Imperial College London

Department of Computing

Imperial College of Science, Technology and Medicine

Gaussian Processes for Hybridisation of Analytical and Data-Driven Approaches for Design of Experiments

Simon Olofsson

Submitted:
6 December 2019

Revised:
9 May 2020

Submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Computing of the Diploma of Imperial College London

Declaration of Originality

I herewith certify that all material in this dissertation which is not my own work has been properly acknowledged.

Copyright

Abstract

In many areas of science and engineering, gathering data and making measurements of a system is costly and time-consuming. Whether the data comes from real-life experiments or computer models, we wish to maximally utilise already existing data to make informed and optimal decisions. The decisions might have to do with where next to evaluate the system, or how to control the system. Our focus is on design of experiments to aid model discrimination, i.e. discarding inadequate members of a set of rival models.

Many models are too complex to analyse to the extent we sometimes wish. To discriminate between parametric models, we often wish to compute function gradients. However, if our function involves evaluating complex legacy code or stochastic simulations, then function gradients are not readily available to us. The function is effectively a black box, where we can input variable values and collect the output function value, without knowing exactly what happens inside the box.

For situations involving black-box functions or models, we turn to black-box methods. Our approach is to construct probabilistic surrogate models using Gaussian process regression. A Gaussian process is a distribution over functions, yielding a Gaussian distribution at each test point, with a mean and variance conditioned on previous function or model evaluations.

We use the surrogate model method to tackle design of experiments for model discrimination. Given our surrogate models, we can utilise existing analytical methods to solve our problems, taking the inherent uncertainty in variables and about our surrogate models into account. Using literature case studies, we demonstrate how our method balances accuracy and computational complexity in solving both the design of experiments and model discrimination problems. We do this for both static and dynamic models. Open-source Python packages *GPdoemd* and *doepy* implement our methods.

Acknowledgements

I want to thank my supervisor, Dr Ruth Misener, for all her support and hard work during my time at Imperial College. She is an inspiring academic and goes to great lengths to look after her students. I am grateful for her belief in me, her guidance during my studies, for helping me improve as a researcher and writer of scientific texts, and for teaching me invaluable skills for my future career in or outside academia.

I also want to thank Prof. Marc Deisenroth. He agreeed to supervise my M.Sc. thesis project that first introduced me to Bayesian inference and Gaussian processes. When Ruth received funding for a new student, Marc introduced us and vouched for me. His continued support and supervision during my postgraduate studies have been extremely valuable.

I want to given thanks to all the members of the Computational Optimisation Group I have had the fortune to meet, work and laugh with: Prof. Berc Rustem, Dr Panos Parpas, Dr Chin (Clint) Pang Ho, Dr Vahan Hovhannisyan, Dr Sei Howe, Dr Juan Campos Salazar, Dr Dimitrios Letsios, Dr Georgia Kouyialis, Dr Jan Kronqvist, Radu Baltean, Miten Mistry, Johannes Wiebe, Francesco Ceccon, Alexander Thebelt and Natasha Page. Additionally, I want to thank the members of the Statistical Machine Learning Group for many interesting discussions and late dinners at Jacob's: Dr Sesh Kumas, Dr Benjamin Chamberlain, Dr Hugh Salimbeni, Sanket Kamthe, Riccardo Moriconi, Steindór Sæmundsson, Janith Petangoda, James Wilson and Alexander Terenin. I am also grateful for all the other people who have helped make the Huxley 302 office a great place to spend 3.5 years: Dr Jeremy Cohen, Francesco Borderi, Haoyang Wang and Mehdi Bahri. Everyone has contributed to making this work possible, and I have made friends for life.

I want to acknowledge the funding from the ModLife project, funded through the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement no. 675251. The ModLife ITN has given me friendship and opportunity to travel and present my work at many different venues. I want to thank all the other Early Stage Researchers for all the fun we have had: Resul Al, Łukasz Ruszczynski, Tanmoy Deb, António Grilo, Puneet Mishra, Eduardo Schultz, Mattia Turchi, Hector Forero-

Hernandez, Alireza Ehsani, Mark Jones, Frederico Montes, Chryssa Kappatou, Atakan Sahin and Behnam Khadem. I also want to thank Prof. Alexander Mitsos, Dr Adel Mhamdi and RWTH Aachen for hosting me in April-May 2017, and Dr Sebastian Niedenführ, Dr Lukas Hebing and Bayer AG for hosting me in March-April 2018.

Obviously, I want to give huge thanks to my family for believing in me and supporting me through this journey. It has meant a lot to have you visit, video call and cook me traditional Swedish food. *Detta hade inte varit möjligt utan er.* I also want to thank the Loaders, for accepting me into your family and creating a home away from home. Last, but definitely not least, I want to thank my partner, Catherine. You have shown me unwavering support, and put up with my frustration when the research was not going my way, and many evenings and weekends spent working. I am forever grateful to you!

Simon Olofsson
London, December 2019

Contents

List of Figures

List of Tables

List of Publications

M. Mehrian, Y. Guyot, I. Papantoniou, S. Olofsson, M. Sonnaert, R. Misener, and L. Geris. Maximizing neotissue growth kinetics in a perfusion bioreactor: an in silico strategy using model reduction and Bayesian optimization. *Biotechnol Bioeng*, 115(3):617–629, 2018.

S. Olofsson, M. Mehrian, L. Geris, R. Calandra, M. P. Deisenroth, and R. Misener. Bayesian multi-objective optimisation of neotissue growth in a perfusion bioreactor set-up. In A. Espuña, M. Graells, and L. Puigjaner, editors, *Proceedings of the 27th European Symposium on Computer Aided Process Engineering (ESCAPE)*, volume 40 of *Comput Aided Chem Eng*, pages 2155–2160. Elsevier, Barcelona, Spain, 2017.

S. Olofsson, M. P. Deisenroth, and R. Misener. Design of experiments for model discrimination hybridising analytical and data-driven approaches. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proc Mach Learn Res*, pages 3908–3917. PMLR, Stockholm, Sweden, 2018a.

