Czech Technical University in Prague
Faculty of Nuclear Sciences and Physical Engineering

Sequential Monte Carlo methods for optimal control design

Research project

Author: Miroslav Zima
Supervisor: Ing. Václav Šmídl , Ph.D
Year: 2010

Declaration

I declare that I have written my research project myself and I have used only cited materials.

In Prague .....

.....
Miroslav Zima

Acknowledgment

I would like to thank my supervisors Ing. Václav Šmídl, Ph.D for supervising. I have greatly benefited from his insight and assistance which helped to develop the project.

Miroslav Zima

Title:
Sequential Monte Carlo methods for optimal control design

Author: Miroslav Zima

Specialization: Mathematical Engineering
Supervisor: Ing. Václav Šmídl , Ph.D
Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic

Abstract: This project deals with application of sequential Monte Carlo method on area of optimal control. Discussed methods are applied on a model of permanent magnet synchronous machine drive. Also, new regulator, based on catious and dual control principles, was for the system obtaining comparable results with conventional PID regulator.

Key words: Estimation, sequential Monte Carlo Method, optimal contro

Contents

List of abbreviations

ASIR	Auxiliary Sequential Importance Sampling
CC	Cautious Control
CEC	Certainty Equivalence Control
EKF	Extended Kalman Filter
IS	Importance Sampling
KF	Kalman Filter
OLFC	Open Loop Feedback Control
PF	Particle Filter
PID	Proportional Integral Derivative
PI	Proportional Integral
PMSM	Permanent Magnet Synchronous Machine
SIDP	Stochastic Iterative dynamic programming
SIR	Sequential Importance Resampling
SIR_{opt}	Sequential Importance Resampling with optimal importance distribution
$\text{SIR}_{\text{prior}}$	Sequential Importance Resampling with prior importance distribution
SIS	Sequential Importance Sampling
SMC	Sequential Monte Carlo

Introduction

In wide range of subjects, we are dealing with problem of making inferences about a hidden state of dynamical stochastic system using only noisy observations provided by the system. This framework occurs for instance in finance [15], signal processing [9], or control theory [27]. Due to the system dynamics, we would like to make inferences sequentially every time where a measurement is received. In this case, a recursive filter is a convenient solution. A recursive filtering approach means that received data can be processed sequentially rather than as a batch so that it is not necessary to store the complete data set nor to reprocess existing data if a new measurement becomes available. Process of estimation is usually done in two stages: prediction and update. In prediction stage the estimate available in current time is propagated using the system model and the observation model. The estimate is then updated through Bayes rule by comparing predicted and measured values.

If the system is linear with Gaussian noise, the optimal solution for state estimation can be computed analytically by so called Kalman filter [18]. In more general cases, the optimal solution can not be computed in closed for thus some approximation technique have to be used. For example extended Kalman filter uses local linearization. Nonetheless, in highly nonlinear cases, the linearization is not appropriate approximation and consequently the estimates based on linearization technique may be inaccurate. This disadvantage can be reduced for example by using so called unscented transformation which is an approach of unscented Kalman filter [16]. However Kalman filter and all its derivatives are based on Gaussian noise assumption and the provided filtering density is also Gaussian.

Another approach is in so called particle filters. Particle filters belongs to the class of simulation filters which recursively approximate the filtering distribution by the cloud of points or 'particles' with point mass distribution. This is a principle of sequential Monte Carlo methods. Main advantage of particle filters is that they are not based on any assumption on linearity of the system or Gaussian distributions, so they can be used in variety of application where standard approach based on Kalman filter suffers.

The topic of this research project lies in the parameter estimation using particle filters and particularly in its application in problematics of optimal control under uncertainty. The problem of optimal control is formulated in the first chapter showing that it consist of two subproblems: 1) to estimate uncertain parameters of the system which is to be controlled and 2) to make a decision which results in desired system behavior. The estimation problem is the subject of the second chapter while

the third chapter deals with the problem of decision making under uncertainty. Comparison of presented estimation techniques includes the fourth chapter. Comparison schemes are focused on different results of particle filters and Kalman filter based methods. It is assumed that particle filters will be more accurate in highly nonlinear situations and in scenarios where Gaussian approximation is far from reality.

Chapter 1

Optimal control problem

In both technical applications and real life, we are tended to making decisions which should be based on knowledge about system of interest. If we would like to control system successfully, we are dealing with two subproblems: 1) to observe the system and 2) to design control action (based on our knowledge) which leads to desired system behavior. However, these two subproblems are often in conflict - the most knowledge about system is obtained when its behavior is unexpected. Moreover, in practice, due to the presence of noise, our knowledge will not be ever absolute.

This chapter aims to a mathematical formulation of optimal control problem. The problem is then decomposed into filtering and decision making which are discussed in later chapters.

1.1 Problem formulation

In control theory [3], it is convention to describe a dynamical system by means of a time discrete state-space model. We consider a Markovian state-space model given by

$$x_t = f_t(x_{t-1}, u_{t-1}) + w_{t-1} \qquad t \geq 1, \tag{1.1}$$

where t is time index, $x_t \in \mathbb{R}^{n_x}$ is the state of the system, $u_t \in U_t \subset \mathbb{R}^{n_u}$ is the control action , $w_t \in \mathbb{R}^{n_w}$ is i.i.d. random variable and $f_t : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ are arbitrary known functions. Initial state x_0 is assumed to be distributed according to some prior density $p(x_0)$. Information about the system are provided by an observation

$$y_t = h_t(x_t) + v_t \qquad t \geq 0, \tag{1.2}$$

here $y_t \in \mathbb{R}^{n_y}$ is the observation, $v_t \in \mathbb{R}^{n_v}$ is i.i.d. random variable and $h_t : \mathbb{R}^{n_y} \times \mathbb{R}^{n_v} \to \mathbb{R}^{n_y}$ are arbitrary known functions.

Suppose, that we are interested in system control within control horizon $0 : n$. Then, the aim is to design control sequence $u_{0:n-1}$, which will lead to desired system behavior. For determination how accurate is some proposed control sequence, a known real function $g(x_{1:t}, u_{0:n-1})$, called loss function, is adopted.

Due to the noise presence and fact that we can observe only y_t, the loss function can not be evaluated directly, thus we are interested in its expectation

$$J = \mathrm{E}\{g(x_{1:t}, u_{0:t-1})\}. \tag{1.3}$$

Finally, the problem of optimal control is to minimize the expectation loss (1.3) with respect to $u_{0:t-1} \in U_{0:t-1}$.

1.1.1 Sequential parameter estimation

According to the system (1.1) and observation (1.2), the model can be viewed as hidden Markov model described by (without any loss of generality we omit the dependence on u_t)

$$p(x_t|x_{t-1}) \qquad t \geq 0, \tag{1.4}$$
$$p(y_t|x_t) \qquad t \geq 0, \tag{1.5}$$

where we denote $p(x_0) := p(x_0|x_{-1})$ for notation convenience.

The aim of the estimation in time t is to compute $p(x_{0:t}|y_{0:t})$ and expectation

$$\mathrm{E}\{g_t\} = \int g_t(x_{0:t}) p(x_{0:t}|y_{0:t}) dx_{0:t} \tag{1.6}$$

for any $p(x_{0:t}|y_{0:t})$ integrable function $g_t : \mathbb{R}^{(t+1)\times n_x} \to \mathbb{R}$. Particullarly, we will be interested in computation of marginal $p(x_t|y_{0:t})$.

1.1.2 Decision making

Generally, the decision making can be done in two different ways. At first, whole action sequence can be computed before simulation and during simulations there is no gain from feedback of the system (open-loop control). This off-line approach, is reasonable only in cases where measurement is not available or is very unreliable. This case is off our interest.

In closed-loop approach, the control actions are computed on-line. It means, that u_{t-1} is computed based on current estimate of x_{t-1} which is sequentially update using measurement y_t. This situation is schematically depicted in the figure 1.1. For closed-loop control it is necessary to have some parametric formula for control action. This parametric formula is usually computed off-line and only evaluation using specific values is performed on-line. Particularly in time-critical applications it is necessary to be able evaluate control actions effectively.

Of course, control sequence proposed by any parametric formula along control horizon should minimize the expectation loss (1.3). Even if we be able to compute expectation loss for every $u_{0:t-1} \in U_{0:t-1}$, direct minimization over action space $U_{0:t-1}$ will be impossible already for very simple problems. Due to this complications, many approximation techniques or problem specifications have been proposed,

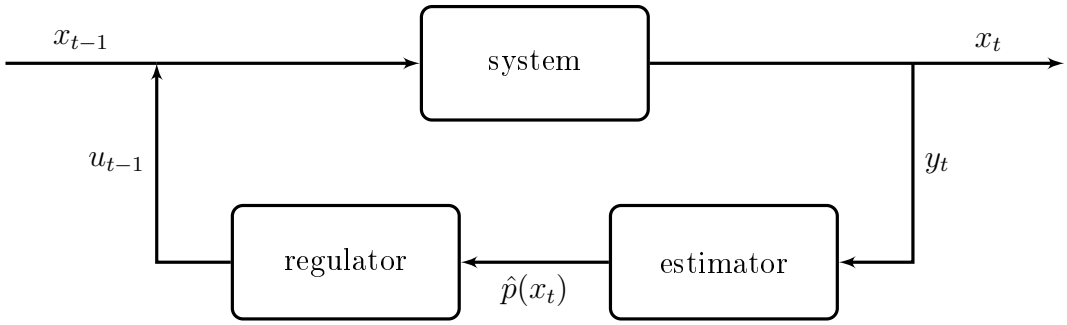

Figure 1.1: Close-loop control.

some general approaches are presented in corresponding chapter. Using an approximation of optimal control, it is often crucial to use closed-loop control because we can evaluate how the proposed control action leads to the desired state and possibly tune the parametric formula to be more effective. On this very simple (but powerful) idea is based e.g. PID regulator [17].

Chapter 2

Sequential parameter estimation

In this chapter, we present general framework for sequential parameter estimation. The main interest lies in sequential Monte Carlo approach. For later comparison, Kalman filter based techniques are also briefly presented.

2.1 Parametr estimation with perfect Monte Carlo simulation

Before we induce the sequential Monte Carlo, we briefly introduce a traditional Monte Carlo method. Suppose, that the aim is to estimate the expectation for function $g_t : \mathbb{R}^{(n+1)\times n_x} \to \mathbb{R}$

$$\mathrm{E}\{g_t\} = \int g_t(x_{0:t}) p(x_{0:t}|y_{0:t}) dx_{0:t} \tag{2.1}$$

The idea of perfect Monte Carlo is to approximate density $p(x_{0:t}|y_{0:t})$ by an empirical estimate

$$\hat{p}(x_{0:t}|y_{0:t}) = \frac{1}{N}\sum_{i=1}^{N} \delta(x_{0:t} - x_{0:t}^{(i)}), \tag{2.2}$$

where $\{x_{0:t}^{(i)}\}_{i=1}^{N}$ are random samples drawn from density $p(x_{0:t}|y_{0:t})$ and δ is the Dirac delta function. A Monte Carlo estimate is obtained by substituting the approximation (2.2) into (2.1), thus

$$\mathrm{E}\{g_t\} \approx \hat{\mathrm{E}}\{g_t\} = \int g_t(x_{0:t}) \hat{p}(dx_{0:t}|y_{0:t}) dx_{0:t} = \frac{1}{N}\sum_{i=1}^{N} g_t(x_{0:t}^{(i)}). \tag{2.3}$$

Due to the strong law of large numbers, $\hat{\mathrm{E}}(g_t)$ converges to $\mathrm{E}(g_t)$ almost surely and, if the variance σ of $g_t(x_{0:t})$ is finite, a central limit theorems holds

$$\sqrt{N}(\hat{\mathrm{E}}(g_t) - \mathrm{E}(g_t)) \to \mathcal{N}(0, \sigma^2) \qquad \text{if } N \to \infty. \tag{2.4}$$

The major problem of this approach is the assumption that we can sample from density $p(x_{0:t}|y_{0:t})$.

