)|y(0{:}t-1))}{q(x^i(t)|x^i(t-1), y(t))q(x^i(t-1)|y(0{:}t-1))} \\ &\propto w^i(t-1)\frac{p(y(t)|x^i(t))p(x^i(t)|x^i(t-1))}{q(x^i(t)|x^i(t-1), y(t))}. \end{aligned} \tag{2.8}$$

Each time a new measurement $y(t)$ is received, the updated particle $x^i(t)$ is drawn from $q(x(t)|x^i(t-1), y(t))$. Then, formula (2.8) is used to update the weights. In many cases, $p(y(t)|x^i(t))$ and $p(x^i(t)|x^i(t-1))$ can be computed easily from (2.1b) and (2.1a), respectively.

Finally, these weights lead to the approximation

$$p(x(t)|y(0{:}t)) \approx \sum_{i=1}^{N_s} w^i(t)\delta(x(t) - x^i(t)), \tag{2.9}$$

which can be used to compute

$$\hat{z}(t|y(0{:}t)) = \mathbb{E}[z(t)|y(0{:}t)] = \mathbb{E}[h(x(t))|y(0{:}t)] \approx \sum_{i=1}^{N_s} w^i(t)h(x^i(t)). \tag{2.10}$$

This method is called the sequential importance sampling algorithm.

### 2.4.3 Degeneracy problem and resampling

A major problem with the sequential importance sampling algorithm is that, after a few iterations, almost all particles will have an almost zero weight [LSSC14]. As a consequence, most of the computational effort will often have a nearly zero contribution to the approximation (2.10). This phenomenon is called *degeneracy problem*. Two options exist to limit the degeneracy problem: a good choice of importance density and resampling. The first consists in choosing a distribution $q(x(t)|x(t-1), y(t))$ that minimizes the variance of the weights. However, this most often leads to a distribution that is difficult to evaluate, and from which it is difficult to

sample. For this reason, the importance density is often chosen as

$$q(x(t)|x^i(t-1), y(t)) = p(x(t)|x^i(t-1)).$$

Unless otherwise specified, we will proceed in this way in this thesis. This choice of importance density allows rewriting the update rule (2.8) as

$$w^i(t) \propto w^i(t-1)p(y(t)|x^i(t)). \tag{2.11}$$

The second method to tackle the degeneracy problem is *resampling*. It consists in focusing on the particles with a high weight. To do this, the idea is to sample (with replacement) $N_s$ new particles from the current discrete weighted approximation of $p(x(t)|y(0{:}t))$ given by (2.9). The new particles are then i.i.d. according to (2.9) and their weights are all $1/N_s$. Note that since all weights $w^i(t-1)$ are equal to $1/N_s$, equation (2.11) can be rewritten

$$w^i(t) \propto p(y(t)|x^i(t)). \tag{2.12}$$

In general, the approximation provided by the particle filter is biased when the number of particles is finite (the bias being $O(1/N_s)$) [DJ09, Section 3.2]. However, we neglect this aspect in this thesis.

### 2.4.4 Summary

Let us summarize the sequential importance resampling filter algorithm. This is the method to which we refer when a particle filter is mentioned in this thesis.

1. **Initialization:** $N_s$ particles $\{x^i(0)\}_{i=1}^{N_s}$ are generated according to the distribution $\mathcal{X}_0$, and their weights $\{w^i(0)\}_{i=1}^{N_s}$ are set to $1/N_s$.
2. **Weighting:** After a measurement $y(t)$ is received at time $t \in \mathbb{N}$, the normalized weights $\{w^i(t)\}_{i=1}^{N_s}$ of these particles are computed according to (2.12).
3. **Resampling:** New particles $\{x^i(t)\}_{i=1}^{N_s}$ are resampled according to (2.9) and all weights $\{w^i(t)\}_{i=1}^{N_s}$ are reset to $1/N_s$.
4. **Estimation:** The particles and weights are used to approximate $\hat{z}(t|y(0{:}t))$ using (2.10).
5. **Mutation:** The particles are evolved according to the dynamics of the system, i.e., $x^i(t+1) \sim p(x(t+1)|x^i(t))$. Back to step 2.

Note that if we want to obtain an approximation of $\hat{z}(t+1|y(0{:}t))$, we can use the following approximation after step 5, and before going back to step 2:

$$\hat{z}(t+1|t) \approx \sum_{i=1}^{N_s} w^i(t+1) h(x^i(t+1)).$$

### 2.4.5 Example

Let's illustrate the particle filter with an example. This is a system often used as a Benchmark for particle filtering [AMGC02, CPS92, Kit96, KLJF02]. It will be used in Chapter 4, it is described by

$$x(t+1) = \frac{x(t)}{2} + \frac{25x(t)}{1+x(t)^2} + 8\cos(1.2t) + w(t), \tag{2.13a}$$

$$y(t) = \frac{x(t)^2}{20} + v(t), \tag{2.13b}$$

$$z(t) = x(t), \tag{2.13c}$$

$$x(0) \sim \mathcal{N}(0, 5^2), \tag{2.13d}$$

where $w(t)$, $v(t)$ and $x(0)$ are randomly distributed according to independent zero mean Gaussian distributions with standard deviations $\sigma_w = 1$, $\sigma_v(t) = \sin(0.25t) + 2$ and $\sigma_{x_0} = 5$, respectively. Note that this system is time-varying, however, the particle filter algorithm can be adapted straightforwardly.

Figure 2.2 shows a trajectory $z(t)$ over 100 time steps and its estimation $\hat{z}(t|y(0:t))$ by the particle filter algorithm.

## 2.5 Linear quadratic regulator and duality

In this section, we present the linear quadratic regulator (LQR). We show how to calculate it and in which sense the LQR is the dual of the Kalman filter presented in Section 2.3. In Chapter 3, we will use this duality to show how the optimal intermittent Kalman filter is related to the optimal intermittent LQR. The presentation of this section is inspired by [SYL19, Section 2].

We consider the following linear system,

$$\begin{aligned} x(t+1) &= \tilde{A}(t)x(t) + \tilde{B}(t)u(t), \\ x(0) &= \tilde{x}_0, \end{aligned}$$

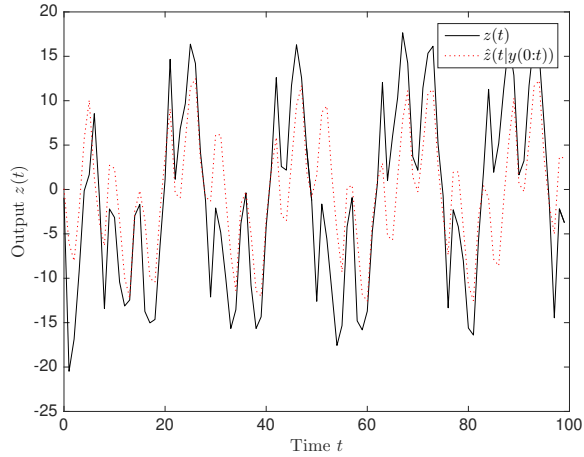



**Fig. 2.2** Trajectory $z(t)$ for the nonlinear dynamical system described by (2.13), and its MSE estimate $\hat{z}(t|t)$ given by the particle filter.

and the goal is to choose the sequence of inputs $u(0), \ldots, u(T-1)$ such that the following quadratic optimality criterion is minimized:

$$\min_{u(\cdot)} \left\{ x(T)^\top \tilde{Q}_f x(T) + \sum_{t=0}^{T-1} x(t)^\top \tilde{Q}(t) x(t) + u(t)^\top \tilde{R}(t) u(t) \right\}.$$

The optimal control law is given by [SYL19, Lemma 2.2]

$$u(t) = -\tilde{K}(t)^\top \tilde{A}(t) x(t),$$

where the $\tilde{K}(t)$ are the LQR gains computed as

$$\tilde{K}(t) = \tilde{P}(t+1)\tilde{B}(t) \left[ \tilde{B}(t)^\top \tilde{P}(t+1)\tilde{B}(t) + \tilde{R}(t) \right]^{-1}, \tag{2.14}$$

and $\tilde{P}(t)$ is obtained recursively (backward in time) from the following discrete Riccati equation

$$\tilde{P}(t) = \tilde{A}(t)^\top \tilde{P}(t+1)\tilde{A}(t) + \tilde{Q}(t) - \tilde{A}(t)^\top \tilde{K}(t)\tilde{B}(t)^\top \tilde{P}(t+1)\tilde{A}(t), \tag{2.15}$$

and using the terminal condition

$$\tilde{P}(T) = \tilde{Q}_f. \tag{2.16}$$

### 2.5.1 The duality

To highlight the duality between Kalman filter and LQR, let us rewrite equation (2.4h) by substituting (2.4e):

$$\begin{aligned} P(t+1|t) &= A(I - K(t)C)P(t|t-1)A^\top + GQG^\top \\ &= AP(t|t-1)A^\top + GQG^\top - AK(t)CP(t|t-1)A^\top. \end{aligned} \tag{2.17}$$

Then, the duality lies in the following observation. The Kalman equations (2.4a), (2.4d) and (2.17), give respectively the LQR equations (2.16), (2.14) and (2.15) under the transformation

$$A(t) = \tilde{A}(T-t)^\top,\ Q(t) = \tilde{Q}(T-t),\ R(t) = \tilde{R}(T-t), \tag{2.18a}$$

$$G = I,\ C(t) = \tilde{B}(T-t)^\top,\ \bar{P}_0 = \tilde{Q}_f. \tag{2.18b}$$

Consequently, the following holds

$$\tilde{P}(T-t) = P(t|t-1), \tag{2.19a}$$

$$\tilde{K}(T-t) = K(t). \tag{2.19b}$$

This shows that it is possible to convert an LQR problem instance into a Kalman estimation problem instance using (2.18). Then, we can solve the estimation problem, and deduce the solution of the LQR problem using (2.19).

The duality between optimal control and estimation has also been studied in the more general case of nonlinear and non-Gaussian dynamics [Tod08].

# 3

# Optimal intermittent Kalman filter

In this chapter, we address the problem of optimal measurement budget allocation to estimate the state of a linear discrete-time dynamical system over a finite horizon. More precisely, our aim is to select the measurement times in order to minimize the variance of the estimation error over a finite horizon. In addition, we investigate the closely related problem of finding a trade-off between the number of measurements and the signal to noise ratio.

First, the optimal measurement budget allocation problem is reduced to a deterministic combinatorial program. Then, we propose a genetic algorithm implementing a count preserving crossover to solve it. On the theoretical side, we provide a one-dimensional analysis that indicates that the benefit of using irregular measurements grows when the system is unstable or when the process noise becomes important. Then, using the duality between estimation and control, we show that the problem of selecting optimal control times for a linear quadratic regulator (LQR) can be reduced to our initial problem.

Finally, numerical implementations demonstrate that using measurement times optimized by our genetic algorithm gives better estimate than regularly spaced measurements. Our method is applied to a discrete version of a continuous-time system and the impact of the discretization time

step is studied. It reveals good convergence properties, showing that our method is well suited to both continuous-time and discrete-time setups.

This chapter assumes basic knowledge of the Kalman filter (see Section 2.3) and of the duality between estimation and control (see Section 2.5).

The results presented in this chapter have been reported in

- Antoine Aspeel, Axel Legay, Raphaël M Jungers, and Benoit Macq. Optimal measurement budget allocation for Kalman prediction over a finite time horizon by genetic algorithms. *EURASIP Journal on Advances in Signal Processing*, 2021(1):1–24, 2021.

## 3.1 Introduction

In most applications of the Kalman filter, the measurements are acquired in a regular way, i.e., they are equally spaced in time. The claim of this chapter is that when the number of measurements is restricted, Kalman prediction will provide better results if the measurements can be selected at the optimal times instead of a regular sampling. This chapter addresses the problem of selecting the optimal measurement times to predict the state of a stochastic linear dynamical system from noisy measurements, over a finite time horizon, when the number of measurements is fixed. The optimal measurement times are obtained by minimizing the mean prediction error variance over the complete horizon. In this chapter, we focus on prediction instead of filtering in order to address real-time applications. Indeed, in our targeted applications, prediction allows real-time tracking by compensating for the measurement delay. However, the proposed method can be adapted straightforwardly for filtering applications.

The optimal measurement times can be formulated as the solution of a combinatorial optimization problem. Evolutionary algorithms are a potential effective approach to solve that kind of problem [KBH94]. We propose different variants of genetic algorithms (GA) to solve this problem and demonstrate their efficiency compared to a random trial approach. GA provides an effective approach because they are able to sample a population broadly.

### 3.1.1 Contribution

Despite the attention paid to the related problem of sensor scheduling (see Section 1.1.2 for a detailed state of the art), the problem of selecting a given

number of measurement times to minimize the average over a finite horizon of the estimation error variance is, to the best of our knowledge, not resolved in the multivariate case. This chapter covers that topic. The work we consider as the closest to the content of this chapter is that of Vitus *et al.* published in 2012 [VZA+12]. It tackles the discrete time sensor scheduling problem in the linear and Gaussian case, without considering any sensing cost. Their objective is to choose at each time step which sensor to activate among a set of available sensors, while our objective is to choose when to measure.

To summarize, the contributions presented in this chapter are the following: (i) Three different GAs are proposed and compared to efficiently solve the combinatorial optimization problem; (ii) an analysis of the impact of the model (stability, process noise and measurement noise variances) on the obtained solutions in the one-dimensional case is proposed and interpreted; (iii) links are illustrated between the solutions of the problem and the rank of the observability matrix of the system; (iv) the discretization of a continuous-time system is considered and numerical experiments show that our method is well-suited to both discrete-time and continuous-time frameworks; (v) a related problem concerning the trade-off between quality and quantity of measurements is mathematically formalized and illustrated by an example; and (vi) the optimal intermittent LQR problem is handled through the duality between estimation and control.

### 3.1.2 Chapter organization

The rest of this chapter is organized as follows: Section 3.2 formalizes the problem mathematically (Sections 3.2.1 to 3.2.3) and presents different algorithms to solve it (Section 3.2.4). Then, a continuous-time version of the problem is presented (Section 3.2.5). The problem of optimal intermittent LQR is addressed and we show that it can be solved by reduction to the problem of the optimal intermittent Kalman predictor (Section 3.2.6). Finally, a related problem concerning the compromise between the quality and the quantity of measurements is presented (Section 3.2.7).

Section 3.3 compares the different proposed algorithms (Section 3.3.1); numerical examples are extensively studied (Section 3.3.2); links with the observability matrix are illustrated (Section 3.3.3); a continuous-time example is presented with a particular focus on the impact of the discretization time step (Section 3.3.4); then, the impact of the system characteristics (stability, variances of the process noise and the measurement noise) on the

obtained solution is illustrated (Section 3.3.5); and finally, the compromise between quality and quantity is illustrated by an example (Section 3.3.6).

Finally, Section 3.4 presents our conclusions and proposes further avenues of work.

### 3.1.3 Notation

The set of real numbers and the empty set are denoted $\mathbb{R}$ and $\emptyset$, respectively. $I_n$ and $0_{m\times n}$ denote the identity matrix of $\mathbb{R}^{n\times n}$ and the null matrix of $\mathbb{R}^{m\times n}$, respectively. The dimensions will sometimes be omitted when they are non-ambiguous.

Matrices are noted in upper case, e.g., $X$, and vectors in lower case, e.g., $x$. The transpose of a matrix $X$ (or of a vector $x$), is noted $X^\top$ (or $x^\top$). If $X$ is a square matrix, its inverse (if invertible) and its trace are noted $X^{-1}$ and $\mathrm{Tr}[X]$, respectively. The Euclidean norm of a vector $x$ is noted $\|x\|$. If $\Omega$ is a set, $|\Omega|$ is its cardinality, while if $a \in \mathbb{R}$ is a scalar, $|a|$ is its absolute value.

A Gaussian distribution of mean $\mu \in \mathbb{R}^n$ and positive semi-definite covariance matrix $\Sigma \in \mathbb{R}^{n\times n}$ is noted $\mathcal{N}(\mu, \Sigma)$. In addition, $x \sim \mathcal{N}(\mu, \Sigma)$ indicates that the vector $x \in \mathbb{R}^n$ is distributed according to $\mathcal{N}(\mu, \Sigma)$. Finally, $\mathbb{E}[x]$ is the expectation of a random variable $x$.

## 3.2 Materials and methods

### 3.2.1 Problem description

We consider the discrete time framework: $t = 0, \ldots, T$. The set of measurement times is denoted $\mathcal{M} \subset \{0, \ldots, T-1\}$ and is constrained to contain only $N$ measurements, i.e., $|\mathcal{M}| = N$ with $N \leq T$. The dynamics, the measurements, and the quantity to estimate are described by the following equations,

$$x(t+1) = Ax(t) + b + Gw(t) \quad t = 0, \ldots, T-1, \tag{3.1a}$$
$$y(t) = Cx(t) + d + v(t) \quad t \in \mathcal{M}, \tag{3.1b}$$
$$z(t) = Dx(t) \quad t = 0, \ldots, T, \tag{3.1c}$$
$$x(0) \sim \mathcal{N}(\bar{x}_0, \bar{P}_0), \tag{3.1d}$$

where $w(t)$ and $v(t)$ are independent zero-mean white Gaussian noises with covariances $\mathbb{E}[w(t_1)w(t_2)^\top] = Q\delta(t_1 - t_2)$ and $\mathbb{E}[v(t_1)v(t_2)^\top] = R\delta(t_1 - t_2)$, and $\delta(\cdot)$ is Kronecker's delta. Column vectors $x(t)$, $y(t)$, $z(t)$

and $w(t)$ have sizes $n_x$, $n_y$, $n_z$ and $n_w$ respectively. The dimensions of other quantities are deduced from compatibility. Matrices and vectors $A$, $b$, $G$, $D$, $C$, $d$, $\bar{x}_0$, $\bar{P}_0$, $Q$ and $R$ are known and could be time-dependent. Equation (3.1a) describes the dynamic of internal state $x$, equation (3.1b) expresses the measurement $y$, and equation (3.1c) gives the quantity to estimate $z$. Note that (3.1b) holds only for $t \in \mathcal{M}$, where $\mathcal{M}$ is the set of measurement times. Relation (3.1d) indicates that the initial state follows a known normal distribution with mean $\bar{x}_0$ and covariance matrix $\bar{P}_0$.

Let us introduce the optimal *a priori* mean squared estimator of $z(t)$ according to $\mathcal{M}$, namely, $\hat{z}_{\mathcal{M}}(t|t-1) \coloneqq \mathbb{E}[z(t)|y(\tau) : \tau \in \mathcal{M}, \tau < t]$ (see Section 2.2 for a brief introduction to mean squared error estimation). The set $\mathcal{M}$ is chosen to minimize the variance of the prediction error norm averaged for each $t$ from 1 to $T$, it is

$$\min_{\mathcal{M} \subset \{0,\cdots,T-1\}} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left[\|z(t) - \hat{z}_{\mathcal{M}}(t|t-1)\|^2\right] \text{ subject to } |\mathcal{M}| = N, \quad (3.2)$$

where $\|\cdot\|$ is the Euclidean norm. Note that two measurements cannot be taken simultaneously. This restriction has a practical interest because in many applications there is only one measuring device.

### 3.2.2 Intermittent Kalman predictor

In this section, we show that problem (3.2) can be reformulated in a more explicit way, thanks to the Kalman formalism. The latter provides explicit formula to compute, for a given measurement set $\mathcal{M}$, the optimal mean squared estimator $\hat{x}_{\mathcal{M}}(t|t-1)$ of $x(t)$ from which one can compute $\hat{z}_{\mathcal{M}}(t|t-1)$.

Let us introduce $\hat{x}_{\mathcal{M}}(t|t-1) \coloneqq \mathbb{E}[x(t)|y(\tau) : \tau \in \mathcal{M}, \tau < t]$ as the optimal *a priori* mean squared estimator of $x(t)$ and $\hat{x}_{\mathcal{M}}(t|t) \coloneqq \mathbb{E}[x(t)|y(\tau) : \tau \in \mathcal{M}, \tau \leq t]$ as the optimal *a posteriori* mean squared estimator of $x(t)$. In addition, let us define the *a priori* and the *a posteriori* covariance matrices as $P(t|t-1) \coloneqq \mathbb{E}[(x(t) - \hat{x}_{\mathcal{M}}(t|t-1))(x(t) - \hat{x}_{\mathcal{M}}(t|t-1))^\top]$ and $P(t|t) \coloneqq \mathbb{E}[(x(t) - \hat{x}_{\mathcal{M}}(t|t))(x(t) - \hat{x}_{\mathcal{M}}(t|t))^\top]$, respectively. The classical Kalman filtering theory (see Section 2.3 for a reminder) states how to update these four quantities recursively in the case where a measurement is acquired at each time step, i.e., when $N = T$.

We consider the intermittent case by replacing the measurement matrix $C$ with the null matrix when no measurement is available, i.e., when

$t \notin \mathcal{M}$. In view of equation (3.1b), it models the fact that $y(t)$ does not contain any information about the state $x(t)$.[1] Then the equations of the intermittent Kalman predictor are the *time update* equations

$$P(t+1|t) = AP(t|t)A^\top + GQG^\top, \tag{3.3a}$$
$$\hat{x}_\mathcal{M}(t+1|t) = A\hat{x}_\mathcal{M}(t|t) + b, \tag{3.3b}$$
$$\hat{z}_\mathcal{M}(t+1|t) = D\hat{x}_\mathcal{M}(t+1|t), \tag{3.3c}$$

for $t = 0, \dots, T-1$. In addition, the *measurement update* equations are

$$K(t) = \begin{cases} P(t|t-1)C^\top[CP(t|t-1)C^\top + R]^{-1} & \text{if } t \in \mathcal{M} \\ 0 & \text{else,} \end{cases} \tag{3.4a}$$
$$P(t|t) = (I - K(t)C)P(t|t-1), \tag{3.4b}$$
$$\hat{x}_\mathcal{M}(t|t) = \hat{x}_\mathcal{M}(t|t-1) + K(t)[y(t) - C\hat{x}_\mathcal{M}(t|t-1) - d], \tag{3.4c}$$
$$\hat{z}_\mathcal{M}(t|t) = D\hat{x}_\mathcal{M}(t|t), \tag{3.4d}$$

for $t = 0, \dots, T$. Note that when $t \notin \mathcal{M}$, $y(t)$ is not defined in equation (3.4c) but can be set to any arbitrary value because it is multiplied by $K(t) = 0$. In addition, one can see from (3.4a)-(3.4c) that when no measurement is available, i.e., when $t \notin \mathcal{M}$, each *a posteriori* quantity is simply the *a priori* one, which could have been expected intuitively. The *initialization* of these recurrence equations are

$$P(0|-1) = \bar{P}_0, \quad \hat{x}_\mathcal{M}(0|-1) = \bar{x}_0, \quad \hat{z}_\mathcal{M}(0|-1) = D\bar{x}_0. \tag{3.5}$$

The intermittent Kalman predictor is summarized by equations (3.3) to (3.5).

A *regular Kalman predictor* is a Kalman predictor for which the $N$ measurement times are selected as equally spaced as possible. It is

$$\mathcal{M}_\text{REG} := \left\{ \text{Round}\left[\frac{kT}{N}\right] \middle| \, k = 0, \dots, N-1 \right\}, \tag{3.6}$$

where Round$[\cdot]$ is the rounding operator.

[1] In Sinopoli *et al.* [SSF+04], the intermittent case is considered by replacing the covariance matrix of the measurement noise $R$ by $\lim_{\mu\to\infty} \mu I$ when no measurement is available. This is driven by the fact that an infinite noise measurement is equivalent to no measurement.

### 3.2.3 Optimal intermittent Kalman predictor

The optimal intermittent Kalman predictor is the intermittent Kalman predictor for which the set of measurement times $\mathcal{M}$ is chosen to minimize the mean error variance, i.e., it is the solution of problem (3.2). Note that a regular Kalman predictor is an intermittent Kalman predictor as soon as $N < T$.

Let us denote the *a priori* error $\tilde{z}(t) := z(t) - \hat{z}_{\mathcal{M}}(t|t-1)$. Its covariance matrix can be written

$$\begin{aligned} S(t) &:= \mathbb{E}\left[\tilde{z}(t)\tilde{z}(t)^\top\right] \\ &= \mathbb{E}\left[(z(t) - \hat{z}_{\mathcal{M}}(t|t-1))(z(t) - \hat{z}_{\mathcal{M}}(t|t-1))^\top\right] \\ &= \mathbb{E}\left[(Dx(t) - D\hat{x}_{\mathcal{M}}(t|t-1))(Dx(t) - D\hat{x}_{\mathcal{M}}(t|t-1))^\top\right] \\ &= D\mathbb{E}\left[(x(t) - \hat{x}_{\mathcal{M}}(t|t-1))(x(t) - \hat{x}_{\mathcal{M}}(t|t-1))^\top\right] D^\top \\ &= DP(t|t-1)D^\top. \end{aligned}$$

Then, the variance of the prediction error norm $\|\tilde{z}(t)\|$ can be written in terms of $P(t|t-1)$ as

$$\mathbb{E}[\|\tilde{z}(t)\|^2] = \mathrm{Tr}[S(t)] = \mathrm{Tr}[DP(t|t-1)D^\top],$$

where $\mathrm{Tr}[\cdot]$ is the trace operator.

Thanks to the previous equations, problem (3.2) can be reformulated as

$$\begin{aligned} &\min_{\mathcal{M}\subset\{0,\dots,T-1\}} \frac{1}{T}\sum_{t=1}^{T} \mathrm{Tr}[DP(t|t-1)D^\top] \text{ subject to} \\ &|\mathcal{M}| = N, \text{equations (3.3a), (3.4a) and (3.4b) with } P(0|-1) = \bar{P}_0, \end{aligned} \tag{3.7}$$

where constraint (3.3a) holds for $t = 0, \dots, T-1$ and constraints (3.4a) and (3.4b) hold for $t = 0, \dots, T$.

*Remark* 3.1. The equations that govern covariance matrices, i.e., equations (3.3a), (3.4a) and (3.4b), are independent of the measurements $y(t)$. Consequently, the optimization problem (3.7) can be solved before measurements are made, i.e., offline. A similar observation is made in [MPD67]. In addition, note that these quantities are also independent of $b$ and $d$. ◇

*Remark* 3.2. Contrary to equations (3.1), which are stochastic, problem (3.7) is deterministic.

In addition, the equations involving $\hat{x}_{\mathcal{M}}$ and $\hat{z}_{\mathcal{M}}$ can be ignored during selection of the measurement times, i.e., the resolution of problem (3.7), and used only for prediction. Consequently, using the optimal intermittent Kalman predictor consists in firstly computing the optimal measurement times offline and then, doing online prediction. ◇

*Remark* 3.3. Note that $x(t)$ and $z(t)$ are linearly related by relation (3.1c). However an optimal set of measurement times for estimating $z(1), \ldots, z(T)$, i.e., an optimal solution of problem (3.7), is not necessarily optimal for estimating $x(1), \ldots, x(T)$. Indeed, the objective function of the problem depends on the matrix $D$ that connects $z(t)$ to $x(t)$. ◇

*Remark* 3.4. It is possible to compute the distribution of the squared norm of the prediction error $\|\tilde{z}(t)\|^2$. First, consider the following eigendecomposition, $S(t) = \Sigma\Lambda\Sigma^\top$, where $\Lambda$ is a diagonal matrix whose diagonal entries are the eigenvalues $\lambda_1(t), \ldots, \lambda_{n_z}(t)$ of the symmetric matrix $S(t) = DP(t|t)D^\top$, and where $\Sigma$ is a unitary matrix, i.e., $\Sigma^\top\Sigma = I$.

Define $\zeta := \Lambda^{-1/2}\Sigma^\top\tilde{z}(t)$. Note that $\Sigma$, $\Lambda$ and $\zeta$ depend on $t$ but this dependency is omitted for clarity. Each random variable $\zeta$ follows a standard normal distribution $\mathcal{N}(0, I)$. Indeed, it is a linear transformation of a centered Gaussian random variable so it is also a central Gaussian random variable. In addition, its covariance matrix is identity. It can be computed using the commutativity of diagonal matrices and the fact that $\Sigma$ is unitary, so that we obtain,

$$\begin{aligned}
\mathbb{E}\left[\zeta\zeta^\top\right] &= \mathbb{E}\left[\left(\Lambda^{-1/2}\Sigma^\top\tilde{z}(t)\right)\left(\Lambda^{-1/2}\Sigma^\top\tilde{z}(t)\right)^\top\right] \\
&= \Lambda^{-1/2}\Sigma^\top\mathbb{E}\left[\tilde{z}(t)\tilde{z}(t)^\top\right]\Sigma\Lambda^{-1/2} \\
&= \Lambda^{-1/2}\Sigma^\top\Sigma\Lambda\Sigma^\top\Sigma\Lambda^{-1/2} \\
&= I.
\end{aligned}$$

The random variable $\|\tilde{z}(t)\|^2$ can be expressed in terms of $\zeta$, yielding

$$\begin{aligned}\|\tilde{z}(t)\|^2 &= \tilde{z}(t)^\top \tilde{z}(t) \\ &= (\Sigma\Lambda^{1/2}\zeta)^\top(\Sigma\Lambda^{1/2}\zeta) \\ &= \zeta^\top\Lambda^{1/2}\Sigma^\top\Sigma\Lambda^{1/2}\zeta \\ &= \zeta^\top\Lambda\zeta \\ &= \sum_{i=1}^{n}\lambda_i(t)\zeta_i^2,\end{aligned}$$

where all $\zeta_i^2$ follow an independent chi-squared distribution with one degree of freedom, i.e., $\zeta_i^2 \sim \chi^2$. It shows that $\|\tilde{z}(t)\|^2$ is a linear combination with non-negative coefficients of independent random variables following a chi-squared distribution with one degree of freedom. Then, a closed form formula can be obtained for the cumulative distribution function $\mathbb{P}[\|\tilde{z}(t)\|^2 < \xi]$ using the theorem of [Fle71]. From a practical point of view, this quantity can be efficiently estimated using the method proposed in [Bau13]. ◇

### 3.2.4 Optimization algorithms

Solving the combinatorial optimization problem (3.7) by means of an exhaustive search would require evaluating $T!/(N!(T-N)!)$ times the cost function, which is computationally intractable. In this section, we propose a random trial (RT) algorithm and various genetic algorithms (GAs) to tackle this problem.

The RT algorithm randomly samples a given number of admissible solutions $\mathcal{M}$, computes their cost and keeps the best one.

In the GA nomenclature, a feasible solution $\mathcal{M} = \{t_1, \ldots, t_N\}$ of the optimization problem is called an individual and its corresponding measurement times $t_i$ are called its genes. A set of several individuals is called a population.

The GA [Mit98] implements the following steps.

1. **Initialization:** An initial population is uniformly sampled on the admissible set of problem (3.7).
2. **Evaluation:** For each individual in the population, the cost is evaluated according to the objective function of problem (3.7).

3. **Selection:** Individuals with low costs are preferably selected according to stochastic universal sampling [Mit98].
4. **Crossover:** Selected individuals are mixed to produce new individuals called the offspring. This reconstitutes a complete population. Three different types of crossover are considered: the shuffle crossover (SC), replace crossover (RC) and count preserving crossover (CPC). They are described below.
5. **Mutation:** Each gene of each individual mutates according to a probability. When a mutation occurs, some measurement times are replaced by random times. This step could create duplicates, i.e., an individual could have repeated times $t_i = t_j$ with $i \neq j$. To avoid this situation, the random times are selected uniformly in $\{0, \dots, T-1\} \backslash \mathcal{M}$.
6. **Repeat:** Come back to step 2 until a convergence criterion is reached.

One pass of steps 2) to 5) is called a generation.

Let us present the three different crossover operators. Each of them gives a variant of the genetic algorithm. They are described and a brief example is given.