S. Olofsson, M. P. Deisenroth, and R. Misener. Design of experiments for model discrimination using Gaussian process surrogate models. In M. R. Eden, M. G. Ierapetritou, and G. P. Towler, editors, *Proceedings of the 13th International Symposium on Process Systems Engineering*, volume 44 of *Comput Aided Chem Eng*, pages 847–852. Elsevier, San Diego, CA, USA, 2018b.

S. Olofsson, L. Hebing, S. Niedenführ, M. P. Deisenroth, and R. Misener. GPdoemd: A Python package for design of experiments for model discrimination. *Comp Chem Eng*, 125:54–70, 2019a.

S. Olofsson, M. Mehrian, R. Calandra, L. Geris, M. P. Deisenroth, and R. Misener. Bayesian multi-objective optimisation with mixed analytical and black-box functions: Application to tissue engineering. *IEEE Trans Biomed Eng*, 66:727–739, 2019b.

Open-Source Software

GPdoemd

Python package for design of experiments for model discrimination using Gaussian process surrogate models [Olofsson et al., 2019a].

Available online at `https://github.com/cog-imperial/GPdoemd`

doepy

Python package for design of dynamic experiments.

Available online at `https://github.com/scwolof/doepy`

Notation

General notation

Symbol	Description
a	Scalar variable, or function.
A	Scalar constant.
$\boldsymbol{a}$	Column vector.
$\mathbf{A}$	Matrix or tensor.
$\mathbf{I}$	Identity matrix.
$\boldsymbol{a}^\top, \mathbf{A}^\top$	Transpose of vector $\boldsymbol{a}$ and matrix $\mathbf{A}$.
$\mathbf{A}^{-1}$	Inverse of matrix $\mathbf{A}$, such that $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$.
$\lvert\mathbf{A}\rvert$	Determinant of matrix $\mathbf{A}$.
$\mathrm{tr}(\mathbf{A})$	Trace of matrix $\mathbf{A}$.
$\mathrm{diag}(\boldsymbol{a})$	Diagonal matrix with elements $a_1, \dots, a_{d_a} \in \boldsymbol{a}$ on the diagonal.
$\mathrm{diag}(\mathbf{A})$	Vector $[A_{1,1}, A_{2,2}, \dots, A_{d_A,d_A}]^\top$.
$\mathrm{diag}(\mathbf{A}, \mathbf{B})$	Block-diagonal matrix.
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Multivariate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$.
$\mathcal{N}(\boldsymbol{a} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$	Gaussian probability density function evaluated at $\boldsymbol{a}$.
$\mathbb{R}$	The set of real numbers (0, 1, $-\frac{1}{2}$, π, etc.).
$\mathbb{E}_a[b(a)]$	The expectation of $b(a)$ computed over the distribution of a.
$\mathbb{V}_a[b(a)]$	The variance of $b(a)$ computed over the distribution of a.
$\nabla_a \phi$	Gradient of $\phi(a, \dots)$ with respect to a, evaluated at $\hat{a} = \mathbb{E}_a[a]$.

Design of experiments notation

Symbol	Description
$\boldsymbol{u}$	Design/control variable.
$\boldsymbol{\Sigma}_u$	Design/control variable covariance.
$\boldsymbol{u}_{1:N}$	Chapter 5: Designs for experiments $1, \dots, N$.
$\boldsymbol{u}_{0:T-1}$	Chapter 6: Control inputs at time steps $0, \dots, T-1$.
D_u	Dimensionality of design variable space, $\boldsymbol{u} \in \mathbb{R}^{D_x}$.
$\boldsymbol{x}^{(i)}$	Latent state of model i.
$\boldsymbol{\mu}_t^{(i)}$	Latent state mean of model i at time step t.
$\boldsymbol{\Sigma}_t^{(i)}$	Latent state covariance of model i at time step t.
$D_{x,i}$	Dimensionality of latent state space of model i, $\boldsymbol{x}^{(i)} \in \mathbb{R}^{D_{x,i}}$.
$\boldsymbol{z}$	Chapter 5: Concatenated design variable and model parameter. Chapter 6: Observed state.
$\boldsymbol{y}$	Experimental measurement (observed state with noise).
$\boldsymbol{y}_{1:N}$	Chapter 5: Measurements in experiments $1, \dots, N$.
$\boldsymbol{y}_{1:T}$	Chapter 6: Measurements at time steps $1, \dots, T$.
D_z	Dimensionality of observation space, $\boldsymbol{z}, \boldsymbol{y} \in \mathbb{R}^{D_z}$.
$\boldsymbol{\theta}_i$	Parameters of model i , $\boldsymbol{\theta}_i \in \mathbb{R}^{D_\theta}$.
$\hat{\boldsymbol{\theta}}_i$	Maximum *a posteriori* parameter estimate for model i.
$\boldsymbol{\Sigma}_{\theta,i}$	Model i parameter covariance.
$D_{\theta,i}$	Dimensionality of model i's parameter space, $\boldsymbol{\theta}_i \in \mathbb{R}^{D_{\theta,i}}$.
$\boldsymbol{v}$	Measurement noise.
$\boldsymbol{\Sigma}_y$	Measurement noise covariance.
$\boldsymbol{w}$	Process noise.
$\boldsymbol{\Sigma}_x$	Process noise covariance.
f_i	Chapter 5: Model i. Chapter 6: Model i's transition function.
$\mathcal{M}_i$	Chapter 6: Model i.
M	Number of rival models f_i, $\mathcal{M}_i$; $i = 1, \dots, M$.
$f_{i,(d)}$	Output dimension d of f_i; $d = 1, \dots, D$.
$\mathcal{D}_{**}$	Design criterion, e.g. D_{JR}.
$\mathbf{H}_i$	Observation matrix of model i.