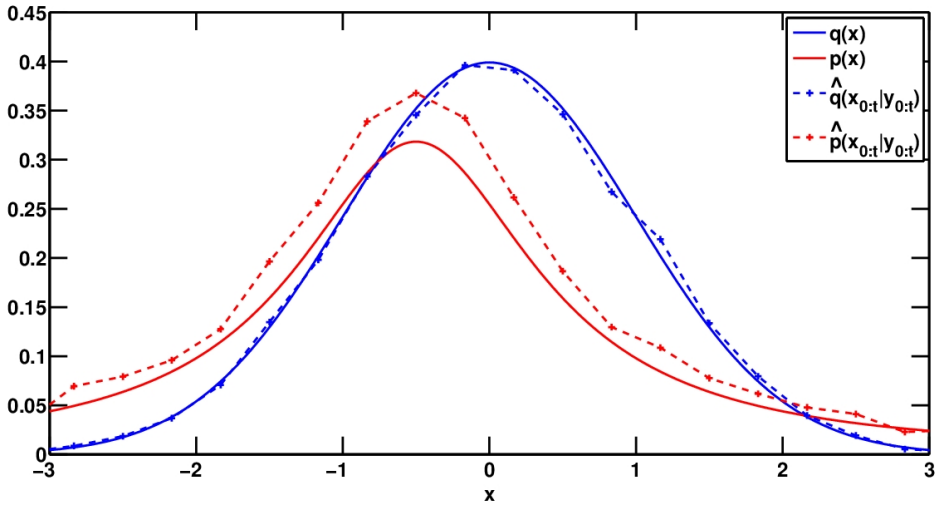

Figure 2.1: Using the IS, the original density $p(x_{0:t}|y_{0:t})$ is approximated by $\hat{p}(x_{0:t}|y_{0:t})$ which is realized by weighted samples $\{x_{0:t}^{(i)}\}_{i=1}^{N}$ drawn from importance density $q(x_{0:t}|y_{0:t})$. Because the probability distribution outside interval covered by particles is not taken into account in approximation, the IS estimate overestimates the probability distribution inside the interval. Here, 10000 samples is used

2.2 Importance sampling

Another approximation of expectation (2.1) can be done using Importance Sampling (IS) method [12]. Unlike previous method, the IS works even if we ca not sample from density $p(x_{0:n}|y_{0:n})$. The idea of the IS is to use a well known density $q(x_{0:n}|y_{0:n})$, so called importance density, instead of $p(x_{0:n}|y_{0:n})$, see figure 2.1. If support of $q(x_{0:n}|y_{0:n})$ includes support of $p(x_{0:n}|y_{0:n})$ the expectation can be expressed as

$$\mathrm{E}\{g_t\} = \int g_t(x_{0:t})p(x_{0:t}|y_{0:t})dx_{0:t} = \frac{\int g_t(x_{0:n})\omega(x_{0:t})q(x_{0:t}|y_{0:t})dx_{0:t}}{\int \omega(x_{0:t})q(x_{0:t}|y_{0:t})dx_{0:t}}, \tag{2.5}$$

where importance weighs $\omega(x_{0:t})$ are defined as

$$\omega(x_{0:t}) = \frac{p(x_{0:t}|y_{0:t})}{q(x_{0:t}|y_{0:t})}. \tag{2.6}$$

The Monte Carlo estimate can be obtained by using random samples $\{x_{0:t}^{(i)}\}_{i=1}^{N}$ drawn from importance density $q(x_{0:t}|y_{0:t})$ in (2.5)

$$\hat{\mathrm{E}}(g_t) = \frac{\frac{1}{N}\sum_{i=1}^{N} g_t(x_{0:t}^{(i)})\omega_t^{(i)}}{\frac{1}{N}\sum_{i=1}^{N}\omega_t^{(i)}} = \sum_{i=1}^{N} g_t(x_{0:t}^{(i)})\tilde{\omega}_t^{(i)}, \tag{2.7}$$

where the normalized importance weighs $\tilde{\omega}_t^{(i)}$ are

$$\tilde{\omega}_t^{(i)} = \frac{\omega_t^{(i)}}{\sum_{i=1}^{N}\omega_t^{(i)}} = \frac{\omega(x_{0:t}^{(i)})}{\sum_{i=1}^{N}\omega(x_{0:t}^{(i)})} \tag{2.8}$$

Again, due to the strong law of large numbers, the estimate $\hat{p}(dx_{0:t}|y_{0:t})$ based on $\hat{q}(dx_{0:t}|y_{0:t})$ and $\{\tilde{\omega}_t^{(i)}\}_{i=1}^N$ converges to the true posterior density $p(dx_{0:t}|y_{0:t})$ if N tends to infinity, and thus even estimated expectation $\hat{\mathrm{E}}(g_t)$ converges to the $\mathrm{E}(g_t)$.

Disadvantage of this approach is that we need to evaluate $p(x_{0:t}^{(i)}|y_{0:t})$ and the importance weights have to be recomputed over the entire state sequence. Consequently, computational complexity increases in time.

2.3 Sequential importance sampling

For $p(x_{0:t}|y_{0:t})$, it can be derived

$$\begin{aligned} p(x_{0:t}|y_{0:t}) &= \frac{p(y_t|x_{0:t}, y_{0:t-1})p(x_{0:t}|y_{0:t-1})}{p(y_t|y_{0:t-1})} \\ &= \frac{p(y_t|x_{0:t}, y_{0:t-1})p(x_{0:t}|y_{0:t-1})}{p(y_t|y_{0:t-1})} \\ &= \frac{p(y_t|x_{0:t}, y_{0:t-1})p(x_t|x_{0:t-1}, y_{0:t-1})p(x_{0:t-1}|y_{0:t-1})}{p(y_t|y_{0:t-1})} \\ &= \frac{p(y_t|x_t)p(x_t|x_{t-1})p(x_{0:t-1}|y_{0:t-1})}{p(y_t|y_{0:t-1})}, \end{aligned}$$

where the first two equalities follow from Bayesian rule, the third from definition of conditional distribution, and the last one from Markovian property. Thus, the density satisfies a recursive formula

$$p(x_{0:t}|y_{0:t}) \propto p(y_t|x_t)p(x_t|x_{t-1})p(x_{0:t-1}|y_{0:t-1}). \tag{2.9}$$

For derivation a recursive formula for importance weights, it is suitable to have an importance density which satisfies

$$q(x_{0:t}|y_{0:t}) = q(x_t|x_{0:t-1}, y_{0:t-1})q(x_{0:t-1}|y_{0:t-1}). \tag{2.10}$$

In this particular case, the $x_{0:t}^{(i)}$ is formed as $x_{0:t}^{(i)} = (x_{0:t-1}^{(i)}, x_t^{(i)})$, where $x_t^{(i)}$ is drawn from $q(x_t|x_{0:t-1}^{(i)}, y_{0:t-1})$. The importance weights for every $x_{0:t}^{(i)}$ can be computed sequentially as

$$\omega_t^{(i)} \propto \omega_{t-1}^{(i)} \frac{p(y_t|x_t^{(i)})p(x_t^{(i)}|x_{t-1}^{(i)})}{q(x_t^{(i)}|x_{0:t-1}^{(i)}, y_{0:t-1})}, \tag{2.11}$$

what gives the Sequential Importance Sampling (SIS) filter, see scheme 2.1. The approximation $\hat{\mathrm{E}}(g_t)$ of the expectation $\mathrm{E}(g_t)$ can be computed according to (2.7).

The advantage of using the SIS filter is that we have to sample only only $x_t^{(i)}$ instead of whole $x_{0:t}^{(i)}$ and we do not have to evaluate $p(x_{0:t}^{(i)}|y_{0:t})$.

2.4 Degeneracy of the SIS filter

For good performance of the SIS filter, it is suitable to have the importance density $q(x_{0:t}|y_{0:t})$ close to the true posterior distribution $p(x_{0:t}|y_{0:t})$. However, as can be

Algorithm 2.1 Sequential Importance Sampling

for $t = 1, 2, \ldots$ **do**
  for $i = 1$ to N **do**
    sample $x_t^{(i)}$ from $q(x_t|x_{0:t-1}^{(i)}, y_{0:t-1})$
    set $x_{0:t}^{(i)} = (x_{0:t-1}^{(i)}, x_t^{(i)})$
  end for
  for $i = 1$ to t **do**
    compute importance weights using recursive formula

$$\omega_t^{(i)} = \omega_{t-1}^{(i)} \frac{p(y_t|x_t)p(x_t|x_{t-1})}{q(x_t|x_{0:t-1}, y_{0:t-1})}$$

  end for
  for $i = 1$ to N **do**
    normalize importance weights

$$\tilde{\omega}_t^{(i)} = \frac{\omega_t^{(i)}}{\sum_{i=1}^{N} \omega_t^{(i)}}$$

  end for
end for

seen from following proposition, the variance of importance weights can only increase over time.

Proposition 1. *The variance of importance weights with both $x_{0:t-1}$ $y_{0:t}$ interpreted as random variables increases over time.*

Proof. [22] □

The variance of importance weights can be reduced by using proper importance density. The optimal one is stated in next proposition.

Proposition 2. *The importance density which minimizes the variance of the importance weight $\omega_t^{(i)}$ conditional upon $x_{0:t-1}^{(i)}$ and $x_{0:t-1}^{(i)}$ is $q(x_t|x_{0:t-1}, y_{0:t}) = p(x_t|x_{t-1}, y_t)$*

Proof. [8] □

Using the optimal density in (2.11), the updating procedure for weights will have the form $\omega_t^{(i)} = \omega_{t-1}^{(i)} p(y_t|x_{t-1}^{(i)})$. However the optimal density has two major drawbacks: 1) it requires the ability of sampling from $p(x_t|x_{t-1}, y_t)$, and 2) to calculate $p(y_t|x_{t-1}^{(i)})$.The second term can be principally evaluated using Chapman-Kolmogorov equation which, due to Markov property, has the form

$$p(y_t|x_{t-1}^{(i)}) = \int p(y_t, x_t|x_{t-1}^{(i)})dx_t = \int p(y_t|x_t, x_{t-1})p(x_t|x_{t-1}^{(i)})dx_t =$$
$$= \int p(y_t|x_t)p(x_t|x_{t-1}^{(i)})dx_t.$$

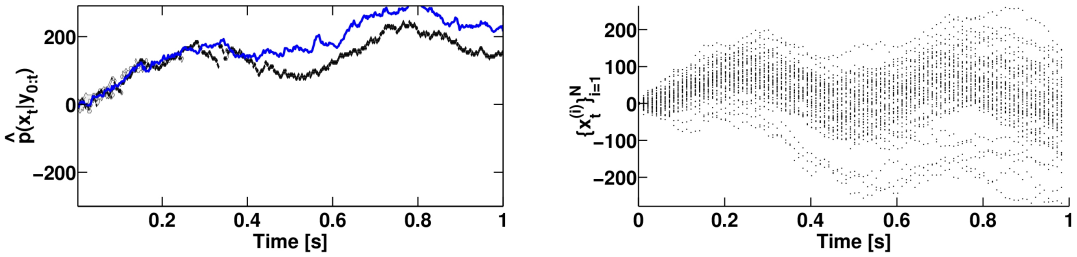

Figure 2.2: Using the SIS filter, the particles are updated by the system equation and weighted with respect to the observation. The left part shows estimated state x_t in particular time, darker color means higher probability. True state is marked by blue line. All the particles in the corresponding times are depicted in the right part. Due to the degeneracy phenomenon, importance weights of majority of particles tend to zero. The number of particles is $N = 50$.

Both $p(x_t|x_{t-1}, y_t)$ and $p(y_t|x_{t-1}^{(i)})$ can be calculated for nonlinear system with linear observation and Gaussian noise, see [8]. However in general case it is not possible and some approximation have to be used. Suitable choice can be e.g. usage of local linearization [8] or prior distribution [25]. This topic, crucial to limit the degeneracy, is discussed in respective chapter on examples.

In practice, after few iteration, major of the normalized importance weights are close to zero due to the degeneracy of the SIS filter, see figure 2.2. Consequently, the major of computation effort is devoted inefficiently to trajectories whose probability is almost zero. Furthermore, any inferences based on these samples will be inaccurate. To overcome this drawback, the resampling procedure is used.

2.5 Resampling

The idea of the resampling procedure is to eliminate particles with small normalized weights and to use copies of the others. In other words, the resampling step produces new particles $\{x_{0:t}^{*(i)}\}_{i=1}^N$ and corresponding weights $\{\tilde{\omega}_t^{*(i)}\}_{i=1}^N$ based on $\{x_{0:t}^{(i)}\}_{i=1}^N$ and $\{\tilde{\omega}_t^{(i)}\}_{i=1}^N$. In which way, the new particles are produced, it depends on used resampling scheme, see [6] and [14] for overview. All resampling procedures discussed below use $\tilde{\omega}_t^{*(i)} = 1/N$, for ilustration example see figure 2.3. Due to $x_{0:t}^{(i)} = (x_{0:t-1}^{(i)}, x_t^{(i)})$, the resampling procedure is performed only with respect to $x_t^{(i)}$. Remaining part $x_{0:t-1}^{*(i)}$ is used accordingly to $x_t^{*(i)}$.

2.5.1 Multinomial resampling

Multinomial resampling, discussed in [25], uses $\{x_t^{*(i)}\}_{i=1}^N$ drawn from point mass distribution $\sum_{i=1}^N \tilde{\omega}_t^{(i)} \delta(x_t - x_t^{(i)})$ where δ is the Dirac delta function. Practical implementation of multinomial resampling uses samples drawn from uniform distribution $U((0;1])$ which determines $N^{(i)}$, the numbers of identical copies of the

original sample $x_{0:t}^{(i)}$. Multinomial resampling is summarized in scheme 2.2.

Algorithm 2.2 Multinomial resampling

for $i = 1$ to N **do**
  compute the cumulative weights $\widehat{\omega}_t^{(i)} = \sum_{j=1}^{i} \tilde{\omega}_t^{(j)}$
end for
for $i = 1$ to N **do**
  sample U^i from uniform distribution $U((0;1])$
end for
order $\{U^{(i)}\}_{i=1}^{N}$ in ascending order
for $i = 1$ to N **do**
  compute $N^{(i)}$ satisfies $\sum_{j=1}^{i} N^{(j)} = \max_{0 \geq l \geq N}\{l | \widehat{\omega}_t^{(i)} > U^{(i)}\}$
end for
for $i = 1$ to N **do**
  for $j = i$ to $i - 1 + N^{(i)}$ **do**
    state $x_{0:t}^{*(i+j)} = x_{0:t}^{(i)}$
  end for
end for

2.5.2 Residual resampling

In residual resampling [13], the number of identical copies for the original sample $x_{0:t}^{(i)}$ is set to $\tilde{N}^{(i)} = \lfloor N\tilde{\omega}_t^{(i)} \rfloor$ for each i. The rest $N - \sum_{j=1}^{N} \tilde{N}^{(j)}$ particles has to be computed using any other resampling scheme. For example by multinomial resampling for weights

$$\bar{\omega}_t^{(i)} = \frac{\widehat{\omega}^{(i)} N - \tilde{N}^{(i)}}{N - \sum_{j=1}^{N} \tilde{N}^{(j)}} \qquad i = 1, \ldots, N. \tag{2.12}$$

Another possible choice for second step of residual resampling is to use one additional copy for first $N - \sum_{j=1}^{N} \tilde{N}^{(j)}$ particles ordered according to $Ni - \tilde{N}^{(i)}$. By this choice, we obtain completely deterministic version of resampling procedure.

Residual resampling proceeds according to the scheme 2.3.

It can be shown (e.g. [6]) that the conditional variance of residual sampling is always smaller than that of multinomial sampling.