**Shuffle crossover (SC)** For each parent pair, the SC [US15] method picks the genes to exchange (each gene has the same probability to be selected) randomly. For example, let two parents (that is, two sets of measurement times $\mathcal{M}$),

$$P_1 = 0\,\underline{1}\,3\,\underline{5}\,\underline{6}\,7 \text{ and } P_2 = 0\,\underline{1}\,2\,\underline{3}\,\underline{5}\,8, \tag{3.8}$$

where _ indicates the location of genes selected for the crossover. The obtained offspring are

$$O_1 = 0\,1\,3\,3\,5\,7 \text{ and } O_2 = 0\,1\,2\,5\,6\,8. \tag{3.9}$$

First, one can observe that $O_1$ contains a duplication of 3, which corresponds to choosing the same measurement time twice. That is not allowed in problem (3.7) and has to be considered a wasted measurement. In other words, fewer measurements are acquired, which is suboptimal, because adding a new distinct measurement can only decrease the cost.

A second observation is that the second offspring $O_2$ does not contain a 3, while both parents $P_1$ and $P_2$ contain one. That means that common

heritage is lost during the crossover, which possibly deteriorates the convergence of the algorithm.

These two observations motivate a careful choice of the crossover method and leads naturally to the next two crossover methods.

**Replace crossover (RC)** The RC method is an SC followed by a random replacement of duplicates, taking care to avoid new duplicates. Duplicates are replaced by picking genes uniformly at random in $\{0, \dots, T-1\} \backslash \mathcal{M}$. The offspring (3.9) become, for example

$$O_1 = 0\,1\,3\,5\,7\,\underline{8} \text{ and } O_2 = 0\,1\,2\,5\,6\,8,$$

where $\underline{\ }$ indicates the new gene. Note that the second offspring remains unchanged because it does not contain duplicates.

With this crossover, the obtained offspring have no duplicates. However, the second offspring $O_2$ does not contain a 3 while this gene was common to both parents. This illustrates that, like SC, this RC can also suffer from a loss of common heritage.

**Count preserving crossover (CPC)** The CPC [US15, HK93] represents each individual by a set (unordered) instead of a vector (ordered). It implements an SC between subsets $P_1 \backslash P_2$ and $P_2 \backslash P_1$. It allows transmission of the common measurement times $P_1 \cap P_2$ to both offspring. For parents in example (3.8), the SC is executed between subsets $P_1 \backslash P_2 = \{6, 7\}$ and $P_2 \backslash P_1 = \{2, 8\}$ (see Figure 3.1). Possible offspring are then

$$O_1 = 0\,1\,\underline{2}\,3\,5\,6 \text{ and } O_2 = 0\,1\,3\,5\,\underline{7}\,8. \tag{3.10}$$

This crossover method makes it possible both to transmit the common genes to all the offspring and to avoid duplicates.

All GA implementations use sigma scaling [Mit98] with the sigma factor equal to 1 (standard value). The crossover operation is applied systematically and mutation occurs with a probability by gene of 0.003 (standard value). The population size is set to 100 individuals and the algorithm stops after 100 generations.

### 3.2.5 Discretization of a continuous-time system

Many real-world problems are modelled as continuous-time systems. For this reason, we study the continuous-time equivalent of problem (3.2). It

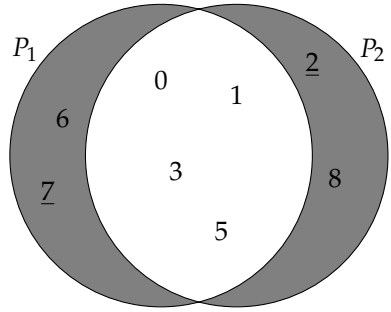



**Fig. 3.1** Venn diagram representing the count preserving crossover on parents $P_1$ and $P_2$ described in (3.8). The crossover allows gene exchanges between the two filled regions. The underlined genes are exchanged, the result is given in (3.10).

is[2],

$$\inf_{0\leq\tau_0<\tau_1<\cdots<\tau_{N-1}\leq\bar{\tau}} \frac{1}{\bar{\tau}}\int_0^{\bar{\tau}} \mathbb{E}[\|z(\tau')-\hat{z}(\tau')\|^2]d\tau', \tag{3.11a}$$

such that

$$\begin{aligned} \frac{dx}{d\tau}(\tau) &= A^c x(\tau) + b^c + G^c w^c(\tau) && \tau\in[0,\bar{\tau}],\\ y(\tau_k) &= Cx(\tau_k) + d + v(\tau_k) && k=0,\dots,N-1,\\ z(\tau) &= D^c x(\tau) && \tau\in[0,\bar{\tau}],\\ x(0) &\sim \mathcal{N}(\bar{x}_0,\bar{P}_0), \end{aligned} \tag{3.11b}$$

where $w^c(\tau)\sim\mathcal{N}(0,Q^c)$ and $v(\tau_k)\sim\mathcal{N}(0,R)$ are independent Gaussian white noises. Finally, $\hat{z}(\tau)=\mathbb{E}[z(\tau)|y(\tau_k):\tau_k<\tau]$ is the optimal mean squared estimator of $z(\tau)$. The exponent $^c$ emphasizes that time is continuous. Note that this is a stochastic differential equation for which derivatives have a non-classical sense. The reader is referred to [Oks13] for details.

This problem can be exactly discretized at a time step $\delta=\bar{\tau}/T$ to fit the formalism of equations (3.1) and (3.2). The corresponding quantities are

[2] An infimum is used because the admissible set of the optimization problem is not compact due to constraint $\tau_k<\tau_{k+1}$. This constraint models the fact that two measurements can not be acquired at the same time. When the time is discretized at time step $\delta$, this constraint becomes $\tau_k+\delta\leq\tau_{k+1}$, which makes the admissible set compact. Then, the corresponding infimum exists and is a minimum.

[DL71],

$$A = e^{A^c \delta},\ D = D^c,\ b = \int_0^\delta e^{A^c \tau} d\tau b^c, \tag{3.12a}$$

$$Q = \int_0^\delta e^{A^c \tau} G^c Q^c (G^c)^\top e^{(A^c)^\top \tau} d\tau,\ G = 1, \tag{3.12b}$$

where $e^{\cdot}$ is the matrix exponential operator.

In Section 3.3.4, we study the impact of the discretization time-step on the obtained solution.

### 3.2.6 Optimal intermittent linear quadratic regulator

Duality between estimation and control problems has been known for decades [SYL19]. Roughly, it states that instead of solving a control problem directly, one can solve a related estimation problem and deduce the solution to the control problem from the solution of the estimation problem. The converse is also possible. For an introduction to duality in the case of linear dynamical systems, see Section 2.5.

In this section, we formalize the problem of optimal intermittent LQR and use duality to show that it can be handled by computing an optimal intermittent Kalman predictor.

Let us consider a controlled linear system with known initial state,

$$x(t+1) = \tilde{A}x(t) + \tilde{\sigma}(t)\tilde{B}u(t),\ \text{for } t = 0, \dots, T-1, \tag{3.13a}$$

$$x(0) = \tilde{x}_0, \tag{3.13b}$$

where $\tilde{\sigma}(t) \in \{0, 1\}$. This control signal $\tilde{\sigma}(\cdot)$ determines the times at which the system is controlled. Note that $\tilde{A}$ and $\tilde{B}$ can be time-dependent.

The *optimal intermittent LQR problem* consists of the following minimization,

$$\tilde{\sigma}^*(\cdot) = \arg\min_{\tilde{\sigma}(\cdot)} V(\tilde{\sigma}(\cdot)) \text{ such that } \sum_{t=0}^{T-1} \sigma(t) = N, \tag{3.14}$$

with

$$V(\tilde{\sigma}(\cdot)) = \min_{u(\cdot)} \left\{ x(T)^\top \tilde{Q}_f x(T) + \sum_{t=0}^{T-1} x(t)^\top \tilde{Q} x(t) + u(t)^\top \tilde{R} u(t) \right\}$$

subject to (3.13),

where $\tilde{Q}$ and $\tilde{R}$ can be time-dependent. This problem has been studied in [SYC12] where a closed form formula is provided under some restrictive conditions.

The following theorem formalizes how to compute this optimal LQR by computing an optimal intermittent Kalman predictor. Note that at each optimal set of measurement times $\mathcal{M}$ can be associated with a signal $\sigma(\cdot)$ such that $\sigma(t) = 1$ if $t \in \mathcal{M}$ and 0 otherwise.

**Theorem 3.5.** *The optimal solution $\tilde{\sigma}^*(\cdot)$ of the problem (3.14) is given by $\tilde{\sigma}^*(t) = \sigma^*(T-t)$ where $\sigma^*(\cdot)$ is the optimal solution of the estimation problem (3.7) solved for parameters*

$$A(t) = \tilde{A}(T-t)^\top,\ Q(t) = \tilde{Q}(T-t),\ R(t) = \tilde{R}(T-t), \tag{3.15a}$$

$$b(t) = 0,\ G = I,\ C(t) = \tilde{B}(T-t)^\top,\ \bar{P}_0 = \tilde{Q}_f,\ d(t) = 0,\ \textit{and} \tag{3.15b}$$

$$D(t) = \delta(T-t)\tilde{x}_0^\top, \tag{3.15c}$$

*where $\delta(T-t)$ is 1 if $t = T$, and 0 otherwise.*

*In addition, the optimal control law is*

$$u(t) = -K_{\sigma^*(\cdot)}(t)^\top A(t)^\top x(t), \tag{3.16}$$

*where $K_{\sigma^*(\cdot)}(t)$ is given by (3.4a).*

*Proof.* See Appendix A. □

### 3.2.7 Quality or quantity of measurements

Instead of considering a fixed number of measurements, an alternative problem is to reach a trade-off between many low-quality measurements and a small number of high-quality measurements. This problem can be handled in the presented formalism.

Allowing the choice of the quantity of measurements corresponds to making the number of measurements $N$ a variable to optimize. However, with only this modification, the optimal solution will always be to take a maximum number of measurements, i.e., $N = T$. To model the compromise between measurement quality and quantity, an increase in the number of measurements $N$ implies an increase in the noise of measurements, i.e., an increase in $R$.

In other words, the measurement covariance matrix $R$ must depends on $N$. We assume $R = f(N)$ for a given function $f : \{1, \dots, T\} \to \mathcal{S}_+^p$

where $\mathcal{S}_+^p$ is the set of $p \times p$ positive semi-definite matrices, and $N_1 \leq N_2$ implies $f(N_1) \preceq f(N_2)$. Then, the problem can be formalized similarly to (3.7), it gives,

$$\min_{1\leq N\leq T} \min_{\mathcal{M}\subset\{0,\dots,T-1\}} \frac{1}{T}\sum_{t=1}^{T} \mathrm{Tr}[DP(t|t-1)D^\top] \text{ subject to}$$
$$|\mathcal{M}| = N, \text{equations (3.3a), (3.4a) and (3.4b)}, P(0|-1) = \bar{P}_0 \text{ and } R = f(N), \tag{3.17}$$

where, as previously, constraint (3.3a) holds for $t = 0, \dots, T-1$ and constraints (3.4a) and (3.4b) hold for $t = 0, \dots, T$.

This problem is studied in Section 3.3.6.

## 3.3 Numerical experiments and discussions

Several results will be illustrated on a discretized spring-mass system. The derivation of these equations will be detailed in Section 3.3.4. For readability, we anticipate the system finally obtained; it is defined by

$$A = \begin{bmatrix} \cos\delta & \sin\delta \\ -\sin\delta & \cos\delta \end{bmatrix}, \; C = D = \begin{bmatrix} 1 & 0 \end{bmatrix}, \; b = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \; G = 1, \tag{3.18a}$$

$$Q = \frac{1}{3200}\begin{bmatrix} \delta - \sin\delta\cos\delta & \sin^2\delta \\ \sin^2\delta & \delta + \sin\delta\cos\delta \end{bmatrix}, \tag{3.18b}$$

where $\delta$ is the discretization time step.

### 3.3.1 Comparison of optimization methods

In this section, the different optimization algorithms presented in Section 3.2.4 are compared. The random trial (RT) method and the three variants of Genetic Algorithms (GA) are compared, i.e., Shuffle Crossover (SC), Replace Crossover (RC) and Count Preserving Crossover (CPC).

Figure 3.2 presents the mean cost and the minimum cost obtained by the four algorithms with respect to the number of cost function evaluations in the case of the spring-mass system described in Section 3.3.4 by equations (3.18) with a discretization time step $\delta = 0.1$ $[s]$. The number of measurements is set to $N = 70$ and the number of time steps is $T = 100$. The mean and minimum costs obtained with RT are also printed. In addition, the cost for regularly spaced measurement times is also printed.

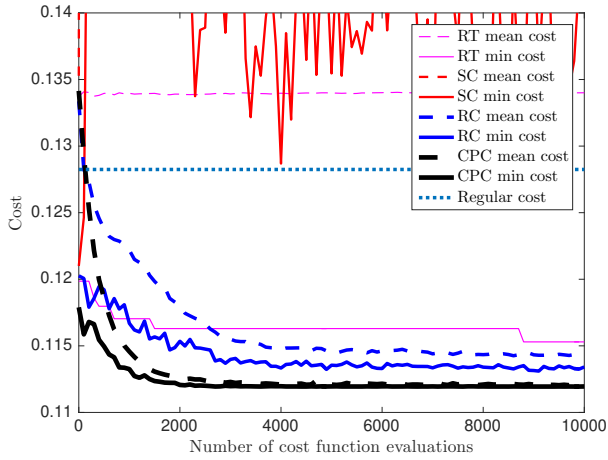



**Fig. 3.2** Comparison of the mean and minimum (min) cost for the random trial (RT) method, the genetic algorithm (GA) with shuffle crossover (SC), the GA with replace crossover (RC), the GA with count preserving crossover (CPC) and the regular Kalman predictor (regular cost) with respect to the number of cost function evaluations. For the genetic algorithms, one generation corresponds to 100 cost function evaluations.

Firstly, one can observe that the minimum and average of the GA with SC has bad convergence behavior (its average cost quickly rises out of the figure). This confirms our previous comments in Section 3.2.4 about the flaws of this crossover operator (creation of duplicates and loss of common heritage). All other algorithms found a solution better than the regular one in few generations. The regular cost is close to the average RT cost, meaning that regular measurement times are 'typical' random measurement times in this example. The GAs with RC and CPC outperform the RT algorithm. Finally, the GA with CPC quickly outperforms all other proposed algorithms. In addition, the average cost converges to the minimal cost only for the CPC. This means that the entire population converges to a single individual -assumed to be optimal-, which is the desired behavior for a GA. Table 3.1 prints the averages and standard deviations of the optimal costs found by the different algorithms over 100 runs. The advantage of the CPC crossover is confirmed and the small standard deviations indicate the reproducibility of the results.

In the following, all experiments use the GA that implements CPC.

**Table 3.1** Average and standard deviation of the optimal costs found by the different algorithms on 100 runs. The algorithms are the random trial (RT) method, the genetic algorithm (GA) with shuffle crossover (SC), the GA with replace crossover (RC), the GA with count preserving crossover (CPC). The cost obtained with regularly spaced measurements (see (3.6)) is also indicated.

| RT | SC | RC | CPC | Regular |
|---|---|---|---|---|
| $0.115 \pm 6 \cdot 10^{-4}$ | $0.120 \pm 10^{-3}$ | $0.113 \pm 10^{-4}$ | $0.112 \pm 2 \cdot 10^{-6}$ | 0.128 |

### 3.3.2 Prediction performance

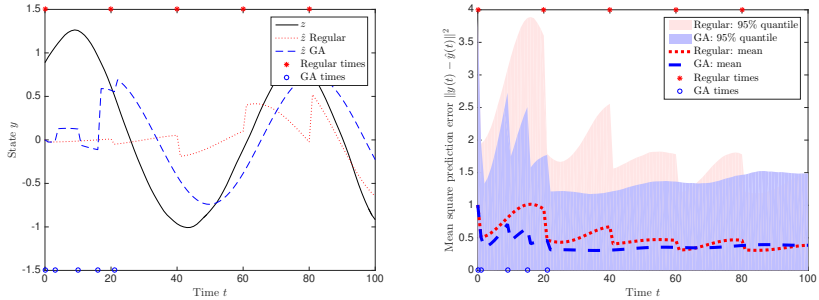
To continue the analysis of our method, predictions using our method are compared with the ones obtained with a Kalman predictor with regularly spaced measurement times (see equation (3.6)). After an analysis on the spring-mass system, we will apply our method to a 50-dimensional system. To conclude this section, some links with the observability matrix will be highlighted.

*Spring-mass system*

The set of measurement times found by the genetic algorithm is denoted $\mathcal{M}_{\text{GA}}$ and the set of regularly spaced measurements is denoted $\mathcal{M}_{\text{REG}}$. The corresponding mean squared tracking errors are denoted, $\text{MSE}(\mathcal{M}_{\text{GA}})$ and $\text{MSE}(\mathcal{M}_{\text{REG}})$, respectively.

For the same system as in the previous section, i.e., equations (3.18) with a discretization time step $\delta = 0.1$ $[s]$ and with $T = 100$ time steps and $N = 5$ measurements, Figure 3.3a presents the signal $z(t)$ and the predictions obtained with the two predictors. In additions, the regular measurement times $\mathcal{M}_{\text{REG}}$ and the GA measurement times $\mathcal{M}_{\text{GA}}$ are printed. One can see that the main difference occurs at the beginning of the tracking. Indeed, our method gets closer faster. Intuitively, this can be understood by the fact that for the intermittent predictor, all measurement times are selected close to the beginning. Over the complete time horizon, the regular approach has a mean squared error of $\text{MSE}(\mathcal{M}_{\text{REG}}) = 0.46$ whereas the mean squared error with our method is $\text{MSE}(\mathcal{M}_{\text{GA}}) = 0.36$. This is a significant improvement. However, these results are valid only for this particular realization of the trajectory.

On the contrary, Figure 3.3b presents the squared error with respect to time over $100,000$ realizations. The mean and the 95% quantile are indicated for the regular Kalman predictor and for our optimal intermittent




**(a)** **(b)**

**Fig. 3.3** (a): Evolution of a particular realization of $z(t)$ and its estimates with both regular Kalman predictor (Regular) and optimal intermittent Kalman predictor (GA). (b): Mean squared prediction error with respect to time and 95% quantile over 100,000 realizations for the regular Kalman predictor (Regular) and for our optimal intermittent Kalman prediction method (GA). Regular measurement times and the GA measurement times are also printed on both graphs. Simulations are realized on the system described by equations (3.18) with $T = 100$ time steps and $N = 5$ measurements.

Kalman predictor method. One can see that the remarks about Figure 3.3a hold for the mean and the 95% quantile behavior: our method rapidly produces a better estimate than the Kalman predictor with regular measurements.

Let us introduce the *benefit* as $\mathcal{B} = \text{MSE}(\mathcal{M}_{\text{REG}}) - \text{MSE}(\mathcal{M}_{\text{GA}})$. A positive benefit indicates that our method outperforms the regular Kalman predictor. Figure 3.4 presents the histogram of the benefit $\mathcal{B}$ computed over 100,000 realizations. It shows that the mean benefit is 0.12 and the benefit is positive in 64% of the cases, i.e., our method outperforms the regular Kalman predictor. The results are summarized in the second column of Table 3.2. Note that the optimal costs correspond to the average mean squared errors.

*A large dimensional system*

We propose here to apply our method to a 50-dimensional system. To this end, we generate a random system as follows: each entry of the matrix $A \in \mathbb{R}^{50\times 50}$ is picked independently at random from a Gaussian distribution with standard deviation 0.25. We consider $b = 0$ and $G = Q = I$. Ten com-

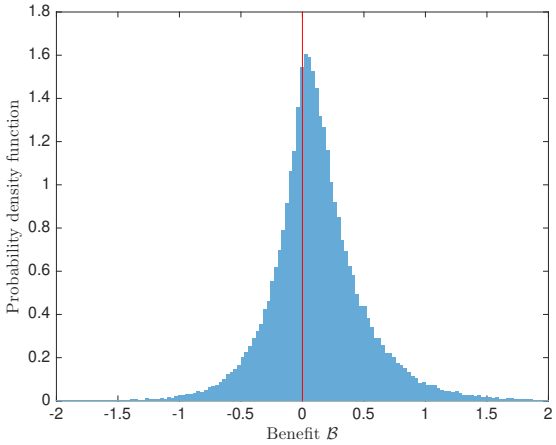
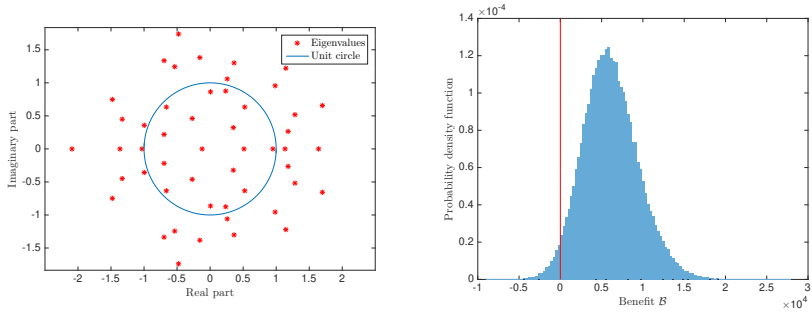



**Fig. 3.4** Histogram of the benefit $\mathcal{B} = \text{MSE}(\mathcal{M}_{\text{REG}}) - \text{MSE}(\mathcal{M}_{\text{GA}})$ computed over 100,000 realizations for system (3.18) with $T = 100$ time steps and $N = 5$ measurements. The histogram approximates the probability density function of the benefit. The red vertical line indicates a null benefit. The mean and standard deviation of the benefit are $0.12 \pm 0.38$ and the benefit is positive in 64% of the realizations.

**Table 3.2** Results for the spring-mass system (Section 3.3.2) and the random system (Section 3.3.2). The cost using regularly spaced measurements and the optimal cost found by the GA are indicated. The mean squared error are indicated (mean and standard deviation over 100,000 realizations). The benefit $\mathcal{B} = \text{MSE}(\mathcal{M}_{\text{REG}}) - \text{MSE}(\mathcal{M}_{\text{GA}})$ is also indicated (mean, standard deviation and proportion of positive).

| | Spring-mass system | Random system |
|---|---|---|
| Cost regular | 0.52 | 16817.80 |
| Cost GA | 0.39 | 10621.31 |
| $\text{MSE}(\mathcal{M}_{\text{REG}})$ (mean $\pm$ std) | $0.51 \pm 0.42$ | $16506.12 \pm 3212.83$ |
| $\text{MSE}(\mathcal{M}_{\text{GA}})$ (mean $\pm$ std) | $0.39 \pm 0.33$ | $10421.00 \pm 1866.53$ |
| Benefit $\mathcal{B}$ (mean $\pm$ std) | $0.12 \pm 0.38$ | $6085.12 \pm 3391.64$ |
| Proportion of positive benefit $\mathcal{B}$ | 64% | 97% |




**(a)** **(b)**

**Fig. 3.5** Outcomes of the experiment described in Section 3.3.2. (a): Eigenvalues of the randomly sampled matrix $A$. (b): Histogram of the benefit $\mathcal{B} = \text{MSE}(\mathcal{M}_{\text{REG}}) - \text{MSE}(\mathcal{M}_{\text{GA}})$ computed over 100,000 realizations for the problem described in Section 3.3.2. The histogram approximates the probability density function of the benefit. The red vertical line indicates a null benefit. The mean and standard deviation of the benefit are $6085.12 \pm 3391.64$ and the benefit is positive for 97% of the realizations.

ponents of the state $x(t)$ are uniformly picked at random to form the objective vector $y(t)$. In other words, each row of the matrix $D \in \{0,1\}^{10\times 50}$ contains only one 1 and all lines are different. The measurement matrix $C \in \{0,1\}^{10\times 50}$ is picked at random in the same way but independently of $D$. We consider $d = 0$ and the covariance matrix of the process noise is the identity, i.e., $R = I$. Finally, we consider $\bar{x}_0 = 0$ and $\bar{P}_0 = I$. This system is studied over $T = 50$ time steps and $N = 25$ measurements are allowed.

The eigenvalues of the randomly sampled matrix $A$ are depicted in Figure 3.5a. Among the 50 eigenvalues, 17 are stable and 33 are unstable. Their modulus varies from 0.12 to 2.09.

The optimal measurement times for this problem are computed using the GA with the CPC. Then, the mean squared prediction error using the optimal intermittent Kalman predictor is compared to the one with regular measurements on 100,000 realizations. The average and standard deviation of the mean squared error in the regular case $\text{MSE}(\mathcal{M}_{\text{REG}})$ are $16506.12 \pm 3212.83$. Using the optimal measurement times, the average and standard deviation of $\text{MSE}(\mathcal{M}_{\text{GA}})$ are $10421.00 \pm 1866.53$. The benefit $\mathcal{B} = \text{MSE}(\mathcal{M}_{\text{REG}}) - \text{MSE}(\mathcal{M}_{\text{GA}})$ is computed and its distribution is depicted in Figure 3.5b. The mean benefit is 6085.1 and the benefit is positive for 97% of the realizations. These results are summarized in the third

column of Table 3.2.

### 3.3.3 Links with observability matrix

To make the results more intuitive, let us present a simple system defined by

$$A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \; D = Q = \bar{P}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \; C = \begin{bmatrix} 1 & 0 \end{bmatrix}, \; R = G = 1, \; b = d = 0, \tag{3.19}$$

and consider $T = 20$ time steps and $N = 10$ measurements.

Jungers *et al.* show in [JKH17] that the observability matrix of a linear system with missing measurements is given by

$$\mathcal{O}_T(A, C, \mathcal{M}) = \begin{bmatrix} \sigma(0)C \\ \sigma(1)CA \\ \vdots \\ \sigma(T-1)CA^{T-1} \end{bmatrix},$$

where $\sigma(t) = 1$ if $t \in \mathcal{M}$ and 0 otherwise. For system (3.19), because the transition matrix $A$ is a 90 degree rotation matrix, the observability matrix has the following structure,

$$\mathcal{O}_T(A, C, \mathcal{M}) = \begin{bmatrix} \sigma(0) & 0 & -\sigma(2) & 0 & \sigma(4) & \cdots & 0 \\ 0 & -\sigma(1) & 0 & \sigma(3) & 0 & \cdots & \sigma(19) \end{bmatrix}^\top. \tag{3.20}$$

From definition (3.6), a regular measurement set is $\mathcal{M}_{\text{REG}} = \{0, 2, \ldots, 18\}$, which leads to the following observability matrix

$$\mathcal{O}_T(A, C, \mathcal{M}_{\text{REG}}) = \begin{bmatrix} 1 & 0 & -1 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}^\top. \tag{3.21}$$

With such regular set of measurement times, the observability matrix is rank deficient. That means that even in the absence of any noise, the state can not be completely determined.

Bearing in mind that the trace of a matrix is the sum of its eigenvalue, then the objective function of problem (3.7) can be rewritten $\sum_{t=1}^{T}(\lambda_1(t) + \lambda_2(t))/T$. Figure 3.6, presents the eigenvalues $\lambda_1(t) > \lambda_2(t)$ of the prediction error

covariance matrix $S(t) = DP(t|t-1)D^\top$ with respect to time. They are presented for regular measurements $\mathcal{M}_{\text{REG}}$, for the optimal measurements found by the genetic algorithm but also when a measurement is acquired at each time, i.e., $\mathcal{M} = \{0, \dots, T-1\}$ and when no measurement is acquired, i.e., $\mathcal{M} = \emptyset$. In addition, the measurement times are printed for both the regular and the intermittent cases.

One can observe that the largest eigenvalue $\lambda_1$ has the same evolution with regular measurements as when no measurement are used. This shows that the regular measurements have no effect on the largest eigenvalue $\lambda_1$. This observation is compatible with the rank deficiency of the observability matrix (3.21). On the contrary, in the case of the measurement times found by the genetic algorithm, both eigenvalues remain significantly smaller than the ones without measurements. In addition, the measurement times selected by the GA are composed of five pairs of successive measurements. The structure of the observability matrix (3.20) ensures that each line is non zero.

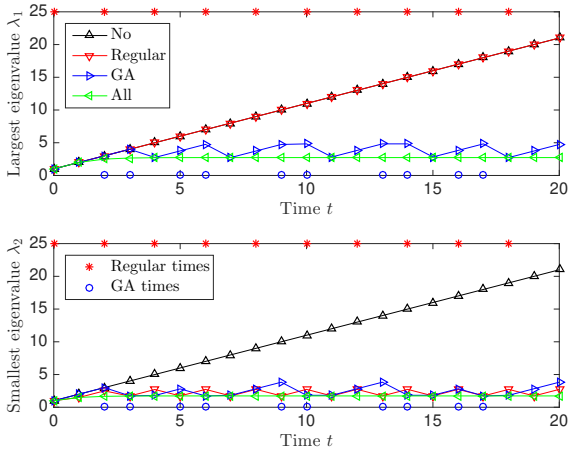



**Fig. 3.6** The two eigenvalues $\lambda_1 > \lambda_2$ of the covariance matrix of the prediction error $S(t)$ with respect to time $t$ when no measurement is acquired, i.e., $\mathcal{M} = \emptyset$ (No); when regular measurement times $\mathcal{M}_{\text{REG}}$ are used (Regular); when optimal measurement times $\mathcal{M}_{\text{GA}}$ are used (GA); and when there is measurement at each times, i.e., $\mathcal{M} = \{0, \dots, T-1\}$ (All). Regular measurement times and the GA measurement times are also printed. The considered system is described by (3.19) with $T = 20$ time steps and $N = 10$ measurements for the regular and the GA cases.

### 3.3.4 Continuous-time example

This section presents an example of the continuous-time case presented in Section 3.2.5. More precisely, we are interested in the impact of the number of discretization steps $T$ (or equivalently, the impact of the discretization step size $\delta$) on the obtained solution.

We want to solve the problem defined by equations (3.11) for the spring-mass system subject to a random force $w^c$,

$$\frac{d^2x}{d\tau^2}(\tau) = -\frac{k}{m}x(\tau) + \frac{1}{m}w^c(\tau) \text{ with } w^c(\tau) \sim \mathcal{N}(0,1), \tag{3.22a}$$

$$y(\tau_k) = x(\tau_k) + v(\tau_k) \text{ with } v(\tau_k) \sim \mathcal{N}(0,1), \tag{3.22b}$$

$$z(\tau) = x(\tau), \tag{3.22c}$$

$$x(0) \sim \mathcal{N}(0,1),\ \frac{dx}{d\tau}(0) \sim \mathcal{N}(0,1), \tag{3.22d}$$

where $m$ is the mass, expressed in $[kg]$ and $k$ is the stiffness of the spring, expressed in $[N/m]$. $x(\tau)$ is the elongation of the mass (in $[m]$) at time $\tau$ (in $[s]$). In the formalism of (3.11b), it corresponds to

$$A^c = \begin{bmatrix} 0 & 1 \\ -k/m & 0 \end{bmatrix},\ D^c = C = \begin{bmatrix} 1 & 0 \end{bmatrix},\ b^c = \bar{x}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix},\ G^c = \begin{bmatrix} 0 & 0 \\ 0 & 1/m \end{bmatrix},$$

$$Q^c = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix},\ \bar{P}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ d = 0,\ R = 1.$$

The corresponding discrete system can be obtained thanks to (3.12) and, in the particular case $k = m$, it gives equation (3.18a) and

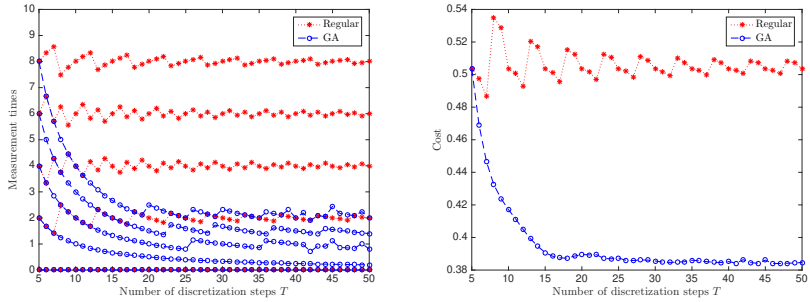
$$\begin{aligned} Q &= \int_0^\delta \frac{1}{m^2} \begin{bmatrix} \sin^2\tau & \sin\tau\cos\tau \\ \sin\tau\cos\tau & \cos^2\tau \end{bmatrix} d\tau \\ &= \frac{1}{2m^2} \begin{bmatrix} \delta - \sin\delta\cos\delta & \sin^2\delta \\ \sin^2\delta & \delta + \sin\delta\cos\delta \end{bmatrix}. \end{aligned}$$

We set $m = 40\ [kg]$ and $k = 40\ [N/m]$ and finally obtain equations (3.18), which had been anticipated at the beginning of Section 3.3.