Gaussian process notation

Symbol	Description
$\mathcal{GP}(\cdot,\cdot)$	Gaussian process prior
$m(\cdot)$	Mean function.
$k(\cdot,\cdot)$	Covariance function.
$\mu(\cdot)$	Gaussian process model's posterior mean.
$\sigma^2(\cdot), \Sigma(\cdot)$	Gaussian process model's posterior variance.
ρ^2	Covariance function signal variance.
λ_j	Length scale for input dimension j.
$\mathbf{\Lambda}$	Diagonal matrix $\mathrm{diag}(\lambda_1^2, \dots, \lambda_D^2)$ of squared length scales.
$\boldsymbol{\eta}$	Measurement noise.
σ_η^2	Measurement noise variance.
$\mathbf{X}$	Training inputs, $\mathbf{X} \in \mathbb{R}^{N \times D_x}$.
$\boldsymbol{y}$	Training targets (labels), single output, $\boldsymbol{y} \in \mathbb{R}^N$.
$\mathbf{Y}$	Training targets (labels), multi-output, $\mathbf{Y} \in \mathbb{R}^{N \times D_y}$.
$\mathbf{K}$	Training inputs covariance matrix, $\mathbf{K} \in \mathbb{R}^{N \times N}$.
N	Number of training data.
P	Number of inducing points (sparse GP regression).
$\phi_{(d)}$	Dimension d of some function ϕ, used for multi-output models.

Model discrimination notation

Symbol	Description
$\pi_{i,N}$	Normalised Gaussian posterior probability of model i after N experiments.
χ_i^2	χ^2 score of model i.
ω_i	Akaike weight of model i.
$\mathcal{D}$	Data set with experimental inputs and measurements.
N	Number of measurements, $N = \lvert\mathcal{D}\rvert$.

"All experiments are designed experiments, it is just that some are poorly designed and some are well-designed."

— Unknown

1 Introduction

Systems biology, pharmaceutical engineering, biochemical engineering and many other scientific and engineering fields deal with noisy and uncertain processes. Modelling these processes is often difficult, and exacerbated by the difficulty of observing mechanisms and reactions on the cellular and molecular level or inside living test subjects. Researchers and engineers can devise different hypotheses about underlying system mechanisms to explain a system's behaviour. These hypotheses are formulated as mathematical parametric models. The development of accurate mechanistic models depends on informative experimental data to guide model discrimination and parameter estimation [Asprey and Macchietto, 2000]. Model discrimination, the process of discarding inaccurate models, forms part of the core of scientific endeavour as it is fundamentally about figuring out how the world around us works.

As a motivating example to illustrate the real-world importance of model discrimination, consider healthcare. A patient's health and safety is an important concern, and healthcare is a heavily regulated industry. In the USA and the EU, private and public regulatory bodies exist on federal/union and state levels [Field, 2008; Hervey, 2010]. Healthcare companies applying to market a new drug or medical device must submit extensive technical information to the regulatory bodies. In the USA and the EU, the Food & Drug Administration (FDA) and European Medicines Agency (EMA) handle these applications, respectively. The FDA require that the technical information contains e.g. the chemical composition of the drug, how the drug affects the human body (pharmacodynamics), how the body affects the drug (pharmacokinetics), and methods for drug manufacturing, packaging and quality control

[U.S. Food & Drug Administration, 2018]. Likewise, applying to market a new medical device requires submitting extensive technical information. A medical device can be any appliance, software or material used for non-pharmacological diagnosis, monitoring or treatment of a disease or condition [European Union, 2017]. An important part of proving the efficacy and safety of a new drug or medical device is showing that its effects on the patient can be predicted and interpreted, e.g. via mathematical and computational models.

These transparency and interpretability requirements, combined with limited amounts of available experimental data, make models learnt solely from observed data unsuitable for proving the efficacy and safety of a new drug or medical device to the regulatory bodies. Hence researchers use explicit parametric models. For drugs, the pharmacokinetics (how the human body affects the drug) is often modelled using systems of ordinary differential equations. These models elucidate how the drug is absorbed and distributed through the body (e.g. brain, kidneys and liver) under different dosing profiles and methods of drug administration (e.g. orally or intravenously) (see Figure 1.1). The EMA pharmacokinetic model guidelines state that regulatory submissions should include detailed descriptions of the models, i.e. justification of model assumptions, parameters and their biochemical plausibility, as well as parameter sensitivity analyses [European Medicines Agency, 2011, p. 16; 2016, p. 3].

DiMasi et al. [2016] estimate the average pre-approval R&D cost for new drugs to $2.56B, of which $1.1B is spent in the pre-clinical stage (in 2013 US dollars). A key problem is to identify a mathematical model that usefully explains and predicts the behaviour of the pharmacokinetic process [Galvanin et al., 2013; Heller et al., 2018]. Successful model discrimination early in the drug development may lower costs, whereas inaccurate models passing the pre-clinical stage can incur significant costs [Scannell and Bosley, 2016; Plenge, 2016]. Hence, inaccurate models should be discarded as early as possible in the drug development process. But researchers will often have insufficient experimental data to discriminate between the models. Additional experiments are generally expensive and time-consuming. Pharmacokinetic experiments typically take 8–24 hours for mice and rats, and 12–100 hours for simians and humans [Ogungbenro and Aarons, 2008; Tuntland et al., 2014]. Observability of mechanisms and reactions on the cellular and molecular level or inside living test