2.5.3 Systematic resampling

Systematic resampling [19] needs only one sample U drawn from uniform distribution $U((0;1/N])$. The numbers of copies $\{N^{(i)}\}_{i=1}^{N}$ are computed similarly to multinomial resampling schemes using equidistant values

$$U^{(i)} = U + \frac{l-1}{N} \qquad i = 1, 2, \ldots, N. \tag{2.13}$$

Algorithm 2.3 Residual resampling

for $i = 1$ to N **do**
  compute $\tilde{N}^{(i)} = \lfloor N\tilde{\omega}_t^{(i)} \rfloor$
end for
for $i = 1$ to N **do**
  set altered weights according to (2.12)
end for
for $i = 1$ to $\sum_{j=1}^{N} \tilde{N}^{(j)}$ **do**
  for $j = i$ to $i - 1 + \tilde{N}^{(i)}$ **do**
    state $x_{0:t}^{*(i+j)} = x_{0:t}^{(i)}$
  end for
end for
get rest $N - \sum_{j=1}^{N} \tilde{N}^{(j)}$ particles from multinomial resampling for $\{x_{0:t}^{(i)}, \bar{\omega}_t^{(i)}\}_{i=1}^{N}$.

Although, due to only one random sample needed, systematic resampling is less computationally expensive than previous methods, each resamplingd particles are (conditionally) dependent and they are sensitive on permutation of the original ones. Thus, studying of systematic resampling method is much harder than for other methods.

2.5.4 Regularized resampling

Using samples from point mass distribution $\sum_{i=1}^{N} \tilde{\omega}_t^{(i)} \delta(x_t - x_t^{(i)})$, it is possible that after resampling step, many particles will have no descendants. In extreme case, there will be only one type of particles obtained from a single one. Possible approach to overcome this impoverishment of diversity is to use regularized resampling [7]. In regularized resampling, instead of point mass distribution a continuous approximation of posterior distribution is used

$$\hat{p}(x_t|y_{1:t}) = \sum_{i=1}^{N} \tilde{\omega}_t^{(i)} K(\frac{x_t - x_t^{(i)}}{b}). \tag{2.14}$$

Here K is kernel density function and $b > 0$ is scalar parameter, called Kernel bandwidth. The Kernel density is symmetric function with zero mean and finite variance. The kernel K and the parameter b are optimally chosen as minimizers of mean square error between posterior density and its approximation (2.14) defined as

$$\mathrm{E}\left[\int \hat{p}(x_t|y_{1:t}) - p(x_t|y_{1:t})\right], \tag{2.15}$$

where E is expectation evaluated with respect to the samples. In particular case with equal normalized weights, the optimal kernel is the Epanechnikov kernel [10]

$$K_{opt} = \begin{cases} \frac{n_x+2}{2c_{n_x}}(1 - ||x||^2) & \text{if } ||x|| < 1, \\ 0 & \text{otherwise,} \end{cases} \tag{2.16}$$

where c_{n_x} is volume of unit sphere in $\mathbf{R}^{n_x}$. Additionally, if the distribution is Gaussian with unit covariance matrix, the corresponding bandwidth is

$$h_{opt} = \left[\frac{8(n_x + 4)(2\sqrt{\pi})^{n_x}}{c_{n_x}} \right]^{\frac{1}{n_x+4}}. \tag{2.17}$$

Due to easy computation and good performance in empirical simulations (e.g. [26]), this bandwidth is used even in non Gaussian cases. Consequently, regularized resampling can be performed by the algorithm summarized in scheme 2.4.

Algorithm 2.4 Regularized resampling

for $\{x_t^{(i)}, \tilde{\omega}_t^{(i)}\}_{i=1}^N$ calculate the empirical covariance S_t
perform decomposition $D_t D_t^T = S_t$
get $\{\tilde{x}_t^{*(i)}\}_{i=1}^N$ as a resample of $\{x_t^{(i)}\}_{i=1}^N$ using any resampling procedures
for $i = 1$ to N **do**
  draw $e^{(i)} \sim K_{opt}$
  regularize $x_t^{*(i)} = \tilde{x}_t^{*(i)} + b_{opt} D_t e^{(i)}$
end for

2.6 Sequential importance resampling

Sequential Importance Resampling (SIR) filter is obtained by usage of a resampling procedure in original SIS filter when degeneracy of the SIS filter is above some certain threshold. One of first particle filter of this type was so called bootstrap filter [25] which was based on using prior density as importance density and taking multinomial resampling after each step.

For estimating a level of the degeneracy (and as a criterion for usage of a resampling procedure) an effective sample size introduced in [20] is used

$$N_{eff} = \frac{N}{1 + \mathrm{Var}(\omega(x_{0:t}))}. \tag{2.18}$$

Exact evaluation of N_{eff} is impossible but an estimate $\widehat{N_{eff}}$ based on computed $\omega_t^{(i)}$ is given by

$$\widehat{N_{eff}} = \frac{1}{\sum_{i=1}^N (\tilde{\omega}_t^{(i)})^2}. \tag{2.19}$$

The resampling step is induced whenever $\widehat{N_{eff}}$ is bellow some fixed threshold N_{tresh}, see the scheme 2.5. The adantage of using the SIR filter in simulation from figure 2.2 is presented n the figure 2.3

Despite of overcoming the degeneracy phenomenon, after resampling, particles are no longer statistically independent. However in [4], the central limit theorem was stated at lest for scheme where resampling is used after each step. Also practical problems occur when resampling procedure is used because, in contrary to the SIS filter, SIR filter is not fully paralellizable since during resampling all particles are combined.

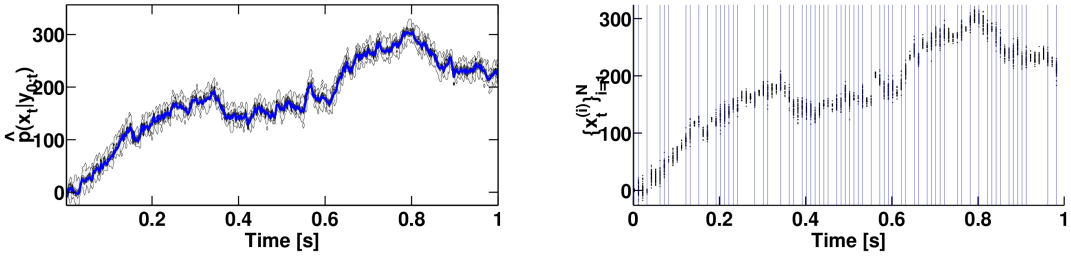

Figure 2.3: Simulation from the figure 2.2 where the SIS filter is replaced by the SIR filter. If the importance weights are distributed very unevenly (marked by blue lines in the right part), the resampling step is used for overcoming the degeneracy. Here, multinomial resampling with $N_{thresh} = N/5 = 12$ is used.

Algorithm 2.5 Sequential Importance Resampling

for $t = 1, 2, \ldots$ **do**
  update $\{x_{0:t-1}^{(i)}, \tilde{\omega}_{t-1}^{(i)}\}_{i=1}^{N}$ to $\{x_{0:t}^{(i)}, \tilde{\omega}_{t}^{(i)}\}_{i=1}^{N}$ using one step of the SIS filter
  compute estimate of effective sample size

$$\widehat{N_{eff}} = \frac{1}{\sum_{i=1}^{N} (\tilde{\omega}_t^{(i)})^2}.$$

  if $\widehat{N_{eff}} < N_{tresh}$ **then**
    update $\{x_{0:t}^{(i)}, \tilde{\omega}_{t}^{(i)}\}_{i=1}^{N}$ to $\{x_{0:t}^{*(i)}, \tilde{\omega}_{t}^{*(i)}\}_{i=1}^{N}$ using a resampling procedure
  end if
end for

2.7 Auxiliary sampling importance resampling

The goal of Auxiliary Sampling Importance Resampling (ASIR) filter, presented in [24], is to design a variant of the SIR filter which would be more robust against outliers. The main idea is to use a higher dimensional importance density $q(x_t, i|y_{1:t})$ from which are sampled pairs of $\{x_t^{(i)}, i^j\}_{i=1}^N$. Here, i^j denotes index of antecedent particle of $x_t^{(i)}$ at iteration $t-1$. Straightforward application of Bayes rule and corresponding definitions of i and $\omega_{t-1}^{(i)}$ gives

$$\begin{aligned} p(x_t, i|y_{0:t}) &= \frac{p(y_t|x_t, i, y_{0:t-1})p(x_t, i|y_{0:t-1})}{p(y_t|y_{0:t-1})} \\ &\propto p(y_t|x_t, y_{0:t-1})p(x_t, i|y_{0:t-1}) \\ &= p(y_t|x_t, y_{0:t-1})p(x_t|i, y_{0:t-1})p(i|y_{0:t-1}) \\ &= p(y_t|x_t)p(x_t|x_{t-1})\omega_{t-1}^{(i)} \end{aligned}$$

The importance density is defined to satisfy similar proportionality

$$q(x_t, i|y_{0:t}) \propto p(y_t|\mu_t^{(i)})p(x_t|x_{t-1})\omega_{t-1}^{(i)}, \tag{2.20}$$

where $\mu_t^{(i)}$ is some characterization of x_t given $x_{t-1}^{(i)}$. Suitable choice could be e.g. mean value or random sample from $p(x_t|x_{t-1}^{(i)})$. The importance density is also chosen to satisfy

$$q(x_t|i, y_{0:t}) = p(x_t|x_{t-1}^{(i)}), \tag{2.21}$$

and thus

$$q(x_t, i|y_{0:t}) = q(i|y_{0:t})q(x_t|i, y_{0:t}). \tag{2.22}$$

Combing together with (2.20), we obtain

$$q(i|y_{1:t}) \propto p(y_t|\mu_t^{(i)})\omega_{t-1}^{(i)}. \tag{2.23}$$

Using the previous, the weights are updated according to

$$\omega_t^{(j)} = \omega_{t-1}^{(i_j)}\frac{p(x_t^{(i)}, i_j|y_{0:t})}{q(x_t^{(i)}, i|y_{0:t})} \propto \omega_{t-1}^{(i_j)}\frac{p(y_t|x_t^{(i)})}{p(y_t|\mu_t^{(i_j)})}. \tag{2.24}$$

Algorithm of the ASIR filter is summarized in the scheme 2.6. Note, that it is not necessary to produce whole samples $\{x_t^{(i)}, i_j\}_{i=1}^N$.

Following previous scheme, it can be seen, that the ASIR filter is similar to the bootstrap filter [25]. Both algorithms uses prior density as importance density and resampling procedure during each step. Motivation for the ASIR filter was to improve performance of the SIR filter in cases with outliers. The reason, why the ASIR filter is more robust, is that the algorithm performs resampling step first and then sample only with respect to particles which are most likely to be close to the true state. Consequently, the weights after importance sampling step will be distributed more evenly. However, if the process noise is large, a single point does not characterize $p(x_t|x_{t-1}^{(i)})$ well, and the ASIR filter resamples based on apoor approximation of $p(x_t|x_{t-1}^{(i)})$. In such scenarios, the use of the ASIR filter then degrades performance.

Algorithm 2.6 Auxiliary Sampling Importance Resampling

for $t = 1, 2, \ldots$ **do**
  for $i = 1$ to N **do**
    calculate $\mu_t^{(i)}$
    set $\omega_t^{(i)} \propto p(y_t|\mu_t^{(i)})\omega_{t-1}^{(i)}$
  end for
  compute normalized importance weights $\{\tilde{\omega}_t^{(i)}\}_{i=1}^N$ (from $\{\omega_t^{(i)}\}_{i=1}^N$)
  determine the $\{i_j\}_{i=1}^N$ using a resampling scheme with $\{\tilde{\omega}_t^{(i)}\}_{i=1}^N$
  for $j = 1$ to N **do**
    sample $x_t^{(j)}$ from $q(x_t|i_j, y_{0:t}) = p(x_t|x_{t-1}^{(i_j)})$
    set $x_{0:t}^{(j)} = (x_{0:t-1}^{(i_j)}, x_t^{(j)})$
  end for
  for $j = 1$ to N **do**
    compute (second stage) importance weights using

$$\omega_t^{(j)} \propto \frac{p(y_t|x_t^{(j)})}{p(y_t|\mu_t^{(i_j)})}$$

  end for
  compute normalized importance weights $\{\tilde{\omega}_t^{(i)}\}_{i=1}^N$
end for

2.7.1 Illustrative example

For brief illustration of advantage of the ASIR filter, we consider a system described by

$$\begin{aligned} x_t &= 1.2x_{t-1} + w_{t-1} \\ y_t &= x_t + v_t \end{aligned} \qquad t = 1, \ldots, 20 \tag{2.25}$$

where $w_{t-1} \sim \mathcal{N}(0, \sigma_w^2)$, $v_t \sim \mathcal{N}(0, \sigma_v^2)$, $x_0 \sim \mathcal{N}(0, \sigma_0^2)$, $\sigma_w = 0.01$, $\sigma_v = 0.05$ and $\sigma_0 = 0.001$. In $t = 5$, the outlier is simulated by $w_4 = 0.5$. Due to the similarities depicted above, bootstrap filter is used for comparison. Multinomial resampling was used both in the ASIR filter and the bootstrap filter. Two possible realizations are shown in figure 2.4. After 1000 simulations, mean square error using the ASIR filter was lesser of 35% than with bootstrap filter.

More comprehensive study of the ASIR filter with illustrative examples was presented in [24].

2.8 Kalman filter based estimators

Classical approach to sequential parameter estimation is the well known Kalman filter [18]. The Kalman filter was derived as optimal filter in the case of linear system with Gaussian noise. However, various extensions for nonlinear cases have been proposed. For later comparison, we briefly present Kalman filter and its most commonly used extension, the so called Extended Kalman filter. Till nowadays, many

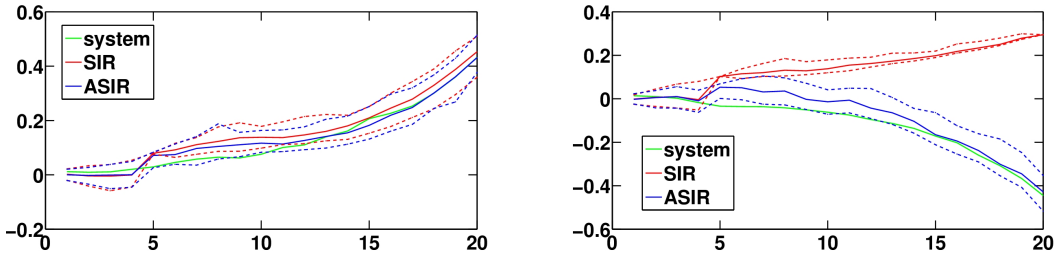

Figure 2.4: In the left part of the figure, typical realization of the scenario is presented. Dotted lines describe range of the particles. It can be seen, that the ASIR filter reduces error caused by the outlier in $t = 5$ slightly better. In extreme case, the estimation using the bootstrap filter is no longer possible, see the right part.

other extensions of the original Kalman filter has been proposed (e.g. Unscended KF [16] or Gausian Sum Filters [21]), but all of them are based on Gaussian densities, which can be limiting in particular applications.