In this section, the time horizon is $\bar{\tau} = 100\ [s]$ and the number of measurements is fixed to $N = 5$.

The measurement times are computed by solving the discretized problem using the GA. Then, the quality of the obtained solution is assessed

by looking at the cost function of the continuous-time problem (3.11). It is done thanks to the continuous-discrete Kalman filter [LXP17]. The results are presented in Figure 3.7. Figure 3.7a presents the measurement times found by the GA and the measurement times for the regular Kalman predictor.




**(a)** **(b)**

**Fig. 3.7** For the discrete-time system corresponding to the continuous system (3.22), graph (a) presents the regular measurement times $\mathcal{M}_{\text{REG}}$ given by (3.6) and the times $\mathcal{M}_{\text{GA}}$ found by the genetic algorithm (GA) with respect to the number of discretization steps $T$. Graph (b) presents the corresponding cost function values (3.11a).

Firstly, the measurement times found by the GA seem to converge when the number of discretization steps $T$ increases (or equivalently, when $\delta$ decreases). In addition, they are more concentrated at the beginning. Furthermore, one can observe that even for the regular Kalman predictor, the measurement times vary with the number of discretization steps $T$. This is due to the fact that the measurement times are constraints to belong in the discrete time set. Algebraically, it is the Round operator in equation (3.6). When the number of discretization steps $T$ increases, the rounding effect decreases.

Figure 3.7b depicts the corresponding cost values for the continuous-time problem (3.11). It shows that the cost for optimal intermittent measurements is systematically smaller than in the regular case. This should be expected because the regular measurement times are in the admissible domain of problem (3.11). If the regular measurements had had smaller cost function value, the GA would probably have selected it. The roughly decreasing and converging behavior of the cost function with respect to

the number of discretization steps shows that our method can be used for continuous-time systems using an appropriate discretization scheme.

### 3.3.5 Impact of stability, process noise variance and measurement noise variance

In this section, the impact of the system's stability, process noise variance $Q$ and measurement noise variance $R$ on the solutions are studied in the one-dimensional case. More precisely, we compare the intermittent Kalman predictor to the regular Kalman predictor for different values of $Q$ and $R$. The experiments are restricted to one-dimensional system, for the stable case $|A| < 1$, the marginally stable case $|A| = 1$ and the unstable case $|A| > 1$. Finally, we comment on the performance in the different cases.

The studied system is a simple version of (3.1) where all quantities are one-dimensional and with $b = 0, G = 1, D = 1, C = 1, d = 0$. This problem is studied for all $A \in \{1/2, 1, 2\}$ and for $Q$ and $R$ in the range $[0.01, 100]$. Results are presented in Figure 3.8. For graphs (a)-(c), $A = 1/2$; for graphs (d)-(f), $A = 1$ and for graphs (g)-(i), $A = 2$. In graphs (a), (d), (g), $Q$ and $R$ vary. In graphs (b), (e), (h), $R = 50$ and $Q$ varies. Finally, in graphs (c), (f), (i), $Q = 50$ and $R$ varies.

In all cases, the optimal intermittent Kalman predictor gives a smaller cost than with regular measurements. For the stable system, i.e., $A = 1/2$ in graphs (a)-(c), the intermittent Kalman predictor gives results very similar to the regular Kalman predictor. In other cases, i.e., $A \in \{1, 2\}$ in graphs (d)-(i), the improvement due to the use of intermittent measurements instead of regular ones is greater. This improvement increases when $R$ increases. The unstable case, i.e., $A = 2$ in graphs (g)-(i), is the one in which the improvement is the most significant. This improvement increases when either $Q$ or $R$ increase.

### 3.3.6 Quality or quantity of measurements: An example

In this section, we study the compromise between few precise measurements and many more noisy measurements presented in Section 3.2.7. The problem (3.17) is studied for the dynamical system (3.18) with $\delta = 0.1[s]$ for $T = 100$ time steps. The measurement noise variance is related to the number of measurements as $R = f(N) = N^{\alpha}$ for different $\alpha > 0$. The larger the $\alpha$, the more an additional measurement increases the measurement noise.

Figure 3.9 presents the cost (3.17) for all $N \in \{1, \ldots, T\}$ and for the regular measurements (3.6) and the irregular measurements found by the

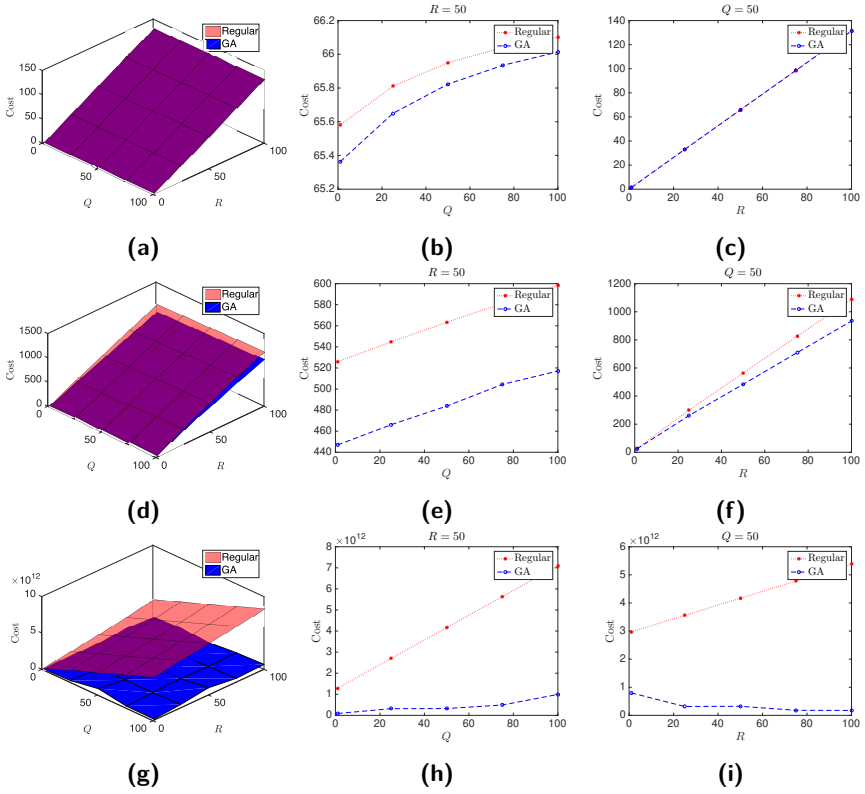



**Fig. 3.8** Cost function of (3.7) for both regular measurement times and times found by the GA for varying process noise variance $Q$ and/or measurement noise variance $R$. It corresponds to a one-dimensional version of model (3.1) with $b = 0$, $G = 1$, $D = 1$, $C = 1$, $d = 0$. For (a)-(c), $A = 1/2$; for (d)-(f), $A = 1$; for (g)-(i), $A = 2$. In (a), (d), (g) $Q$ and $R$ vary. In (b), (e), (h) $R = 50$ and $Q$ varies. In (c), (f), (i) $Q = 50$ and $R$ varies.

genetic algorithm. In each case, the minimum is indicated. Results are presented for all values of $\alpha \in \{0.2, 0.5, 0.75, 1, 1.25, 2, 5\}$. Firstly, observe that the irregular measurements found by the GA give smaller cost than regular measurements for all $\alpha$ and all $N$.

For the tested values of $\alpha$, the optimal number of measurement times for the regular measurements is $N = 1$ when $\alpha \geq 1.25$ and $N = 20 = T$ when $\alpha \leq 1$. In comparison, for the intermittent measurements the optimal number of measurements is $N = 1$ only for $\alpha \geq 2$ and increases progressively when $\alpha$ decreases. These observations illustrate that when measurements are expensive, i.e., $\alpha$ is high, only few measurements will be acquired. Conversely, when measurements are cheap, i.e., $\alpha$ is small, many measurements will be acquired.

Another observation is that the cost difference between the regular measurements and the intermittent varies with $N$. When $N = T$, the regular and intermittent measurements sets are the same because the only admissible set is the set with all measurement times, i.e., $\mathcal{M} = \{0, \ldots, T-1\}$. More generally, the size of the admissible domain of the optimization problem (3.17) is smaller when $N$ is close to 1 or $T$. Using intermittent measurements is most justified when $N$ is between 10 and 60.

## 3.4 Conclusions

This chapter addresses the problem of selecting the optimal measurement times for Kalman prediction over a finite time horizon. A random trial algorithm and three variants of genetic algorithms are proposed to solve it and the genetic algorithm that implements a count preserving crossover is shown to outperform the others. The tracking performances are extensively demonstrated on a numerical example, showing an improvement in 64% of the trajectories in comparison with regularly spaced measurement times for a spring-mass system. For a 50-dimensional system, our optimal intermittent Kalman predictor outperformed the regular Kalman predictor for 97% of the trajectories.

Then, the case of continuous-time systems is considered through a spring-mass system. The solution is shown to converge with respect to the discretization step size, making our method suitable for both discrete-time and continuous-time systems. Finally, the influence of noise variance is studied in the one-dimensional case, showing that the improvement is the most significant in the unstable case and when the process noise becomes important.

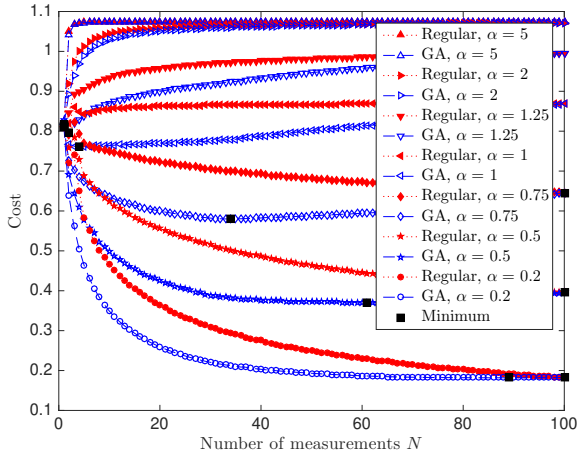



**Fig. 3.9** Cost (3.17) for all $N \in \{1, \dots, T\}$ and for the regular measurements (3.6) and the irregular measurements found by the genetic algorithm (GA). The minimum cost with respect to $N$ is depicted in each case. The considered dynamic system is (3.18) with $T = 100$ and $f(N) = N^{\alpha}$. Results are presented for $\alpha \in \{0.2, 0.5, 0.75, 1, 1.25, 2, 5\}$.

Thanks to duality, we have shown that the problem of selecting optimal control times for a LQR can be reduced to the proposed problem. Then, we studied the optimal compromise between a lot of noisier measurements and less more precise measurements.

In Chapter 4, the method presented here is generalized to the case of nonlinear and non-Gaussian dynamical systems. In Chapter 6, we show that the optimal intermittent Kalman predictor can be used to track mobile tumors for radiation therapy.

# 4

# Optimal intermittent particle filter

In Chapter 3, we addressed the problem of choosing the measurement times that minimize the expected mean squared estimation error in the case of a linear Gaussian dynamical system. In this Chapter 4, we are interested in the case of a nonlinear and non-Gaussian dynamical system.

Contrary to the linear and Gaussian case covered in Chapter 3, we will see that in this more general setting, there is no equivalence between time-triggered and self-triggered measurements. Indeed, the measurement times cannot be computed *a priori*, i.e., independently of the measurements already acquired, without loss of optimality. For that reason, we propose three different optimal intermittent filters. For the first, the self-triggered filter, the measurement times are given by a policy that determines whether a measurement should be taken based on the measurements already acquired. The second, called the time-triggered filter determines all measurement times at once by solving a combinatorial optimization program before any measurement acquisition. The third one, which we call ex post particle filter, is a recursive version of the time-triggered filter. It leads to self-triggered measurements because each time a new measurement is received, the next measurement time is recomputed to take all the information that is then available into account. We prove that in terms of expected

mean squared error, the self-triggered filter outperforms the ex post filter, which itself outperforms the time-triggered filter. To solve the combinatorial optimization programs, we compare a random trial algorithm, greedy forward and backward algorithms, a simulated annealing algorithm, and a genetic algorithm. Finally, the performance of the proposed methods is illustrated on two examples: a tumor motion model and a common benchmark for particle filtering.

This chapter assumes a basic knowledge of particle filters (see Section 2.4).

The results presented in this chapter are the result of close collaboration with Amaury Gouverneur and were reported in

- Antoine Aspeel, Amaury Gouverneur, Raphaël M Jungers, and Benoît Macq. Optimal measurement budget allocation for particle filtering. In *27th IEEE International Conference on Image Processing*, 2020.
- Antoine Aspeel, Amaury Gouverneur, Raphaël M Jungers, and Benoît Macq. Optimal intermittent particle filter. To appear.

## 4.1 Introduction

Particle filtering is a popular approach to estimate the state of nonlinear and non-Gaussian systems from a set of noisy measurements [AMGC02]. This tool has been largely used, among others, in computer vision [HLCP01, CJDP+06, ATD05], positioning and navigation [GGB+02, Gus10], chemistry [SPM+11, JBOBDT16], mechanics [CZA09], robotics [Thr02], and medicine [RLG+11].

The problem we address is to choose the best measurement times on a finite horizon context. In other words, one has a measurement budget and has to choose when to acquire measurements. The optimality criterion is to minimize the expected filtering *mean squared error* (MSE) over the complete time horizon.

First, the problem of optimal measurement budget allocation is formalized as a nonlinear multistage *stochastic program*. The measurement times are selected one after the other and each measurement time is selected taking the measurements already observed into account, leading to a self-triggered measurement mechanism.

Next, an easier related problem that we call the *time-triggered program* is presented. It corresponds to selecting all the measurement times at once

and offline, i.e., before any measurement acquisition: it is a time-triggered mechanism.

Then, we develop a recursive adaptation of the time-triggered problem, which we call the *ex post program*. It is a self-triggered mechanism, which gives better filtering performance, at the expense of a higher computational cost. In addition, the computations must be carried out online.

The closest work to this chapter regards the sensor scheduling problem (see Section 1.1.2) in the nonlinear and non-Gaussian case. In [DVAD02, SVDE04], an optimal one-step ahead policy is proposed to maximize the expected differential entropy between the filtering and the prediction density. Intuitively, it maximizes the information obtained at the current measurement time. In [YN18], the authors choose a set of sensors at each time step in order to trade off between the number of active sensors and the current posterior Cramér-Rao lower bound (PCRLB). This is justified by the fact that the PCRLB is a lower bound on the mean squared estimation error.

### 4.1.1 Contribution

This chapter addresses the problem of optimal allocation of a measurement budget in the discrete-time nonlinear case with disturbance and measurement noise processes following arbitrary distributions.

Our contributions in this chapter are that (i) we show that the problem of optimal intermittent filtering can be modeled as a stochastic program (see Problem 4.1); (ii) we formulate the time-triggered filter; (iii) based on this, we propose a recursive version of optimal intermittent filter (see Problem 4.3); (iv) we prove Theorem 4.4, which states that the expected mean squared filtering error is smaller or equal for the self-triggered filter than for the ex post filter, which in turn, yields an expected mean squared error that is smaller or equal than that of the time-triggered filter; (v) optimization algorithms are tested (greedy forward, greedy backward, simulated annealing, and genetic algorithm); and (vi) the results are presented (among others) on a new model inspired by real-world tumor tracking applications.

Overall, the two main contributions of this chapter are to propose an efficient algorithm to solve the problem of optimal measurement times selection; and to show the interest of intermittent measurements in particle filtering.

### 4.1.2 Chapter organization

The rest of this chapter is organized as follows: Section 4.2 presents how to define filtering with intermittent measurements (Section 4.2.1); different versions of optimal intermittent filters (Section 4.2.2) and how to compute them numerically with different optimization algorithms (Sections 4.2.3 and 4.2.4). Results and discussions are presented in Section 4.3, where a model of tumor motion is proposed for benchmarking, in addition to a common benchmark for particle filters (Section 4.3.1), the optimization algorithms are compared (Section 4.3.2), and the filtering performance of the time-triggered and ex post particle filters are compared with a regular particle filter (Section 4.3.3). Finally, Section 4.4 concludes and discusses possible improvements and perspectives.

### 4.1.3 Notation

The set of real numbers and the empty set are denoted $\mathbb{R}$ and $\emptyset$, respectively.

Vectors are noted in lower case, e.g., $x$. The transpose of a vector $x$ is noted $x^\top$ and its Euclidean norm is $\|x\|$.

A Gaussian distribution of mean $\mu \in \mathbb{R}^n$ and positive semi-definite covariance matrix $\Sigma \in \mathbb{R}^{n\times n}$ is noted $\mathcal{N}(\mu, \Sigma)$. For a compact set $\Omega \subset \mathbb{R}^n$, $\mathcal{U}(\Omega)$ denotes a uniform distribution on $\Omega$. In addition, $x \sim \mathcal{X}$ indicates that the vector $x \in \mathbb{R}^n$ is distributed according to the distribution $\mathcal{X}$. For random vectors $x$, $y$ and $z$, $\mathbb{E}_x[z|y]$ denotes the expectation of $z$ knowing $y$ with respect to the distribution of the random variable $x$.

Finally, the following notations are used: for $j \leq k$, $t_{j:k} \coloneqq \{t_j, t_{j+1}, \dots, t_k\}$ and $y(t_{j:k}) \coloneqq \begin{bmatrix} y(t_j) & y(t_{j+1}) & \dots & y(t_k) \end{bmatrix}$.

## 4.2 Materials and methods

### 4.2.1 Intermittent filter

A discrete time stochastic nonlinear dynamical system describes the evolution of a state $x(t)$ over the finite time horizon $t = 0, \dots, T$. One wants to estimate a quantity $z(t)$ related to $x(t)$ and has access to previously acquired noisy measurements $y(t)$ of $x(t)$. Measurements are not available at each time step. More precisely, only $N$ measurements are available, at

times $t_1, \dots, t_N \in \{0, \dots T\}$[1]. This is modelled as

$$x(t+1) = f_t(x(t), w(t)) \quad \text{for } t = 0, \dots, T-1, \tag{4.1a}$$
$$y(t_j) = g_{t_j}(x(t_j), v(t_j)) \quad \text{for } j = 1, \dots, N, \tag{4.1b}$$
$$z(t) = h_t(x(t)) \quad \text{for } t = 0, \dots, T, \tag{4.1c}$$
$$x(0) \sim \mathcal{X}_0, \tag{4.1d}$$

with $t_i \neq t_j$ if $i \neq j$, and $x(t) \in \mathbb{R}^{n_x}$, $y(t) \in \mathbb{R}^{n_y}$ and $z(t) \in \mathbb{R}^{n_z}$. In addition, $w(t)$ and $v(t)$ are random processes with known probability density functions. Functions $f_t(\cdot, \cdot)$, $g_t(\cdot, \cdot)$ and $h_t(\cdot)$ are known and have compatible dimensions. The initial state $x(0)$ follows a known distribution $\mathcal{X}_0$.

For instance, in a tumor tracking problem based on X-ray images, $x(t) \in \mathbb{R}^6$ can be a state vector containing the tumor's position and velocity in 3-dimensional space, $y(t) \in \mathbb{R}^2$ can be the 2-dimensional projection of the target and $z(t) \in \mathbb{R}^3$ the position of the mass center in 3-dimensional space.

We define the *intermittent filter* estimate,

$$\hat{z}(t|y(t_{1:j})) := \mathbb{E}_{x(0),w(0),\dots,w(t-1)}[z(t)|y(t_k), \forall t_k \leq t], \tag{4.2}$$

where $\mathbb{E}_X[\cdot]$ holds for the expectation operator according to the random variable $X$. The name "intermittent" emphasizes that measurements are not accessible at each time step. It is a mean squared error estimate in the sense of Section 2.2. In this chapter, we write it $\hat{z}(t|y(t_{1:j}))$ and not $\hat{z}(t|t_j)$ to make explicit the dependence in the measurements $y(t_{1:j})$. Section 4.2.3 will explain how an intermittent particle filter can be used to compute an approximation of the intermittent filter estimate.

Let us write the expected filtering error variance,

$$\mathcal{E}[t|y(t_{1:j})] := \\ \mathbb{E}_{x(0),w(0),\dots,w(t-1)}[\|z(t) - \hat{z}(t|y(t_{1:j}))\|^2 | y(t_k), \forall t_k \leq t], \tag{4.3}$$

where $\|\cdot\|$ is the Euclidean norm. A numerical approach to estimate this quantity is described in Section 4.2.3.

*Remark* 4.1. From definition (4.2), it holds that $\hat{z}(t|y(t_{1:j})) = \hat{z}(t|y(t_{1:j-1}))$ when $t < t_j$. Similarly, from definition (4.3), if $t < t_j$, then $\mathcal{E}[t|y(t_{1:j})] =$

[1] In this chapter, the notation $t_1, \dots, t_N$ will be used to denote measurement times, instead of the notation $\mathcal{M}$ used in Chapter 3. This will make the expressions simpler. Note that the link between the two is $\mathcal{M} = \{t_1, \dots, t_N\}$.

$\mathcal{E}[t|y(t_{1:j-1})]$. This ensures the causality of the filter by avoiding future measurements of influencing the present estimation. Otherwise, these two quantities are random variables because they depend on $y(t_1), \ldots, y(t_j)$ and therefore on $v(t_1), \ldots, v(t_j)$, which are random. ◇

*Remark* 4.2. Instead of the estimate of $z(t)$ defined in (4.2), one could use any estimate that can be defined from the distribution of $x(t)|(y(t_k), \forall t_k \leq t)$, for example, the *maximum a posteriori estimate* could replace (4.2) using the method from [SBD+09]. This flexibility is allowed because we will use a particle filter, which estimates the whole distribution of $x(t)|(y(t_k), \forall t_k \leq t)$.

Moreover, since we are going to use black-box optimization algorithms, another quality criterion could be used instead of the error variance (4.3), for example, the squared Euclidean norm could be replaced by the $1-$norm or the $\infty-$norm. ◇

### 4.2.2 Optimal intermittent filters

An *optimal intermittent filter* is an intermittent filter whose measurement times have been chosen according to a certain optimality criterion. In this section, we present three different optimality criteria for selecting the measurement times. Each optimality criterion induces an optimal intermittent filter.

#### *Self-triggered filter*

In the self-triggered filter, each measurement time $t_{j+1}$ is chosen taking the already acquired measurements $y(t_{1:j})$ into account.

Formally, the *self-triggered filter* is the intermittent filter whose measurement times are the solution of the following stochastic program.

**Problem 4.1** (Self-Triggered Program)**.**

$$
\begin{aligned}
V_0(\emptyset;\emptyset) = \min_{t_1} \Bigg\{ & \sum_{t=0}^{t_1-1} \mathcal{E}[t|\emptyset] \\
& + \mathbb{E}_{y(t_1)} \Bigg[ \min_{t_2} \Bigg\{ \sum_{t=t_1}^{t_2-1} \mathcal{E}[t|y(t_1)] + \cdots \\
& + \mathbb{E}_{y(t_{N-1})} \Bigg[ \min_{t_N} \Bigg\{ \sum_{t=t_{N-1}}^{t_N-1} \mathcal{E}[t|y(t_{1:N-1})] \\
& + \mathbb{E}_{y(t_N)} \Bigg[ \sum_{t=t_N}^{T} \mathcal{E}[t|y(t_{1:N})] \Bigg] \Bigg\} \Bigg] \cdots \Bigg\} \Bigg] \Bigg\},
\end{aligned}
$$

*subject to* $0 \leq t_1 < t_2 < \cdots < t_N \leq T$.

Introducing the notation $t_0 \coloneqq 0$, the problem can be stated recursively as,

$$
\begin{aligned}
V_j(t_{1:j}; y(t_{1:j})) = \min_{t_{j+1}} \Bigg\{ & \sum_{t=t_j}^{t_{j+1}-1} \mathcal{E}[t|y(t_{1:j})] \\
& + \mathbb{E}_{y(t_{j+1})} \left[ V_{j+1}(t_{1:j+1}; y(t_{1:j+1})) \right] \text{ with } t_j < t_{j+1} \leq T \Bigg\},
\end{aligned} \tag{4.4}
$$

with the terminal condition,

$$
V_N(t_{1:N}; y(t_{1:N})) = \sum_{t=t_N}^{T} \mathcal{E}[t|y(t_{1:N})]. \tag{4.5}
$$

Solving this multistage stochastic program requires finding an optimal policy for each $t_{j+1}$. Such optimal policies are functions that associate the next optimal measurement time with the previously acquired measurements, i.e., it is $(y(t_1), \ldots, y(t_j)) \mapsto t_{j+1}$.

Under certain restrictive assumptions, including linearity and finite stochastic outcomes, stochastic multistage programs can be solved using scenario trees and duality [SDR14]. For more complicated problems such as ours (nonlinear, continuous random variables), one approach is to use deep reinforcement learning [FLHI+18].

Indeed, this stochastic program can be seen as a partially observable Markov decision process (POMDP). The problem is partially observable

because the state $x(t)$ is not directly available. At each time $t$, the action is either to measure the system or not. Then, depending on this choice, an observation, i.e., a measurement $y(t_j)$, is received (or not) and used to compute $\hat{z}(t|y(t_j))$ (or $\hat{z}(t|y(t_{j-1}))$). A reward $-\|z(t) - \hat{z}(t|y(t_{1:j}))\|^2$ (or $-\|z(t) - \hat{z}(t|y(t_{1:j-1}))\|^2$) is then computed to train the reinforcement learning agent. To solve such a POMDP, one option is to use deep Q-learning. There are mainly two methods to implement a deep Q-network in the partially observable setting. The first is to use a Q-network with a history, the second is to use a recurrent Q-network [RB19].

However, deep reinforcement learning often requires a lot of trial and error and fine tuning of some parameters (network architecture, learning rate, training discount factor, value of $\epsilon$ when epsilon-greedy is used, batch size, etc). This makes the implementation of such methods time consuming and often requires a large amount of computing power. For these reasons, the use of deep reinforcement learning to solve Problem 4.1 is left for future work.

Here, we propose another approach, which is to optimize the variables all at once and offline, i.e., before any measurement acquisition. We use this approximation and we call it the time-triggered approach.

*Time-triggered filter*

The time-triggered filter is optimal (in the sense of the expected MSE) if all the measurement times must be chosen before receiving any measurement. Each measurement time $t_j$ is independent of the measurements that preceded it, $y(t_{1:j-1})$.

Formally, the *time-triggered filter* is the intermittent filter whose measurement times are the solution of the following time-triggered program.

**Problem 4.2** (Time-Triggered Program)**.**

$$J_0(\emptyset;\emptyset) \coloneqq \min_{0\leq t_1<\cdots<t_N\leq T} \mathbb{E}_{y(t_{1:N})}\left[\sum_{t=0}^{T} \mathcal{E}[t|y(t_{1:N})]\right].$$

Because each optimal measurement time $t_j$ is independent of the previously acquired measurements $y(t_{1:j-1})$, all these optimal measurement times $t_{1:N}$ can be computed at once, before any measurement acquisition. This can be a decisive advantage for certain real-world applications if the time between two discrete time steps is too short to solve an optimization program. Unlike Problem 4.1, here the optimization variables are no longer functions but natural numbers.

*Ex post filter*

To enhance filtering performance, once the measurements $y(t_{1:j})$ have been acquired, we may want to use them to recompute the next measurement time $t_{j+1}$ including the information previously acquired, i.e., the previous measurements. This requires solving $N$ different optimization programs, one for each measurement time.

Formally, the *ex post filter* is the intermittent filter for which each measurement time $t_{j+1}$ (for $j = 0, \dots, N-1$) is the solution of the following $(j+1)^{\text{th}}$ ex post program:

**Problem 4.3** ($(j+1)^{\text{th}}$ Ex Post Program)**.** *When measurements $y(t_{1:j})$ have been acquired, the next measurement time $t_{j+1}$ is computed by solving*

$$J_j(t_{1:j}; y(t_{1:j})) \coloneqq \min_{t_{j+1} < \cdots < t_N \leq T} \mathbb{E}_{y(t_{j+1:N})} \Bigg[ \sum_{t=t_j}^{T} \mathcal{E}[t | y(t_{1:N})] \Bigg], \text{ such that } t_{j+1} > t_j, \tag{4.6}$$

*where $t_0 \coloneqq 0$ but the constraint is replaced by $t_1 \geq 0$ when $j = 0$.*

In this $(j+1)^{\text{th}}$ program, only the measurement time $t_{j+1}$ is used by the ex post filter. To avoid ambiguities, it will sometimes be noted $t^o_{j+1}$. The other variables of the $(j+1)^{\text{th}}$ program, i.e., the $t_{j+2:N}$, are not the measurement times of the ex post filter because they will be replaced by the solutions of the next programs. Such ex post programs can be seen as a recursive version of the time-triggered program, i.e., Problem 4.2.

*Remark* 4.3. Setting $j = 0$ in Problem 4.3 gives exactly Problem 4.2. Consequently, the optimal $t_1$s are the same for these two problems. ◇

If $t_{1:j}$ are the $j$ first measurement times and $y(t_{1:j})$ the corresponding measurements, we denote $F_j(t_{1:j}; y(t_{1:j}))$ the remaining cost-to-go from

time $t_j$ using the ex post filter. For $j = 0, \dots, N-1$, it is

$$\begin{aligned} F_j(t_{1:j}; y(t_{1:j})) &= \sum_{t=t_j}^{t_{j+1}^o - 1} \mathcal{E}[t|y(t_{1:j})] \\ &+ \mathbb{E}_{y(t_{j+1}^o)} \Bigg[ \sum_{t=t_{j+1}^o}^{t_{j+2}^o - 1} \mathcal{E}[t|y(t_{1:j}), y(t_{j+1}^o)] + \cdots \\ &+ \mathbb{E}_{y(t_{N-1}^o)} \Bigg[ \sum_{t=t_{N-1}^o}^{t_N^o - 1} \mathcal{E}[t|y(t_{1:j}), y(t_{j+1:N-1}^o)] \\ &+ \mathbb{E}_{y(t_N^o)} \Bigg[ \sum_{t=t_N^o}^{T} \mathcal{E}[t|y(t_{1:j}), y(t_{j+1:N}^o)] \Bigg] \Bigg] \cdots \Bigg], \end{aligned}$$

and for $j = N$, it is

$$F_N(t_{1:N}; y(t_{1:N})) = \sum_{t=t_N}^{T} \mathcal{E}[t|y(t_{1:N})]. \tag{4.7}$$

With this definition, the following recursive relation holds,

$$\begin{aligned} F_j(t_{1:j}; y(t_{1:j})) &= \sum_{t=t_j}^{t_{j+1}^o - 1} \mathcal{E}[t|y(t_{1:j})] \\ &+ \mathbb{E}_{y(t_{j+1}^o)} \Big[ F_{j+1}(t_{1:j}, t_{j+1}^o; y(t_{1:j}), y(t_{j+1}^o)) \Big]. \end{aligned} \tag{4.8}$$

In the following theorem, we establish that the expected mean squared filtering error is smaller or equal for the self-triggered filter than for the ex post filter, which itself is smaller or equal than for the time-triggered filter.

**Theorem 4.4.** $V_0(\emptyset;\emptyset) \leq F_0(\emptyset;\emptyset) \leq J_0(\emptyset;\emptyset)$.

*Proof.* See Appendix B. □

*Remark* 4.5. If the dynamical system (4.1) is linear and Gaussian, i.e., functions $f_t(\cdot,\cdot), g_t(\cdot,\cdot)$ and $h_t(\cdot)$ are linear and distributions of $w(t), v(t)$ and $x(0)$ are Gaussian, then for given measurement times $t_1, \dots, t_N$, the optimal filtered estimate $\hat{z}(t|y(t_{1:j}))$ is given explicitly by the Kalman filter [Kal60, Theorem 2].