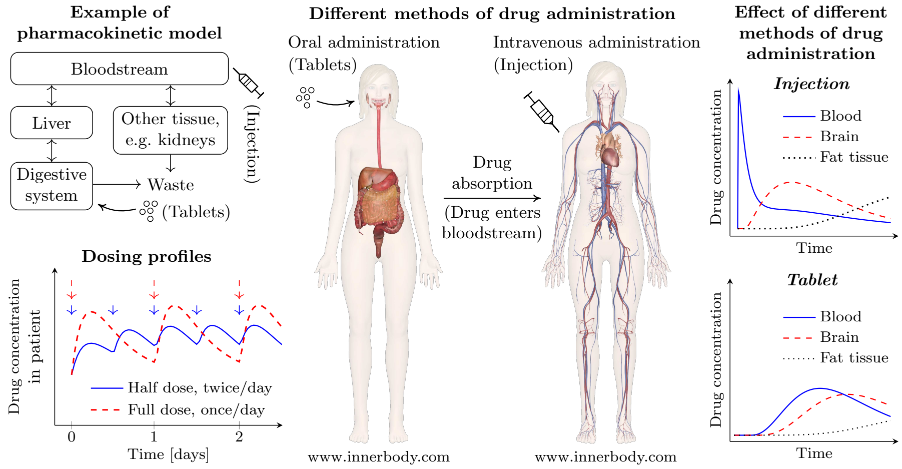


Figure 1.1: Different aspects of pharmacokinetic models. (Top left) simple pharmacokinetic model with four compartments: digestive system, liver, bloodstream and other tissue. The drug can enter as tablets into the digestive system or as an injection into the bloodstream, and leave the system by being excreted as waste. (Bottom left) the effect on drug concentration in a patient from two different dosing profiles: (i) half the dose twice per day, or (ii) the full dose once per day. (Centre) two different methods of drug administration: orally in the form of tablets, or intravenously as an injection, and (Right) their different effects: an injection has a quicker effect than tablets.

subjects is limited. To minimise the number of additional experiments required for model discrimination, it is important to design experiments yielding maximally informative outcomes.

The example presented here demonstrates the importance of mechanistic modelling, and some of the challenges faced by practitioners. Model discrimination is only one of those challenges. The optimal design of experiments to aid model discrimination is a non-trivial problem, particularly since there are many sources of uncertainty that need to be accounted for (e.g. measurement and process noise). Although literature stretches back at least as far as Hunter and Reiner [1965], some important challenges remain to be solved. We tackle some of these challenges in this work.

1.1 Objective

This thesis demonstrates how surrogate models can be used to find optimal experimental designs to aid discrimination between rival black-box models. The novel surrogate-based approach helps bridge the gap between existing approaches, and tries to combine their advantages in terms of computational speed and modelling flexibility.

1.2 Outline

Figure 1.2 illustrates the structure of the manuscript and how the chapters are linked.

Chapter 2 gives an introduction to Gaussian process regression and approximate inference, which will be used for surrogate model-based solutions when the models are not written down in analytical form.

Chapter 3 introduces existing work on optimal design of experiments for model discrimination and the two main approaches for solving the problem under uncertainty: the analytical approach and the data-driven approach. These two approaches are contrasted, and advantages and disadvantages of both approaches discussed.

Chapter 4 describes a new design criterion, based on the quadratic Jensen-Rényi divergence. Design criteria are maximised in order to find optimal designs using the analytical approach to design of experiments. The new design criterion is compared to existing design criteria in literature, and the trade-offs of different design criteria discussed.

In Chapter 5, we show how design of experiments for model discrimination can be performed even for cases of black-box models, using Gaussian process surrogate models. The software package GPdoemd is introduced.

Chapter 6 extends the GP surrogate model methodology of the previous chapter to design of dynamic experiments, where the models are explicitly time-dependent.

Chapter 7 provides a discussion of trade-offs we have made and possible future directions for research, as well as some conclusions.

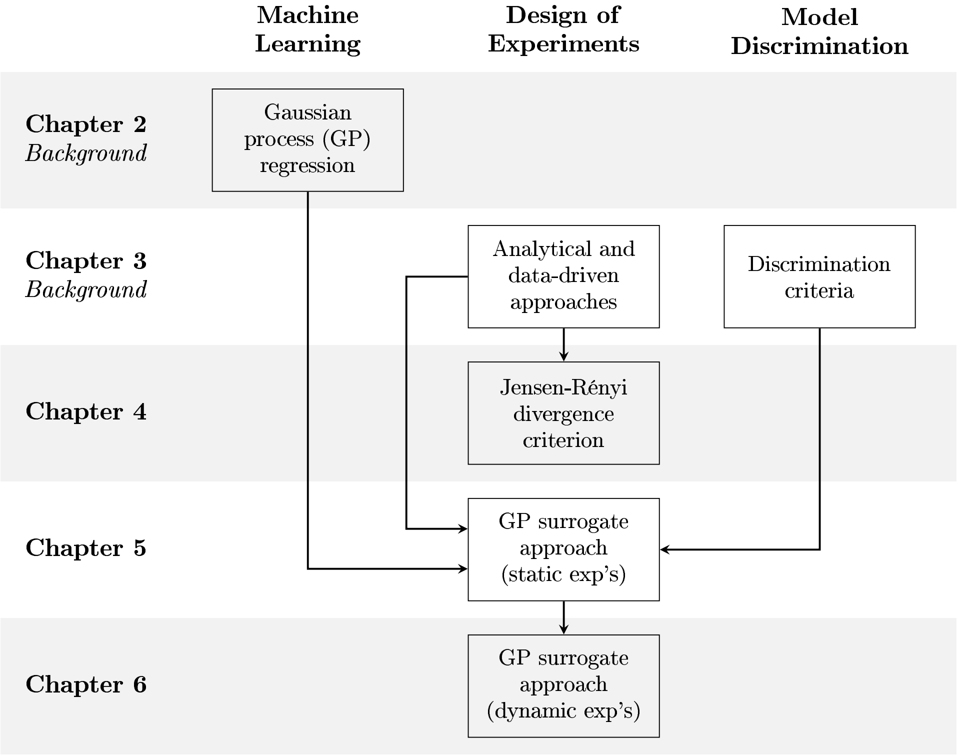

Machine Learning
Design of Experiments
Model Discrimination
Chapter 2
Background
Gaussian process (GP) regression
Chapter 3
Background
Analytical and data-driven approaches
Discrimination criteria
Chapter 4
Jensen-Rényi divergence criterion
Chapter 5
GP surrogate approach (static exp's)
Chapter 6
GP surrogate approach (dynamic exp's)

Figure 1.2: Structure of the manuscript and illustration of how the chapters are linked. The three themes of the thesis is (i) machine learning, (ii) design of experiments, and (iii) model discrimination. Chapter 4 makes a novel contribution to design of experiments, Chapter 5 combines the three themes to tackle design of static experiments for model discrimination, and Chapter 6 extends this to design of dynamic experiments.