2.8.1 Kalman filter

In 1960, the solution of optimal estimator of linear system with Gaussian noises was derived in [18] and was named after its autor as Kalman filter. The estimator is optimal in sense of mean square error, thus the estimate $\hat{x}$ proposed by Kalman filter minimizes

$$\mathrm{E}\{(x_t - \hat{x}_t)^2 | y_{0:t}\} \tag{2.26}$$

between all possible estimators of x.

Due to assumption of linearity, the system is described by

$$x_t = A_t x_{t-1} + B_t u_{t-1} + w_{t-1} \qquad t \geq 1, \tag{2.27}$$

$$y_t = H_t x_t + v_{t-1}, \tag{2.28}$$

where $w_t \sim \mathcal{N}(0, Q_t)$, $v_t \sim \mathcal{N}(0, R_t)$ and matrices A_t, B_t, Q_t and R_t are supposed to be known.

It was proven in [18] that estimate of x_t based on $y_{0:t}$ is distributed according to $\mathcal{N}(\hat{x}_{t|t}, P_{t|t})$ and can be computed sequentially as

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t(y_{t+1} - H_t \hat{x}_{t|t-1}), \tag{2.29}$$

where

$$\hat{x}_{t|t-1} = A_t \hat{x}_{t-1|t-1} + B_t u_{t-1}, \tag{2.30}$$

$$P_{t|t-1} = A_t P_{t-1|t-1} A_t^T + Q_t, \tag{2.31}$$

$$K_t = P_{t|t-1} H_t^T (H_t P_{t|t-1} H_t^T + R_t)^{-1}, \tag{2.32}$$

$$P_{t|t} = (I - K_t H_t) P_{t|t-1}. \tag{2.33}$$

Although very strict assumptions udner which the Kalman filter is the optimal estimator, it is still widely used in applications, e.g. [30].

2.8.2 Extended Kalman filter

If the system is nonlinear, Kalman filter can be still used on linearized system. This straightforward extension is called the Extended Kalman Filter (EKF). Suppose, that system is described as

$$x_f = f_t(x_{t-1}, u_{t-1}) + w_{t-1}, \tag{2.34}$$
$$y_t = h_t(x_t) + v_t, \tag{2.35}$$

where $w_t \sim \mathcal{N}(0, Q_t)$,$v_t \sim \mathcal{N}(0, R_t)$ and both functions f_t, h_t and matrices Q_t, R_t are supposed to be known.

The EKF is obtained simply from the original KF by replacing (2.29) and (2.30) by

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t(y_{t+1} - h_t(\hat{x}_{t|t-1})), \tag{2.36}$$
$$\hat{x}_{t|t-1} = f_t(\hat{x}_{t-1|t-1}, u_{t-1}) + w_{t-1} \tag{2.37}$$

where the rest of EKF uses equations from KF with

$$A_t = \frac{\partial f}{\partial x}\Big|_{x=\hat{x}_{t-1|t-1},u_{t-1}}, \tag{2.38}$$
$$H_t = \frac{\partial f}{\partial u}\Big|_{x=\hat{x}_{t|t-1}}, \tag{2.39}$$
$$\tag{2.40}$$

Extended Kalman filter is useful especially in cases of weak nonlinearities or if the linearizing point is near to the true state. In both mentioned cases, the linearization is sufficiently accurate and the EKF often estimates the true state well. In other cases, the convergence of EKF estimates to the true state is not guaranteed. Also, EKF suffers if the true posterior density is far from the Gaussian.

Chapter 3

Decision making

In this general formulation, the problem is practically unsolvable. Reasonable specification (with perspective to various applications) is to assume the loss function to be additive over time, thus

$$g(x_{1:N}, u_{0:N-1}) = \sum_{t=0}^{N-1} g_t(x_{t+1}, u_t). \tag{3.1}$$

for some known real functions g_t. Under the assumption of additivity, the expectation loss can be written as

$$J(x_0) = \mathrm{E}_{w_{0:N-1}} \left\{ \sum_{t=0}^{N-1} g_t(x_{t+1}, \mu_t(x_t)) \right\}. \tag{3.2}$$

As was pointed out in [11], the minimization of additive loss can be (theoretically) done by dynamic programming. Dynamic programming is based on optimality principle which states that the loss $J(x_0)$ on horizon N will be minimal if and only if all losses $J_k(x_k)$ on horizon $N-k$ will by minimal. Thus, the original problem can be rewritten as recursive problem

$$\begin{aligned} J_N(x_N) &= 0, \\ J_t(x_t) &= \min_{u_t \in U(x_t)} \mathrm{E}_{w_t} \left\{ g_k(x_{t+1}, u_t) + J_{t+1}(x_{t+1}) \right\}, \qquad t = 0, \ldots, N-1. \end{aligned} \tag{3.3}$$

Consequently, the minimization proceeds in backward direction for $k = N, \ldots, 0$ storing u_k for all possible x_k.

3.1 Optimal regulator

Optimal regulator proposes control policy (i.e. the sequence $\{u_{0:N-1}\}$) which minimizes the expectation loss (3.2). The optimal policy do not have to exist or to be unique, the sufficient condition is for example compactness of U and convexity of expectation loss J. However, analytical approach often suffers even for very simple system.

The exact solution is known for linear system with quadratic loss and Gaussian densities as so called Linear Quadratic Gaussian control (LQG), see [3]. The LQG control consist from Kalman filter (optimal Linear Quadratic Estimator, LQE) and Linear Quadratic Regulator (LQR).

In more general cases, both expectation and minimization can be performed only by some approximation technique.

3.2 Dual control

In [11], it was discussed that optimal control policy should not only to control the system to the desired state, it should also have some probing ability which provide better system identification and as consequence allow more accurate control actions in future steps. These two requirements on optimal policy are often in contradiction and this is what gives the name dual control. This principle could be very helpful in suboptimal control policy design – for example, if we have some control policy mu_t which does not satisfies the duality principle, we can incorporate the probing term by defining new policy as

$$\tilde{\mu}_t = \mu_t + \mu_t^{\text{prob}}, \tag{3.4}$$

where μ_t^{prob} is the probing term. In some cases, reasonable choice for probing term can be scaled white noise.

3.3 PID regulator

A PID regulator in its standard form proposes control actions composed from proportional, integral and derivative terms which gives the abbreviation PID. This regulator is well known since early 20th century and is the most widely used controller in process control until today [1]. Reason for its wide usage is simplicity and good performance in various applications.

The PID regulator produces control actions equal to

$$u(t) = P\left(e(t) + \frac{1}{I}\int_0^t e(\tau)\mathrm{d}\tau + D\frac{\mathrm{d}e(t)}{\mathrm{d}t}\right), \tag{3.5}$$

where P is the proportional gain, I the integral time constant, D the derivative time constant and $e(t)$ is error between measured process variable and its desired value. Functionality of respective terms can be described as follows

- The proportional term – providing an overall control action proportional to the error signal through the all-pass gain factor.
- The integral term – reducing steady-state errors through low-frequency compensation by an integrator.

- The derivative term – improving transient response through high-frequency compensation by a differentiator.

Thus, through the integral and derivative term, the PID regulator can be understood as a controller that takes also the past, and the future error into consideration.

For optimum performance of the regulator, parameters P, I, and D have to be set properly. Nonetheless, it is generally impossible to outline optimal values for parameters theoretically. Due to this fact, they are obviously tuned manually or by some adaptive method, see e.g. [1]. More extensive introduction into PID regulator problematics can be found in [17].

It should be pointed out that the relation between the PID regulator and the original problem is only through the parameters P, I, and D. As a consequence, it is hard to say how to change the parameters of the regulator when the parameters of the system are changed. This hidden relation makes the detailed study of the regulator practically impossible.

From (3.5), it is clear that the PID regulator does not provide dual control.

3.4 Cautious control and Certainty equivalence principle

Cautious control and Certainty equivalence principle are commonly used approaches for simplification of the original problem (3.3), see [3].

Cautious Control (CC) is obtained by restriction of the original optimization problem to horizon of length $N = 1$. The name origins from fact that the optimization of control action does not incorporate the advantage of probing. The simplification by certainty equivalence principle replaces all the random variables in (3.3) by their mean values, this gives Certainty Equivalence Control (CEC). Of course, both approaches can be combined.

Both approximation techniques provide control policy which are not dual, however these techniques are often the only ones which are able to propose control policy based on original problem and which allows online computations.

3.5 Methods based on stochastic approximations

3.5.1 Stochastic iterative approximations of dynamic programming (SIDP)

Method of Stochastic iterative Approximations of Dynamic programming (SIDP) was proposed in [28]. It is based on two main principles:

- solving the dynamic programming (3.3) in several iterations rather than in only one step,
- using Monte Carlo approximation of expectations in (3.3).

The first approach is so called Iterative Dynamic Programming (IDP). In [28], it was shown that under relatively general assumptions, control policy iterations provided by the SIDP algorithm converges to the optimal policy regardless of the initial policy. Moreover using IDP, it is sufficient to use less points in discretization of the space because only the part of the space which could be reached in current iteration has to be discretized.

The algorithm proceeds offline and provides the control policy in a form of control actions for every discretized point and every time step. Then, controlling is performed using these prepared control actions. In the original article, the control actions outside the discretized points are linearly interpolated.

However, SIDP algorithm has exponential computational complexity in horizon length and thus can be applied on systems with long transient response only with huge computational effort, see [28] for evidence. Moreover, if the system noise is relatively large, convergence of the algorithm is very slow or even unstable.

3.5.2 Stochastic approximations of policy gradient

Stochastic approximations of policy gradient is method presented in [27], here is also the proof of optimality of the algorithm. Principally, the computation of optimal control action uses stochastic approximations of the gradient of (3.2) with respect to the control actions $u_{0:N-1}$ and then the gradient descent algorithm is utilized for finding the optimum.

In the original article [27], Open-Loop Feedback Control (OLFC) approach is used. The approach lies in optimizing the (3.2) at horizon $t = k, \ldots, N$ during every time step k, see [3]. The disadvantage of OLFC approach is that it is inapplicable on real-time applications because the computation of single step of the algorithm is relatively time consuming.

Chapter 4

Simulations

In this chapter, the application of previously presented estimating and controlling techniques on real model is presented.

4.1 Model of permanent magnet synchronous machine drive

We will be interested in model of Permanent Magnet Synchronous Machine drive with surface magnets on the rotor (abbreviated as PMSM). Following description is adopted from [23].

4.1.1 Time continuous model

The model is described by conventional equations in stationary reference frame

$$\frac{\mathrm{d}i_\alpha}{\mathrm{d}t} = -\frac{R}{L}i_\alpha + \frac{\Psi}{L}\omega\sin\theta + \frac{u_\alpha}{L} \tag{4.1}$$

$$\frac{\mathrm{d}i_\alpha}{\mathrm{d}t} = -\frac{R}{L}i_\beta - \frac{\Psi}{L}\omega\cos\theta + \frac{u_\beta}{L} \tag{4.2}$$

$$\frac{\mathrm{d}\omega}{\mathrm{d}t} = \frac{k_p p_p^2 \Psi}{J}(i_\beta\cos\theta - i_\alpha\sin\theta) - \frac{B}{J}\omega - \frac{p_p}{J}T \tag{4.3}$$

$$\frac{\mathrm{d}\theta}{\mathrm{d}t} = \omega. \tag{4.4}$$

Here, i_α, i_β, u_α and u_β represent stator current and voltage in the stationary reference frame, respectively; ω is electrical rotor speed and θ is electrical rotor position. R and L is stator resistance and inductance respectively, Ψ is the flux of permanent magnets on the rotor, B is friction and T is load torque, J is moment of inertia, p_p is the number of pole pairs, k_p is the Park constant. Voltage constraint is

$$\sqrt{u_\alpha^2 + u_\beta^2} \leq 100, \tag{4.5}$$

original term	substitution	value in simulations
$1 - \frac{R}{L}\Delta t$	a	0.9898
$\frac{\Psi}{L}\Delta t$	b	0.0072
$\frac{\Delta t}{L}$	c	0.0361
$1 - \frac{B}{J}\Delta t$	d	1
$\frac{k_p p_p^2 \Psi}{J}\Delta t$	e	0.0149

Table 4.1: Parameters of the PMSM.

where v

The goal is to design control policy for voltages which will result into desired rotor speed $\bar{\omega}$. The loss function is quadratic

$$\int_0^T v(u_\alpha^2(t) + u_\beta^2(t)) + (\omega(t) - \bar{\omega}(t))^2 \mathrm{d}t, \tag{4.6}$$

where v is known constant. During simulations, we use $v = 0.1$.