In addition, the variance of the filtering error $\mathcal{E}[t|y(t_{1:j})]$ is independent of the measurements $y(t_{1:j})$, which implies that the Kalman gains can be computed offline (see Remark 3.1 for details). Consequently, the expectation operators in Problems 4.1, 4.2 and 4.3 are equivalent to identity operators, which gives the same problem three times. In conclusion, in the linear Gaussian case, the optimal solutions of Problems 4.1, 4.2 and 4.3 are the same, i.e., $V_0(\emptyset;\emptyset) = F_0(\emptyset;\emptyset) = J_0(\emptyset;\emptyset)$. ◇

The difference between Problems 4.1 and 4.3 may not seem obvious. Indeed, they both require computing the measurement times online. To clarify the difference, we propose an example where these two problems give different solutions.

*Example* 4.6. Consider three time steps, i.e., $T = 2$, with $N = 2$ measurements and consider the system illustrated in Figure 4.1 and summarized as

$$y(0) = z(0) = x(0) = \begin{cases} 1, & \text{with probability } 1/2 \\ -1, & \text{with probability } 1/2, \end{cases} \tag{4.9a}$$

$$y(1) = z(1) = x(1) = \begin{cases} w(0), & \text{if } x(0) = 1 \\ 0, & \text{if } x(0) = -1, \end{cases} \tag{4.9b}$$

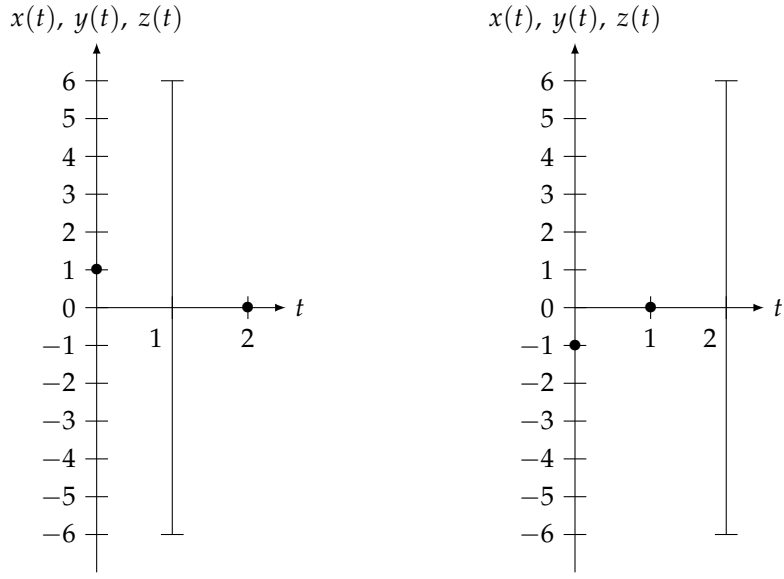
$$y(2) = z(2) = x(2) = \begin{cases} 0, & \text{if } x(1) \neq 0 \\ w(1), & \text{if } x(1) = 0, \end{cases} \tag{4.9c}$$

where $w(0)$ and $w(1)$ independently follow a uniform distribution between $-6$ and 6, i.e., $w(0), w(1) \sim \mathcal{U}([-6, 6])$.

First observe that it is possible to make no filtering error. To do this, set the first measurement time to $t_1 = 0$ and then, if $y(0) = 1$, set the second to $t_2 = 1$ and on the contrary, if $y(0) = -1$, set the second to $t_2 = 2$. With this schedule, a measurement is acquired whenever there is an uncertainty, which guarantees that there is no filtering error. It is the solution of Problem 4.1.

However, we will show that the solutions of both Problems 4.2 and 4.3 are to choose $t_1 = 1$ and $t_2 = 2$, which results in a filtering error at time $t = 0$.

Doing an exhaustive search for Problem 4.2 shows that choosing $t_1 = 1$ and $t_2 = 2$ is optimal. This is due to the fact that if there is no measurement at $t = 1$ or at $t = 2$, a large error will be made half the time because of the uniformly distributed $x(t)$. Then, because the optimal $t_1$s for Problems 4.3




**(a)** If $x(0) = 1$. **(b)** If $x(0) = -1$.

**Fig. 4.1** Illustration of the system (4.9). Vertical lines represent uniform distributions $\mathcal{U}([-6, 6])$. Each of scenarios (a) and (b) has a 50/50 chance of occurring, depending on whether $x(0) = 1$ or $-1$.

and 4.2 are the same (see Remark 4.3), $t_1 = 1$ is also optimal for the former. But then, the only remaining possibility for $t_2$ is $t_2 = 2$. This shows that the optimal solutions of both Problems 4.2 and 4.3 are $t_1 = 1$ and $t_2 = 2$, which gives a filtering error at time $t = 0$.

This example illustrates that the optimal time $t_1$ of Problem 4.3 is computed by assuming that the time $t_2$ cannot depend on $y(t_1)$. ◇

### 4.2.3 Numerical approximations

In this section, we present the numerical approximations, which are used to make Problems 4.2 and 4.3 tractable. Firstly, we briefly explain how to approximate numerically the estimate $\hat{z}(t|y(t_{1:j}))$ using a particle filter and then how to approximate the objective functions of these two problems. We reserve the term "intermittent filter" for the ideal filter and use "intermittent particle filter" for its approximation. Let us emphasize that Theorem 4.4 applies only in the ideal case.

#### *Approximation of the estimate*

The estimate $\hat{z}(t|y(t_{1:j}))$ is not straightforward to compute but can be approximated using a particle filter. See Section 2.4 for a detailed description about particle filtering, or Section 2.4.4 only for a summary.

Essentially, after (i) an initialization step, the considered particle filter algorithm alternates between (ii) a weighting step, (iii) a resampling step, (iv) an estimation step, and (v) a mutation step. To deal with intermittent measurements, the weighting step (ii) and the resampling step (iii) are skipped when no measurement is available, i.e., when $t \neq t_k$ for all $k$. Unfortunately, if measurement times are widely spaced, it can happen that all particles have numerically zero weights (because the weights are not updated at time steps where no measurement is accessible). In this case, the same weight is assigned to each particle.

In what follows, we will make a slight abuse of notation by using the same notation $\hat{z}(t|y(t_{1:j}))$ for the estimate (4.2) and for its approximation computed by the particle filter.

#### *Approximation of the objective functions*

The objective functions of Problems 4.2 and 4.3 contain an expectation operator, which makes these functions challenging to evaluate. To tackle this problem, we use a Monte Carlo approximation of the expectation.

Because the objective function of Problem 4.2 is the same as the objective function of Problem 4.3 for $j = 0$ (see Remark 4.3), we focus on the

objective function of Problem 4.3 for any $j$, which includes Problem 4.2.

The Monte Carlo approximation of the objective function of Problem 4.3 is represented in Figure 4.2. Assuming that the first $j$ measurements $y(t_{1:j})$ have already been acquired and that future measurement times $t_{j+1:N}$ are fixed, one can simulate $K$ realizations of the state $\{x^k(t)\}_{t=0,\dots,T}^{k=1,\dots,K}$ drawn according to equations (4.1a) and (4.1d). The corresponding quantities that remain to be estimated $\{z^k(t)\}_{t=t_j,\dots,T}^{k=1,\dots,K}$ can be computed using equation (4.1c). Similarly, corresponding measurements $\{y^k(t_l)\}_{l=j+1,\dots,N}^{k=1,\dots,K}$ can be simulated thanks to equation (4.1b). For each $k$, the simulated measurement sequence can be concatenated to the known measurements to give a complete measurement sequence $y(t_1),\dots,y(t_j),y^k(t_{j+1}),\dots,y^k(t_N)$, from which the particle filter computes the estimates $\{\hat{z}^k(t|y(t_1),\dots,y^k(t_N))\}_{t=t_j,\dots,T}$ of $\{z^k(t)\}_{t=t_j,\dots,T}$. Then, thanks to definition (4.3), the Monte Carlo estimator is computed as

$$\begin{aligned}
&\mathbb{E}_{y(t_{j+1:N})}\left[\sum_{t=t_j}^{T}\mathcal{E}[t|y(t_{1:N})]\right] = \mathbb{E}_{y(t_{j+1:N})}\Bigg[\\
&\sum_{t=t_j}^{T}\mathbb{E}_{x(0),w(0),\dots,w(T-1)}\left[\|z(t)-\hat{z}(t|y(t_{1:N}))\|^2\Big|y(t_l),\forall t_l\le t\right]\Bigg]\\
&\approx \frac{1}{K}\sum_{k=1}^{K}\sum_{t=t_j}^{T}\|z^k(t)-\hat{z}^k(t|y(t_{1:j}),y^k(t_{j+1:N}))\|^2. \qquad (4.10)
\end{aligned}$$

Computing this Monte Carlo approximation requires running the particle filter $K$ times, which can be computationally expensive.

The fact that we have access only to an approximation of the objective function requires particular attention during the optimization. Note that since the particle filter provides a biased approximation, the approximation of the cost function may be biased as well. However, accounting for this bias is outside the scope of this thesis and is left to future work.

### 4.2.4 Optimization algorithms

Solving the combinatorial optimization Problem 4.2 (or equivalently, in light of Remark 4.3, Problem 4.3 with $j = 0$) with an exhaustive search would require evaluating the objective function $\frac{(T+1)!}{(T+1-N)!N!}$ times, i.e., the number of subsets of size $N$ in a set of size $T+1$. This is computationally intractable for large $N$ and $T$.

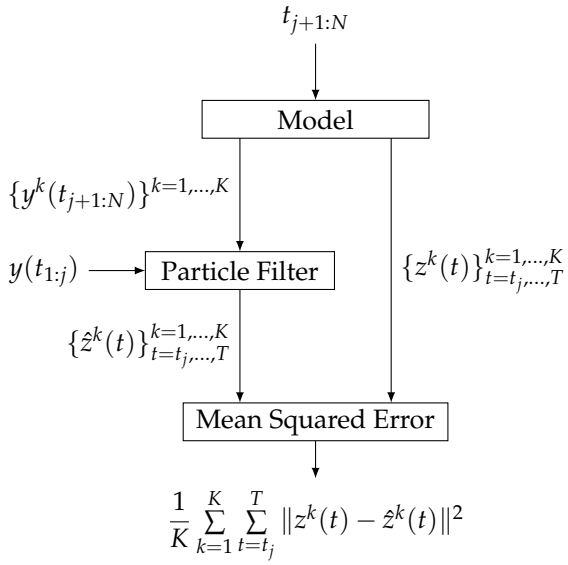



**Fig. 4.2** Representation of the Monte Carlo algorithm that estimates $\mathbb{E}_{y(t_{j+1:N})}[\sum_{t=t_j}^{T}\mathcal{E}[t|y(t_{1:N})]]$. The outputs generated by the Model block are drawn according to equations (4.1) and the distributions of $v(t)$ and $w(t)$. The particle filter computes an estimate $\hat{z}^k(t) = \hat{z}^k(t|y(t_{1:j}), y^k(t_{j+1:N}))$ of $z^k(t)$ from previous intermittent measurements. The Mean Qquared Error block computes the mean squared of the difference between its inputs; it is the right-hand side of equation (4.10).

In this section, we present five different heuristic optimization algorithms to find an approximate solution for Problems 4.2 and 4.3 efficiently.

*Random trial algorithm (RT)*

The random trial algorithm samples measurement time sets uniformly at random and evaluates their corresponding costs. The measurement time set with minimum cost is returned.

*Greedy forward algorithm (GF)*

The greedy forward algorithm begins with an empty measurement set. Sequentially, it adds the measurement times one at a time such that, at each iteration, the added measurement time minimizes the cost. It stops when the set contains $N$ measurement times.

*Greedy backward algorithm (GB)*

The greedy backward algorithm works in the opposite direction from the greedy forward algorithm. It begins with a complete measurement set. Sequentially, it takes out the measurement times one at a time such that, at each iteration, the dropped measurement time minimizes the cost. It stops when the set contains $N$ measurement times.

*Simulated annealing algorithm (SA)*

The implementation of the SA algorithm is done according to [DCM19, Section 1.2.3]. The initial temperature parameter is $T^\circ = 10$ and the geometric temperature decay rate is 0.9, i.e., after each generation, $T^\circ := 0.9 \cdot T^\circ$. If a mutation reduces the cost, it is kept. If it increases the cost, it is kept with probability $\exp(-(\text{cost increase})/T^\circ)$. The mutations are executed by prohibiting the duplication of measurement times and the mutation probability is 0.1 by measurement time.

*Genetic algorithm (GA)*

The implementation of the genetic algorithm is based on [Mit98]. The selection of the individuals kept for the next generation is done according to stochastic universal sampling [Mit98, Section 5.4]. To ensure that the number of measurement times remains constant over generations, a count preserving crossover operator is implemented [US15, Section 3.6]. It prevents the production of individuals $(t_{j+1}, \ldots, t_N)$ for which $t_k = t_l$ with $k \neq l$. For the same reason, the mutations are executed by prohibiting the duplication of measurement times. The algorithm uses sigma scaling with a unitary sigma coefficient [Mit98, Section 5.4], as in Chapter 3. Crossover

probability is 1 and mutation probability per measurement time is 0.003.

When the RT, SA or GA algorithm is used to solve the $N$ ex post Problems 4.3, the solution found for the $j^{\text{th}}$ problem can be added to the initial population when solving the $(j+1)^{\text{th}}$ problem. This can speed up the convergence of the algorithm but also reduce the exploration capacity of these methods. Here, we do not use this acceleration trick.

As mentioned in Section 4.2.3, the objective functions of Problems 4.2 and 4.3 can be evaluated approximately only. This makes minimization difficult if it is too sensitive to bad cost function estimations. To face this issue, the SA algorithm and the GA return the best individual of the last generation instead of the best individual in all generations.

In Section 4.3, the performances of these five algorithms are compared. It will show that the GA outperforms the other algorithms on the studied example.

## 4.3 Numerical examples and discussions

A MATLAB implementation of the presented algorithms and the code that generate the figures is available on GITHUB at GITHUB.COM/AMAURYGOUVERNEUR/OPTIMAL_MEASUREMENT_TIMES_FOR_PARTICLE_FILTERING.

### 4.3.1 Examples parameters

In this section, first we present two dynamical systems used for the results, then we describe the parameters of the simulations and the performance indicators.

*Tumor motion model*

To illustrate the performances of our method, we study a model describing one-dimensional tumor motion. As mentioned in introduction, the problem of tracking tumors with X-rays requires doing the best of each X-ray acquisition in order to spare healthy tissues (see Chapter 6 for details on mobile tumor tracking).

The duration between each discrete time step is $\delta$. The tumor position to estimate $z(t)$ is a shifted sinusoidal signal with a time-varying amplitude $a(t)$, a time-varying shift $b(t)$, and a constant frequency $\omega$. Both $a(t)$ and $b(t)$ are bounded random walks. Bounds ensure that values remain realistic over time. In addition, the constant oscillation frequency $\omega$ is picked

uniformly at random at the beginning of the process. Each measurement $y(t)$ is a noisy version of the position $z(t)$.

Let us define the state $x(t) = \begin{bmatrix} a(t) & b(t) & \omega(t) \end{bmatrix}^\top \in \mathbb{R}^3$ and the process noise $w(t) = \begin{bmatrix} w_a(t) & w_b(t) \end{bmatrix}^\top \in \mathbb{R}^2$. The dynamic of the system is defined as follows:

$$\begin{aligned} x(t+1) &= \begin{bmatrix} a(t+1) \\ b(t+1) \\ \omega(t+1) \end{bmatrix} \\ &= f(x(t), w(t)) && \text{(4.11a)} \\ &:= \begin{bmatrix} \text{clip}(a(t) + w_a(t), \underline{a}, \bar{a}) \\ \text{clip}(b(t) + w_b(t), \underline{b}, \bar{b}) \\ \omega(t) \end{bmatrix}, \\ y(t) &= g_t(x(t), v(t)) = a(t)\sin(\omega(t)t\delta) + b(t) + v(t), && \text{(4.11b)} \\ z(t) &= h_t(x(t)) = a(t)\sin(\omega(t)t\delta) + b(t), && \text{(4.11c)} \\ x(0) &= \begin{bmatrix} a(0) \\ b(0) \\ \omega(0) \end{bmatrix} \sim \mathcal{U}\left([\underline{a}, \bar{a}] \times [\underline{b}, \bar{b}] \times [\underline{\omega}, \bar{\omega}]\right). && \text{(4.11d)} \end{aligned}$$

Equations (4.11a), (4.11b), and (4.11c) hold for $t = 0, \ldots, T-1$, $t \in \{t_1, \ldots, t_N\}$, and $t = 0, \ldots, T$, respectively. Noises $w_a(t)$, $w_b(t)$ and $v(t)$ are independent zero mean Gaussian noises and have a unitary standard deviation, i.e., $\sigma_a = \sigma_b = \sigma_v = 1\ [mm]$. The clipping function is defined as $\text{clip}(x, \underline{x}, \bar{x}) := \min(\max(x, \underline{x}), \bar{x})$. Equation (4.11d) indicates that the initial state is uniformly distributed at random on the indicated domain.

We consider $N = 11$ measurements in the range of 0 to $T = 30$. To make this model more realistic, the values of $\underline{a}$, $\bar{a}$, $\underline{b}$, and $\bar{b}$ are inspired from [DVB+18, Table 1] and the values of $\underline{\omega}$ and $\bar{\omega}$ are inspired from [WVB+11, Table 1]. The parameters of the system are

$$\begin{aligned} &\underline{a} = 8.8\ [mm],\ \bar{a} = 24\ [mm], \\ &\underline{b} = -5.8\ [mm],\ \bar{b} = 5.8\ [mm], \\ &\underline{\omega} = 1.3\ [rad/s],\ \bar{\omega} = 2.1\ [rad/s], \\ &\delta = 0.25\ [s],\ \sigma_a = \sigma_b = \sigma_v = 1\ [mm], \\ &T = 30,\ N = 11. \end{aligned}$$

The importance density used by the particle filter is the prior density,

except for $\omega(t)$ where we use $\omega(t+1) = \text{clip}(\omega(t) + w_\omega(t), \underline{\omega}, \bar{\omega})$ with $w_\omega(t)$, an independent zero mean Gaussian noise with standard deviation $\sigma_\omega = 0.005$.

*A common benchmark for particle filters*

We also test our method on the following widely studied model [AMGC02, CPS92, Kit96, KLJF02] (it has already been presented in Section 2.4.5):

$$x(t+1) = \frac{x(t)}{2} + \frac{25x(t)}{1+x(t)^2} + 8\cos(1.2t) + w(t), \tag{4.12a}$$

$$y(t) = \frac{x(t)^2}{20} + v(t), \tag{4.12b}$$

$$z(t) = x(t), \tag{4.12c}$$

$$x(0) \sim \mathcal{N}(0, 5^2), \tag{4.12d}$$

where $x(t), y(t), z(t), w(t), v(t) \in \mathbb{R}$. Equations (4.12a), (4.12b) and (4.12c) hold for $t = 0, \ldots, T-1$, $t \in \{t_1, \ldots, t_N\}$, and $t = 0, \ldots, T$, respectively. Quantities $w(t)$, $v(t)$ and $x(0)$ are randomly distributed according to independent zero mean Gaussian distributions with standard deviations $\sigma_w = 1$, $\sigma_v(t) = \sin(0.25t) + 2$ and $\sigma_{x_0} = 5$, respectively. As for the tumor model, $N = 11$ measurements are allowed and the horizon is set at $T = 30$. For this model, the importance density used by the particle filter is the prior density.

*Performance indicators*

To test the proposed methods, we simulate many realizations of $\{y(t)\}_{t=0,\ldots,T}$ and $\{z(t)\}_{t=0,\ldots,T}$ according to system dynamic on which our filtering method is applied. Then, the filtering mean squared error, $\text{MSE} = \frac{1}{T+1}\sum_{t=0}^{T} \|z(t) - \hat{z}(t)\|^2$, is computed and is compared with the filtering mean squared error $\text{MSE}_{\text{REG}}$ obtained with regular measurement times,

$$t_{1:N} = \left\{ \text{Round}\left[\frac{kT}{N-1}\right] \middle| \, k = 0, \ldots, N-1 \right\}, \tag{4.13}$$

**Table 4.1** Values of the model parameters, test parameters, and optimization parameters for the time-triggered Problem 4.2 and the ex post Problem 4.3.

| | Parameters | Time-triggered | Ex post |
|---|---|---|---|
| Optimization | # draws $K$ | 1000 | 200 |
| | # particles | 200 | 100 |
| | pop. size | 50 | 30 |
| | # generations | 25 | 15 |
| Simulations | # simulations | 100,000 | 500 |
| | # particles | 1000 | 1000 |

where Round$[\cdot]$ is the rounding operator[2]. More precisely, we look at the gain

$$G := \log_{10}\left(\frac{\mathrm{MSE}_{\mathrm{REG}}}{\mathrm{MSE}}\right). \tag{4.14}$$

It illustrates the merits of using optimal measurement times instead of regular ones. A positive gain indicates that the considered intermittent particle filter outperforms the regular one.

As performance indicators, we look at the mean and median gain and the proportion of positive gain over all the simulations. Notice that the mean gains can be seen as an estimation of the expected gain and the proportion of positive gains as an estimation of the probability of outperforming the regular measurement filter.

*Optimization and simulations parameters*

As the measurement times that are the solution of the ex post Problem 4.3 are a function of the previous measurements, the optimization algorithm has to be run on each simulation. This increases the computation time significantly. Therefore, the solutions of Problems 4.2 and 4.3 are studied separately and the number of simulations and the optimization parameters have been set accordingly. The values of the optimization parameters and the simulation parameters are given in Table 4.1. Note that the number of particles used for the optimization is different from the number of particles for the filtering.

[2]Definition (4.13) is not the same as definition (3.6) because here we are interested in filtering, while Chapter 3 is about prediction.

### 4.3.2 Comparison of optimization algorithms

The different optimization algorithms proposed in Section 4.2.4 are tested on the time-triggered Problem 4.2 for the tumor motion model (4.11).

For all the optimization algorithms, the evolution of the cost of Problem 4.2 with respect to the number of cost function evaluations is illustrated in Figure 4.3. Because the number of cost function evaluations of the GF and GB algorithm is fixed, they are represented by points in the figure.

One can observe that all five optimization algorithms return a measurement subset with associated expected MSE significantly lower than the regularly spaced measurements. The final expected MSE of the SA, RT, GF and GB optimization algorithms all lie close to 6. Note that the GF and GB algorithms give good results for a relatively small number of evaluations of the cost function. Consequently, they can be a good option if computing resources are limited. The GA and SA algorithms perform notably better with a final minimum cost around 5.4. One can note that the difference between GA average and minimum cost decreases over generations before reaching quasi-convergence, meaning that most of the individuals are identical, which is the desired convergence behavior for a GA.

As a result of these observations, the GA is used as the optimization algorithm in the rest of the chapter.

### 4.3.3 Filtering performance

In this section, we analyze the filtering performance obtained using the time-triggered (Problem 4.2) and ex post (Problem 4.3) measurement times. First, we report the results on the tumor motion model (equations (4.11)), then we present the results for the common benchmark (equations (4.12)).

*Results for the tumor motion model (equations (4.11))*

The histogram of the gains $G$ for the time-triggered Problem 4.2 and for the ex post Problem 4.3 can be found in Figure 4.4. The time-triggered gain is computed over 100,000 simulations and the ex post gain is computed over 500 simulations. The vertical line corresponds to the null gain. These histograms can be interpreted as approximations of the probability density of the gains $G$. The proportion of positive gain for the GA algorithm can be read as the area of the histogram above 0. The corresponding performance indicators are reported in Table 4.2 for both the time-triggered particle filter (related to Problem 4.2) and ex post particle filter (related to Problems 4.3).

The mean and median gains are positive for our time-triggered and ex

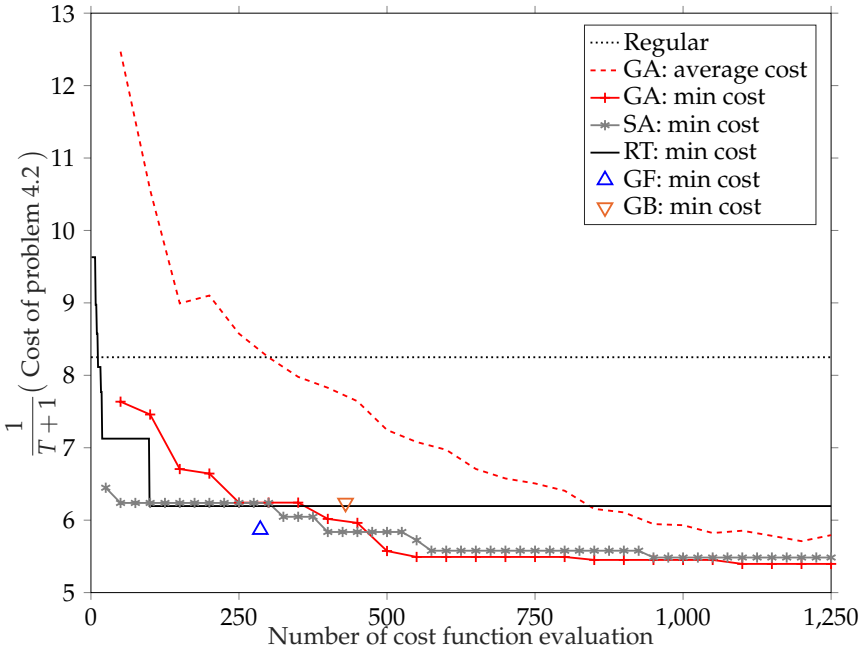



**Fig. 4.3** Evolution of the minimum cost of the time-triggered Problem 4.2 with respect to the number of cost function evaluations for the different optimization algorithms: genetic algorithm (GA), simulated annealing (SA), random trial (RT), greedy forward (GF), and greedy backward (GB). It illustrates the quality of minimization for given computational resources. The evolution of the average cost of the population of the GA algorithm at each generation and the cost of regularly spaced measurement times are shown as well.

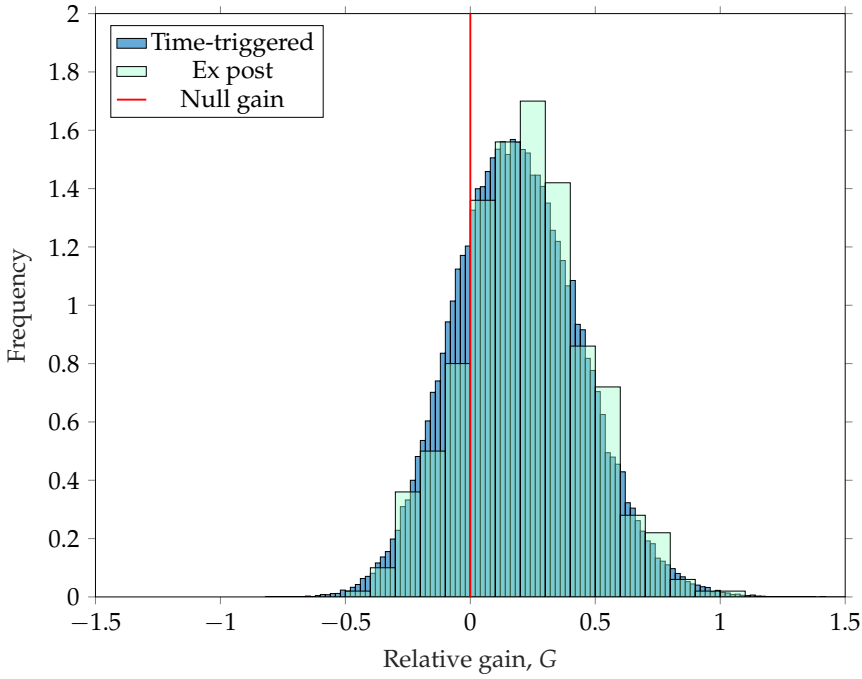



**Fig. 4.4** Histograms of the gain $G$ (see equation (4.14)) for the tumor motion model (equations (4.11)). The histograms have been obtained over 100,000 and 500 simulations for the time-triggered and ex post particle filters, respectively.

post methods, which indicates that they generally outperform the regular particle filter. In addition, the proportion of positive gain indicates that our time-triggered and ex post methods outperform the regular particle filter in 76.8% and 82.2% of the cases, respectively.

Figure 4.5 represents a particular trajectory $z(t)$ and the trajectory filtered by a particle filter with regular measurement times (see equation (4.13)), as well as with the time-triggered and ex post particle filters. In all three cases, the measurement times are indicated. As expected, we obtain better filtering performance using the ex post particle filter (the gain is $G_{\text{ex post}} = 0.20$) than the time-triggered particle filter (the gain is $G_{\text{time-triggered}} = 0.15$), the largest mean squared filtering error being obtained with the regular measurements.

**Table 4.2** Performance indicators for the time-triggered and ex post solutions on the tumor motion model (equations (4.11)). The considered system, the performance indicators, and the optimization and simulation parameters are described in Section 4.3.1.

| | Time-triggered | Ex post |
|---|---|---|
| Mean gain (± std) | 0.190 (± 0.253) | 0.220 (± 0.244) |
| Median gain | 0.182 | 0.216 |
| Proportion positive gain | 76.8 % | 82.2 % |

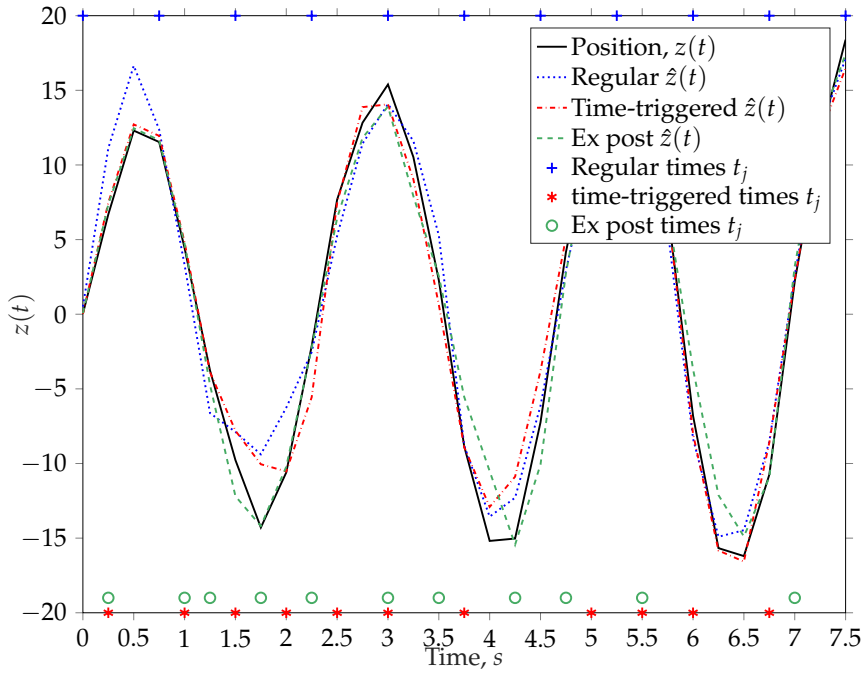



**Fig. 4.5** Comparison of a particular realization $z(t)$ with the filtered values $\hat{z}(t)$ obtained with a particle filter with regular measurement times and both the time-triggered and the ex post particle filters. The gains on the complete sequence are $G_{\text{time-triggered}} = 0.150$ and $G_{\text{ex post}} = 0.203$. Results are simulated from the tumor motion model (equations (4.11)).