2 Gaussian Process Regression

This chapter gives an introduction to Gaussian process (GP) regression. GPs are universal function approximators that can be trained on observed data. This makes them useful surrogates for expensive-to-evaluate black-box models.

2.1 Bayesian Inference

Bayesian statistics provides a framework for describing epistemological uncertainty (due to limited data or knowledge) using the same mathematical tools commonly employed to aleatory uncertainty (due to inherent randomness). Bayesian statistics combine prior beliefs about random variables with observed data to form posterior beliefs. The goal is to be able to account for both epistemological and aleatory uncertainty when making decisions. As an example, Bayesian statistics allows us to use the same mathematical language to describe uncertainty due to unknown system mechanisms or uncertain model parameters (epistemological uncertainty) and measurement or process noise (aleatory uncertainty). Engineers, chemists, biologists and others have to deal with many sources of uncertainty.

Assume two random variables A and B. Their joint prior distribution is denoted $p(A, B)$. Given the joint distribution, the marginal distribution $p(A)$ of A may be computed by integrating over B

$$p(A) = \int p(A, B)\mathrm{d}B\,.$$

The joint distribution $p(A, B)$ can also be written as the product of a conditional distribution, e.g. $p(B|A)$, and a marginal distribution, e.g. $p(A)$. This relationship yields what is commonly referred to as Bayes' rule[1]

$$p(A\,|B) = \frac{p(B\,|A)\,p(A)}{p(B)}\,.$$

If we know the statistical relationship between two random variables, Bayes' rule provides the means of updating our beliefs about random variables given observations. If we observe one random variable, we can compute the new *posterior* distribution for the other, e.g. the posterior distribution $p(A|B = b)$ of A given that we have observed $B = b$

$$p(A\,|B = b) = \frac{p(b\,|A)\,p(A)}{p(b)}\,.$$

The posterior-update version of Bayes' rule is often written in plain words as

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}\,.$$

Updating prior beliefs given data forms the backbone of Bayesian inference. One challenge is to find prior distributions that allow for efficient computation of the posterior distributions of random variables. The Gaussian distribution is one such prior distribution.

2.2 Gaussian Distribution

Let $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denote a random variable $\boldsymbol{x} \in \mathbb{R}^{D_x}$ with distribution $p(\boldsymbol{x})$ given by the multivariate Gaussian probability density function

$$p(\boldsymbol{x}) = (2\pi)^{-D_x/2}\,|\boldsymbol{\Sigma}|^{-1/2} \exp\left(\tfrac{-1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu})\right)\,, \tag{2.2.1}$$

with mean $\boldsymbol{\mu} \in \mathbb{R}^{D_x}$ and positive definite covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{D_x \times D_x}$.

The Gaussian distribution is perhaps the most commonly used approximation in science and

[1] Also known as Bayes' theorem or Bayes' law.

engineering for describing random variables. There are two main reasons for this. First, the Gaussian distribution often yields analytically tractable results. Secondly, the Central Limit Theorem shows that as sample sizes go to infinity, the distribution of sample means will be Gaussian distributed, which means that the Gaussian distribution is a useful approximation for scenarios where the effects of multiple random variables are combined.

Let the two random variables A and B have Gaussian joint prior distribution

$$\begin{bmatrix} A \\ B \end{bmatrix} \sim p(A,B) = \mathcal{N}\left(\begin{bmatrix} \mu_A \\ \mu_B \end{bmatrix}, \begin{bmatrix} \sigma_A^2 & \sigma_{AB}^2 \\ \sigma_{AB}^2 & \sigma_B^2 \end{bmatrix} \right) . \tag{2.2.2}$$

The marginal distributions are given by $p(A) = \mathcal{N}(\mu_A, \sigma_A^2)$ and $p(B) = \mathcal{N}(\mu_B, \sigma_B^2)$. For the covariance matrix in Equation (2.2.2) to be positive definite, $\sigma_A^2 > \sigma_{AB}^2$ and $\sigma_B^2 > \sigma_{AB}^2$ must be satisfied.

Given observation $B = y_B$ of the random variable B, the posterior distribution (i.e. the updated belief) $p(A|B)$ for A given B can be written in closed form as

$$\begin{aligned} p(A \mid B) &= \mathcal{N}\left(\mu_{A|B}, \sigma_{A|B}^2\right) , \\ \mu_{A|B} &= \mu_A + \tfrac{\sigma_{AB}^2}{\sigma_B^2}(y_B - \mu_B) , \\ \sigma_{A|B}^2 &= \sigma_A^2 - \tfrac{\left(\sigma_{AB}^2\right)^2}{\sigma_B^2} . \end{aligned} \tag{2.2.3}$$

Figure 2.1 shows an example of prior joint and marginal distributions for two random variables A and B, and the posterior distribution given an observation y_B.

Three important observations:

1. Since $\sigma_A^2 > \sigma_{AB}^2$ and $\sigma_B^2 > \sigma_{AB}^2$, the posterior variance $0 < \sigma_{A|B}^2 < \sigma_A^2$. The posterior variance is smaller than the prior variance, i.e. information has been gained from observing $B = y_B$.

2. $\sigma_{A|B}^2$ is not dependent on the observed value y_B, i.e. the reduction in variance happens regardless of what value is observed.