4.1.2 Discretized model

Discretization of the model was performed using Euler method with the following result:

$$i_{\alpha,t+1} = (1 - \frac{R}{L}\Delta t) i_{\alpha,t} + \frac{\Psi}{L}\Delta t \omega_t \sin\theta_t + \frac{\Delta t}{L} u_{\alpha,t} \tag{4.7}$$

$$i_{\beta,t+1} = (1 - \frac{R}{L}\Delta t) i_{\beta,t} - \frac{\Psi}{L}\Delta t \omega_t \cos\theta_t + \frac{\Delta t}{L} u_{\beta,t} \tag{4.8}$$

$$\omega_{t+1} = (1 - \frac{B}{J}\Delta t)\omega_t + \Delta t \frac{k_p p_p^2 \Psi}{J}(i_{\beta,t}\cos\theta_t - i_{\alpha,t}\sin\theta, t) - \frac{p_p}{J} T \Delta t \tag{4.9}$$

$$\theta_{t+1} = \theta_t + \omega_t \Delta t. \tag{4.10}$$

For simplifying the notation, we make a substitution summarized in table 4.1 (we consider parameters of the model known). It results in

$$i_{\alpha,t+1} = a i_{\alpha,t} + b\omega_t \sin\theta_t + c u_{\alpha,t} \tag{4.11}$$

$$i_{\beta,t+1} = a i_{\beta,t} - b\omega_t \cos\theta_t + c u_{\beta,t} \tag{4.12}$$

$$\omega_{t+1} = d\omega_t + e(i_{\beta,t}\cos\theta_t - i_{\alpha,t}\sin\theta_t) \tag{4.13}$$

$$\theta_{t+1} = \theta_t + \omega_t \Delta t. \tag{4.14}$$

The state variables and the voltages can be aggregated into $x_t = (i_{\alpha,t}, i_{\beta,t}, \omega_t, \theta_t)^T$ and $u_t = (u_{\alpha,t}, u_{\beta,t})^T$. Constraint on u_t is then

$$||u_t|| \leq 100. \tag{4.15}$$

Discretized loss function is of the form

$$\sum_0^N u_t^T \Gamma u_t + (x_t - \bar{x}_t)^T \Xi (x_t - \bar{x}_t), \tag{4.16}$$

where we denoted

$$\Gamma = \begin{pmatrix} v & 0 \\ 0 & v \end{pmatrix}, \qquad \Xi = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{4.17}$$

The sensor-less control scenario arise when sensors of the speed and position are missing (from various reasons). The only observable variables are thus $(i_{\alpha,t}, i_{\beta,t})$, however, only up to some precision. In relation to the chapter 1, the system is modeled as Markovian state-space model given by

$$x_t = g(x_{t-1}, u_{t-1}) + w_{t-1} \qquad t \geq 0, \tag{4.18}$$

$$y_t = Hx_t + v_t, \tag{4.19}$$

where $H = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$, $w_{t-1} \sim \mathcal{N}(0, Q)$ and $v_t \sim \mathcal{N}(0, R)$. In later simulations, we use

$$Q = \mathrm{diag}(0.0013, 0.0013, 5 \times 10^{-6}, 10^{-10}), \tag{4.20}$$

$$R = \mathrm{diag}(0.0006, 0.0006). \tag{4.21}$$

Values for matrices Q and R and parameters in table 4.1 are adopted from [23], where real prototype of the PMSM was analyzed.

4.2 Application of presented estimating techniques on the PMSM

In this section, estimating techniques described in chapter 2 are applied on the PMSM.

4.2.1 Particle filters

As was mentioned in section 2.4, the importance density which minimizes the variance of the importance weight is $p(x_t|x_{t-1}, y_t)$ and particular case of Gaussian state space model with non-linear system equation allows analytic solution. Following results are adopted from [7]. Defining

$$S^{-1} = Q^{-1} + H^T R H, \tag{4.22}$$

$$m_t = S(Q^{-1} g(x_{t-1}) H^T R y_t), \tag{4.23}$$

one can obtain

$$x_t|(x_{t-1}, y_t) \sim \mathcal{N}(m_t, S), \tag{4.24}$$

and

$$p(y_t|x_{t-1}) \propto exp\left(-\frac{1}{2}(y_t - Hg(x_{t-1}))^T (R + HQH^T)^{-1}(y_t - Hg(x_{t-1}))\right). \tag{4.25}$$

We denote SIR_{opt} the version of SIR filter with optimal importance density. For later comparison, we also denote $\text{SIR}_{\text{prior}}$ the version of SIR filter in which the prior density as importance density is used.

Although SIR filters can be theoretically used for arbitrary large state space, estimation in higher dimension states is less accurate because of need of larger amount of particles for sufficient coverage of the space. Because of this, we use estimation by SIR filters only for unobservable (ω_t, θ); as estimator of $(i_{\alpha,t}, i_{\beta,t})$, we use only observed values. It incorporates errors caused by noise in observations, however, nearly the same level of accuracy can be achieved by usage of significantly less particles (about 25x lesser).

We also tested the ASIR filter, nonetheless, similar state space reduction as before is not functioning well. It is due to the fact, that usage of only single point for approximation of currents increases noises and as was mentioned in [2] in the case of too noisy observations the performance of the ASIR filter is unreliable. Consequently, inside the ASIR filer, the whole state x_t is approximated by particles.

Using both $\text{SIR}_{\text{prior}}$ and ASIR filter, performance using true matrix R was insufficient; both filters was too restrictive during weight updating step and often converge to a wrong state. Performance was improved by using covariance matrix $\tilde{R} = \rho R$ instead of R, where $\rho > 1$ is a real parameter. Amplification of ρ increases robustness of respective particle filter but in case of convergence it also increases average error of estimation. Proper ρ should be chosen after some initial simulations. Based on several quantitative test, we will use $\rho = 100$ in case of $\text{SIR}_{\text{prior}}$ and even $\rho = 1000$ for the ASIR filter. Moreover, we found that the failure of estimation also occurs in case where SIR_{opt} is used but the number of failures were much more lesser than in cases of the $\text{SIR}_{\text{prior}}$ and ASIR filter. Nonetheless, we will incorporate the same approach to add robustness by using $\rho = 10$.

Moreover, we found that number of particles can be lowered (approximately 5x) without loss of accuracy if larger variance on θ is used. Thus in all particle filters, we use $\tilde{\sigma}_\theta^2 = 10^{-4}$ instead of true $\sigma_\theta^2 = 10^{-10}$.

Using small number of particles, it may happen that estimation process is corrupted when particles approximating initial state are generated very unevenly. For overcoming this problem, particles approximating initial distribution are chosen equidistantly in interval $[-\pi, \pi]$ instead of random samples

Performance using different resampling procedures was nearly the same in our simulations, thus, we use only deterministic resampling for its computational effectiveness and fully deterministic version of residual resampling in case when we would like to omit effects caused by random realization inside particle filter.

In later simulations, we want to be able to estimate θ wherever can be in interval $(-\pi, \pi)$. As consequence, if we have no additional initial information, we have to have whole interval $(-\pi, \pi)$ discretized in initial distribution on θ. This may cause a problem when the true state is close to one of the boundary values – large weight will obtain both particles with θ close to $-\pi$ and π because of periodicity of trigonometric functions. It may degrade estimations based on this approximation,

in extreme case for example $\cos\mathrm{E}\{\theta\} \approx -1$, although $\mathrm{E}\{\cos\theta\} \approx 1$. This particular problem makes CEC regulator unable to reaching desired state when initial state is close to π (or $-\pi$). This can be overcame by utilizing not the $\cos\mathrm{E}\{\theta\}$ but $\mathrm{E}\{\cos\theta\}$. Another (and also more general) possibility how to overcome the problem is to make approximation on θ continuous in π when it is useful. It can done by transformation

$$\Phi(\theta) = \begin{cases} \theta & \text{if } \theta > 0, \\ \theta + 2\pi & \text{otherwise.} \end{cases} \tag{4.26}$$

The transformation overcomes the discontinuity in π. As suitable indicator for introducing the transformation, we utilize standard deviation of estimates based on distributions before and after transformation (denoted σ_1 and σ_2). If $\sigma_1 \geq 1.5\sigma_2$, we utilize the transformation. Practical usage of the transformation is shown in figure **??**

4.2.2 EKF

By linearization of the model in $\hat{x}$, and using notation from section 2.8, we obtain (note, that equation for observation is already linear)

$$A_t = \begin{pmatrix} a & 0 & b\,\sin\hat{\theta}_t & b\hat{\omega}_t\,\cos\hat{\theta}_t \\ 0 & a & -b\,\cos\hat{\theta}_t & b\hat{\omega}_t\,\sin\hat{\theta}_t \\ -e\,\sin\hat{\theta}_t & e\,\cos\hat{\theta}_t & d & -e(\hat{i}_{\beta,t}\,\sin\hat{\theta}_t + \hat{i}_{\alpha,t}\,\cos\hat{\theta}_t) \\ 0 & 0 & \Delta t & 1 \end{pmatrix}. \tag{4.27}$$

Having initial distribution $x_0 \sim \mathcal{N}(\hat{x}_{0|0}, P_{0|0})$, computation of EKF can proceed directly using equations from section 2.8.

As will be shown in subsection 4.4.2, the EKF is unable to filter proposed density properly in cases where only weak prior information is available. However, as is generally known (e.g [5]), performance of the EKF can be improved by using some matrices $\tilde{Q}$ and $\tilde{R}$ during estimation step instead of original matrices Q and R. Estimation with matrices containing greater diagonal elements is more sensitive on differences between predicted and observed values and thus estimation is less conservative, although proposed estimates are still only Gaussian distributions. These properties was observed also in our simulations, however, no settings of the matrices resulted into sufficient reliability. This issue is shortly discussed in respective subsection.

4.3 Application of presented controling techniques on the PMSM

In this section, implementation of PI, CC and CEC regulator is presented. Control policy proposed by SIDP (or extensions of previous controlling techniques enhanced by SIDP) does not work well, so we omit detail of its implementation on the PMSM.

Algorithm of stochastic policy gradient was not implemented because, due to the computational complexity, it is improper for application on the PMSM.

4.3.1 PI regulator

Historically, first PID regulators consisted from only of two components – the proportional one and the integral one. Derivative term was added to stabilize system against overshoot produced by the integral component. However, the derivative term slows the transient response and causes instability of PID regulator if noises are sufficiently large, see [1]. Due to this fact and the fact that the transient response of the PMSM is also very large (affected by small Δt), we omit the derivative term in implementation of the PID regulator on the PMSM.

The classical PI regulator control is based on transformation to $d-q$ reference frame (for detailed derivation see [29])

$$i_d = i_\alpha \cos(\theta) + i_\beta \sin(\theta), \tag{4.28}$$
$$i_q = i_\beta \cos(\theta) - i_\alpha \sin(\theta). \tag{4.29}$$

For desired ω, firstly, we compute target i_q current, denoted $\bar{i}_q$. It is derived using the PI regulator

$$\bar{i}_q = \mathrm{PI}(\bar{\omega} - \omega, P_i, I_i), \tag{4.30}$$

where arbitrary PI controller is defined as follows

$$x = \mathrm{PI}(\epsilon, P, I) = P\epsilon + I(S_{t-1} + \epsilon), \tag{4.31}$$
$$S_t = S_{t-1} + \epsilon. \tag{4.32}$$

This current needs to be achieved through voltages u_d, u_q which are again obtained from PI regulators

$$u_d = \mathrm{PI}(-i_d, P_u, I_u), \tag{4.33}$$
$$u_q = \mathrm{PI}(\bar{i}_q - i_q, P_u, I_u). \tag{4.34}$$

Because of magnetic field caused by rotor motion, voltages are compensated by

$$u_d = u_d - L_s \omega \bar{i}_q, \tag{4.35}$$
$$u_q = u_q + \Psi_{pm} \omega. \tag{4.36}$$

Conversion to u_α, u_β is done by

$$u_\alpha = |U| \cos\phi, \tag{4.37}$$
$$u_\beta = |U| \sin\phi, \tag{4.38}$$

where

$$|U| = \sqrt{u_d^2 + u_q^2} \qquad \phi = \begin{cases} \arctan\frac{u_d}{u_q} + \theta & \text{if } u_d \geq 0, \\ \arctan\frac{u_d}{u_q} + \theta + \pi & \text{if } u_d < 0. \end{cases} \tag{4.39}$$

In order to satisfy the constraint (4.15), if $U > 10$, we firstly set $U := 10$.

Based on several initial simulations, constants for PI regulators were set on

$$P_i = 3, \qquad I_i = 0.00375, \tag{4.40}$$

$$P_u = 20, \qquad I_u = 0.5. \tag{4.41}$$

It should be noted that the PI regulator is based on the value of the true state . Having only an estimate of the state, we are tended to use some characteristic of probability distribution of the state, for example mean value. Another possibility could be maximum value or random sample from the distribution, however, these are more sensitive on accuracy of the estimate and, in a case of large uncertainty, they often propose unstable control actions. Due to this we will aim only on the PI regulator computed for mean value.

4.3.2 Cautious control

Because arbitrary input of u_t can cause changes firstly in ω_{t+2}, we derive cautious control by minimization of

$$u_t^T \Gamma u_t + \mathrm{E}\big((x_{t+1} - \bar{x}_{t+1})^T \Xi (x_{t+1} - \bar{x}_{t+1}) + (x_{t+2} - \bar{x}_{t+2})^T \Xi (x_{t+2} - \bar{x}_{t+2})\big). \tag{4.42}$$

We note that the first term in expectation can be omitted due to the fact that its derivative with respect to u_t is zero, however, we keep it for better readability.

We outline only unconstrained minimization of (4.42). If obtained control action u_t does not satisfy the constraint (4.15), we define

$$u_t := 10 \frac{u_t}{||u_t||}. \tag{4.43}$$

Unconstrained minimization can be performed by setting first derivative of (4.42) with respect to u_t to zero, thus using the symmetry of Γ and Ξ

$$u_t^T \Gamma + \mathrm{E}\left((x_{t+1} - \bar{x}_{t+1})^T \Xi \frac{\partial x_{t+1}}{\partial u_t} + (x_{t+2} - \bar{x}_{t+2})^T \Xi \frac{\partial x_{t+2}}{\partial u_t}\right) = 0. \tag{4.44}$$

Due to that the control action at $t+1$ does not affect any change in ω, x_{t+2} can be expressed as

$$x_{t+1} = A_t x_t + C u_t + w_t \tag{4.45}$$

$$x_{t+2} = A_{t+1} x_{t+1} + w_{t+1} = A_{t+1} A_t x_t + A_{t+1} C u_t + A_{t+1} w_t + w_{t+1}, \tag{4.46}$$

where A_t and C are

$$A_t = A(x_t) = \begin{pmatrix} a & 0 & b\sin\theta_t & 0 \\ 0 & a & -b\cos\theta_t & 0 \\ -e\sin\theta_t & e\cos\theta_t & d & 0 \\ 0 & 0 & \Delta t & 1 \end{pmatrix}, \qquad C = \begin{pmatrix} c & 0 \\ 0 & c \\ 0 & 0 \\ 0 & 0 \end{pmatrix}. \tag{4.47}$$

It should be mentioned, that although system equation (4.46) is expressed only using linear operations, it does not mean that the system is linear because matrix A_t is nonlinear function of x_t.