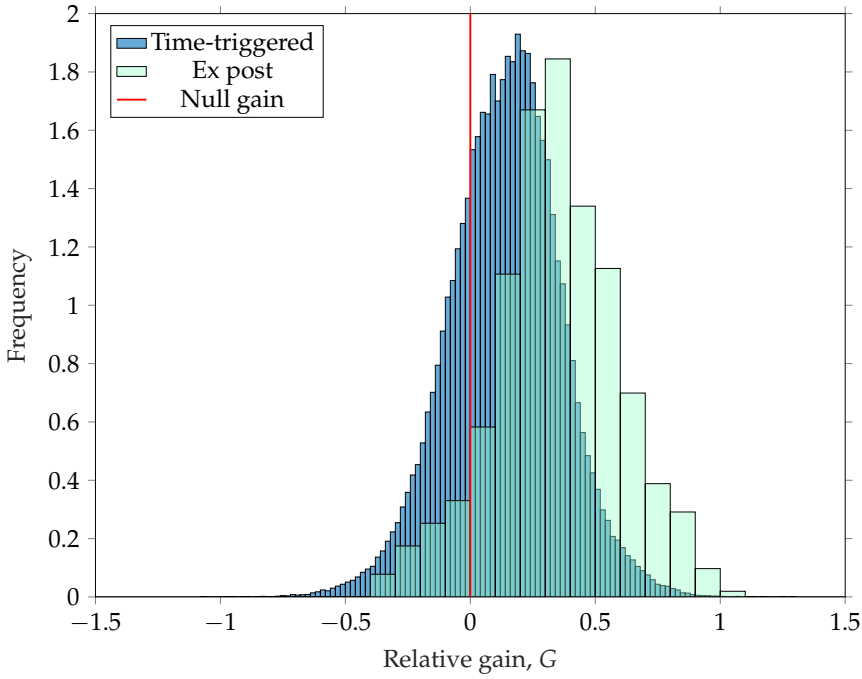



**Fig. 4.6** Histograms of the gain $G$ (see equation (4.14)) for the common benchmark (equations (4.12)). The histograms have been obtained over 100,000 and 500 simulations for the time-triggered and ex post particle filters, respectively.

*Results for the common benchmark (equations* (4.12)*)*

Figure 4.6 represents the histogram of the time-triggered and ex post gains $G$ for the common benchmark. The corresponding performance indicators are reported in Table 4.3. It is observed that both the time-triggered and ex post methods outperform the particle filter with regular measurement times. In addition, the performance difference between the time-triggered and ex post methods is important.

Finally, Figure 4.7 shows a trajectory of the common benchmark as well as the tracking obtained by the regular, time-triggered and ex post particle filters. The measurement times are also displayed.

**Table 4.3** Performance indicators for the time-triggered and ex post solutions on the common benchmark model (equations (4.12)). The considered system, the performance indicators, and the optimization and simulation parameters are described in Section 4.3.1.

| | Time-triggered | Ex post |
|---|---|---|
| Mean gain (± std) | 0.138 (± 0.222) | 0.345 (± 0.251) |
| Median gain | 0.146 | 0.337 |
| Proportion positive gain | 74.6 % | 91.7 % |

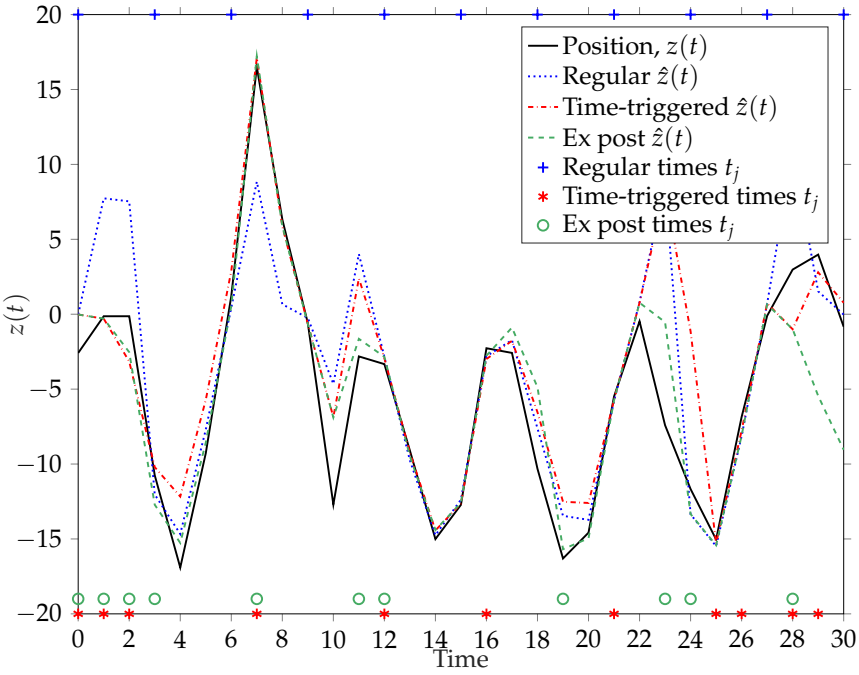



**Fig. 4.7** Comparison of a particular realization $z(t)$ with the filtered values $\hat{z}(t)$ obtained with a particle filter with regular measurement times and both the time-triggered and the ex post particle filters. The gains on the complete sequence are $G_{\text{time-triggered}} = 0.115$ and $G_{\text{ex post}} = 0.340$. Results are simulated from the common benchmark (equations (4.12)).

## 4.4 Conclusions

In this chapter, the problem of the optimal intermittent particle filter has been addressed. This consists in selecting the measurement times of the particle filter according to a certain optimality criterion.

We have proposed three variants of expected mean squared error minimization giving rise to three different intermittent filters: the self-triggered filter, the time-triggered filter, and the ex post filter.

A theorem has been proven that orders the performance of these three intermittent filters. Then, different algorithms to compute the measurement times of the time-triggered and ex post particle filters have been proposed: the random trial, greedy forward, greedy backward, simulated annealing, and genetic algorithms. These different optimization algorithms were compared on a tumor motion model inspired by real data. The genetic and simulated annealing algorithms are the two optimizers showing the best performance.

Finally, on this same tumor model, our time-triggered and ex post particle filters were compared with a particle filter with regular measurement times. The time-triggered and ex post particle filters outperform the regular particle filter in 76.8% and 82.2% of the cases, respectively. For a second example, the regular particle filter is outperformed in 74.6% and 91.7% of the cases by the time-triggered and ex post particle filters, respectively. Overall, our results demonstrate the added value of selecting measurement times according to system dynamics for the optimal estimation of a state from limited measurements.

Further work will focus on robustness analyses to determine how model uncertainties affect the performance of the proposed methods for situations in which the system dynamics are known only approximately. Other continuations of this work will consist of experimental validation on real-world data, for example, in the case of mobile tumor tracking.

We will also use deep Q-learning to solve the stochastic program described by Problem 4.1 and implement the self-triggered particle filter.

# 5

# Beyond mean squared error: Optimal sampling for robust estimation

In Chapter 3 we were interested in the case of a linear Gaussian system and we showed how to choose the measurement times in order to minimize the expected mean squared error. Here, we are also interested in linear systems, but we suppose that the disturbances are bounded and the objective is to choose the measurement times in order to minimize the worst-case estimation error.

Instead of solving this robust estimation problem directly, we will solve a control problem from which we can deduce the solution of the original problem, in the spirit of the duality principle (see Section 2.5). For this control problem, we simultaneously optimize the measurement times and the control times. Even if the control times have no interesting counterpart for the initial robust estimation problem, we present this generalization because it has applications in control.

Using Q-parameterization [SB10, YJB76] and a polytope containment method [BTN99, ST20], we prove in this chapter that the co-design of an affine feedback controller, a measurement schedule and a control schedule can be exactly formulated as a mixed integer linear program (MILP) with

2 binary variables per time step. As a consequence, this problem can be solved efficiently, even when an exhaustive search for measurement and control times would have been impossible in a reasonable amount of time.

The results presented in this chapter are the result of close collaboration with Kwesi Rutledge and were reported in

- Antoine Aspeel, Kwesi Rutledge, Raphaël M Jungers, Benoit Macq, and Necmiye Özay. Optimal control for linear networked control systems with information transmission constraints. In *The 60th IEEE International Conference on Decision and Control*, 2021.

## 5.1 Introduction

In this chapter, we are interested in deriving formal guarantees about reachability of a target set when performing control and measurement of a single agent over a network.

To our knowledge, the method proposed in this chapter is the first derivation of robust output feedback control design with measurement and control budget constraints. Other areas, such as parsimonious control [DSW13], also have results defining how to minimize the number of control actions that are taken in a distributed system setting. In the context of Q-parameterization, the quadratic invariance property has been studied. It has been shown that it is a necessary [LL11] and sufficient [RL06] condition for convexity. In this chapter, we have similar results for linearity. Then, the combinatorial structure allowing to co-design the controller, the measurement and control times will lead to a MILP and not only to a mixed integer convex program. Consequently, branch an bounds methods can be used to find an optimal solution efficiently on practical examples.

Note that some alternatives to Q-parameterization exist, e.g., disturbance rejection control [Gao06], or system level synthesis [WMD19].

### 5.1.1 Contribution

In this chapter, we develop a controller, which satisfies constraints on the amount of bandwidth used on the network by jointly optimizing (i) a measurement schedule, (ii) a control schedule, and (iii) the gains of the controller, while guaranteeing that the output variable remains in a safety set. The controller contains memory and a zero-order hold structure, which is partially illustrated in Figure 5.1. When the uncertainty sets in this problem

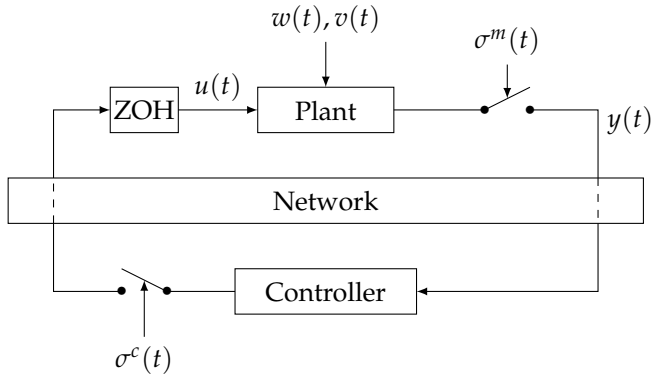



**Fig. 5.1** Block diagram representing the interaction between the plant (5.1) and the controller (5.2). ZOH means zero-order hold.

are defined as polytopes, this problem can be formulated as a robust optimization using the polytope containment methods discussed in [BTN99, ST20]. The output feedback controller, which would normally lead to complex nonconvex constraints can be parameterized linearly in terms of the optimization variables, using the methods of Q-parameterization [SB10, YJB76]. Then, the unique binary choices associated with the schedules introduce binary variables, making the optimization problem a MILP, which is tractable to solve on practical problems.

### 5.1.2 Chapter organization

The rest of this chapter is organized as follows: in Section 5.2, the main control problem we are interested in is defined. In Section 5.3 several related problems are presented. In particular, Section 5.3.2 shows the estimated counterpart of these problems and how to reduce them to control problems. In Section 5.4, we propose a solution to the problem using Q-parameterization and Mixed-Integer Linear Programming. Section 5.5 demonstrates our method on two examples and Section 5.6 concludes and proposes future works. The proofs of the lemmas are in Appendix C.

### 5.1.3 Notation

The set of real numbers and the empty set are denoted $\mathbb{R}$ and $\emptyset$, respectively. The set of $m \times n$ matrices with nonnegative real entries is $\mathbb{R}_+^{m\times n}$. Matrices are noted in upper case, e.g., $X$, and vectors in lower case, e.g., $x$.

For a matrix $X$, $X^\top$ is the transpose of $X$. The symbol $I_n$ represents the identity matrix of size $n$, $0_{m\times n}$ is the zero $m \times n$ matrix, $\mathbb{1}_n$ is the $n$ dimensional vector of ones. For the sake of brevity, dimensions are sometimes omitted when they can be inferred from compatibility. The Kronecker product is $\otimes$. Inequalities between vectors are considered element-wise. For $A \in \mathbb{R}^{m\times n}$ and $\mathcal{I} \subset \{1,\dots,m\}$, we write $A_{\mathcal{I},:}$ the $|\mathcal{I}| \times m$ matrix whose rows correspond to the rows of $A$ with indices in $\mathcal{I}$. The notation $A_{:,\mathcal{J}}$ is used similarly for the columns. For integers $i \leq j$, we write $i{:}j = \{i, i+1, \dots, j\}$.

In this chapter, calligraphic letters represent polytopes (except $\mathcal{I}$ and $\mathcal{J}$, which represent set of indices). For two polytopes $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A} \times \mathcal{B}$ is their Cartesian product; for a positive integer $n$, $\mathcal{A}^n$ is the $n$-th Cartesian power of $\mathcal{A}$; and $\partial\mathcal{A}$ denotes the boundary of $\mathcal{A}$.

## 5.2 Problem statement

We consider the following dynamical system

$$x(t+1) = Ax(t) + Bu(t) + w(t), \quad w(t) \in \mathcal{W}, \tag{5.1a}$$

$$y(t) = \begin{cases} Cx(t) + v(t), & \sigma^m(t) = 1 \\ \emptyset, & \sigma^m(t) = 0 \end{cases}, \quad v(t) \in \mathcal{V}, \tag{5.1b}$$

$$z(t) = Dx(t) + d. \tag{5.1c}$$

Quantities $w(t)$, $v(t)$, $x(0)$ are unknown but are contained in known sets $\mathcal{W}$, $\mathcal{V}$, $\mathcal{X}_0$, respectively. The measurement scheduling signal $\sigma^m(t) \in \{0,1\}$ determines if a measurement is acquired at time $t$. Let us emphasize the differences with Chapter 3. While in equations (3.1) the uncertainties follow distributions, in (5.1), the uncertainties are distribution-free, but are bounded. In addition, there is an input signal $u(t)$ in (5.1a).

We want to design the input signal $u(t)$ from the previous measurements $y(t)$. We restrict to linear controllers of the form

$$u(t) = \begin{cases} f_t + \sum_{\tau \leq t \text{ s.t. } \sigma^m(\tau)=1} F_{(t,\tau)} y(\tau), & \sigma^c(t) = 1, \\ u(t-1), & \sigma^c(t) = 0, \end{cases} \tag{5.2}$$

with $u(-1) = 0$. Column vectors $x(t)$, $y(t)$, $z(t)$ and $u(t)$ are respectively in $\mathbb{R}^{n_x}$, $\mathbb{R}^{n_y}$, $\mathbb{R}^{n_z}$ and $\mathbb{R}^{n_u}$. Other dimensions can be deduced from compatibility. The control scheduling signal $\sigma^c(t) \in \{0,1\}$ determines when a new control input is sent to the plant. Figure 5.1 represents the interaction

between equations (5.1) and (5.2).

*Remark* 5.1. When $\sigma^c(t) = 0$, instead of zero-order hold, we could set the input to zero, i.e., $u(t) = 0$. Both options are presented in [SSF+07], while [HTVdWN10] focuses on zero-order hold. ◇

We are interested in controlling the output $z(t)$ during a finite horizon of length $T$. Sensors and actuators are supposed to be constrained in their number of uses. This kind of situation occurs for example to save energy for sensors and actuators. Formally, we have a maximum number of measurements $N_m \geq 0$, and a maximum number of new control inputs $N_c \geq 0$. It is

$$\sum_{t=0}^{T-1} \sigma^m(t) \leq N_m \quad \text{and} \quad \sum_{t=0}^{T-1} \sigma^c(t) \leq N_c. \tag{5.3}$$

*Remark* 5.2. These constraints can be replaced by any linear constraints, i.e., $\sum_{t=0}^{T-1} \left(c_{it}^m \sigma^m(t) + c_{it}^c \sigma^c(t)\right) \leq b_i$, for $i = 1, \ldots, I$ where $c_{it}^m$, $c_{it}^c$ and $b_i$ are known constants. For example, instead of considering two separated budgets, one for the measurements and the other for the controls, we could consider a common budget $\sum_{t=0}^{T-1} \left(c^m \sigma^m(t) + c^c \sigma^c(t)\right) \leq N$. Budgets over a sliding window of length $\bar{t} \geq 1$ can also be considered, i.e., $\sum_{\tau=t-\bar{t}}^{t-1} \sigma(\tau) \leq N$, for $t = \bar{t}, \ldots, T$. ◇

A safety set $\mathcal{Z}$ is given for which our objective is to ensure $z(t) \in \mathcal{Z}$. In addition, we want the control input to stay inside a known set $\mathcal{U}$, i.e., $u(t) \in \mathcal{U}$. Furthermore, we include the following assumption.

*Assumption* 1. The sets $\mathcal{X}_0$, $\mathcal{W}$, $\mathcal{V}$, $\mathcal{U}$ and $\mathcal{Z}$ are (not necessarily bounded) convex polyhedra. ◇

It is assumed to have these polyhedra in H-representation. For $\mathcal{A} \in \{\mathcal{X}_0, \mathcal{W}, \mathcal{V}, \mathcal{U}, \mathcal{Z}\}$, we write $\mathcal{A} = \{x | H_{\mathcal{A}} x \leq h_{\mathcal{A}}\}$ and $n_{\mathcal{A}}$ the number of rows in the matrix $H_{\mathcal{A}}$ and in the column vector $h_{\mathcal{A}}$.

The problem we are interested in is to find measurement times, control times, control gains and control offsets, that keep the output variable $z(t)$ in the safety set $\mathcal{Z}$ during the complete horizon, i.e., for $t = 0, \ldots, T$.

**Problem 5.1** (Safety)**.**

$$\textit{Find } \{\sigma^m(t)\}_{t=0}^{T-1},\ \{\sigma^c(t)\}_{t=0}^{T-1},\ \{f_t\}_{t=0}^{T-1},\ \{F_{(t,\tau)}\}_{t=0,\,\tau=0}^{T-1,\,t},$$

*such that*

- *The dynamics* (5.1) *holds,*

- *The controller* (5.2) *is used,*
- $\sigma^m(t), \sigma^c(t) \in \{0,1\}$ *for* $t = 0, \dots, T-1$*,*
- *Budget constraints (5.3) hold,*
- *For all* $w(t) \in \mathcal{W}$*,* $v(t) \in \mathcal{V}$*, for* $t = 0, \dots, T-1$*, and for all* $x(0) \in \mathcal{X}_0$*, it holds that* $z(t) \in \mathcal{Z}$ *for* $t = 0, \dots, T$*, and* $u(t) \in \mathcal{U}$ *for* $t = 0, \dots, T-1$*.*

*Remark* 5.3. It is possible to add an objective function for minimizing a cost related to measurements and controls, e.g., $\sum_{t=0}^{T-1} (c^m \sigma^m(t) + c^c \sigma^c(t))$, where $c^m$ and $c^c$ are given non negative constants.

In addition, one could add the constraint $x(T) \in \mathcal{X}_0$. Then, a solution to Problem 5.1 can be applied periodically to keep $z(t) \in \mathcal{Z}$ and $u(t) \in \mathcal{U}$ for all $t \geq 0$. ◇

## 5.3 Related problems

In this section, problems related to Problem 5.1 are presented.

### 5.3.1 Safety as long as possible

In this problem, one wants to keep the output safe, i.e., $z(t) \in \mathcal{Z}$ as long as possible. We assume to have an *a priori* upper bound $\bar{T}$ on the largest such time. This can be formalized as follows.

**Problem 5.2** (Safety as long as possible)**.**

$$\max_{\substack{T \in \{0,\dots,\bar{T}\} \\ \{\sigma^m(t)\}_{t=0}^{\bar{T}-1}, \{\sigma^c(t)\}_{t=0}^{\bar{T}-1} \\ \{f_t\}_{t=0}^{\bar{T}-1}, \{F_{(t,\tau)}\}_{t=0,\,\tau=0}^{\bar{T}-1,\,t}}} T,$$

*such that*

- *Equations (5.1) and (5.2) hold for* $t = 0, \dots, \bar{T}-1$*,*
- $\sigma^m(t), \sigma^c(t) \in \{0,1\}$ *for* $t = 0, \dots, \bar{T}-1$*,*
- *Budget constraints (5.3) hold with* $\bar{T}$ *instead of* $T$*,*
- *For all* $w(t) \in \mathcal{W}$*,* $v(t) \in \mathcal{V}$*, for* $t = 0, \dots, T-1$*, and for all* $x(0) \in \mathcal{X}_0$*, it holds that* $z(t) \in \mathcal{Z}$ *for* $t = 0, \dots, T$*, and* $u(t) \in \mathcal{U}$ *for* $t = 0, \dots, T-1$*.*

Problem 5.2 consists of finding the largest $T$ such that Problem 5.1 is feasible. Note that if Problem 5.1 is feasible for some $T'$, then it is also feasible for all $T \leq T'$. Then, a binary search algorithm can be used to find the optimal $T$ by solving $O\left(\log_2(\bar{T})\right)$ instances of Problem 5.1. In Section 5.5.2, we solve Problem 5.2 on an example.

### 5.3.2 From control to estimation

This section presents the estimation counterparts of Problems 5.1 and 5.2 and how to reduce these estimation problems to the control problems.

One wants to estimate the output $z(t)$ of the system (5.1) from noisy measurements $y(\tau)$ ($\tau \leq t$). To this end, we consider the following estimator

$$\hat{x}(t+1) = A\hat{x}(t) + Bu(t) - \gamma(t), \tag{5.4a}$$
$$\hat{y}(t) = C\hat{x}(t), \tag{5.4b}$$
$$\hat{z}(t) = D\hat{x}(t) + d, \tag{5.4c}$$

with feedback:

$$\gamma(t) = f_t + \sum_{\tau \leq t \text{ s.t. } \sigma^m(\tau)=1} F_{(t,\tau)}\bar{y}(\tau), \tag{5.5}$$

where $\bar{y}(\tau) = y(\tau) - \hat{y}(\tau)$.

Taken together, (5.1) and (5.4) may be combined to define a dynamical system for the differences $\bar{x}(t) = x(t) - \hat{x}(t)$, and the estimation error $\bar{z}(t) = z(t) - \hat{z}(t)$

$$\bar{x}(t+1) = A\bar{x}(t) + \gamma(t) + w(t) \tag{5.6a}$$
$$\bar{y}(t) = \begin{cases} C\bar{x}(t) + v(t), & \sigma^m(t) = 1, \\ \emptyset, & \sigma^m(t) = 0. \end{cases} \tag{5.6b}$$
$$\bar{z}(t) = D\bar{x}(t). \tag{5.6c}$$

*Remark* 5.4. Equations (5.5) and (5.6) give (5.1) and (5.2) under transformation

$$B = I_{n_x},\ d = 0_{n_z \times 1},\ \sigma^c(t) = 1,\ x(t) = \bar{x}(t),$$
$$y(t) = \bar{y}(t),\ z(t) = \bar{z}(t),\ u(t) = \gamma(t).$$

Then, solving Problem 5.1 with $N_c = T - 1$ and ignoring the constraints $u(t) \in \mathcal{U}$, gives an observer, which keeps the estimation error $\bar{z}(t)$ in $\mathcal{Z}$ for all $t = 0, \dots, T$, for all $x(0) \in \mathcal{X}_0$, $w(t) \in \mathcal{W}$, $v(t) \in \mathcal{V}$. On the other hand, solving problem 5.2 gives an observer, which keeps the error $\bar{z}(t)$ in $\mathcal{Z}$ as long as possible, for all $x(0) \in \mathcal{X}_0$, $w(t) \in \mathcal{W}$, $v(t) \in \mathcal{V}$. Note that in these cases, $\sigma^c(t)$ are constants (equal to 1) and not optimization variables. ◇

We see that the control setting is more general than the estimation setting because we do not choose the control times $\sigma^c(t)$ in the estimation case. The dual version of the control times would be to replace the equation (5.5) by

$$\gamma(t) = \begin{cases} f_t + \sum_{\tau \leq t \text{ s.t. } \sigma^m(\tau)=1} F_{(t,\tau)} \bar{y}(\tau), & \sigma^c(t) = 1, \\ \gamma(t-1), & \sigma^c(t) = 0. \end{cases}$$

However, choosing $\sigma^c(t) = 0$ has no interest here because there is no reason not to update the estimate. Unlike the control problem, having $\gamma(t) \neq \gamma(t-1)$ does not imply any communication between the filtering device and the plant.

## 5.4 Methods

In this section, we show that a feasible solution of Problem 5.1 can be obtained by solving a MILP with $2T$ binary variables. In Section 5.4.1, we show that missing measurements and controls can be modeled as linear indicator constraints on the gains $F_{(t,\tau)}$ and $f_t$. Section 5.4.2 recalls the Q-parameterization approach, which leads to express the trajectories of $z(t)$ and $u(t)$ as linear transformation of the uncertainties $w(t)$, $v(t)$ and $x(0)$. It leads to the introduction of new design variables $Q$ and $r$. In Section 5.4.3 we use polytope containment techniques to handle the robustness constraints $z(t) \in \mathcal{Z}$ and $u(t) \in \mathcal{U}$ for all uncertainties $w(t)$, $v(t)$ and $x(0)$. This leads to linear constraints in terms of the new decision variables $Q$ and $r$. Then, it is shown in Section 5.4.4 that the indicator constraints on $F_{(t,\tau)}$ and $f_t$ for missing measurements and controls can be expressed as linear indicator constraints in terms of the new design variables $Q$ and $r$. Finally, Section 5.4.5 combines the previous results to prove that Problem 5.1 is equivalent to a MILP.

### 5.4.1 Missing measurements and controls

There are different ways to deal with missing measurements. A common way is to set the measurement matrix $C$ to zero when no measurements are taken, i.e., when $\sigma^m(t) = 0$, and treat the system as time-varying [JKH18]. In our case, the measurement times are decision variables and such an approach would be equivalent to considering part of the dynamics of the system as a variable of the problem. This would lead to non-linearities.

For this reason, we deal with missing measurements in a different way. We consider that a measurement is available at each time step but that the measurements that should be missing are forbidden to the controller. For this reason, the gains associated with prohibited measurements are set to zero, i.e., $\sigma^m(\tau) = 0$ implies $F_{(t,\tau)} = 0$ for all $t$. Intuitively, we do not consider missing measurements, but prohibited measurements instead. Consequently, the choice of measurement times does not translate as a choice on the dynamics of the system, but as a constraint on the controller's gains. This idea is taken up in the following lemma.

**Lemma 5.5** (Forbidden measurements)**.** *For given $x(0)$, $w(t)$, $v(t)$, $\sigma^m(t)$ and $\sigma^c(t)$ for $t = 0, \dots, T-1$, the sequences $\{z(t)\}_{t=0}^{T}$ and $\{u(t)\}_{t=0}^{T-1}$ generated by (5.1) and (5.2) are the same as the ones generated by*

$$x(t+1) = Ax(t) + Bu(t) + w(t), \tag{5.7a}$$
$$y(t) = Cx(t) + v(t), \tag{5.7b}$$
$$z(t) = Dx(t) + d \tag{5.7c}$$
$$u(t) = f_t + \sum_{\tau \le t} F_{(t,\tau)} y(\tau), \tag{5.7d}$$

*when for all $\tau = 0, \dots, T-1$,*

$$\sigma^m(\tau) = 0 \Rightarrow \Big( F_{(t,\tau)} = 0, \ \textit{for all} \ \ t = \tau, \dots T-1 \Big), \tag{5.8}$$

*and for all $t = 0, \dots, T-1$,*

$$\sigma^c(t) = 0 \Rightarrow \begin{cases} f_t = f_{t-1}, \\ F_{(t,\tau)} = F_{(t-1,\tau)}, \textit{ for } \tau = 0, \dots, t \end{cases} \tag{5.9}$$

*with $f_{-1} = 0$ and $F_{(-1,\tau)} = F_{(t,t-1)} = 0$.*

*Proof.* See Appendix C.1. □

### 5.4.2 Trajectories and Q-parameterization

This section follows the presentation of the Q-parameterization from [RYO20]. First, let us define the dynamical variables of (5.7) in terms of trajectories

$$x = \begin{bmatrix} x(0)^\top & x(1)^\top & \cdots & x(T)^\top \end{bmatrix}^\top, \tag{5.10a}$$
$$z = \begin{bmatrix} z(0)^\top & z(1)^\top & \cdots & z(T)^\top \end{bmatrix}^\top, \tag{5.10b}$$
$$u = \begin{bmatrix} u(0)^\top & u(1)^\top & \cdots & u(T-1)^\top \end{bmatrix}^\top, \tag{5.10c}$$
$$w = \begin{bmatrix} w(0)^\top & w(1)^\top & \cdots & w(T-1)^\top \end{bmatrix}^\top, \tag{5.10d}$$
$$v = \begin{bmatrix} v(0)^\top & v(1)^\top & \cdots & v(T-1)^\top \end{bmatrix}^\top, \tag{5.10e}$$
$$f = \begin{bmatrix} f_0^\top & f_1^\top & \cdots & f_{T-1}^\top \end{bmatrix}^\top, \tag{5.10f}$$
$$\sigma^m = \begin{bmatrix} \sigma^m(0) & \sigma^m(1) & \cdots & \sigma^m(T-1) \end{bmatrix}^\top, \tag{5.10g}$$
$$\sigma^c = \begin{bmatrix} \sigma^c(0) & \sigma^c(1) & \cdots & \sigma^c(T-1) \end{bmatrix}^\top, \tag{5.10h}$$

$$F = \begin{bmatrix} F_{(0,0)} & 0 & 0 & \cdots & 0 \\ F_{(1,0)} & F_{(1,1)} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ F_{(T-1,0)} & F_{(T-1,1)} & F_{(T-1,2)} & \cdots & F_{(T-1,T-1)} \end{bmatrix}. \tag{5.11}$$

Note that $F$ and $f$ characterize the control input and are decision variables. The state trajectory $x$ (along with several other trajectories) is a nonlinear function of the decision variables $F$ and $f$:

$$\begin{aligned} x = (H + SF(I - \bar{C}SF)^{-1}\bar{C}H)w + SF(I - \bar{C}SF)^{-1}v + \\ (I + SF(I - \bar{C}SF)^{-1}\bar{C})Jx(0) + S(I + Q\bar{C}S)f, \end{aligned} \tag{5.12}$$

where

$$J = \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^T \end{bmatrix}, \; H = \begin{bmatrix} 0_{n_x \times n_x} & 0 & 0 & \cdots & 0 \\ I & 0 & 0 & \cdots & 0 \\ A & I & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A^{T-1} & A^{T-2} & A^{T-3} & \cdots & I \end{bmatrix},$$

$$\bar{C} = \begin{bmatrix} I_T \otimes C & 0_{Tn_y \times n_x} \end{bmatrix},$$

$$S = \begin{bmatrix} 0_{n_x \times n_u} & 0 & 0 & \cdots & 0 \\ B & 0 & 0 & \cdots & 0 \\ AB & B & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A^{T-1}B & A^{T-2}B & A^{T-3}B & \cdots & B \end{bmatrix}.$$

Thus, in view of (5.12), efficiently searching the set of feasible gains ($F$ and $f$) for Problem 5.1 would be a search over a nonconvex set. However, from [RYO20, Theorem 2, equation (17)], the following also holds

$$x = (H + SQ\bar{C}H)w + SQv + (I + SQ\bar{C})Jx(0) + Sr, \tag{5.13a}$$
$$u = Q\bar{C}Hw + Qv + Q\bar{C}Jx(0) + r, \tag{5.13b}$$

where the following Q-parameterization mapping is used

$$Q = F(I - \bar{C}SF)^{-1}, \tag{5.14a}$$
$$r = (I + Q\bar{C}S)f, \tag{5.14b}$$

and gives a $n_x \times n_y$ block lower triangular matrix $Q$. Conversely, if $Q$ is $n_x \times n_y$ block lower triangular, then this mapping is invertible and the inverse mapping is

$$F = (I + Q\bar{C}S)^{-1}Q, \tag{5.15a}$$
$$f = (I + Q\bar{C}S)^{-1}r. \tag{5.15b}$$

This mapping allows to express our problem in terms of $Q$ and $r$ instead of $F$ and $f$.

In addition, let us write

$$\bar{D} = I_{T+1} \otimes D,\ \bar{d} = \mathbb{1}_{T+1} \otimes d.$$

Then, using $z = \bar{D}x + \bar{d}$ and (5.13), one can write

$$\begin{bmatrix} z \\ u \end{bmatrix} = \begin{bmatrix} P_{zw} & P_{zv} & P_{zx_0} \\ P_{uw} & P_{uv} & P_{ux_0} \end{bmatrix} \begin{bmatrix} w \\ v \\ x(0) \end{bmatrix} + \begin{bmatrix} \tilde{z} \\ \tilde{u} \end{bmatrix}, \tag{5.16}$$

where

$$\begin{array}{lll} P_{zw} = \bar{D}(H + SQ\bar{C}H), & P_{zv} = \bar{D}SQ, & P_{zx_0} = \bar{D}(I + SQ\bar{C})J, \\ P_{uw} = Q\bar{C}H, & P_{uv} = Q, & P_{ux_0} = Q\bar{C}J, \end{array} \tag{5.17}$$

$$\tilde{z} = \bar{D}Sr + \bar{d} \quad \text{and} \quad \tilde{u} = r. \tag{5.18}$$

Note that these quantities depend linearly on the new decision variables $Q$ and $r$, which is at the heart of Q-parameterization.