3. If the correlation between A and B goes to 1, then $\mu_{A|B} = y_B$ and $\sigma_{A|B}^2 = 0$.

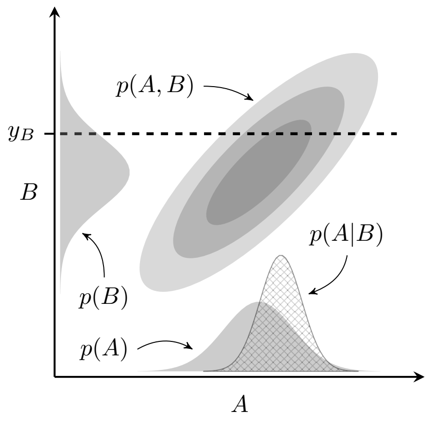


Figure 2.1: Prior joint distribution $p(A,B)$, marginal distributions $p(A)$ and $p(B)$, and the posterior distribution $p(A|B)$ after $B = y_B$ has been observed. The posterior distribution $p(A|B)$ is the slice of $p(A,B)$ at $B = y_B$.

The third observation will be important in the next section, where we parameterise the random variables. If A and B are dependent on deterministic parameters $\boldsymbol{x}_A$ and $\boldsymbol{x}_B$, then we will expect strong correlation between A and B if $\boldsymbol{x}_A = \boldsymbol{x}_B$.

2.3 Gaussian Process Inference

In Section 2.1 and Section 2.2, there is a finite set $\{A, B\}$ of random variables. But Bayesian inference can be extended to random processes and generalised to the infinite-dimensional case. A random process is a collection of random variables, where each random variable is indexed in some way. Typically, this indexation takes the form of random variables as functions $g(\boldsymbol{x})$ dependent on an external factor $\boldsymbol{x}$. The goal of Bayesian inference for random processes is to find distributions over functions.

A Gaussian process (GP) is a collection of random variables, any finite subset of which is jointly Gaussian distributed [Rasmussen and Williams, 2006]. A GP prior on a function $g : \mathbb{R}^{D_x} \to \mathbb{R}$

$$g \sim \mathcal{GP}\left(m(\cdot), k(\cdot, \cdot)\right) ,$$

is completely specified by its mean function m and covariance function k. GPs are (data-driven) Bayesian non-parametric models that can infer posterior distributions for both scalar and vector functions g.

2.3.1 Inference for Scalar Function

The function $g : \mathbb{R}^{D_x} \to \mathbb{R}$ takes a vector input and produces a scalar output. Let $\mathbf{X}$ and $g(\mathbf{X})$ denote a set of N input locations $\mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]^\top$ in matrix form, and the corresponding function values $g(\mathbf{X}) = [g(\boldsymbol{x}_1), \ldots, g(\boldsymbol{x}_N)]^\top$ in vector form. Let $\boldsymbol{m} = [m(\boldsymbol{x}_1), \ldots, m(\boldsymbol{x}_N)]^\top$ and $\mathbf{K}$ denote the prior mean and covariance at the input locations, with entries $[\mathbf{K}]_{i,j} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $i, j \in \{1, \ldots, N\}$. Then the function value $g(\boldsymbol{x})$ at a test point $\boldsymbol{x}$ and the function values $g(\mathbf{X})$ at input locations $\mathbf{X}$ have joint Gaussian distribution

$$\begin{bmatrix} g(\boldsymbol{x}) \\ g(\mathbf{X}) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(\boldsymbol{x}) \\ \boldsymbol{m} \end{bmatrix}, \begin{bmatrix} k(\boldsymbol{x}, \boldsymbol{x}) & \boldsymbol{k}^\top \\ \boldsymbol{k} & \mathbf{K} \end{bmatrix} \right),$$

where $\boldsymbol{k} = [k(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, k(\boldsymbol{x}, \boldsymbol{x}_N)]^\top$ is the cross-covariance at the test point and input locations.

Given observations $g(\mathbf{X})$ at input locations $\mathbf{X}$, the posterior distribution for $g(\boldsymbol{x})$ at test point $\boldsymbol{x}$ becomes [Rasmussen and Williams, 2006]

$$\begin{aligned} p(g(\boldsymbol{x}) \,|\, \boldsymbol{x}, \mathbf{X}, g(\mathbf{X})) = \mathcal{N}\left(\mu(\boldsymbol{x}), \sigma^2(\boldsymbol{x})\right), \\ \mu(\boldsymbol{x}) = m(\boldsymbol{x}) + \boldsymbol{k}^\top \mathbf{K}^{-1}(g(\mathbf{X}) - \boldsymbol{m}), \\ \sigma^2(\boldsymbol{x}) = k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}^\top \mathbf{K}^{-1} \boldsymbol{k}. \end{aligned} \tag{2.3.1}$$

GP regression utilises GP priors on unknown functions and the expressions in Equation (2.3.1) to infer function value distributions at unexplored input locations. Note that the expressions for the posterior mean and variance are directly equivalent to the expressions for the posterior mean and variance in Equation (2.2.3). It follows from Equation (2.3.1) and observation 3 in Section 2.2 that the posterior variance $\sigma^2(\boldsymbol{x}_i) = 0$ at input locations $\boldsymbol{x}_i \in \mathbf{X}$.

In reality, observations are usually affected by measurement noise. If only noisy observations

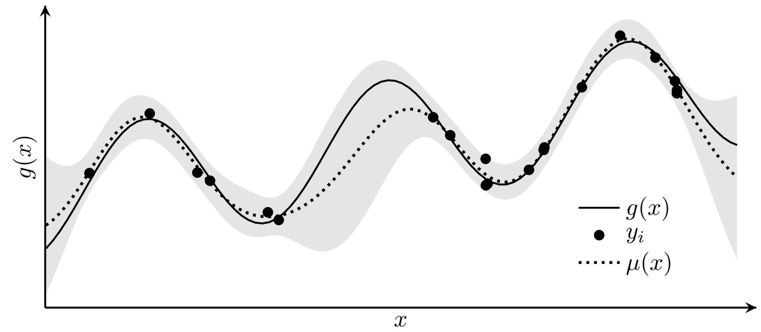


Figure 2.2: GP regression given noisy observations $\boldsymbol{y}$ (black dots) of an underlying function g (solid line). The mean (dotted line) and two standard deviations (grey area) of the GP prediction are plotted.