After substitution in (4.42) and small simplification, we obtain

$$\begin{aligned} u_t^T\Gamma + \mathrm{E}\big((A_t x_t - \bar{x}_{t+1})^T\Xi + (A_{t+1}A_t x_t - \bar{x}_{t+2})^T\Xi A_{t+1}\big)C + \\ + u_t^T C^T \mathrm{E}\left(\Xi + A_{t+1}^T\Xi A_{t+1}\right)C = 0. \end{aligned} \tag{4.48}$$

Denoting

$$\Lambda_2 = \mathrm{E}\big((A_t x_t - \bar{x}_{t+1})^T\Xi + (A_{t+1}A_t x_t - \bar{x}_{t+2})^T\Xi A_{t+1}\big)C, \tag{4.49}$$

$$\Sigma_2 = C^T \mathrm{E}\left(\Xi + A_{t+1}^T\Xi A_{t+1}\right)C, \tag{4.50}$$

we have CC action u_t in the form

$$u_t^T = \Lambda_2(\Gamma + \Sigma_2)^{-1}. \tag{4.51}$$

Nonetheless, proposed control policy will not work properly. It is due to the fact, that matrix Λ_2 has elements multiplied by $ce \approx 5.10^{-4}$ meanwhile dominant elements of matrix in denominator are close to $v = 0.1$. Then, proposed control action u_t will be close to zero as can be viewed clearly from following example. Suppose $i_{\alpha,t}, i_\beta, \omega_t$ known and equal to zero and $\bar{\omega}_{t+2} = 10\mathrm{rad.s}^{-1}$. Under this assumption, the norm of proposed control action is bounded by

$$||u_t||^2 = u_t^T u_t \leq \left(\frac{10ce}{v}\right)^2 \mathrm{E}\left\{\begin{pmatrix} \sin\theta_t \\ \cos\theta_t \end{pmatrix}^T\right\} \mathrm{E}\left\{\begin{pmatrix} \sin\theta_t \\ \cos\theta_t \end{pmatrix}\right\} < 5.10^{-4}\ . \tag{4.52}$$

There are several possibilities how to overcome this. The most straightforward possibility is to artificially decrease v when u_t is computed. Based on experiments which are not included, decrease of v under 10^{-4} yields into control policy which is able to reach desired $\bar{\omega}_t$. However, this solution of the problem is improper because control policy does not take into consideration future effects of u_t. This policy often overshoot the desired value of velocity and results into damped oscillations.

More natural way for overcoming the problem of too small control actions obtained from (4.51) is to incorporate the effect of u_t at longer horizon. For arbitrary $k \in \mathbb{N}$, we can extend the loss (4.42) by future loss and write

$$J = u^T\Gamma u_t + \mathrm{E}\left(\sum_{k=1}^{n}(x_{t+k} - \bar{x}_{t+k})^T\Xi(x_{t+k} - \bar{x}_{t+k})\right). \tag{4.53}$$

Similarly as before, future states can be evaluated as

$$x_{t+1} = A_t x_t + C u_t + w_t \tag{4.54}$$

$$\begin{aligned} x_{t+k} &= A_{t+k-1}x_{t+k-1} + w_{t+k-1} = \\ &= A_{t+k-1}A_{t+k-2}x_{t+k-2} + A_{t+k-1}w_{t+k-2} + w_{t+k-1} = \ldots = \\ &= \left(\prod_{l=0}^{k-1} A_{t+l}\right)x_t + \left(\prod_{l=1}^{k-1} A_{t+l}\right)C u_t + \sum_{l=0}^{k-1}\left(\left(\prod_{m=l+1}^{k-1} A_{t+m}\right)w_{t+l}\right), \end{aligned} \tag{4.55}$$

where the products have to be understood by matrix multiplication in correct order. After simplification, condition for unconstrained minimization have the form

$$u_t^T\Gamma + \mathrm{E}\left\{\sum_{k=1}^{n}\left(\left(\left(\prod_{l=0}^{k-1}A_{t+l}\right)x_t - \bar{x}_{t+k}\right)^T \Xi\left(\prod_{l=1}^{k-1}A_{t+l}\right)\right)\right\}C + \\ + u_t^T C^T \mathrm{E}\left\{\sum_{k=1}^{n}\left(\left(\prod_{l=1}^{k-1}A_{t+l}\right)^T \Xi\left(\prod_{l=1}^{k-1}A_{t+l}\right)\right)\right\}C = 0. \tag{4.56}$$

Thus, we can use previous notation denoting

$$\Lambda_n = \mathrm{E}\left\{\sum_{k=1}^{n}\left(\left(\left(\prod_{l=0}^{k-1}A_{t+l}\right)x_t - \bar{x}_{t+k}\right)^T \Xi\left(\prod_{l=1}^{k-1}A_{t+l}\right)\right)\right\}C, \tag{4.57}$$

$$\Sigma_n = C^T\mathrm{E}\left\{\sum_{k=1}^{n}\left(\left(\prod_{l=1}^{k-1}A_{t+l}\right)^T \Xi\left(\prod_{l=1}^{k-1}A_{t+l}\right)\right)\right\}C, \tag{4.58}$$

and to obtain CC action as

$$u_t^T = \Lambda_n(\Gamma + \Sigma_n)^{-1}. \tag{4.59}$$

Computation of Λ_n and Σ_n is relatively time consuming and for n for which the proposed control law could be sufficient, the online computation is impossible. Nonetheless, the computation can be greatly speed up by following approximation. Let n is sufficiently small that $\{\theta_{t+k}\}_{k=0}^{n}$ is constant sequence up to some desired level and effects of magnetic induction in equations for the currents can be neglected. In other words, we use the approximation

$$A_t \approx A_{t+l} \approx \tilde{A} = \begin{pmatrix} a & 0 & 0 & 0 \\ 0 & a & 0 & 0 \\ -e\sin\theta_t & e\cos\theta_t & d & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad l = 0, \ldots, n \tag{4.60}$$

Due to this approxiamtion, we can express

$$\prod_{l=0}^{k-1}A_{t+l} \approx \tilde{A}^k = \begin{pmatrix} a^k & 0 & 0 & 0 \\ 0 & a^k & 0 & 0 \\ -eS_k\sin\theta_t & eS_k\cos\theta_t & d^k & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{4.61}$$

where

$$S_k = \sum_{l=0}^{k-1}a^l d^{k-1-l} = d^{k-1}\sum_{l=0}^{k-1}\left(\frac{a}{d}\right)^l = \frac{d^k - a^k}{d-a}. \tag{4.62}$$

After simplification, we obtain Λ_n and Σ_n approximated as

$$\Lambda_n \approx \gamma_n \mathrm{E}\left\{\begin{pmatrix} i_{\alpha,t} \\ i_{\beta,t} \end{pmatrix}^T \begin{pmatrix} \sin^2\theta_t & -\sin\theta_t\cos\theta_t \\ -\sin\theta_t\cos\theta_t & \cos^2\theta_t \end{pmatrix}\right\} + \\ + \sum_{k=1}^{n}\left(ceS_{k-1}\left(d^k \mathrm{E}\left\{\omega_t \begin{pmatrix} \sin\theta_t \\ \cos\theta_t \end{pmatrix}^T\right\} - \bar{\omega}_{t+k}\mathrm{E}\left\{\begin{pmatrix} \sin\theta_t \\ \cos\theta_t \end{pmatrix}^T\right\}\right)\right) \tag{4.63}$$

$$\Sigma_n \approx \delta_n \mathrm{E}\left\{\begin{pmatrix} \sin^2\theta_t & -\sin\theta_t\cos\theta_t \\ -\sin\theta_t\cos\theta_t & \cos^2\theta_t \end{pmatrix}\right\}, \tag{4.64}$$

where for $d \neq 1$ (for $d = 1$ is the computation even simpler)

$$\gamma_n = ce^2 \sum_{k=1}^{n} S_k S_{k-1} = c\left(\frac{e}{d-a}\right)^2 \sum_{k=0}^{n-1}\left(d(d^2)^k - (a+d)(ad)^k + (a^2)^k\right) = \\ = c\left(\frac{e}{d-a}\right)^2 \left(d\frac{1-d^{2n}}{1-d^2} - (a+d)\frac{1-(ad)^n}{1-ad} + a\frac{1-a^{2n}}{1-a^2}\right), \tag{4.65}$$

$$\delta_n = c^2e^2 \sum_{k=1}^{n} S_{k-1}^2 = \left(\frac{ce}{d-a}\right)^2 \sum_{k=0}^{n-1}((d^2)^k - 2(ad)^k + (a^2)^k) = \\ = \left(\frac{ce}{d-a}\right)^2 \left(\frac{1-d^{2n}}{1-d^2} - 2\frac{1-(ad)^n}{1-ad} + \frac{1-a^{2n}}{1-a^2}\right). \tag{4.66}$$

For comparison with the case without incorporating the future loss, we consider similar example as before in (4.52). For $\bar{\omega}_{t+k} = 10\mathrm{rad.s}^{-1}$ for $k = 1, \ldots, n$, we obtain an estimate

$$||u_t||^2 = u_t^T u_t \approx \left(\frac{10ce}{v}\sum_{k=1}^{n} S_{k-1}\right)^2 \mathrm{E}\left\{\begin{pmatrix} \sin\theta_t \\ \cos\theta_t \end{pmatrix}^T\right\} \mathrm{E}\left\{\begin{pmatrix} \sin\theta_t \\ \cos\theta_t \end{pmatrix}\right\}. \tag{4.67}$$

This expression gives an intuitive recipe for appropriate n. Successful policy is obtained by choice $n \geq 50$. In later simulations we use $n = 80$, what corresponds to incorporating the loss generated in horizon $t = 0.01\mathrm{s}$. However, the quality of control is only a little sensitive and control actions are nearly the same for large range of n.

Approximation of (4.59) by (4.63) and (4.64) can be computed very efficiently. Expectations can be approximated by estimates proposed by particle filter. Furthermore, constants γ_n and δ_n can be computed offline.

More crucial problem with control policy generated according to (4.59) is its caution. To ilustrate this, suppose, that $i_{\alpha,t}, i_\beta$ is known and equal to zero, ω_t is independent on θ and $\theta \sim \mathrm{U}(-\pi, \pi)$. This assumption holds for example for $t = 0$ if we have no additional initial information.From (4.63) or (4.67), we can see that the norm of proposed control action will be zero regardless on difference between actual and desired state.

4.3.3 Certainty equivalence control

The problem from the end of subsection 4.3.2 can be overcome by utilizing principle of Certainty Equivalence Control (CEC). According to that, we replace all the random variables in (4.63) and (4.64) by their mean values.

4.3.4 Dual control

Another possibility how to handle the excesive caution of CC regulator is to add some probing term according to (3.4). At first, we denote

$$\alpha(x_t) = \left|\left| \mathrm{E}\left\{ \begin{pmatrix} \sin\theta_t \\ \cos\theta_t \end{pmatrix} \right\} \right|\right|_2 . \tag{4.68}$$

Because $0 \leq \alpha(x_t) \geq 1$, where the lower bound is reached for uniform distribution and the upper bound for Dirac distribution, $\alpha(x_t)$ is proper measure for introducing some probing term. As consequence, we modify original CC regulator as follows

$$u_t = \alpha(x_t)\mathrm{CC}(x_t) + (1 - \alpha(x_t))u_t^{\mathrm{prob}}, \tag{4.69}$$

where $\mathrm{CC}(x_t)$ is control action computed by (4.59) and u_t^{prob} is some probing term.

Based on our experience with the system, reasonable choice of probing term is generator of constant voltage whose phase is changing monotonically

$$u_t^{\mathrm{prob}} = U_0 \begin{pmatrix} \sin\left(\frac{2\pi}{T}t + \phi_0\right) \\ \cos\left(\frac{2\pi}{T}t + \phi_0\right) \end{pmatrix}. \tag{4.70}$$

Parameters of the probing term has to be tuned, we utilize $U_0 = 10$, $T = 240$ (corresponds to turning by 2π every 0.06s) and $\phi_0 = 0$. Although here is necessity of tuning the regulator, every parameter has simple meaning and proper setting can be done intuitively after few initial simulations.

4.3.5 SIDP

The SIDP algorithm was implemented and tested without obtaining any sufficient results. The algorithm was also implemented for improving previously presented regulators by perturbations, however, again with no measurable improvements. Based on author's opinion, it is due to the fact, that optimization in (3.3) is performed step by step, although due to the long transient response and relatively large noise, a profit from only one single control action is hard to evaluate. Successful control policy should give similar control action for sufficiently long horizon to cause any measurable effects and, as a consequence, to improve the identification and to allow better control in future. To obtain this behavior by optimizing step by step, it would take large amount of iterations of algorithm.

Nonetheless, in contrary to original PID, CC and CEC regulator, the SIDP algorithm could provide control actions which would have the dual character naturally. This attractive property could motivate develop of some modifications of SIDP.

4.4 Results

4.4.1 Test scenario

The most challenging problem with the PMSM is starting phase without any initial information on θ. Difficulties arise due to the symmetry of the system and long delays between actions and measurable responses. The performance of particle filters and the EKF is compared.