### 5.4.3 Robustness constraints by polytope containment

Problem 5.1 contains the robust constraints $z(t) \in \mathcal{Z}$ and $u(t) \in \mathcal{U}$ for all $w(t)$, $v(t)$ and $x(0)$. To deal with such constraints, we use polytope containment techniques. To this end, we will need the following extension of the Farkas' lemma.

**Lemma 5.6** (H-Polytope in H-Polytope)**.** *Let $\mathcal{A} = \{x \in \mathbb{R}^n \mid H_{\mathcal{A}}x \leq h_{\mathcal{A}}\}$, $\mathcal{B} = \{x \in \mathbb{R}^n \mid H_{\mathcal{B}}x \leq h_{\mathcal{B}}\} \subset \mathbb{R}^n$, $H_{\mathcal{A}} \in \mathbb{R}^{n_{\mathcal{A}} \times n}$, $H_{\mathcal{B}} \in \mathbb{R}^{n_{\mathcal{B}} \times n}$. We have $\mathcal{A} \subseteq \mathcal{B}$ if and only if*

$$\exists \Lambda \in \mathbb{R}_+^{n_{\mathcal{B}} \times n_{\mathcal{A}}} \text{ such that } \Lambda H_{\mathcal{A}} = H_{\mathcal{B}},\ \Lambda h_{\mathcal{A}} \leq h_{\mathcal{B}}.$$

*Proof.* See [Hen89]. □

We use Lemma 5.6 to prove the following lemma, which allows us to handle the robustness constraint $z(t) \in \mathcal{Z}$.

**Lemma 5.7** (Safety constraint)**.** *Let the sequence $\{z(t)\}_{t=0}^T$ be generated by (5.7). The relation $z(t) \in \mathcal{Z}$ holds for all $t = 0, \ldots, T$, for all $w(t) \in \mathcal{W}$, for all $v(t) \in \mathcal{V}$ and for all $x(0) \in \mathcal{X}_0$ if and only if there exists $\Lambda \in \mathbb{R}_+^{(T+1)n_{\mathcal{Z}} \times (T(n_{\mathcal{W}}+n_{\mathcal{V}})+n_{\mathcal{X}_0})}$ such that*

$$\Lambda \begin{bmatrix} I_T \otimes H_{\mathcal{W}} & 0 & 0 \\ 0 & I_T \otimes H_{\mathcal{V}} & 0 \\ 0 & 0 & H_{\mathcal{X}_0} \end{bmatrix} = (I_{T+1} \otimes H_{\mathcal{Z}}) \begin{bmatrix} P_{zw} & P_{zv} & P_{zx_0} \end{bmatrix}, \tag{5.19}$$

*and*

$$\Lambda \begin{bmatrix} \mathbb{1}_T \otimes h_{\mathcal{W}} \\ \mathbb{1}_T \otimes h_{\mathcal{V}} \\ h_{\mathcal{X}_0} \end{bmatrix} \leq \mathbb{1}_{T+1} \otimes h_{\mathcal{Z}} - (I_{T+1} \otimes H_{\mathcal{Z}})\,\tilde{z}, \tag{5.20}$$

*where $P_{zw}$, $P_{zv}$, $P_{zx_0}$ are defined in (5.17) and $\tilde{z}$ is defined in (5.18).*

*Proof.* See Appendix C.2. □

We can state a similar result for the bounded input constraint $u(t) \in \mathcal{U}$.

**Lemma 5.8** (Bounded inputs)**.** *Let the sequence $\{u(t)\}_{t=0}^{T-1}$ be generated by (5.7). The relation $u(t) \in \mathcal{U}$ holds for all $t = 0, \dots, T-1$, for all $v(t) \in \mathcal{V}$, for all $w(t) \in \mathcal{W}$ and for all $x(0) \in \mathcal{X}_0$ if and only if there exists $\Gamma \in \mathbb{R}_+^{Tn_\mathcal{U} \times (T(n_\mathcal{W}+n_\mathcal{V})+n_{\mathcal{X}_0})}$ such that*

$$\Gamma \begin{bmatrix} I_T \otimes H_\mathcal{W} & 0 & 0 \\ 0 & I_T \otimes H_\mathcal{V} & 0 \\ 0 & 0 & H_{\mathcal{X}_0} \end{bmatrix} = (I_T \otimes H_\mathcal{U}) \begin{bmatrix} P_{uw} & P_{uv} & P_{ux_0} \end{bmatrix}, \tag{5.21}$$

*and*

$$\Gamma \begin{bmatrix} \mathbb{1}_T \otimes h_\mathcal{W} \\ \mathbb{1}_T \otimes h_\mathcal{V} \\ h_{\mathcal{X}_0} \end{bmatrix} \leq \mathbb{1}_T \otimes h_\mathcal{U} - (I_T \otimes H_\mathcal{U})\, \tilde{u}, \tag{5.22}$$

*where $P_{uw}$, $P_{uv}$, $P_{ux_0}$ are defined in (5.17) and $\tilde{u}$ is defined in (5.18).*

*Proof.* See Appendix C.3. □

*Remark* 5.9. Note that Lemmas 5.7 and 5.8 give linear constraints on $P_{zw}$, $P_{zv}$, $P_{zx_0}$, $P_{uw}$, $P_{uv}$ and $P_{ux_0}$, which are linear functions of $Q$ and $r$. Consequently, these constraints are linear in the decision variables $Q$ and $r$. Thus, a feasibility problem, which once contained non-convex constraints on the decision variables $F$ and $f$ can be transformed into an equivalent feasibility problem with convex constraints on the decision variables $Q$ and $r$. ◇

### 5.4.4 Missing measurements and controls revisited

The two following Lemmas state how the forbidden measurements constraint (5.8) and the missing control constraint (5.9) can be expressed linearly in terms of the new design variables $Q$ and $r$.

**Lemma 5.10** (Missing measurements)**.** *Let a binary measurement signal $\sigma^m \in \{0,1\}^T$ and let $\{F_{(t,\tau)}\}_{t=0,\,\tau=0}^{T-1,\,t}$ be some control gains. Then, (5.8) holds for $\tau = 0, \dots, T-1$ if and only if*

$$\sigma^m(\tau) = 0 \Rightarrow Q_{:,\mathcal{J}_\tau} = 0, \text{ for } \tau = 0, \dots, T-1, \tag{5.23}$$

*where $Q$ is defined by (5.11) and (5.14a) and where $\mathcal{J}_\tau = 1 + \tau n_y : (\tau+1)n_y$.*

*Proof.* See Appendix C.4. □

**Lemma 5.11** (Missing controls)**.** *Let a binary control signal $\sigma^c \in \{0,1\}^T$ and let $\{f_t\}_{t=0}^{T-1}$ and $\{F_{(t,\tau)}\}_{t=0,\,\tau=0}^{T-1,\,t}$ be some control offsets and control gains. Then, (5.9) holds for all $t = 0, \dots, T-1$ if and only if the following indicator constraints hold,*

$$\text{for } t = 0, \dots, T-1,\ \sigma^c(t) = 0 \Rightarrow \begin{cases} Q_{\mathcal{I}_t,:} = Q_{\mathcal{I}_{t-1},:}, \\ r_{\mathcal{I}_t} = r_{\mathcal{I}_{t-1}}, \end{cases} \tag{5.24}$$

*where $\mathcal{I}_t = 1 + tn_u : (t+1)n_u$, $Q_{\mathcal{I}_{-1},:} = 0$ and $r_{\mathcal{I}_{-1}} = 0$.*

*Proof.* See Appendix C.5. □

Similar results can be obtained with quadratic invariance. Remark that Lemmas 5.10 and 5.11 prove that the constraints are linear (and not just convex) in $Q$ and $r$.

### 5.4.5 Main result

Before stating our main result, recall that indicator constraints such that (5.23) and (5.24) can be handled as mixed integer linear constraints thanks to the Big-M formulation. The Big-M formulation uses the fact that for $\sigma \in \{0,1\}$, the indicator constraint $\sigma = 0 \Rightarrow f(x) = 0$ is equivalent to $-\sigma\bar{M} \leq f(x) \leq \sigma\bar{M}$ for $\bar{M}$ sufficiently large.

**Theorem 5.12.** *Problem 5.1 is feasible if and only if the following MILP is feasible.*

**Problem 5.3.**

$$\text{Find } \sigma^m,\ \sigma^c,\ Q,\ r,\ \Lambda \geq 0,\ \Gamma \geq 0,$$

*such that*

- $\sigma^m(t), \sigma^c(t) \in \{0,1\}$ *for* $t = 0, \dots, T-1$,
- *Budget constraints (5.3) hold,*
- *Matrix $Q$ is $n_x \times n_y$ block lower triangular,*
- *Safety constraints (5.19) and (5.20) hold,*
- *Bounded input constraints (5.21) and (5.22) hold,*
- *Measurement compatibility constraint (5.23) holds,*

- *Control compatibility constraint (5.24) holds.*

*Moreover, the feasible values of $\sigma^m$ and $\sigma^c$ are the same for both problems and the feasible $Q$ and $r$ are linked by transformations (5.15) to the feasible values of $\{f_t\}_{t=0}^{T-1}$ and $\{F_{(t,\tau)}\}_{t=0,\,\tau=0}^{T-1,\,t}$ for Problem 5.1.*

*Proof.* From Lemma 5.5, the sequences $\{z(t)\}_{t=0}^{T}$ and $\{u(t)\}_{t=0}^{T-1}$ generated by (5.1) and (5.2) are the same than the ones generated by (5.7) when (5.8) and (5.9) hold.

Then, Lemma 5.7 indicates that (5.7) and the robustness constraint $z(t) \in \mathcal{Z}$ for $t = 0, \dots, T$ hold if and only if (5.19) and (5.20) hold. Similarly, Lemma 5.8 shows that the constraint $u(t) \in \mathcal{U}$ for $t = 0, \dots, T-1$ holds if and only if (5.21) and (5.22) hold.

From Lemma 5.10, (5.8) holds if and only if (5.23) holds, and thanks to Lemma 5.11, (5.9) holds if and only if (5.24) holds. Finally, the indicator constraints can be handled using the Big-M method. □

*Remark* 5.13. In Problem 5.1, one may want to shrink the safety set $\mathcal{Z}$ as much as possible to have a solution as safe as possible. To do so, we introduce a scaling factor $\alpha \geq 0$ as an optimization variable to minimize and we replace the constraint $z(t) \in \mathcal{Z}$ by

$$z(t) - c \in \alpha(\mathcal{Z} - c) \coloneqq \{\alpha(z - c) | H_{\mathcal{Z}} z \leq h_{\mathcal{Z}}\}, \tag{5.25}$$

where $c \in \mathbb{R}^{n_z}$ is the scaling center and is not a design variable. It is typically chosen in the interior of $\mathcal{Z}$. Using the change of variable $\tilde{z} \coloneqq \alpha(z - c)$, constraint (5.25) can be rewritten

$$\begin{array}{rrcl} & z(t) - c & \in & \{\tilde{z} | H_{\mathcal{Z}} \tilde{z} \leq \alpha(h_{\mathcal{Z}} - H_{\mathcal{Z}} c)\} \\ \Leftrightarrow & H_{\mathcal{Z}}(z(t) - c) & \leq & \alpha(h_{\mathcal{Z}} - H_{\mathcal{Z}} c) \\ \Leftrightarrow & H_{\mathcal{Z}} z(t) & \leq & \alpha(h_{\mathcal{Z}} - H_{\mathcal{Z}} c) + H_{\mathcal{Z}} c, \end{array}$$

which can be interpreted as a constraint of the form $z(t) \in \tilde{\mathcal{Z}}$. Consequently, this constraint can be encoded by replacing equation (5.20) by

$$\Lambda \begin{bmatrix} \mathbb{1}_T \otimes h_{\mathcal{W}} \\ \mathbb{1}_T \otimes h_{\mathcal{V}} \\ h_{\mathcal{X}_0} \end{bmatrix} \leq \mathbb{1}_{T+1} \otimes \Big(\alpha(h_{\mathcal{Z}} - H_{\mathcal{Z}} c) + H_{\mathcal{Z}} c\Big) - (I_{T+1} \otimes H_{\mathcal{Z}})\, \tilde{z},$$

which is still linear in $\Lambda$, $\tilde{z}$, but also in $\alpha$. Consequently, this modified problem also reduces to a MILP.

For the corresponding estimation problem described in Section 5.3.2, if we choose $\mathcal{Z} = [-1,1]^{n_z}$ and $c = 0_{n_z \times 1}$, then it corresponds to solve

$$\min_{\{\sigma^m(t)\}_{t=0}^{T-1}, \{f_t\}_{t=0}^{T-1}, \{F_{(t,\tau)}\}_{t=0,\,\tau=0}^{T-1,\,t}} \quad \max_{\substack{t=0,\dots,T \\ v(t)\in\mathcal{V}, w(t)\in\mathcal{W}, x(0)\in\mathcal{X}_0}} \|\bar{z}(t)\|_\infty,$$

which is minimizing the worst case estimation error.

Finally, instead of minimizing the scaling of $\mathcal{Z}$, one could minimize the scaling of $\mathcal{U}$, or a combination of both. ◇

Solving such a MILP can be done efficiently on many practical situations using a branch and bound approach. This is illustrated in Section 5.5. In addition, this problem is solved offline, before any measurements are received. Then, the feasible solution can be efficiently used online.

## 5.5 Numerical examples and discussions

A JULIA code using GUROBI [GO21] generating the figures and implementing the algorithms is available at `https://github.com/kwesiRutledge/measurement-scheduling0/tree/master/examples/`. The reported computation times have been obtained on a laptop with an `Intel(R) Core(TM) i5-7267U 3.1 GHz` processor.

### 5.5.1 A Double Integrator Drone System

The task of remote monitoring of an area (e.g., for the remote reading of pressure guages in power plants) is one of the more recent topics of interest in cyber-physical systems [BDT18]. Frequently implemented on collections of robots including drones, quadripeds, and more, remote monitoring controllers must be scalable enough to be implemented on large numbers of systems at once while also avoiding exchanging too many messages and burdening the network. It offers an excellent problem to solve with our method because each robot's tasks can be encoded as reachability problems or reach-avoid problems (i.e. reach a target set while avoiding an unsafe set), which are readily handled with this method.

If one assumes that a drone's $x$ position $p_t^x$ and $y$ position $p_t^y$ are controlled by a simple force input, then the following dynamics may be written

$$\begin{aligned} \ddot{p}_t^x &= u_t^x, \\ \ddot{p}_t^y &= u_t^y. \end{aligned}$$

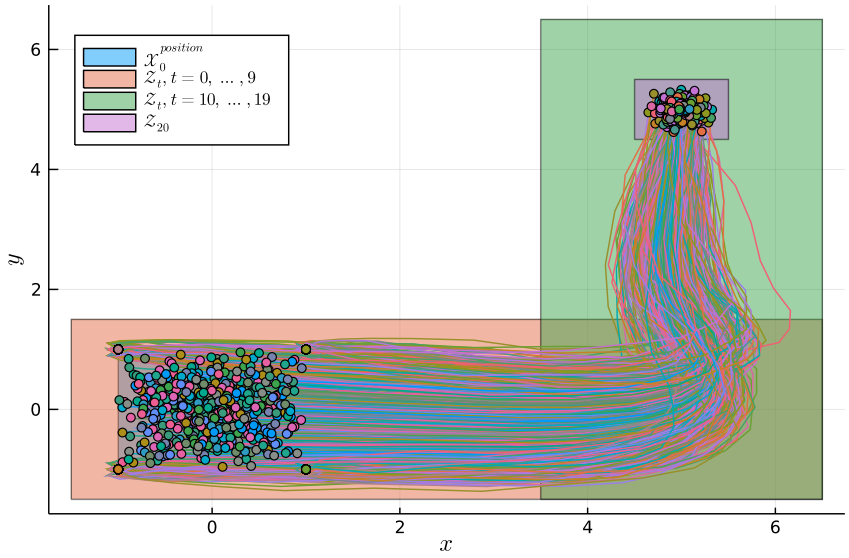



**Fig. 5.2** Representation of 1000 trajectories $z(t)$ sampled from the system described in Section 5.5.1. The admissible set of initial positions is indicated as $\mathcal{X}_0^{\text{position}}$ and the three different safety sets $\mathcal{Z}_t$ are represented.

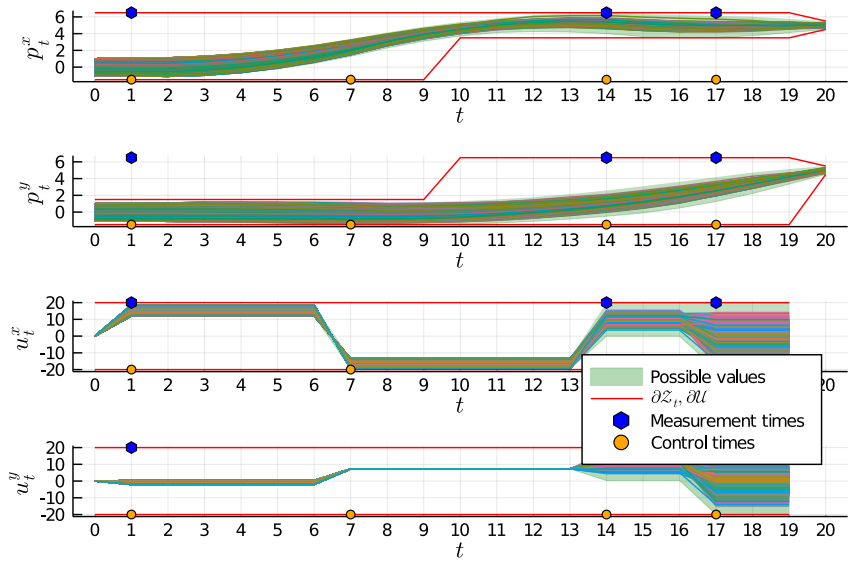



**Fig. 5.3** Representation of 1000 simulated trajectories ($z(t)$ and $u(t)$) of the system described in Section 5.5.1. Measurement and control times are indicated. The boundaries of the safety sets $\partial\mathcal{Z}_t$ and the boundaries of the set of admissible inputs $\partial\mathcal{U}$ are depicted. Finally, the sets of possible values for $z(t)$ and $u(t)$ for all possible noises are also depicted.

The state and the input of the system are respectively

$$x(t) = \begin{bmatrix} p_t^x & p_t^y & \dot{p}_t^x & \dot{p}_t^y \end{bmatrix}^\top, \; u(t) = \begin{bmatrix} u_t^x & u_t^y \end{bmatrix}^\top,$$

and the dynamics is

$$\dot{x}(t) = \begin{bmatrix} 0_{2\times 2} & I_2 \\ 0_{2\times 2} & 0_{2\times 2} \end{bmatrix} x(t) + \begin{bmatrix} 0_{2\times 2} \\ I_2 \end{bmatrix} u(t).$$

Finally, the output variable and the measurements are

$$z(t) = \begin{bmatrix} p_t^x & p_t^y \end{bmatrix}^\top, \; y(t) = z(t) + v(t).$$

We consider an exactly discretized version of this system with a discretization time step of 0.1. In addition, a process noise is considered. The time horizon is $T = 20$ time steps with $N_m = 3$ measurements and $N_c = 4$ control inputs. In addition, we define $\mathcal{W} = [-0.05, 0.05]^4$, $\mathcal{V} = [-0.05, 0.05]^2$, $\mathcal{X}_0 = [-1, 1]^2 \times \{0\}^2$ and $\mathcal{U} = [-20, 20]^2$. Finally, the safety set $\mathcal{Z}_t$ is time varying and takes 3 different values for respectively $t = 0, \ldots, 9$; $t = 10, \ldots, 19$; and $t = 20$. They are represented in Figure 5.2. This problem is solved in 292 seconds.

Figure 5.3 shows 1000 simulated trajectories. For half of them, the $x(0)$, $w(t)$ and $v(t)$ are uniformly sampled in the polytopes. For the other half of the trajectories, each uncertain variable is randomly sampled among the vertices of the respective noise polytopes in order to promote extreme trajectories. The measurement and control times are indicated, as well as the boundaries of the safety sets $\partial\mathcal{Z}_t$ and the boundaries of admissible inputs sets $\partial\mathcal{U}$. In addition, the set of possible values for $z(t)$ and $u(t)$ for all admissible uncertainty $w(t)$, $v(t)$ and $x(0)$ are represented. The same trajectories are depicted in Figure 5.2 where the set of admissible initial positions of the drone is indicated (and written $\mathcal{X}_0^{\text{position}}$), in addition to the time-varying safety sets.

We can verify that all simulated trajectories respect the constraints $z(t) \in \mathcal{Z}$ and $u(t) \in \mathcal{U}$. In addition, the measurement and control times are not regularly spaced. In fact, if we impose the measurement and control times to be spaced in time as regularly as possible, i.e., $\sigma^m(t) = 1$ for $t \in \{0, 10, 19\}$ and $\sigma^c(t) = 1$ for $t \in \{0, 6, 13, 19\}$, then the problem is infeasible.

Finally, the left part of Table 5.1 presents the evolution of the solver time

**Table 5.1** Left: Problem 5.1 is solved for the system described in Section 5.5.1 with time horizon $T = 20$ with a varying number of drones (and thus with varying state dimension $n_x$). Right: Problem 5.1 is solved for the system described in Section 5.5.2 with a varying time horizon $T$ (and the number of measurement and control times schedulings $\sigma^m$ and $\sigma^c$) with $N_m = N_c = T/2$ measurements and controls.

| Number of Drones | State Dimension $n_x$ | Solver Time [s] |
|---|---|---|
| 1 | 4 | 275.88 |
| 4 | 16 | 332.74 |
| 8 | 32 | 1118.96 |
| 12 | 48 | 2846.47 |

| $T$ | Number Schedules | Solver Time [s] |
|---|---|---|
| 2 | 4 | 0.03 |
| 10 | 63504 | 0.81 |
| 20 | $3.41 \cdot 10^{10}$ | 19.96 |
| 30 | $2.41 \cdot 10^{16}$ | 3284.71 |

with respect to the number of drones to be controlled in parallel. The state dimension $n_x$ is also indicated. As expected, the solving time grows with the state dimension. Note that for a state space in 48 dimensions, the problem can be solved in less than 50 minutes. In addition, these computations are executed offline, which make the computation time not critical.

### 5.5.2 Planar Linear Inverted Pendulum Model (Simplified Walker Dynamics)

The task of walking is one which typically does not require constant observation in order to safely execute. Humans routinely walk without knowledge of the exact position of their foot or angle of their torso, yet most controllers for walking robots require full state observation at all times.

To show the utility of our method, we analyze a small example where a robot model (the planar linear inverted pendulum) can be safely controlled by a policy, which infrequently observes the state.

The planar linear inverted pendulum model as discussed in [PKT17] can be represented as a state space model with the following equations:

$$\ddot{x}^{cm} = \frac{g}{\bar{z}^{cm}}(x^{cm} + r^{foot}u), \tag{5.26}$$

where $x^{cm}$ is the lateral position of the pendulum's center of mass, $\bar{z}^{cm}$ is the height of the pendulum's center of mass (which is assumed to be constant), $r^{foot}$ is the radius of the foot, and $u$ controls the center of pressure on the foot. We consider $\bar{z}^{cm} = 1$, $r^{foot} = 0.5$ and $g = 9.81$.

We solve Problem 5.2 for an exactly discretized version of (5.26) with a 0.1 discretization time step. We have $x(t) = \begin{bmatrix} x_t^{cm} & \dot{x}_t^{cm} \end{bmatrix}^\top$, $y(t) = x(t) +$

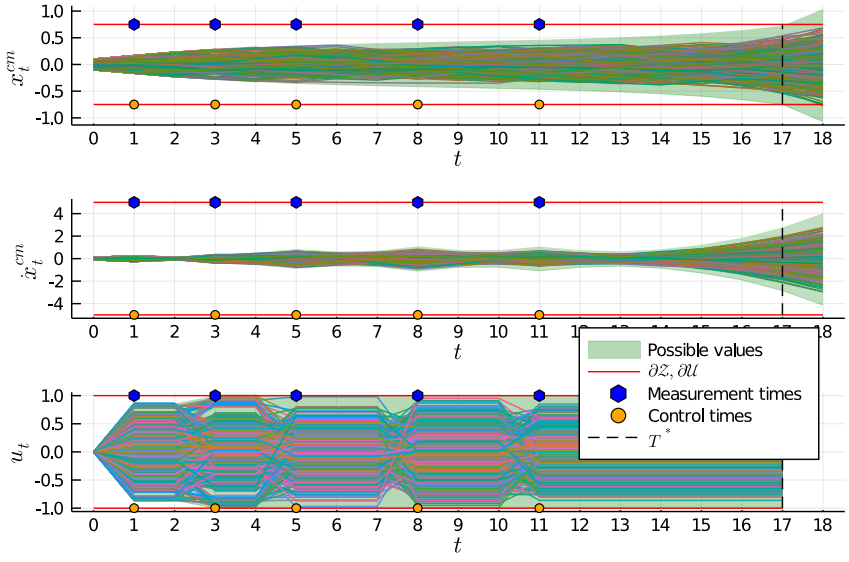



**Fig. 5.4** Representation of 1000 simulated trajectories of system (5.26). Measurement and control times are indicated. The boundaries of the safety set $\partial\mathcal{Z}$ and the boundaries of the set of admissible inputs $\partial\mathcal{U}$ are depicted. Finally, the sets of possible values for $z(t)$ and $u(t)$ for all possibles noises are also depicted.

$v(t)$ and $z(t) = x(t)$. We consider $\mathcal{W} = [-0.05, 0.05]^2$, $\mathcal{V} = [-0.01, 0.01]^2$, $\mathcal{X}_0 = [-0.1, 0.1]^2$, $\mathcal{U} = [-1, 1]$, $\mathcal{Z} = [-0.75, 0.75] \times [-5, 5]$, $N_m = N_c = 5$. We consider the upper bound $\bar{T} = 20$. This problem is solved in 284 seconds. The largest $T$ for which the problem is feasible is $T^* = 17$. Figure 5.4 presents the obtained trajectories. For half of them, the uncertainties are randomly selected among the vertices of the polytopes to promote extreme trajectories.

For the same system, the right part of Table 5.1 presents the evolution of the solving time for Problem 5.1 according to the time horizon $T$ when the number of measurement and control times is $N_m = N_c = T/2$. In addition, the number of measurement and control scheduling is indicated (assuming that the budget constraints (5.3) are tight). This table shows that the solving time increases with the time horizon but much slower than the number of possible measurement and control times scheduling. Again, these computations are performed offline.

*Remark* 5.14. Although these time horizons may be modest for many applications, a long time horizon $T'$ can be decomposed into multiple "components" (e.g., $T_1 + T_2 = T'$) where smaller problems are solved with appropriate initial, intermediate conditions and budget constraints to derive a result for the long time horizon $T'$. ◇

## 5.6 Conclusions

In this chapter, we have addressed the problem of co-designing a linear controller with memory, a measurement schedule, and a control schedule that guarantees safety. By proving that such a problem is equivalent to a MILP, we make this problem computationally tractable. This is illustrated on two examples. The dual problem of co-designing an estimate and choosing the measurement times in order to minimize the worst-case estimation error is also presented.

This chapter opens many avenues of continuation for our future research. First, we want to adapt this method to the case where polytopes are zonotopes. In this case, we hope to reduce the computation time using methods such as zonotope order reduction techniques [YS18]. Secondly, we plan to develop an online version of this algorithm, where observation and control times are computed on the fly, while the system is running, and thus leveraging the current observations at each time step (in the spirit of the online version of the particle filter presented in Section 4.2.2). In such a

self-triggered setting, the problem will be solved again after each measurement in order to incorporate the information recently acquired. Thirdly, another way to continue this research is to relax the binary constraints and do sparsity promoting through 1-norm regularization. This would lead to a linear program instead of a MILP, allowing a faster approximate resolution. Fourthly, we will consider the case where several sensors and actuators are available. In this context, the objective is to choose when each of them should be activated. Finally, another generalization would be to consider an uncertain linear dynamics, where the matrices $A$, $B$, $C$, $D$ and vector $d$ are unknown but contained in known polytopes.

# 6

# Case study: tumor tracking

In this chapter, we show how the optimal intermittent Kalman predictor presented in Chapter 3 can be used in the context of radiation therapy, a cancer treatment method whose objective is to irradiate the tumor with an ionizing beam.

In radiation therapy, tumor tracking is a challenging task that allows a better dose delivery. One practice is to acquire X-ray images in real-time during treatment, that are used to estimate the tumor location. This information is used to predict the close future tumor trajectory. Kalman prediction is a classical approach for this task. The main drawback of X-ray acquisition is that it irradiates the patient, including its healthy tissues. X-ray acquisitions are usually acquired regularly, i.e., at a constant rate. In this chapter, we propose a new approach, which relaxes this constraint in order to take measurements when they are the most useful. Our aim is for a given budget of X-ray acquisitions to optimize the tracking process. This idea naturally brings us to the optimal intermittent Kalman predictor presented in Chapter 3, for which measurement times are selected to minimize the mean squared prediction error. This optimization problem can be solved directly when the respiratory model has been identified and the optimal measurement times can be computed at once. We created a test benchmark on trajectories validated on one patient with a lung tumor. This new prediction method is compared to the regular Kalman predictor and a relative improvement of 9.8% is observed on the root mean squared

position estimation error.

The results presented in this chapter have been reported in

- Antoine Aspeel, Damien Dasnoy, Raphaël M Jungers, and Benoît Macq. Optimal intermittent measurements for tumor tracking in X-ray guided radiotherapy. In *Medical Imaging 2019: Image-Guided Procedures, Robotic Interventions, and Modeling*, volume 10951, page 109510C. International Society for Optics and Photonics, 2019.

- Antoine Aspeel, Axel Legay, and Benoît Macq. Genetic algorithms for optimal intermittent measurements for tumor tracking. In *The International Conference on the Use of Computers in Radiation Therapy*, 2019. (No proceedings).

## 6.1 Introduction

Mobile tumors are particularly challenging in radiation therapy. The motion creates uncertainty about the position of the tumor, obliging the physician to increase margins of the treatment that will irradiate healthy tissues. This effect is particularly important for particle therapy that is less robust to planning errors. A current response to this problem is to do image-guided radiation therapy. For particle therapy, it mainly consists in real-time X-ray acquisitions. In that case, X-ray images are acquired during the treatment to track the tumor in real-time at the cost of unwanted irradiation of the patient due to X-rays. There is a trade-off between more radiation by imaging with smaller margins or less radiation by imaging but with larger margins. For medical reasons (ALARA: As low as reasonably achievable), the measurement budget (amount of images) has to be restricted. These images feed a prediction model of the close future of tumor motion, which is needed to compensate the delay between image acquisition and hardware reaction, including the computational time of the predictive algorithm. For now, these images are taken at constant rate.

### 6.1.1 Contribution

The purpose of this chapter is (i) to develop a predictive tumor tracking algorithm based on intermittent measurements; and (ii) to select the best times to take the X-ray measurements in order to have the best tumor position estimation during its irradiation. To the best of our knowledge, this

is the first time that intermittent X-ray measurements are considered for mobile tumor tracking.

The main idea is to optimally use each X-ray image allowing *irregular sampling*, i.e., variable rate imaging. This additional degree of freedom permits a better use of each X-ray image and then, a better estimation of the tumor position without additional dose. More precisely, we consider that the number $N$ of X-ray images allowed during a fraction is fixed by the clinician. These $N$ measurements will be made when they are the more useful in order to reduce the error variance of the predicted tumor's location. The framework of Kalman filtering theory has still been used by Sharp *et al.* [SJSS04] in the context of tumor tracking in radiation therapy.