$\boldsymbol{y} = g(\mathbf{X}) + \boldsymbol{\eta}$ can be made of the function g, with $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \sigma_\eta^2\mathbf{I})$ zero-mean Gaussian measurement noise with variance σ_η^2, the function value $g(\boldsymbol{x})$ at a test point $\boldsymbol{x}$ and the observed values $\boldsymbol{y}$ at locations $\mathbf{X}$ have joint Gaussian distribution

$$p(g(\boldsymbol{x}), \boldsymbol{y} \,|\, \boldsymbol{x}, \mathbf{X}) = \mathcal{N}\left(\begin{bmatrix} m(\boldsymbol{x}) \\ \boldsymbol{m} \end{bmatrix}, \begin{bmatrix} k(\boldsymbol{x}, \boldsymbol{x}) & \boldsymbol{k}^\top \\ \boldsymbol{k} & \mathbf{K} + \sigma_\eta^2\mathbf{I} \end{bmatrix}\right),$$

The posterior distribution for $g(\boldsymbol{x})$ given noisy observations $\boldsymbol{y}$ is given by [Rasmussen and Williams, 2006]

$$\begin{aligned} p(g(\boldsymbol{x}) \,|\, \boldsymbol{x}, \mathbf{X}, \boldsymbol{y}) &= \mathcal{N}\left(\mu(\boldsymbol{x}), \sigma^2(\boldsymbol{x})\right), \\ \mu(\boldsymbol{x}) &= m(\boldsymbol{x}) + \boldsymbol{k}^\top(\mathbf{K} + \sigma_\eta^2\mathbf{I})^{-1}(g(\mathbf{X}) - \boldsymbol{m}), \\ \sigma^2(\boldsymbol{x}) &= k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}^\top(\mathbf{K} + \sigma_\eta^2\mathbf{I})^{-1}\boldsymbol{k}. \end{aligned} \tag{2.3.2}$$

For the case of noisy observations, the posterior variance $\sigma^2(\boldsymbol{x}) > 0$ for all $\boldsymbol{x} \in \mathbb{R}^{D_x}$.

Figure 2.2 illustrates GP regression. Given noisy observations $y_i = g(x_i) + \eta_i$, $i = 1, \ldots, N$, of the function g, Equation (2.3.2) computes a mean prediction $\mu(x)$ and corresponding variance $\sigma^2(x)$. We see that the variance is lower in regions near data than in regions further away from data.

2.3.2 Inference for Vector Function

For many applications, the unknown function g has multiple output dimensions: $g : \mathbb{R}^{D_x} \to \mathbb{R}^{D_y}$, $D_y \geq 2$. We index the output dimensions of g, such that $y_{(d)}$ denotes the d^{th} dimension of observation $\boldsymbol{y}$. To avoid later confusion with other indices, the output dimension indices will be written inside brackets through this manuscript. Multiple output dimensions makes GP inference of the posterior distribution of g more complex. Similar to before, the input locations in matrix form are $\mathbf{X} \in \mathbb{R}^{N \times D_x}$ and the corresponding observations $\mathbf{Y} \in \mathbb{R}^{N \times D_y}$, with $[\mathbf{Y}]_{i,(d)} = y_{(d),i}$, $i = 1, \ldots, N$ and $d = 1, \ldots, D_y$.

One possible approach is to place independent GP priors $g_{(d)} \sim \mathcal{GP}(m_{(d)}(\cdot), k_{(d)}(\cdot, \cdot))$ on each output dimension $d = 1, \ldots, D_y$ of g. The assumption is that the output function values $g_{(1)}(\boldsymbol{x}), \ldots, g_{(D_y)}(\boldsymbol{x})$ are approximately independent for a given input $\boldsymbol{x}$. Using the independent GP priors, the posterior predictive distribution at a test point $\boldsymbol{x}$ is given by

$$\begin{aligned} p(g(\boldsymbol{x}) \mid \boldsymbol{x}, \mathbf{X}, \mathbf{Y}) &= \mathcal{N}\left(\mu(\boldsymbol{x}), \Sigma(\boldsymbol{x})\right) , \\ \mu(\boldsymbol{x}) &= \left[\mu_{(1)}(\boldsymbol{x}), \ldots, \mu_{(D_y)}(\boldsymbol{x})\right]^\top , \\ \Sigma(\boldsymbol{x}) &= \operatorname{diag}\left(\sigma^2_{(1)}(\boldsymbol{x}), \ldots, \sigma^2_{(D_y)}(\boldsymbol{x})\right) . \end{aligned} \tag{2.3.3}$$

where the means $\mu_{(d)}(\boldsymbol{x})$ and variances $\sigma^2_{(d)}(\boldsymbol{x})$, $d = 1, \ldots, D_y$, are given by Equation (2.3.1) and Equation (2.3.2), for the case of noise-free or noisy measurements, respectively.

A different possible GP prior on g uses a linear combination of scalar base functions [Helterbrand and Cressie, 1994; Seeger et al., 2004; Lawrence, 2015; Álvarez, 2017]

$$g(\boldsymbol{x}) = \sum_j \boldsymbol{a}_j \tilde{g}_j(\boldsymbol{x}) .$$

Independent GP priors $\tilde{g}_j \sim \mathcal{GP}(m_j(\cdot), k_j(\cdot, \cdot))$ are put on the base functions $\tilde{g}_j : \mathbb{R}^{D_x} \to \mathbb{R}$. The linear coefficients $\boldsymbol{a}_j$ are found from data. This yields a semi-parametric model.

In this manuscript, the former approach (independent GP priors on each output) is used, in order to simplify computations.