The goal is to linearly increase the velocity during 0.1s from 0 up to 10rad.s^{-1} and to keep this value for next 0.1s. Initial state is drawn randomly from distribution representing initial estimate.

$$\begin{aligned} i_{\alpha,0} &\sim \mathcal{U}(-0.01, 0.01) \\ i_{\alpha,0} &\sim \mathcal{U}(-0.01, 0.01) \\ \omega &\sim \mathcal{U}(-0.01, 0.01) \\ \theta &\sim \mathcal{U}(-\pi, \pi). \end{aligned} \tag{4.71}$$

4.4.2 Qualitative comparison of filters

In this subsection, the qualitative results of estimating by SIR_{opt} filter and the EKF are presented. Results for the ASIR filter and the $\text{SIR}_{\text{prior}}$ are omitted because are (qualitatively) near the same as for SIR_{opt}. As consequence, comparison of these variants of particle filter is included in later subsection where quantitative results are presented.

For SIR_{opt}, we use $N = 60$, $N_{thresh} = N/5 = 12$ and deterministic resampling procedure. Control actions are generated from PI regulator (mean value of estimated distribution is used).

TODO-predelat obrazek Result of one realization is summarized in figure 4.4.2. In contrast to the SIR_{opt} filter, controlling based on the EKF yields to relatively small currents. It is due to the fact, that mean value of proposed estimate match with desired state and, thus, only small control actions are necessary. However, small actions are not very informative which cause failure of the EKF. As consequence of the failure, the EKF convergences to the state with angle shifted by π from the true state. It causes turning in opposite direction. On the other hand, in cases where better initial information is available (and thus better performance of the EKF is assumed), the EKF would be able to reach desired state very effectively. Another remarkable result is that the highest currents are reached just after estimate proposed by the SIR_{opt} filter becomes sufficiently accurate.

TODO-predelat obrazek Wrong convergence of the EKF is caused by the assumption that the posterior distribution can be sufficiently approximated by Gaussian distribution. Comparison with estimated posterior distribution proposed by the SIR_{opt} filter shows that the assumption of Gaussianity is improper (at least during first 0.05s), see figures 4.2 and 4.3. From figure 4.2, it is clear that the posterior distri-

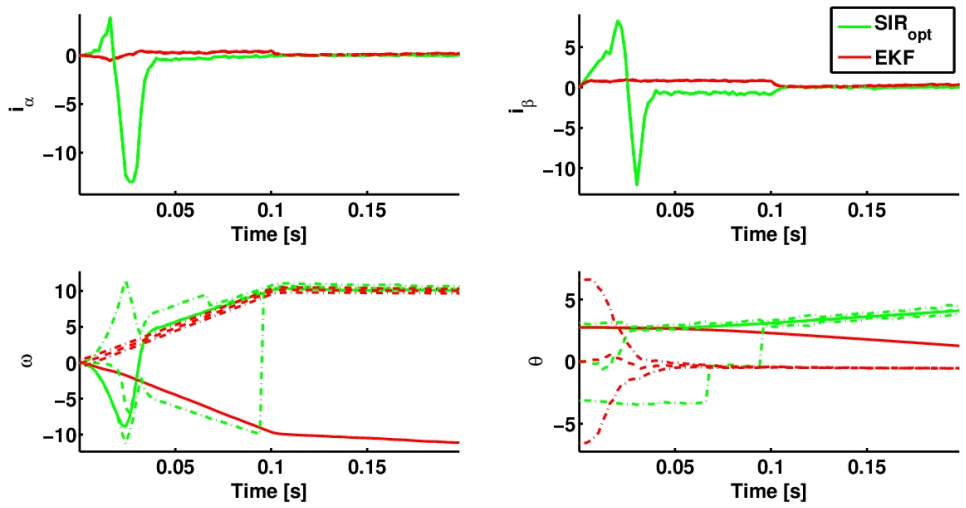

Figure 4.1: Qualitative comparison of different filters. For ω_t and θ_t, the true state is marked by full line, the estimate by dashed line, and bound of estimated distribution (in case of the $\mathrm{SIR}_{\mathrm{opt}}$) and variance (in case of the KF) by dash-dot line.

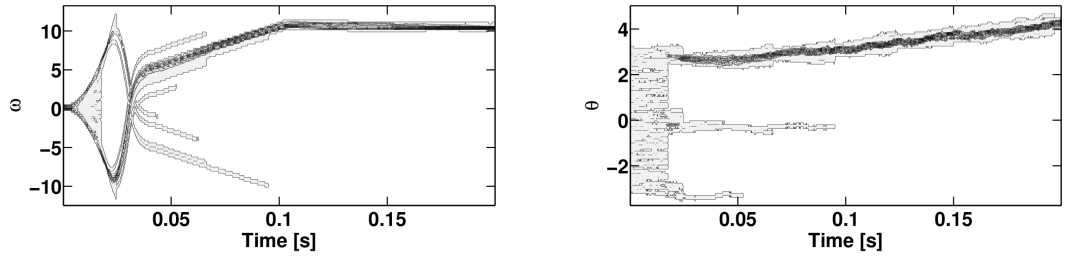

Figure 4.2: Estimated posterior distribution of ω and θ in particular times, darker color means higher probability. Time steps, in which resample step was used, can be identified by sharp edges in estimates of posterior distribution.

bution is far from the Gaussian at least during first 0.05s. Proper approximation would be rather sum of Gaussian distributions in both variables. Nonetheless, it seems that the Gaussian approximation could be sufficient after convergence during initial stage. It should be pointed out, that figure 4.2 presents only projections of joint probability distribution of both variables in particular times. The joint distribution is plotted in figure 4.3.

TODO-predelat obrazek Although in previous simulation the EKF did not converge to the true state, if the initial estimate approximates initial state well, the EKF provides good results. In contrast, dependence between convergence of particle filter and difference between initial estimate and initial state is not significant. These results are summarized in figure 4.4.

Intuitive idea of complexity and stability of particle filter (the $\mathrm{SIR}_{\mathrm{opt}}$ filter in our case) can be obtained from figure 4.5 where two realization of ω during the same simulation are presented. Initial state, initial estimate and system noise realization

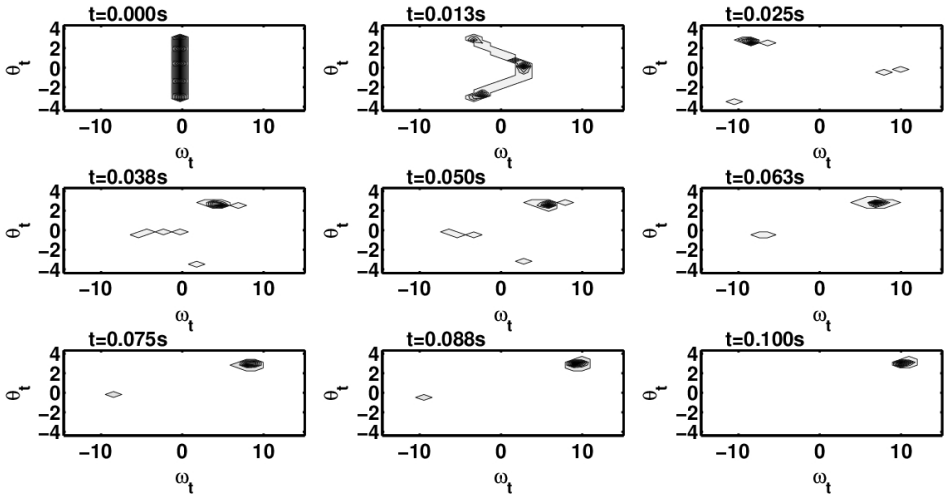

Figure 4.3: The joint distribution of ω and θ in particular times of the simulation, darker color means higher probability. After convergence of the estimate ($t > 0.075s$), ω stays on desired value and θ is shifting due to the rotor motion.

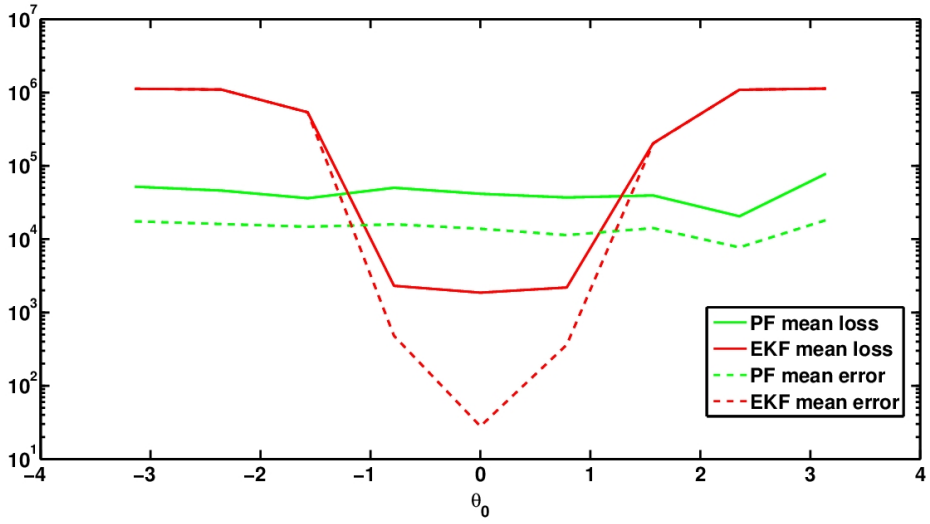

Figure 4.4: Dependence between initial θ and qulity of estimation. All the values are averages from 10 simulations.

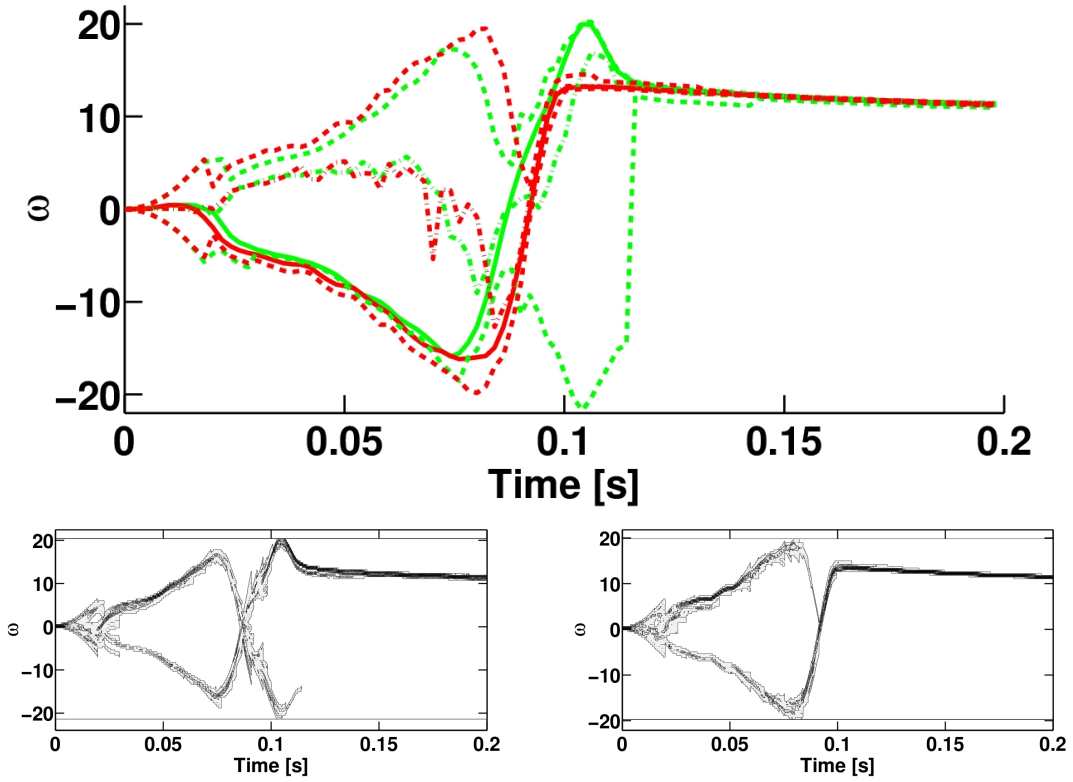

Figure 4.5: Two realizations of the same simulation using $\mathrm{SIR}_{\mathrm{opt}}$ filter. For ω_t and θ_t, the true state is marked by full line, the estimate by dashed line, and bound of estimated distribution by dash-dot line. In histograms, darker color means higher probability.

are in both cases the same. Differences are caused only by random realizations inside particle filter. Note, that differences are amplified by the fact, that the control actions are based on different estimates.

TODO-all

4.4.3 Qualitative comparison of control

This subsection deals with qualitative results of previously described regulators. For excluding influence of noise, all the realizations of noise in the system are set to zero. The $\mathrm{SIR}_{\mathrm{opt}}$ filter with deterministic variant of residual resampling is used for filtering. Also, random realizations inside the filter are the same in all simulations.

Figure 4.6 shows results when different regulators are used. In the CEC and PI regulator, mean value of estimate is utilized for computing control action. As was mentioned at the end of subsection 4.3.2, having no additional information about initial state, the CC regulator proposes control action equal to zero. Due to this fact, the system will stay unchanged and the situation will be repeated in next time steps. As consequence, the CC regulator is unable to change the state. In practice, the state of the system is influenced by noise realizations and through this, the

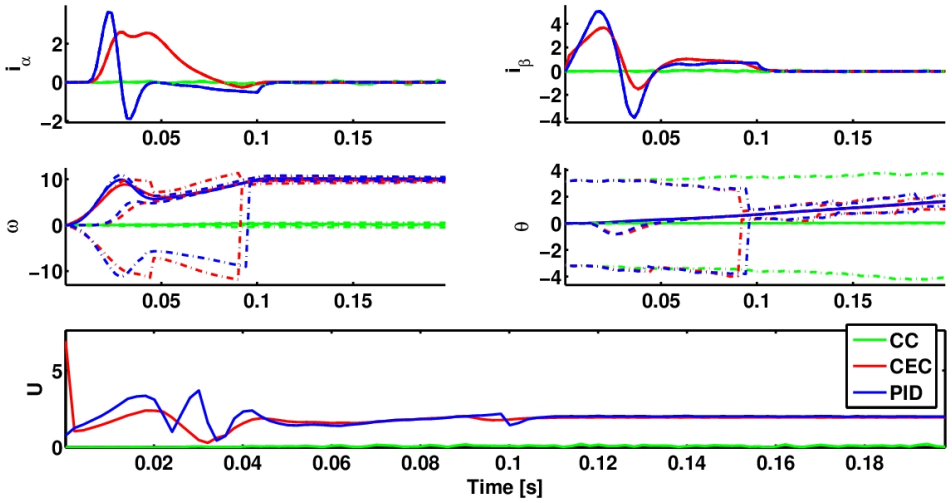

Figure 4.6: Qualitative comparison of different regulators. For ω_t and θ_t, the true state is marked by full line, the estimate by dashed line, and bound of estimated distribution (in case of the $\mathrm{SIR}_{\mathrm{opt}}$) and variance (in case of the KF) by dash-dot line. The last figure shows norm of control actions in respective times ($U = ||u||$).

system will deflected what (after sufficiently long time) gives additional information which eliminates zero control. From the figure, it can also be seen the advantage of incorporating the future effects of control action – although evolution of the state controlled by the CEC or PI regulator is nearly the same, the PI regulator proposes more agressive control actions which yields into more rapid changes in currents without any valuable effect. As consequence, we assume that the controling using the PI regulator will generate higher loss. After reaching the desired state, regulators only compensate induced currents caused by rotor motion.