### 6.1.2 Chapter organization

The rest of this chapter is organized as follows: Section 6.2 develops how to use the method in clinical settings, the technical developments and the kind of data; Section 6.3 presents and discuss of the prediction quality; finally Section 6.4 concludes with a summary of the results and further work.

### 6.1.3 Notation

The notations used in this chapter are the same as in Chapter 3. See Section 3.1.3 for a reminder.

## 6.2 Materials and methods

X-ray image acquisition is not directly suited for real-time because tumor localization and the measurement process encompass some delay. This delay imposes not only to estimate the current tumor position, but also to predict its future motion. Another physical restriction is the time between two X-ray acquisitions that can not be arbitrarily small. Here, these two delays are assumed to be equal. The time is discretized with a time step equal to that quantity. If the horizon prediction is not equal to the delay between X-ray images, the presented method can straitforwardly be adapted. In clinical workflow, X-ray images of the patient are acquired during the treatment. Following Sharp *et al.* [SJSS04], the first images are used for system identification and then, they are used for real-time tumor tracking.

In a first phase, these images are acquired regularly, i.e., at constant rate. These successive noisy measurements $y(t)$ of the tumor location $z(t)$

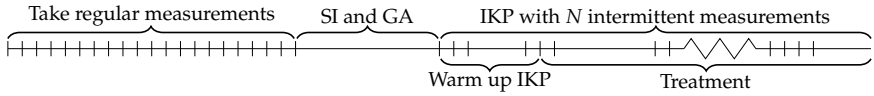



**Fig. 6.1** Time line of a fraction. Each phase is specified and vertical dashes represent measurement times. SI is system identification, GA is genetic algorithm and IKP is intermittent Kalman predictor.

are used to model the tumor motion. We call it *system identification* (SI) and it is described in Section 6.2.1.

Secondly, the model of the tumor motion is used to find the set $\mathcal{M}$ of $N$ optimal times to take measurements with X-ray images. It corresponds to finding the *mean squared estimator* (MSE) among all the *intermittent Kalman predictors* (IKP). These measurement times are found by approximately solving a combinatorial optimization problem using a *genetic algorithm* (GA). A complete description of the optimal intermittent Kalman predictor was presented in Chapter 3 (See Sections 3.2.1 to 3.2.3 for the optimal intermittent Kalman filter and Section 3.2.4 for the genetic algorithm).

In a last phase, the tumor motion is predicted in real-time by the IKP based on measurements taken at the times found by the GA. At the beginning of this phase, some time is left to warm up the predictor. The dose delivery begins when the predictor is warmed up. This procedure is represented on Figure 6.1.

### 6.2.1 System identification

We assume that the tumor position $z(t)$ can be modelled as a linear discrete-time stochastic system

$$\begin{cases} x(t+1) &= Ax(t) + b + Gw(t) & \text{(Internal states)} \\ z(t) &= Cx(t) + d & \text{(Tumor position)} \end{cases} \quad t = 0, \dots, T-1 \tag{6.1}$$

where $x(0) \sim \mathcal{N}(\bar{x}_0, \bar{P}_0)$, $w(t) \sim \mathcal{N}(0, Q)$ are mutually and time independent white Gaussian noises. $T$ is the duration of the treatment (in number of time steps). $x(t)$, $b$ and $w(t)$ are $n_x \times 1$ vectors; $A$ is a $n_x \times n_x$ matrix; $G$ has compatible dimensions; $z(t)$ and $d$ are $n_z \times 1$ vectors and $C$ is a $n_z \times n_x$ matrix. In practice, $n_z$ is 2 or 3 for 2- and 3-dimensional tracking, respectively. The number of internal states $n_x$ is an hyper-parameter to determine (it is not patient-specific). The system identification problem is to model

the tumor position $z(t)$ on the basis of noisy measurement $y(t)$. The measurement equation is $y(t) = z(t) + v(t)$ for any $t \in \mathcal{M}$ in which the white Gaussian noise $v(t) \sim \mathcal{N}(0, R)$ is time independent and independent of $x(0)$ and $w(\tau)$ for any $\tau$. Each $y(t)$ is a $n_z \times 1$ vector. The previous measurement equation and equations (6.1) can be rewritten to fit the Kalman formalism,

$$\left\{ \begin{array}{rcll} x(t+1) & = & Ax(t) + b + Gw(t) & t = 0, \ldots, T-1 \\ y(t) - d & = & Cx(t) + v(t) & t \in \mathcal{M} \end{array} \right. \tag{6.2}$$

where $b$ can be considered as a constant command. Note that during the first phase (see Figure 6.1), X-rays are acquired regularly and so, during system identification, the second equation has to be considered for all $t$.

The purpose of these equations is to model the tumor motion, thus the parameters of this system are identified from regularly spaced measurements at the beginning of the fraction, they are $A$, $b$, $Q$, $d$, $C$, $R$, $\bar{x}_0$ and $\bar{P}_0$. Note that there is no matrix $B$ as in Section 3.2.1 because the measurements $y(t)$ are a noisy version of $z(t)$. This system identification is done using the maximum likelihood estimation with the expectation maximization algorithm of Digalakis *et al.* [DRO93] as suggested by Sharp *et al.* [SJSS04] in the context of tumor tracking.

### 6.2.2 Warm up

In order to treat the tumor only when its position is accurately estimated, a warm up phase of $T_0$ time steps is introduced (see Figure 6.1). To do this, the measurement times are chosen considering only the error made after the warm up. The objective functions in equations (3.2) and (3.7) become

$$\begin{aligned} &\min_{\mathcal{M} \subset \{0, \cdots, T-1\}} \frac{1}{T} \sum_{t=T_0+1}^{T} \mathbb{E}\left[ \|z(t) - \hat{z}(t|t-1)\|^2 \right] \\ = &\min_{\mathcal{M} \subset \{0, \cdots, T-1\}} \frac{1}{T} \sum_{t=T_0+1}^{T} \mathrm{Tr}[CP(t|t-1)C^\top], \end{aligned}$$

where sums have been truncated.

### 6.2.3 Data description

The tumor position $z(t)$ is obtained from the Figure 1 in Bukovsky *et al.* [BHI+15] and are depicted in Figure 6.2. It is a lung tumor and $y_1$, $y_2$ and

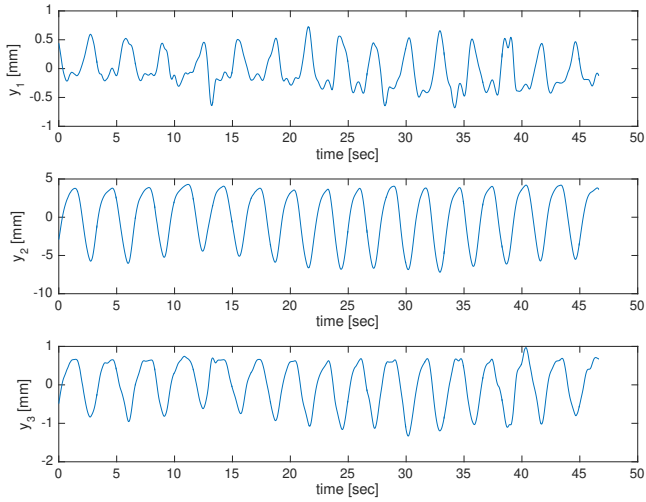



**Fig. 6.2** The lung tumor position $z(t) = \begin{bmatrix} y_1(t) & y_2(t) & y_3(t) \end{bmatrix}^\top$ with respect to time for the studied patient. Figure inspired from Figure 1 of Bukovsky *et al.* [BHI+15].

$y_3$ are the position on the lateral, cephalocaudal and anteroposterior axes, respectively.

The trajectory is considered as the ground truth $z(t) \in \mathbb{R}^3$. A noisy trajectory $y(t)$ is simulated from this ground truth to imitate a X-ray acquisition followed by a tracking software, i.e., $y(t) = z(t) + v(t)$ with $v(t) \sim \mathcal{N}(0, \sigma^2 I)$. The parameter $\sigma^2$ quantifies the quality of the observation process, the more the measurement procedure is precise, the more $\sigma^2$ is small. For example, a high $\sigma^2$ models a low-dose imaging. The prediction algorithm uses $y$ as noisy measurements of the tumor and computes estimate $\hat{z}(t|t-1)$ of the ground truth $z(t)$.

The duration between two discrete time steps is $1/30$ seconds. The system identification is 600 time steps (20 seconds), we fix $T = 800$ (26.7 seconds), and warm up takes $T_0 = 300$ time steps (10 seconds). In the next section, we compare our algorithm to other methods.

## 6.3 Experiments and discussions

Hyper-parameters have been tuned on simulated data. Our method (IKP) is compared to three other methods: *regular measurements without prediction* (RMWP), i.e., the tumor is assumed motionless between measurements; *intermittent measurements without prediction* (IMWP), i.e., measurements are taken at the same time than IKP but tumor is assumed motionless between measurements; and *regular Kalman predictor* (RKP), i.e., with regularly spaced measurement times. Root-mean squared prediction errors are presented in Table 6.1 for these four methods. Values of $\sigma^2 \in \{1, 4, 25, 100\}$ are tested, it corresponds to a standard deviation of 1, 2, 5, 10 $[mm]$, which is a plausible range of measurement error for practical applications. In addition, the comparison is done for different measurement budget $N$ ($N/T$ and mean acquisition rate are printed for more meaning).

Table 6.1 shows that our method (IKP) outperforms the three other predictors in all cases. Globally IMWP gives similar results than RMWP, indicating that the measurement times computed for IKP are well-suited only for Kalman predictions. The two Kalman approaches (RKP and IKP) are highly more precise than when there is no predictor (RMWP and IMWP). For the values presented in that table, the mean relative improvement of IKP on RKP is about[1] 9.8%. If some improvements look marginal, note that a small improvement in tumor localization permits to reduce the margins of the planning target volume. In addition, the tumor being a three-dimensional object, when the margin radius $r$ decreases, the planning target volume decreases in $r^3$, which amplifies the benefit, specifically for precise dose delivery by particle therapy.

Nevertheless, a demonstration on only one patient has to be considered as a proof of concept but is not sufficient for clinical application.

## 6.4 Conclusions

A tumor tracking method based on an optimal choice of measurement times has been developed. The benefit of using well-chosen intermittent measurement times in the case of radiation therapy guided by X-ray images has been shown on a lung tumor. This improvement is particularly interesting in the case of particle therapy because of the reduction errors on

[1] It is $(\mathrm{RMS}_{\mathrm{RKP}} - \mathrm{RMS}_{\mathrm{IKP}})/\mathrm{RMS}_{\mathrm{RKP}}$ averaged on the values of Table 6.1, where RMS means root-mean squared error.

**Table 6.1** Root-mean squared prediction errors [$mm$] for different predictors, different $\sigma^2$ and different $N/T$ (also written $N/T \times 30Hz$, the mean acquisition rate). Compared methods are RMWP: Regular measurements without prediction; IMWP: Intermittent measurements without prediction; RKP: Regular Kalman predictor; and our method, IKP: Intermitent Kalman predictor. Bold characters highlight the best predictor in each situation.

| | $\sigma^2$ [$mm^2$] | 1 | 4 | 25 | 100 |
|---|---|---|---|---|---|
| $N/T$ (mean rate) | Method | | | | |
| 0.1 (3 [$Hz$]) | RMWP | 2.48 | 3.99 | 9.14 | 18.01 |
| | IMWP | 2.22 | 3.93 | 9.05 | 17.90 |
| | RKP | 1.29 | 1.71 | 3.51 | 3.78 |
| | IKP | **1.11** | **1.50** | **3.14** | **3.64** |
| 0.2 (6 [$Hz$]) | RMWP | 1.99 | 3.67 | 8.96 | 17.85 |
| | IMWP | 2.00 | 3.59 | 8.82 | 17.53 |
| | RKP | 1.02 | 1.42 | 3.04 | 3.61 |
| | IKP | **0.98** | **1.15** | **2.86** | **3.54** |
| 0.3 (9 [$Hz$]) | RMWP | 1.88 | 3.56 | 8.77 | 17.48 |
| | IMWP | 1.87 | 3.54 | 8.79 | 17.54 |
| | RKP | 0.95 | 1.31 | 2.77 | 3.47 |
| | IKP | **0.81** | **1.20** | **2.47** | **3.31** |
| 0.4 (12 [$Hz$]) | RMWP | 1.84 | 3.56 | 8.82 | 17.62 |
| | IMWP | 1.84 | 3.52 | 8.83 | 17.64 |
| | RKP | 0.84 | 1.24 | 2.63 | 3.43 |
| | IKP | **0.76** | **1.09** | **2.27** | **3.14** |
| 0.5 (15 [$Hz$]) | RMWP | 1.79 | 3.51 | 8.72 | 17.42 |
| | IMWP | 1.82 | 3.54 | 8.78 | 17.52 |
| | RKP | 0.83 | 1.21 | 2.59 | 3.44 |
| | IKP | **0.73** | **1.06** | **2.29** | **3.23** |

tumor localization has significant effects on the margin reduction (sparing healthy tissues).

The presented method is the optimal intermittent Kalman predictor developed in Chapter 3. The first experiments that have been presented show that this new method seems to outperform the classic Kalman approach, i.e., with regular measurement times. However, we prefer to be cautious about our conclusions because the evidence that has been presented is based on only one patient. Deeper tests are needed to definitely assess the performance of this method on clinical framework, even if the proof of concept seems to be encouraging. Future work could also consider re-identifying the system after each new measurement, as suggested by Riaz *et al.* [RSW+09].

# Conclusions

In this thesis, we studied the problem of the optimal choice of measurement times to estimate the state of a dynamical system over a finite horizon. When the total amount of measurements is reduced, for example because they are expensive, time consuming, energy consuming or dangerous, choosing the best measurement times allows getting the maximum out of each of them. This question led us to the distinction between time-triggered and self-triggered measurements.

In Chapter 3, we were interested in the case of linear and Gaussian systems and our optimality criterion was the expected mean squared estimation error. We have proved that, in this case, the optimal time-triggered and self-triggered mechanisms are the same, and that the optimal measurement times are obtained by solving a combinatorial optimization problem. This led us to propose the optimal intermittent Kalman filter. Several algorithms were tested and compared to solve the combinatorial optimization problem and it was shown that a genetic algorithm with a count preserving crossover gives good performances. Next, it was numerically investigated how system stability, noise variance, and system observability matrix influence the optimal measurement times. In addition, it was demonstrated that the method is also adapted to continuous-time systems via a discretization of time. Finally, by dualizing the problem, we obtained an optimal intermittent linear quadratic regulator.

In Chapter 4, we extended the results of Chapter 3 to nonlinear and non-Gaussian systems. In this case, there is a difference between time-triggered and self-triggered measurements. We then proposed three variants of optimal intermittent particle filters. A theorem ordering the expected mean squared errors obtained with the different methods was proved. Overall, the two main contributions of that chapter was to pro-

pose algorithms to solve the problem of optimal measurement times selection in the nonlinear and non-Gaussian case; and to show the interest of intermittent measurements in particle filtering.

In Chapter 5, we considered linear systems subject to bounded disturbances. Interested in the choice of the measurement times minimizing the worst-case estimation error, we solved a more general control problem: the co-design of (i) the measurement times, (ii) the control times, and (iii) the gains of the controller, while guaranteeing that the output variable remains in a safety set. Using the polytope containment method and Q-parameterization, we proved that this problem can be reduced to a mixed integer linear program with 2 binary variables per time step. We demonstrated through numerical experiments that this allows us to solve the problem efficiently in practical cases.

In Chapter 6, we applied the optimal intermittent Kalman filter developed in Chapter 3 to the problem of tumor tracking for radiation therapy. Considering real data from a lung tumor patient, a more accurate estimation of the tumor position for the same radiation imaging was obtained, allowing a better treatment.

Overall, in this thesis, we developed tools to model, analyze, and compute the optimal sampling for state estimation of stochastic dynamical systems. Through numerous numerical experiments, we have shown that, most of the time, there is an interest in measuring irregularly, rather than regularly.

The above suggests several interesting directions for future research.

First, it would be interesting to study the robustness of optimal intermittent Kalman and particle filters to modeling errors. The objective would be to know how the expected mean squared error evolves with respect to the modeling error. Although the fact that the optimal intermittent Kalman filter was successfully tested on real data is encouraging, further analysis would be beneficial.

Some of our future work will involve solving the partially observable Markov decision process described in Section 4.2.2 using a deep Q-network. This will lead to an optimal self-triggered policy for the choice of measurement times for nonlinear systems. We will be able to compare the performances obtained with different filtering methods, such as particle filtering and recurrent neural networks.

Chapter 5 on the co-design of measurement times, control times, and a controller subject to bounded disturbances also opens up many avenues

of continuation. First, we will try to approximate the mixed integer linear program by a linear program without binary variables thanks to a sparsity promoting approach based on 1-norm regularization. This should speed up the solution of the problem, at the cost of a suboptimal solution, and would allow to tackle larger problems. Then, we will focus on the case of multiple sensors and actuators. The objective is to independently choose when each sensor and actuator should be activated. Finally, we will study the problem where the dynamics of the system is only partially known (e.g., each matrix of the system being in a known polytope).

Concerning radiation therapy for mobile tumors, the methods presented in this thesis could be tested for tumor tracking from intermittent X-ray image acquisitions. Different directions are to be explored and compared: the first one, presented in Chapter 6, consists in combining a linear system identification method with our optimal intermittent Kalman predictor. A proof of concept was presented on a patient, but further tests are needed to attest the approach. A second approach would be to use one of the optimal intermittent particle filters developed in Chapter 4, with a dynamical system modeling tumor motion (e.g., the tumor motion model presented in Section 4.3.1). Finally, a robust estimation could be obtained by combining a linear system identification with the worst-case estimation method proposed in Chapter 5. All these alternatives could be compared on different patients to prove their clinical validity, and the appropriateness of using intermittent X-ray images for tumor tracking.

At the end of the 20th century, solutions emerged to estimate the state of a stochastic system from noisy measurements [Kal60, God19]. Since then, a large research effort has been made to adapt these methods to non-ideal situations closer to real world applications (partially unknown dynamics [YG06, Pap05], decentralized sensors [Ste90, RGT12],...). In this thesis, we have contributed to this effort by considering the constraints that may exist concerning the use of sensors (measurements may be expensive, energy consuming, time consuming or dangerous).

# A

# Proof of Theorem 3.5

THE proof is a consequence of the classical duality between Kalman filter and LQR (see Section 2.5). We want to use Theorem 2.1 from [SYL19]. However, these authors' formalism has two differences from ours.

First, they do not consider missing control times (or missing measurement times). However, this is not restrictive because for a given $\tilde{\sigma}(t)$, equation (3.13a) can reduce to the form of [SYL19] by considering $\tilde{\sigma}(t)\tilde{B}$ as a known time-varying control matrix $\bar{B}(t)$. Similarly, the measurement matrix $C$ in equation (3.1b) can be seen as the null matrix when no measurement is acquired, i.e., when $\sigma(t) = 0$ or equivalently $t \notin \mathcal{M}$.

The second difference with [SYL19] is that they limit the estimation problem to the case where $D = I$, i.e., $z(t) = x(t)$. However, as shown by equations (3.3a), (3.4a) and (3.4b), the prediction error covariance matrix $P(t|t-1)$ does not depend on $D$. Consequently, results from Theorem 2.1 from [SYL19] that concern $P(t|t-1)$ can be used.

Thus, for a given signal $\tilde{\sigma}(\cdot)$, Theorem 2.1 from [SYL19] states that $V(\tilde{\sigma}(\cdot)) = \tilde{x}_0^\top \tilde{P}_{\tilde{\sigma}(\cdot)} \tilde{x}_0$ where $\tilde{P}_{\tilde{\sigma}(\cdot)} = P_{\sigma(\cdot)}(T|T-1)$ and the last is given by equations (3.3a), (3.4a) and (3.4b), under transformations $\sigma(t) = \tilde{\sigma}(T-t)$, (3.15a) and (3.15b).

Then, under transformation (3.15c), the objective function of the esti-

mation problem (3.7) is

$$\begin{aligned}\sum_{t=1}^{T} \mathrm{Tr}[D(t)P_{\sigma(\cdot)}(t|t-1)D(t)^\top] &= \sum_{t=1}^{T} \mathrm{Tr}[\delta(T-t)\tilde{x}_0^\top P_{\sigma(\cdot)}(t|t-1)\tilde{x}_0] \\ &= \tilde{x}_0^\top P_{\sigma(\cdot)}(T|T-1)\tilde{x}_0 \\ &= \tilde{x}_0^\top \tilde{P}_{\tilde{\sigma}(\cdot)}(0)\tilde{x}_0 \\ &= V(\tilde{\sigma}(\cdot)),\end{aligned}$$

where the trace operator has been removed because its argument is one-dimensional. Taking the minimum overall admissible $\sigma(\cdot)$ (on the left-hand side) and the corresponding $\tilde{\sigma}(\cdot)$ (on the right-hand side) gives equality between problems (3.7) and (3.14).

Once the signal $\tilde{\sigma}^*(\cdot)$ is fixed, finding the optimal command $u(t)$ is a classic LQR problem. Theorem 2.1 form [SYL19] gives the relation (3.16).

# B

# Proof of Theorem 4.4

THE structure of the proof consists in (i) observing that once all the measurements are known, cost-to-go is equal for the three filters; then (ii) we show that if the cost-to-go when choosing the measurement time $t_{j+1}$ is smaller with the self-triggered filter than with the ex post filter, which itself is smaller than with the time-triggered filter, then this is also the case when choosing the measurement time $t_j$; and finally (iii) a backward recursive argument shows that this is also true for the initial cost-to-go.

This structure is applied to each inequality separately.

**Proof of the first inequality** From relations (4.5) and (4.7) it follows that for all $t_{1:N}$ and $y(t_{1:N})$,

$$V_N(t_{1:N}; y(t_{1:N})) = F_N(t_{1:N}; y(t_{1:N})). \tag{B.1}$$

For a fixed $j \in \{0, \dots, N-1\}$, assume that for all $t_{1:j+1}$ and $y(t_{1:j+1})$ it holds that

$$V_{j+1}(t_{1:j+1}; y(t_{1:j+1})) \leq F_{j+1}(t_{1:j+1}; y(t_{1:j+1})). \tag{B.2}$$

Then, taking the expectation with respect to $y(t_{j+1})$ and adding a same

term on both sides, we have,

$$\sum_{t=t_j}^{t_{j+1}-1} \mathcal{E}[t|y(t_{1:j})] + \mathbb{E}_{y(t_{j+1})}\left[V_{j+1}(t_{1:j+1}; y(t_{1:j+1}))\right] \leq$$
$$\sum_{t=t_j}^{t_{j+1}-1} \mathcal{E}[t|y(t_{1:j})] + \mathbb{E}_{y(t_{j+1})}\left[F_{j+1}(t_{1:j+1}; y(t_{1:j+1}))\right],$$

for all $t_{1:j+1}$ and $y(t_{1:j})$. Taking the minimum over all $t_{j+1}$ on the left-hand side and fixing $t_{j+1}$ to $t^o_{j+1}$ on the right-hand side, we have,

$$\min_{t_{j+1}} \left\{ \sum_{t=t_j}^{t_{j+1}-1} \mathcal{E}[t|y(t_{1:j})] + \mathbb{E}_{y(t_{j+1})}\left[V_{j+1}(t_{1:j+1}; y(t_{1:j+1}))\right] \right\} \leq$$
$$\sum_{t=t_j}^{t^o_{j+1}-1} \mathcal{E}[t|y(t_{1:j}))] + \mathbb{E}_{y(t^o_{j+1})}\left[F_{j+1}(t_{1:j}, t^o_{j+1}; y(t_{1:j}), y(t^o_{j+1}))\right],$$

for all $t_{1:j}$ and $y(t_{1:j})$. From relation (4.4) on the left-hand side and relation (4.8) on the right-hand side, we have

$$V_j(t_{1:j}; y(t_{1:j})) \leq F_j(t_{1:j}; y(t_{1:j})), \tag{B.3}$$

for all $t_{1:j}$ and $y(t_{1:j})$.

But then, because relation (B.2) implies (B.3) and because of (B.1), we have $V_0(\emptyset;\emptyset) \leq F_0(\emptyset;\emptyset)$.

**Proof of the second inequality** It follows from a similar argument. We define

$$J_N(t_{1:N}; y(t_{1:N})) := F_N(t_{1:N}; y(t_{1:N})). \tag{B.4}$$

Then, for a fixed $j \in \{0, \ldots, N-1\}$, assume that for all $t_{1:j+1}$ and $y(t_{1:j+1})$ it holds that

$$F_{j+1}(t_{1:j+1}; y(t_{1:j+1})) \leq J_{j+1}(t_{1:j+1}; y(t_{1:j+1})). \tag{B.5}$$

Rewriting the right-hand side according to (4.6) gives

$$F_{j+1}(t_{1:j+1}; y(t_{1:j+1})) \leq \min_{t_{j+2}<\cdots<t_N\leq T} \mathbb{E}_{y(t_{j+2:N})}\left[\sum_{t=t_{j+1}}^{T} \mathcal{E}[t|y(t_{1:N})]\right].$$

The right-hand side is upper bounded by

$$\mathbb{E}_{y(t_{j+2:N})}\left[\sum_{t=t_{j+1}}^{T} \mathcal{E}[t|y(t_{1:N})]\right],$$

for all $t_{j+2:N}$. Then one can write

$$F_{j+1}(t_{1:j+1}; y(t_{1:j+1})) \leq \mathbb{E}_{y(t_{j+2:N})}\left[\sum_{t=t_{j+1}}^{T} \mathcal{E}[t|y(t_{1:N})]\right],$$

for all $t_{1:N}$ and $y(t_{1:j+1})$. Taking the expectation with respect to $y(t_{j+1})$ and adding a same term on both sides gives

$$\begin{aligned}&\sum_{t=t_j}^{t_{j+1}-1} \mathcal{E}[t|y(t_{1:j})] + \mathbb{E}_{y(t_{j+1})}\left[F_{j+1}(t_{1:j+1}; y(t_{1:j+1}))\right] \leq \\ &\sum_{t=t_j}^{t_{j+1}-1} \mathcal{E}[t|y(t_{1:j})] + \mathbb{E}_{y(t_{j+1:N})}\left[\sum_{t=t_{j+1}}^{T} \mathcal{E}[t|y(t_{1:N})]\right],\end{aligned} \tag{B.6}$$

for all $t_{1:N}$ and $y(t_{1:j})$. Using the fact that the first term of the right-hand side is independent of $y(t_{j+1:N})$, and then thanks to Remark 4.1, the right-hand side can be rewritten successively as

$$\begin{aligned}&\mathbb{E}_{y(t_{j+1:N})}\left[\sum_{t=t_j}^{t_{j+1}-1} \mathcal{E}[t|y(t_{1:j})] + \sum_{t=t_{j+1}}^{T} \mathcal{E}[t|y(t_{1:N})]\right] = \\ &\mathbb{E}_{y(t_{j+1:N})}\left[\sum_{t=t_j}^{t_{j+1}-1} \mathcal{E}[t|y(t_{1:N})] + \sum_{t=t_{j+1}}^{T} \mathcal{E}[t|y(t_{1:N})]\right].\end{aligned}$$

By combining the two sums on this right-hand side, the inequality (B.6) becomes

$$\sum_{t=t_j}^{t_{j+1}-1} \mathcal{E}[t|y(t_{1:j})] + \mathbb{E}_{y(t_{j+1})}\left[F_{j+1}(t_{1:j+1}; y(t_{1:j+1}))\right] \leq$$
$$\mathbb{E}_{y(t_{j+1:N})}\left[\sum_{t=t_j}^{T} \mathcal{E}[t|y(t_{1:N})]\right],$$

for all $t_{1:N}$ and $y(t_{1:j})$. We set $t_{j+1:N}$ to the value that minimizes this right-hand side. In so doing, $t_{j+1} = t^o_{j+1}$ and the inequality becomes

$$\sum_{t=t_j}^{t^o_{j+1}-1} \mathcal{E}[t|y(t_{1:j})] + \mathbb{E}_{y(t^o_{j+1})}\left[F_{j+1}(t_{1:j}, t^o_{j+1}; y(t_{1:j}), y(t^o_{j+1}))\right] \leq$$
$$\min_{t_{j+1}<\cdots<t_N\leq T} \mathbb{E}_{y(t_{j+1:N})}\left[\sum_{t=t_j}^{T} \mathcal{E}[t|y(t_{1:N})]\right],$$

for all $t_{1:j}$ and $y(t_{1:j})$. Now, using relation (4.8) on the left-hand side and relation (4.6) on the right-hand side, we obtain

$$F_j(t_{1:j}; y(t_{1:j})) \leq J_j(t_{1:j}; y(t_{1:j})), \tag{B.7}$$

for all $t_{1:j}$ and $y(t_{1:j})$.

Finally, because relation (B.5) implies relation (B.7) and because of (B.4), we have $F_0(\emptyset;\emptyset) \leq J_0(\emptyset;\emptyset)$, which concludes the proof.

# C

# Proofs of Chapter 5

THIS chapter contains the proofs of the lemmas of Chapter 5.

## C.1 Proof of Lemma 5.5

If $\sigma^c(t) = 1$, one can use (5.8) to write

$$f_t + \sum_{\tau \leq t \text{ s.t. } \sigma^m(\tau)=1} F_{(t,\tau)} y(\tau) = f_t + \sum_{\tau \leq t} F_{(t,\tau)} y(\tau),$$

then (5.2) and (5.7d) give the same $u(t)$. On the other hand, if $\sigma^c(t) = 0$, one can use (5.9) to write

$$\begin{aligned} f_t + \sum_{\tau \leq t} F_{(t,\tau)} y(\tau) &= f_{t-1} + \sum_{\tau \leq t} F_{(t-1,\tau)} y(\tau) \\ &= f_{t-1} + \sum_{\tau \leq t-1} F_{(t-1,\tau)} y(\tau). \end{aligned}$$

then (5.2) and (5.7d) give the same $u(t)$.

It follows that the sequences of $u(t)$ are the same. But then, so it is for the sequences of $x(t)$ and finally, for the sequences of $z(t)$.

## C.2 Proof of Lemma 5.7

By writing $\eta = \begin{bmatrix} w^\top & v^\top & x(0)^\top \end{bmatrix}^\top \in \mathcal{W}^T \times \mathcal{V}^T \times \mathcal{X}_0$ and $P_{z,:} = \begin{bmatrix} P_{zw} & P_{zv} & P_{zx_0} \end{bmatrix}$ and using (5.16), we can write

$$\begin{aligned}
& z(t) && \in && \mathcal{Z} \text{ for all } t = 0, \dots, T \\
\Leftrightarrow \quad & H_{\mathcal{Z}} z(t) && \leq && h_{\mathcal{Z}} \text{ for all } t = 0, \dots, T \\
\Leftrightarrow \quad & (I_{T+1} \otimes H_{\mathcal{Z}})\, z && \leq && \mathbb{1}_{T+1} \otimes h_{\mathcal{Z}} \\
\Leftrightarrow \quad & (I_{T+1} \otimes H_{\mathcal{Z}})\, (P_{z,:}\eta + \tilde{z}) && \leq && \mathbb{1}_{T+1} \otimes h_{\mathcal{Z}} \\
\Leftrightarrow \quad & (I_{T+1} \otimes H_{\mathcal{Z}})\, P_{z,:}\eta && \leq && \mathbb{1}_{T+1} \otimes h_{\mathcal{Z}} \\
& && && \quad - (I_{T+1} \otimes H_{\mathcal{Z}})\, \tilde{z}.
\end{aligned}$$

The last inequality can be interpreted as requiring that $\eta$ is in a polyhedron (let us call it $\mathcal{P}$). Then, $z(t) \in \mathcal{Z}$ for all $t = 0, \dots, T$ and for all $\eta \in \mathcal{W}^T \times \mathcal{V}^T \times \mathcal{X}_0$ if and only if $\eta \in \mathcal{P}$ for all $\eta$. This is equivalent to $\mathcal{W}^T \times \mathcal{V}^T \times \mathcal{X}_0 \subseteq \mathcal{P}$. Then, using Lemma 5.6, the constraint $z(t) \in \mathcal{Z}$ for all $t = 0, \dots, T$ and for all $\eta \in \mathcal{W}^T \times \mathcal{V}^T \times \mathcal{X}_0$ holds if and only if there exists $\Lambda \in \mathbb{R}_+^{(T+1)n_{\mathcal{Z}} \times (T(n_{\mathcal{W}}+n_{\mathcal{V}})+n_{\mathcal{X}_0})}$ such that (5.19) and (5.20) hold.