4.4.4 Quantitative results

In this section, quantitative results of proposed filters and estimators on the simulation of starting phase are presented. For better distinction between cases in which particle filter (or regulator) converged to wrong value and cases where estimation (or regulation) only takes more time, we extend the second phase of the simulation where desired value of ω is kept constant to 0.3s. Simulation is repeated 1000 times. In every single simulation, initial state was drawn according to **??** and realizations of random variables was the same for all tested filters or regulators.

Quantitative comparison of previously discussed regulators is shown in figure 4.7. Obtained histograms states that the CC regulator with added probing term reach both the lowest tracking loss and estimation error. The most remarkable outline of the figure is that the amount of estimation failures significantly decreases, using modified CC regulator. For estimation, the SIR_{opt} filter with $N = 60$, $N_{thresh} = N/5 = 15$, and deterministic variant of residual resampling is used.

As was mentioned before, if the SIR_{prior} and the ASIR filter are tuned properly, they

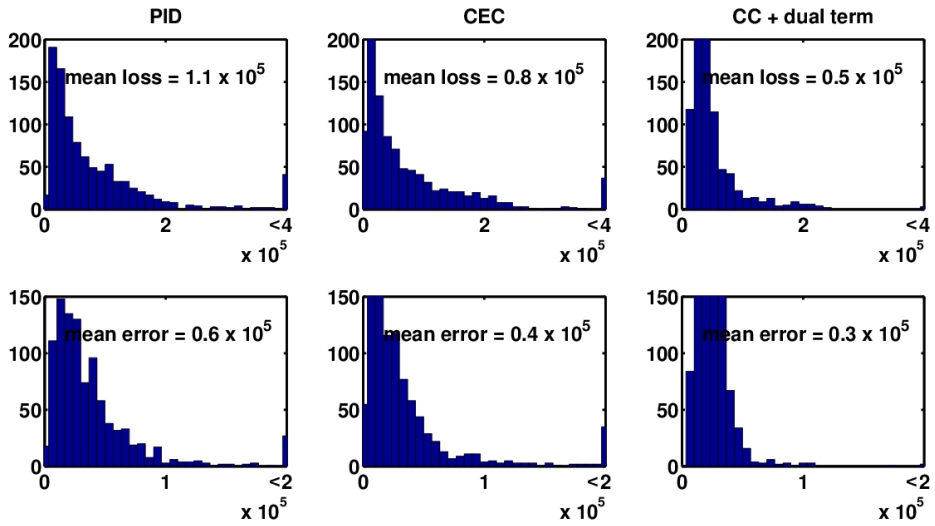

Figure 4.7: Qualitative comparison of different regulators.

propose nearly the same results as the SIR_{opt} filter. To give an evidence, see **??**. Similar results help us to tune the matrix $\tilde{R} = \rho R$ mentioned in previous text – we can see simulations in which estimation failed as those with estimation error $> 2.10^6$. Increase of robustness and thus decrease of the amount of failures can be done by increasing the ρ. However, overall estimation error increases. Some compromise can be reaching be some experiments with these histograms. Consequently, we use values for ρ mentioned before, 100 for SIR_{prior} and 1000 for ASIR.

In case of the PMSM, computation time for provide estimate is very limiting (only $\Delta t = 0.000125$s). Thus, number of particles should be small, thus we have performed the same Monte Carlo study for different number of particles. As regulator, the CC with probing term was used, the state was estimated by SIR_{opt} filter. Results of the same simulation for $N = 60$ are in the last column of figure 4.7. In the all cases deterministic variant of residual resampling with $N_{thresh} = N/5$ was utilized.

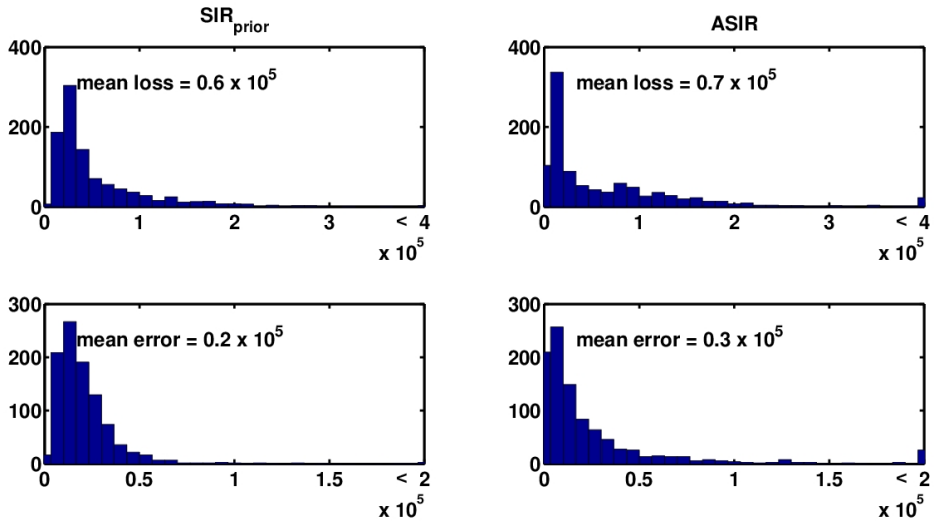

Figure 4.8: Qualitative comparison of different filters. Comparable results for the SIR_{opt} are shown in the first part of figure ??

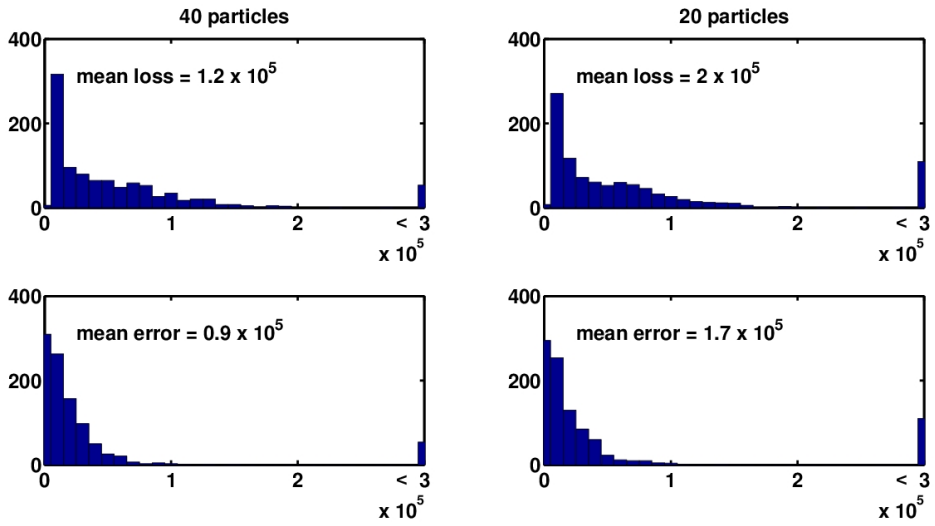

Figure 4.9: Tracking loss for different numbers of particles ($= N$). Control actions are generated by CC regulator CC with probing term.

Conclusions

The scope of this research project was to study methods of particle filtering, especially in framework of controlling under uncertainty. Successful control should well identify hidden parameters of the system and, by some regulator, to propose control actions which yields into desired system state. The identify step can be viewed as sequential estimating problem and its solution using sequential Monte Carlo method was discussed. On the other hand, regulating step is a variant of minimization problem. The problem is often analytically unsolvable and, moreover, also its approximations based on some general idea like dynamic programming or gradient method are to time consuming. Consequently, we are often tended to use some more special regulators, e.g the PID, CEC or CC regulator.

After discussing the principles of estimating and controlling in control theory framework, we was interested in application of the methods on model of PMSM. For the system, several particles filters was implemented and compared with the EKF. The best estimation of posterior distribution was proposed by $\mathrm{SIR}_{\mathrm{opt}}$, however, also other particle filters proved to be able in estimating well. For controlling, the PI, CC and CEC regulator was implemented and tested. The CC and CEC regulators was outlined upon specific additional conditions which yield into simple and effective control law. During comparison, the most successful regulator was CC with added probing term. In addition, quantitative results proved that successful estimation can be achieved using only few particles.

Further research would be to utilize proposed estimation and controlling techniques on a PMSM model which would better match with real motor. Another direction would be to propose some general algorithm which would incorporate probing in controlling, less artificial than the presented one. One possibility could be to propose some extension of the SIDP or stochastic policy gradient algorithm which would be able to perform time consuming computations more effectively and offline.

References

[1] K.H. Ang, G. Chong, and Y. Li. PID control system analysis, design, and technology. *Control Systems Technology, IEEE Transactions on*, 13(4):559–576, 2005.

[2] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *Signal Processing, IEEE Transactions on*, 50(2):174–188, 2002.

[3] D.P. Bertsekas. *Dynamic Programming and Optimal Control, vol. 1*. Athena Scientific, 1995.

[4] C. Berzuini, N.G. Best, W.R. Gilks, and C. Larizza. Dynamical Conditional Independence Models and Markov Chain Monte Carlo Methods. *Journal of the American Statistical Association*, 92(440), 1997.

[5] S. Bolognani, L. Tubiana, and M. Zigliotto. Extended Kalman filter tuning in sensorless PMSM drives. *Industry Applications, IEEE Transactions on*, 39(6):1741–1747, 2003.

[6] R. Douc and O. Cappé. Comparison of resampling schemes for particle filtering. In *Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on*, pages 64–69. IEEE, 2005.

[7] A. Doucet, N. De Freitas, and N. Gordon. *Sequential Monte Carlo methods in practice*. Springer Verlag, 2001.

[8] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and computing*, 10(3):197–208, 2000.

[9] A. Doucet and X. Wang. Monte Carlo methods for signal processing: a review in the statistical signal processing context. *Signal Processing Magazine, IEEE*, 22(6):152–170, 2005.

[10] VA Epanechnikov. Nonparametric estimation of a multidimensional probability density. *Teoriya Veroyatnostei i ee Primeneniya*, 14(1):156–161, 1969.

[11] A. A. Feldbaum. Dual control theory i-iv. 1960.

[12] P.W. Glynn and D.L. Iglehart. Importance sampling for stochastic simulations. *Management Science*, pages 1367–1392, 1989.

[13] T. Higuchi. Monte Carlo filter using the genetic algorithm operators. *Journal of Statistical Computation and Simulation*, 59, 1996.

[14] J.D. Hol, T.B. Schon, and F. Gustafsson. On resampling algorithms for particle filters. In *Nonlinear Statistical Signal Processing Workshop, 2006 IEEE*, pages 79–82. Ieee, 2007.

[15] A. Jasra and P. Del Moral. Sequential Monte Carlo methods for option pricing. *Arxiv preprint arXiv:1005.4797*, 2010.

[16] S.J. Julier and J.K. Uhlmann. A new extension of the Kalman filter to nonlinear systems. In *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*, volume 3, page 26. Citeseer, 1997.

[17] T. Hagglund K. J. Astrom. *PID Controllers:theory, Design, and Tuning*. The International Society for Measurement and Control, 1995.

[18] R.E. Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

[19] G. Kitagawa. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of computational and graphical statistics*, 5(1):1–25, 1996.

[20] A. Kong, J.S. Liu, and W.H. Wong. Sequential imputations and Bayesian missing data problems. *Journal of the American Statistical Association*, 89(425):278–288, 1994.

[21] J.H. Kotecha and P.M. Djuric. Gaussian sum particle filtering. *Signal Processing, IEEE Transactions on*, 51(10):2602–2612, 2003.

[22] J.C. Neddermeyer. Sequential Monte Carlo Methods for General State-Space Models.

[23] Z. Peroutka, V. Smidl, and D. Vosmik. Challenges and limits of extended Kalman Filter based sensorless control of permanent magnet synchronous machine drives. In *Power Electronics and Applications, 2009. EPE'09. 13th European Conference on*, pages 1–11. IEEE, 2009.

[24] M.K. Pitt and N. Shephard. Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, pages 590–599, 1999.

[25] D. Salmond, N. Gordon, and A. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proc. F, Radar and signal processing*, volume 140, pages 107–113, 1993.

[26] J.S. Simonoff. *Smoothing methods in statistics*. Springer Verlag, 1996.

[27] S.S. Singh, N. Kantas, B.N. Vo, A. Doucet, and R.J. Evans. Simulation-based optimal sensor scheduling with application to observer trajectory planning. *Automatica*, 43(5):817–830, 2007.

[28] A.M. Thompson and W.R. Cluett. Stochastic iterative dynamic programming: a Monte Carlo approach to dual control. *Automatica*, 41(5):767–778, 2005.

[29] P. Vas. *Electrical machines and drives: a space-vector theory approach*. Monographs in electrical and electronic engineering. Clarendon Press, 1992.

[30] D. Yuan, X. Cui, D. Fan, W. Feng, and Y. Yu. Application of Kalman Filter Method to the Date Processing of GPS Deformation Monitoring. In *2010 Second International Workshop on Education Technology and Computer Science*, pages 269–272. IEEE, 2010.