## C.3 Proof of Lemma 5.8

The proof is similar to the one of Lemma 5.7. Let $\eta = \begin{bmatrix} w^\top & v^\top & x(0)^\top \end{bmatrix}^\top \in \mathcal{W}^T \times \mathcal{V}^T \times \mathcal{X}_0$ and $P_{u,:} = \begin{bmatrix} P_{uw} & P_{uv} & P_{ux_0} \end{bmatrix}$ and using (5.16), we can write

$$\begin{aligned}
& u(t) && \in && \mathcal{U} \text{ for all } t = 0, \dots, T-1 \\
\Leftrightarrow \quad & H_{\mathcal{U}} u(t) && \leq && h_{\mathcal{U}} \text{ for all } t = 0, \dots, T-1 \\
\Leftrightarrow \quad & (I_T \otimes H_{\mathcal{U}})\, u && \leq && \mathbb{1}_T \otimes h_{\mathcal{U}} \\
\Leftrightarrow \quad & (I_T \otimes H_{\mathcal{U}})\, (P_{u,:}\eta + \tilde{u}) && \leq && \mathbb{1}_T \otimes h_{\mathcal{U}} \\
\Leftrightarrow \quad & (I_T \otimes H_{\mathcal{U}})\, P_{u,:}\eta && \leq && \mathbb{1}_T \otimes h_{\mathcal{U}} - (I_T \otimes H_{\mathcal{U}})\, \tilde{u}.
\end{aligned}$$

The last inequality can be interpreted as requiring that $\eta$ is in a polytope (let us call it $\mathcal{Q}$). Then, $u(t) \in \mathcal{U}$ for all $t = 0, \dots, T-1$ and for all $\eta \in \mathcal{W}^T \times \mathcal{V}^T \times \mathcal{X}_0$ if and only if $\eta \in \mathcal{Q}$ for all $\eta$. This is equivalent to $\mathcal{W}^T \times \mathcal{V}^T \times \mathcal{X}_0 \subseteq \mathcal{Q}$. Then, using Lemma 5.6, the constraint $u(t) \in \mathcal{U}$ for all $t = 0, \dots, T-1$ and for all $\eta \in \mathcal{W}^T \times \mathcal{V}^T \times \mathcal{X}_0$ holds if and only if there exists $\Gamma \in \mathbb{R}_+^{Tn_{\mathcal{U}} \times (T(n_{\mathcal{W}}+n_{\mathcal{V}})+n_{\mathcal{X}_0})}$ such that (5.21) and (5.22) hold.

## C.4 Proof of Lemma 5.10

Let $\tau \in \{0, \dots, T-1\}$ and assume $\sigma^m(\tau) = 0$. From (5.11), we have that for a fixed $\tau$, the matrices $F_{(t,\tau)}$ are in the same columns of $F$ for all $t$. The indices of these columns are $\mathcal{J}_\tau$. Then, $F_{(t,\tau)} = 0$ for all $t = \tau, \dots, T-1$, if and only if $F_{:,\mathcal{J}_\tau} = 0$. Then, looking at (5.15a), one can write

$$F_{:,\mathcal{J}_\tau} = (I + Q\bar{C}S)^{-1} Q_{:,\mathcal{J}_\tau}.$$

Observing that $(I + Q\bar{C}S)^{-1}$ is invertible, it follows that $F_{:,\mathcal{J}_\tau} = 0$ if and only if $Q_{:,\mathcal{J}_\tau} = 0$.

## C.5 Proof of Lemma 5.11

Let $t \in \{1, \dots, T-1\}$ and assume that $\sigma^c(t) = 0$. From (5.10) and (5.11), the constraints $f_t = f_{t-1}$ and $F_{(t,\tau)} = F_{(t-1,\tau)}$ can be written $f_{\mathcal{I}_t} = f_{\mathcal{I}_{t-1}}$ and $F_{\mathcal{I}_t,:} = F_{\mathcal{I}_{t-1},:}$.

Otherwise, from (5.14a), one can write

$$Q_{\mathcal{I}_t,:} - Q_{\mathcal{I}_{t-1},:} = (F_{\mathcal{I}_t,:} - F_{\mathcal{I}_{t-1},:})(I - \bar{C}SF)^{-1}. \tag{C.1}$$

But $(I - \bar{C}SF)^{-1}$ is invertible so $F_{\mathcal{I}_t,:} - F_{\mathcal{I}_{t-1},:} = 0_{n_u \times T n_y}$ if and only if $Q_{\mathcal{I}_t,:} - Q_{\mathcal{I}_{t-1},:} = 0$.

Finally, thanks to (5.14b), one can write

$$r_{\mathcal{I}_t} - r_{\mathcal{I}_{t-1}} = f_{\mathcal{I}_t} - f_{\mathcal{I}_{t-1}} + (Q_{\mathcal{I}_t,:} - Q_{\mathcal{I}_{t-1},:})\bar{C}Sf,$$

where, thanks to the observation following (C.1), the last term is cancelled in both the necessary and the sufficient cases, i.e., when $Q_{\mathcal{I}_t,:} - Q_{\mathcal{I}_{t-1},:} = 0$ and when $F_{\mathcal{I}_t,:} - F_{\mathcal{I}_{t-1},:} = 0$. It leads to $F_{\mathcal{I}_t,:} - F_{\mathcal{I}_{t-1},:} = 0$ and $f_{\mathcal{I}_t} - f_{\mathcal{I}_{t-1}} = 0$ if and only if $Q_{\mathcal{I}_t,:} - Q_{\mathcal{I}_{t-1},:} = 0$ and $r_{\mathcal{I}_t} - r_{\mathcal{I}_{t-1}} = 0$.

The case $t = 0$ is similar, indeed, consider that $u(-1), f_{-1}, F_{(-1,\tau)}$ and all quantities indexed by $\mathcal{I}_{-1}$ are zeros with compatible dimensions.

# Bibliography


[AAMJ17] Alexandre Aksenov, Pierre-Olivier Amblard, Olivier Michel, and Christian Jutten. Optimal measurement times for observing a brownian motion over a finite period using a kalman filter. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 509–518. Springer, 2017.

[AAMJ19] Alexandre Aksenov, Pierre-Olivier Amblard, Olivier Michel, and Christian Jutten. Optimal measurement times for a small number of measures of a brownian motion over a finite period. *arXiv preprint arXiv:1902.06126*, 2019.

[Ado88] George Adomian. *Nonlinear stochastic systems theory and applications to physics*, volume 46. Springer Science & Business Media, 1988.

[AL12] Ludwig Arnold and René Lefever. *Stochastic Nonlinear Systems in Physics, Chemistry, and Biology: Proceedings of the Workshop Bielefeld, Fed. Rep. of Germany, October 5–11, 1980*, volume 8. Springer Science & Business Media, 2012.

[AMGC02] M Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on signal processing*, 50(2):174–188, 2002.

[ANSV13] Vincent Andrieu, Madiha Nadri, Ulysse Serres, and Jean-Claude Vivalda. Continuous discrete observer with updated sampling period. *IFAC Proceedings Volumes*, 46(23):439–444, 2013.

[ATD05] Henrik Andreasson, André Treptow, and Tom Duckett. Localization for mobile robots using panoramic vision, local features and particle filter. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 3348–3353. IEEE, 2005.

[Ath72] Michael Athans. On the determination of optimal costly measurement strategies for linear stochastic systems. *Automatica*, 8(4):397–412, 1972.

[Bau13] Johannes Bausch. On the efficient calculation of a linear combination of chi-square random variables with an application in counting string vacua. *Journal of Physics A: Mathematical and Theoretical*, 46(50):505202, 2013.

[BDT18] Suda Bharadwaj, Rayna Dimitrova, and Ufuk Topcu. Synthesis of surveillance strategies via belief abstraction. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 4159–4166. IEEE, 2018.

[BHI+15] Ivo Bukovsky, Noriyasu Homma, Kei Ichiji, Matous Cejnek, Matous Slama, Peter M Benes, and Jiri Bila. A fast neural network approach to predict lung tumor motion during respiration for radiation therapy applications. *BioMed research international*, 2015, 2015.

[BPW02] M. S. Branicky, S. M. Phillips, and Wei Zhang. Scheduling and feedback co-design for networked control systems. In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, volume 2, pages 1211–1217 vol.2, 2002. doi:10.1109/CDC.2002.1184679.

[BTN99] A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations Research Letters*, 25(1):1–13, 1999. URL: https://www.sciencedirect.com/science/article/pii/S0167637799000164, doi:https://doi.org/10.1016/S0167-6377(99)00016-4.

[CJDP+06] Jung Uk Cho, Seung Hun Jin, Xuan Dai Pham, Jae Wook Jeon, Jong Eun Byun, and Hoon Kang. A real-time object

tracking system using a particle filter. In *2006 IEEE/RSJ international conference on intelligent robots and systems*, pages 2822–2827. IEEE, 2006.

[CLM10] Jinguang Chen, Jiancheng Li, and Lili Ma. Nonlinear filtering with multiple packet dropouts. In *2010 International Conference on Image Analysis and Signal Processing*, pages 83–88. IEEE, 2010.

[CPS92] Bradley P. Carlin, Nicholas G Polson, and David S Stoffer. A monte carlo approach to nonnormal and nonlinear state-space modeling. *Journal of the American Statistical Association*, 87(418):493–500, 1992.

[CRZL20] Zhijian Cheng, Hongru Ren, Bin Zhang, and Renquan Lu. Distributed kalman filter for large-scale power systems with state inequality constraints. *IEEE Transactions on Industrial Electronics*, 2020.

[CZA09] Francesco Cadini, Enrico Zio, and DIANA Avram. Monte carlo-based filtering for fatigue crack growth estimation. *Probabilistic Engineering Mechanics*, 24(3):367–373, 2009.

[DCM19] Daniel Delahaye, Supatcha Chaimatanan, and Marcel Mongeau. Simulated annealing: From basics to applications. In *Handbook of Metaheuristics*, pages 1–35. Springer, 2019.

[DJ09] Arnaud Doucet and Adam M Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of nonlinear filtering*, 12(656-704):3, 2009.

[DL71] Peter Dorato and A Levis. Optimal linear regulators: The discrete-time case. *IEEE Transactions on Automatic Control*, 16(6):613–620, 1971.

[DM97] Pierre Del Moral. Nonlinear filtering: Interacting particle resolution. *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, 325(6):653–658, 1997.

[Dou98] Arnaud Doucet. On sequential simulation-based methods for bayesian filtering. 1998.

[DRO93] Vassilios Digalakis, Jan Robin Rohlicek, and Mari Ostendorf. Ml estimation of a stochastic linear system with the em algorithm and its application to speech recognition. *IEEE Transactions on speech and audio processing*, 1(4):431–442, 1993.

[DSW13] C. De Persis, R. Sailer, and F. Wirth. Parsimonious event-triggered distributed control: A zeno free approach. *Automatica*, 49(7):2116–2124, 2013. URL: https://www.sciencedirect.com/science/article/pii/S0005109813001726, doi:https://doi.org/10.1016/j.automatica.2013.03.003.

[DVAD02] Arnaud Doucet, B-N Vo, Christophe Andrieu, and Manuel Davy. Particle filtering for multi-target tracking and sensor management. In *Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002.(IEEE Cat. No. 02EX5997)*, volume 1, pages 474–481. IEEE, 2002.

[DVB+18] Jennifer Dhont, Jef Vandemeulebroucke, Manuela Burghelea, Kenneth Poels, Tom Depuydt, Robbe Van Den Begin, Cyril Jaudet, Christine Collen, Benedikt Engels, Truus Reynders, et al. The long-and short-term variability of breathing induced tumor motion in lung and liver over the course of a radiotherapy treatment. *Radiotherapy and Oncology*, 126(2):339–346, 2018.

[Fle71] Joseph L Fleiss. On the distribution of a linear combination of independent chi squares. *Journal of the American Statistical Association*, 66(333):142–144, 1971.

[FLHI+18] Vincent François-Lavet, Peter Henderson, Riashat Islam, Marc G Bellemare, and Joelle Pineau. An introduction to deep reinforcement learning. *arXiv preprint arXiv:1811.12560*, 2018.

[FP00] Jan A Freund and Thorsten Pöschel. *Stochastic processes in physics, chemistry, and biology*, volume 557. Springer Science & Business Media, 2000.

[Gao06] Zhiqiang Gao. Active disturbance rejection control: a paradigm shift in feedback control system design. In *2006 American control conference*, pages 7–pp. IEEE, 2006.

[GCHM06] Vijay Gupta, Timothy H Chung, Babak Hassibi, and Richard M Murray. On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage. *Automatica*, 42(2):251–260, 2006.

[GGB+02] Fredrik Gustafsson, Fredrik Gunnarsson, Niclas Bergman, Urban Forssell, Jonas Jansson, Rickard Karlsson, and P-J Nordlund. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on signal processing*, 50(2):425–437, 2002.

[GO21] LLC Gurobi Optimization. Gurobi optimizer reference manual, 2021. URL: `http://www.gurobi.com`.

[God19] Simon Godsill. Particle filtering: the first 25 years and beyond. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7760–7764. IEEE, 2019.

[GOR10] Roman Garnett, Michael A Osborne, and Stephen J Roberts. Bayesian optimization for sensor set selection. In *Proceedings of the 9th ACM/IEEE international conference on information processing in sensor networks*, pages 209–219, 2010.

[GSS93] Neil J Gordon, David J Salmond, and Adrian FM Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings F-radar and signal processing*, volume 140, pages 107–113. IET, 1993.

[Gus10] Fredrik Gustafsson. Particle filter theory and practice with positioning applications. *IEEE Aerospace and Electronic Systems Magazine*, 25(7):53–82, 2010.

[HC06] Ying He and Edwin KP Chong. Sensor scheduling for target tracking: A monte carlo sampling approach. *Digital Signal Processing*, 16(5):533–545, 2006.

[HC19] Jung-Su Ha and Han-Lim Choi. On periodic optimal solutions of persistent sensor planning for continuous-time linear systems. *Automatica*, 99:138–148, 2019.

[Hen89] Jean-Claude Hennet. Une extension du lemme de farkas et son application au probleme de régulation linéaire sous contraintes. *CR Acad. Sci. Paris*, 308(I):415–419, 1989.

[HJT12] W. P. M. H. Heemels, K. H. Johansson, and P. Tabuada. An introduction to event-triggered and self-triggered control. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 3270–3285, 2012. doi:10.1109/CDC.2012.6425820.

[HK93] Stephen J Hartley and Aaron H Konstam. Using genetic algorithms to generate steiner triple systems. In *Proceedings of the 1993 ACM Conference on Computer Science*, pages 366–371. ACM, 1993.

[HLCP01] Carine Hue, J-P Le Cadre, and Patrick Pérez. A particle filter to track multiple objects. In *Proceedings 2001 IEEE Workshop on Multi-Object Tracking*, pages 61–68. IEEE, 2001.

[HNX07] Joo P Hespanha, Payam Naghshtabrizi, and Yonggang Xu. A survey of recent results in networked control systems. *Proceedings of the IEEE*, 95(1):138–162, 2007.

[HTVdWN10] WP Maurice H Heemels, Andrew R Teel, Nathan Van de Wouw, and Dragan Nešić. Networked control systems with communication constraints: Tradeoffs between transmission intervals, delays and performance. *IEEE Transactions on Automatic control*, 55(8):1781–1796, 2010.

[JBOBDT16] Mayank Shekhar Jha, Mathieu Bressel, Belkacem Ould-Bouamama, and Genevieve Dauphin-Tanguy. Particle filter based hybrid prognostics of proton exchange membrane fuel cell in bond graph framework. *Computers & Chemical Engineering*, 95:216–230, 2016.

[JKH17] Raphael M Jungers, Atreyee Kundu, and WPMH Heemels. Observability and controllability analysis of linear systems subject to data losses. *IEEE Transactions on Automatic Control*, 63(10):3361–3376, 2017.

[JKH18] R. M. Jungers, A. Kundu, and W. P. M. H. Heemels. Observability and controllability analysis of linear systems subject to data losses. *IEEE Transactions on Automatic Control*, 63(10):3361–3376, 2018. doi:10.1109/TAC.2017.2781374.

[Kal60] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[KBH94] Sami Khuri, Thomas Bäck, and Jörg Heitkötter. An evolutionary approach to combinatorial optimization problems. In *ACM Conference on Computer Science*, pages 66–73, 1994.

[KCS19] Mojtaba Kordestani, Ali Chibakhsh, and Mehrdad Saif. A control oriented cyber-secure strategy based on multiple sensor fusion. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 1875–1881. IEEE, 2019.

[KDMS20] Mojtaba Kordestani, Maryam Dehghani, Behzad Moshiri, and Mehrdad Saif. A new fusion estimation method for multi-rate multi-sensor systems with missing measurements. *IEEE Access*, 8:47522–47532, 2020.

[KF15] Markus Kögel and Rolf Findeisen. Robust output feedback predictive control with self-triggered measurements. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 5487–5493. IEEE, 2015.

[Kit93] Genshiro Kitagawa. A monte carlo filtering and smoothing method for non-gaussian nonlinear state space models. In *Proceedings of the 2nd US-Japan joint seminar on statistical time series analysis*, pages 110–131, 1993.

[Kit96] Genshiro Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of computational and graphical statistics*, 5(1):1–25, 1996.

[KLJF02] Visakan Kadirkamanathan, Ping Li, Mohamed H Jaward, and Simon G Fabri. Particle filtering-based fault detection in non-linear stochastic systems. *International Journal of Systems Science*, 33(4):259–265, 2002.

[KN19] Hazhar Sufi Karimi and Balasubramaniam Natarajan. Kalman filtered compressive sensing with intermittent observations. *Signal Processing*, 163:49–58, 2019.

[Kri02] Vikram Krishnamurthy. Algorithms for optimal scheduling and management of hidden markov model sensors. *IEEE Transactions on Signal Processing*, 50(6):1382–1397, 2002.

[LC98] Jun S Liu and Rong Chen. Sequential monte carlo methods for dynamic systems. *Journal of the American statistical association*, 93(443):1032–1044, 1998.

[LCP10] Yan Liang, Tongwen Chen, and Quan Pan. Optimal linear state estimator with multiple packet dropouts. *IEEE Transactions on Automatic Control*, 55(6):1428–1433, 2010.

[LFMV14] Sijia Liu, Makan Fardad, Engin Masazade, and Pramod K Varshney. Optimal periodic sensor scheduling in networks of dynamical systems. *IEEE Transactions on Signal Processing*, 62(12):3055–3068, 2014.

[LKCG09] Yun Li, Lucas W Krakow, Edwin KP Chong, and Kenneth N Groom. Approximate stochastic dynamic programming for sensor scheduling to track multiple targets. *Digital Signal Processing*, 19(6):978–989, 2009.

[LL11] Laurent Lessard and Sanjay Lall. Quadratic invariance is necessary and sufficient for convexity. In *Proceedings of the 2011 American Control Conference*, pages 5360–5362. IEEE, 2011.

[LNFD10] Jerome Le Ny, Eric Feron, and Munther A Dahleh. Scheduling continuous-time kalman filters. *IEEE Transactions on Automatic Control*, 56(6):1381–1394, 2010.

[LSSC14] Tiancheng Li, Shudong Sun, Tariq Pervez Sattar, and Juan Manuel Corchado. Fight sample degeneracy and impoverishment in particle filters: A review of intelligent approaches. *Expert Systems with applications*, 41(8):3944–3954, 2014.

[LTL01] HW Joseph Lee, Kok Lay Teo, and Andrew EB Lim. Sensor scheduling in continuous time. *Automatica*, 37(12):2017–2023, 2001.

[LWYG16] Wenshuo Li, Zidong Wang, Yuan Yuan, and Lei Guo. Particle filtering with applications in networked systems: a survey. *Complex & Intelligent Systems*, 2(4):293–315, 2016.

[LXP17] Frank L Lewis, Lihua Xie, and Dan Popa. *Optimal and Robust Estimation: with an Introduction to Stochastic Control Theory*. CRC press, 2017.

[MGS14] Yilin Mo, Emanuele Garone, and Bruno Sinopoli. On infinite-horizon sensor scheduling. *Systems & control letters*, 67:65–70, 2014.

[Mit98] Melanie Mitchell. *An introduction to genetic algorithms*. MIT press, 1998.

[MP14] Nacim Meslem and Christophe Prieur. State estimation based on self-triggered measurements. *IFAC Proceedings Volumes*, 47(3):86–91, 2014.

[MPD67] Lewis Meier, John Peschon, and Robert Dressler. Optimal control of measurement subsystems. *IEEE Transactions on Automatic Control*, 12(5):528–536, 1967.

[Oks13] Bernt Oksendal. *Stochastic Differential Equations: an Introduction with Applications*. Springer Science & Business Media, 2013.

[Pap05] Anastasia Papavasiliou. A uniformly convergent adaptive particle filter. *Journal of Applied Probability*, 42(4):1053–1068, 2005.

[PKT17] Michael Antonio Posa, Twan Koolen, and Russell L Tedrake. Balancing and step recovery capturability via sums-of-squares optimization. 2017.

[RB19] Clément Romac and Vincent Béraud. Deep recurrent q-learning vs deep q-learning on a simple partially observable markov decision process with minecraft. *arXiv preprint arXiv:1903.04311*, 2019.

[RGT12] Matthew Rosencrantz, Geoffrey Gordon, and Sebastian Thrun. Decentralized sensor fusion with distributed particle filters. *arXiv preprint arXiv:1212.2493*, 2012.

[RL06] Michael Rotkowitz and Sanjay Lall. A characterization of convex problems in decentralized control. *IEEE transactions on Automatic Control*, 51(2), 2006.

[RLG+11] AA Abd Rahni, E Lewis, MJ Guy, B Goswami, and K Wells. A particle filter approach to respiratory motion estimation in nuclear medicine imaging. *IEEE Transactions on Nuclear Science*, 58(5):2276–2285, 2011.

[RSW+09] Nadeem Riaz, Piyush Shanker, Rodney Wiersma, Olafur Gudmundsson, Weihua Mao, Bernard Widrow, and Lei Xing. Predicting respiratory tumor motion with multidimensional adaptive filters and support vector regression. *Physics in Medicine & Biology*, 54(19):5735, 2009.

[RYO18] Kwesi J Rutledge, Sze Zheng Yong, and Necmiye Ozay. Optimization-based design of bounded-error estimators robust to missing data. *IFAC-PapersOnLine*, 51(16):157–162, 2018.

[RYO20] Kwesi Rutledge, Sze Zheng Yong, and Necmiye Ozay. Finite horizon constrained control and bounded-error estimation in the presence of missing data. *Nonlinear Analysis: Hybrid Systems*, 36:100854, 2020.

[SB10] Joelle Skaf and Stephen P Boyd. Design of affine controllers via convex optimization. *IEEE Transactions on Automatic Control*, 55(11):2476–2487, 2010.

[SBD+09] Saikat Saha, Yvo Boers, Hans Driessen, Pranab Kumar Mandal, and Arunabha Bagchi. Particle based map state estimation: A comparison. In *2009 12th International Conference on Information Fusion*, pages 278–283. IEEE, 2009.

[SCC11] Ling Shi, Peng Cheng, and Jiming Chen. Optimal periodic sensor scheduling with limited resources. *IEEE Transactions on Automatic Control*, 56(9):2190–2195, 2011.

[SDR14] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on stochastic programming: modeling and theory*. SIAM, 2014.

[SHB16] Touraj Soleymani, Sandra Hirche, and John S Baras. Optimal stationary self-triggered sampling for estimation. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 3084–3089. IEEE, 2016.

[SJSS04] Gregory C Sharp, Steve B Jiang, Shinichi Shimizu, and Hiroki Shirato. Prediction of respiratory tumour motion for real-time image-guided radiotherapy. *Physics in Medicine & Biology*, 49(3):425, 2004.

[SKV+07] Sumeetpal S Singh, Nikolaos Kantas, Ba-Ngu Vo, Arnaud Doucet, and Robin J Evans. Simulation-based optimal sensor scheduling with application to observer trajectory planning. *Automatica*, 43(5):817–830, 2007.

[SPM+11] Arjun V Shenoy, Jagadeesan Prakash, KB McAuley, Vinay Prasad, and SL Shah. Practical issues in the application of the particle filter for estimation of chemical processes. *IFAC Proceedings Volumes*, 44(1):2773–2778, 2011.

[SSF+04] Bruno Sinopoli, Luca Schenato, Massimo Franceschetti, Kameshwar Poolla, Michael I Jordan, and Shankar S Sastry. Kalman filtering with intermittent observations. *IEEE transactions on Automatic Control*, 49(9):1453–1464, 2004.

[SSF+07] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry. Foundations of control and estimation over lossy networks. *Proceedings of the IEEE*, 95(1):163–187, Jan 2007. doi:10.1109/JPROC.2006.887306.

[ST70] A Sano and M Terao. Measurement optimization in optimal process control. *Automatica*, 6(5):705–714, 1970.

[ST20] Sadra Sadraddini and Russ Tedrake. Robust output feedback control with guaranteed constraint satisfaction. In *Proceedings of the 23rd International Conference on Hybrid Systems: Computation and Control*, pages 1–10, 2020.

[Ste90] C Felter Stephen. An overview of decentralized kalman filter techniques. *Transactions of the Technical Section of Canadian Pulp and Paper Association*, pages 79–87, 1990.

[SVDE03] Sumeetpal Singh, Ba-Ngu Vo, Arnaud Doucet, and Robin Evans. Stochastic approximation for optimal observer trajectory planning. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, volume 6, pages 6313–6318. IEEE, 2003.

[SVDE04] Sumeetpal Singh, Ba-Ngu Vo, A Doucet, and R Evans. Variance reduction for monte carlo implementation of adaptive sensor management. In *Proceedings of the international conference on information fusion*, pages 901–908, 2004.

[SXXS08] Shuli Sun, Lihua Xie, Wendong Xiao, and Yeng Chai Soh. Optimal linear estimation for systems with multiple packet dropouts. *Automatica*, 44(5):1333–1342, 2008.

[SYC12] Ling Shi, Ye Yuan, and Jiming Chen. Finite horizon lqr control with limited controller-system communication. *IEEE Transactions on Automatic Control*, 58(7):1835–1841, 2012.

[SYL19] Xinmin Song, Xuehua Yan, and Xiaodi Li. Survey of duality between linear quadratic regulation and linear estimation problems for discrete-time systems. *Advances in Difference Equations*, 2019(1):90, 2019.

[TAJP17] Vasileios Tzoumas, Nikolay A Atanasov, Ali Jadbabaie, and George J Pappas. Scheduling nonlinear sensors for stochastic process estimation. In *2017 American Control Conference (ACC)*, pages 580–585. IEEE, 2017.

[TB10] M. Trivellato and N. Benvenuto. State control in networked control systems under packet drops and limited transmission bandwidth. *IEEE Transactions on Communications*, 58(2):611–622, 2010. doi:10.1109/TCOMM.2010.02.080289.

[Thr02] Sebastian Thrun. Particle filters in robotics. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 511–518. Morgan Kaufmann Publishers Inc., 2002.

[TJP16a] Vasileios Tzoumas, Ali Jadbabaie, and George J Pappas. Near-optimal sensor scheduling for batch state estimation: Complexity, algorithms, and limits. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 2695–2702. IEEE, 2016.

[TJP16b] Vasileios Tzoumas, Ali Jadbabaie, and George J Pappas. Sensor placement for optimal kalman filtering: Fundamental limits, submodularity, and algorithms. In *2016 American Control Conference (ACC)*, pages 191–196. IEEE, 2016.

[TO85] Shogo Tanaka and Tsuyoshi Okita. Optimal timing of observations for state estimation in a one-dimensional linear continuous system. *Automatica*, 21(3):329–331, 1985.

[Tod08] Emanuel Todorov. General duality between optimal control and estimation. In *2008 47th IEEE Conference on Decision and Control*, pages 4286–4292. IEEE, 2008.

[US15] AJ Umbarkar and PD Sheth. Crossover operators in genetic algorithms: A review. *ICTACT Journal on Soft Computing*, 6(1), 2015.

[VZA+12] Michael P Vitus, Wei Zhang, Alessandro Abate, Jianghai Hu, and Claire J Tomlin. On efficient sensor scheduling for linear dynamical systems. *Automatica*, 48(10):2482–2493, 2012.

[WMD19] Y. Wang, N. Matni, and J. C. Doyle. A system-level approach to controller synthesis. *IEEE Transactions on Automatic Control*, 64(10):4079–4093, 2019. doi:10.1109/TAC.2018.2890753.

[WRL10] Siew Fang Woon, Volker Rehbock, and Ryan Loxton. Global optimization method for continuous-time sensor scheduling. *Nonlinear Dynamics and Systems Theory*, 10:175–188, 2010.

[WVB+11] Ruth Wuyts, Elke Vlemincx, Katleen Bogaerts, Ilse Van Diest, and Omer Van den Bergh. Sigh rate and respiratory variability during normal breathing and the role of negative affectivity. *International Journal of Psychophysiology*, 82(2):175–179, 2011.

[XH04] Yonggang Xu and Joao P Hespanha. Optimal communication logics in networked control systems. In *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, volume 4, pages 3527–3532. IEEE, 2004.

[YD16] Chunshan Yang and Zili Deng. Guaranteed cost robust weighted measurement fusion kalman estimators with uncertain noise variances and missing measurements. *IEEE Sensors Journal*, 16(14):5817–5825, 2016.

[YFS19] Chen Yang, Huajing Fang, and Bing Shi. Particle filter with markovian packet dropout and time delay. *Journal of the Franklin Institute*, 356(1):675–696, 2019.

[YG06] Yuanxi Yang and Weiguang Gao. An optimal adaptive kalman filter. *Journal of Geodesy*, 80(4):177–183, 2006.

[YJB76] D. Youla, H. Jabr, and J. Bongiorno. Modern wiener-hopf design of optimal controllers–part ii: The multivariable case. *IEEE Transactions on Automatic Control*, 21(3):319–338, 1976. doi:10.1109/TAC.1976.1101223.

[YJCW20] Chunshan Yang, Jianbo Ji, Shaohang Chen, and Xuemei Wang. Guaranteed cost robust centralized fusion kalman estimators with uncertain noise variances and missing measurements. In *2020 39th Chinese Control Conference (CCC)*, pages 2752–2757. IEEE, 2020.

[YN18] Xiaojun Yang and Ruixin Niu. Adaptive sensor selection for nonlinear tracking via sparsity-promoting approaches. *IEEE Transactions on Aerospace and Electronic Systems*, 54(4):1966–1982, 2018.

[YO21] Liren Yang and Necmiye Ozay. Safety control synthesis for systems with missing measurements. *IFAC-PapersOnLine*, 54(5):97–102, 2021.

[YS18] Xuejiao Yang and Joseph K Scott. A comparison of zonotope order reduction techniques. *Automatica*, 95:378–384, 2018.

[ZHG+19] Xian-Ming Zhang, Qing-Long Han, Xiaohua Ge, Derui Ding, Lei Ding, Dong Yue, and Chen Peng. Networked control systems: A survey of trends and techniques. *IEEE/CAA Journal of Automatica Sinica*, 7(1):1–17, 2019.

[ZHZ17] X. Zhang, Q. Han, and B. Zhang. An overview and deep investigation on sampled-data-based event-triggered control and filtering for networked systems. *IEEE Transactions on Industrial Informatics*, 13(1):4–16, 2017. doi:10.1109/TII.2016.2607150.

[ZZH+14] Lin Zhao, Wei Zhang, Jianghai Hu, Alessandro Abate, and Claire J Tomlin. On the optimal solutions of the infinite-horizon linear sensor scheduling problem. *IEEE Transactions on Automatic Control*, 59(10):2825–2830, 2014.