2.4 Covariance Functions

GP regression uses covariance functions $k(\boldsymbol{x}, \boldsymbol{x}') = \operatorname{cov}(g(\boldsymbol{x}), g(\boldsymbol{x}'))$ to model the similarity between function outputs $g(\boldsymbol{x}), g(\boldsymbol{x}')$ for different inputs $\boldsymbol{x}, \boldsymbol{x}'$. As an example, take the affine function $g_{\text{aff}}(\boldsymbol{x}) = \boldsymbol{a}^\top \boldsymbol{x} + b$, with $\boldsymbol{a} \sim \mathcal{N}(\mathbf{0}, \sigma_a^2 \mathbf{I})$ and $b \sim \mathcal{N}(0, \sigma_b^2)$. The covariance function k corresponding to this g_{aff} is the linear kernel

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sigma_a^2 \boldsymbol{x}^\top \boldsymbol{x}' + \sigma_b^2 \,.$$

GP regression with the linear covariance function is equivalent to Bayesian linear regression.

For a function k to be a valid covariance function, the Gram matrix $\mathbf{K}$, with elements $[\mathbf{K}]_{i,j} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, must be positive definite for all choices of input locations $\{\boldsymbol{x}_i\}_{i=1}^N$ and $N > 0$ [Rasmussen and Williams, 2006].

A covariance function k is said to be stationary if its value only depends on the distance $\|\boldsymbol{x} - \boldsymbol{x}'\|$ between input locations $\boldsymbol{x}$ and $\boldsymbol{x}'$, and not their specific values, i.e. $k(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{x} + \boldsymbol{\delta}, \boldsymbol{x}' + \boldsymbol{\delta})$ for all $\boldsymbol{\delta} \in \mathbb{R}^{D_x}$. It is common to write stationary covariance functions as functions of the distance measure

$$r = \sqrt{(\boldsymbol{x} - \boldsymbol{x}')^\top \boldsymbol{\Lambda}^{-1} (\boldsymbol{x} - \boldsymbol{x}')} \,,$$

where $\boldsymbol{\Lambda} = \operatorname{diag}(\lambda_1^2, \ldots, \lambda_{D_x}^2)$ is a diagonal matrix of squared length scales. Some common stationary covariance functions are

- The radial bases function (RBF) kernel[2]: $k(r) = \rho^2 \exp\left(-\frac{1}{2} r^2\right)$.
- The exponential kernel: $k(r) = \rho^2 \exp\left(-r\right)$.
- The Matérn-3/2 kernel: $k(r) = \rho^2 \left(1 + \sqrt{3} r\right) \exp\left(-\sqrt{3} r\right)$.
- The Matérn-5/2 kernel: $k(r) = \rho^2 \left(1 + \sqrt{5} r + \frac{5}{3} r^2\right) \exp\left(-\sqrt{5} r\right)$.
- The rational quadratic kernel: $k(r) = \rho^2 \left(1 + \frac{1}{2\alpha} r^2\right)^{-\alpha}$.

[2] also known as the squared exponential kernel or Gaussian kernel

where ρ^2 is the signal variance. Different covariance functions correspond to different prior beliefs about the underlying function being modelled, e.g. how quickly it changes or if it is periodic. Selecting the best covariance function is an art, and requires training and experience. A common choice is the RBF kernel, which has some nice properties such as being infinitely differentiable. For more information about the different covariance functions, readers are referred to Rasmussen and Williams [2006]. This manuscript uses the RBF kernel unless otherwise stated.

2.5 Hyperparameter Learning

GPs are Bayesian non-parametric models, but the mean and covariance functions may contain hyperparameters whose values need to set appropriately. For the stationary covariance functions defined in Section 2.4, the hyperparameters consist of the signal variance ρ^2 and the squared lengthscales $\mathbf{\Lambda}$. Often the noise variance σ_η^2 is unknown, and added to the list of hyperparameters. The data $\mathbf{X}$, $\boldsymbol{y}$ is commonly referred to as the *training data*, and is used to learn the GP hyperparameters at training. For simplicity, it is common to normalise the observed data $\boldsymbol{y}$ (such that $\mathrm{avg}(\boldsymbol{y}) = 0$ and $\mathrm{var}(\boldsymbol{y}) = 1$) and use the mean function $m(\cdot) \equiv 0$ [Lawrence, 2015].

The hyperparameters are typically learnt by maximising the log-marginal likelihood

$$\log p(\boldsymbol{y} \,|\, \mathbf{X}, \rho^2, \mathbf{\Lambda}, \sigma_\eta^2) \propto -\boldsymbol{y}^\top (\mathbf{K} + \sigma_\eta^2 \mathbf{I})^{-1} \boldsymbol{y} - \log |\mathbf{K} + \sigma_\eta^2 \mathbf{I}| \,,$$

of the training targets with respect to the hyperparameters. Maximising the log-marginal likelihood can be done using gradient-based non-convex optimisation methods. A fully Bayesian approach places priors on the hyperparameters and integrates over them at prediction. However, this is generally analytically intractable.

For notational simplicity, the conditional dependence on the learnt hyperparameters is not written out in the expressions for the posterior predictive distributions

$$p(g(\boldsymbol{x}) \,|\, \boldsymbol{x}, \mathbf{X}, \boldsymbol{y}) \equiv p(g(\boldsymbol{x}) \,|\, \boldsymbol{x}, \mathbf{X}, \boldsymbol{y}, \rho^2, \mathbf{\Lambda}, \sigma_\eta^2) \,.$$

If the underlying function g is deterministic, then the noise covariance $\sigma_\eta^2 = 0$. This is the case when g is a black-box model that we wish to replace with a GP surrogate model. Equation (2.3.1) gives the expressions for GP regression. However, the matrix $\mathbf{K}$ may be ill-conditioned (e.g. if two training inputs are very close to each other), which can lead to numerical issues [Neal, 1999]. To avoid this, we lower-bound or fix the noise covariance σ_η^2 by a constant, e.g. $\sigma_\eta^2 \geq 1\text{E-}6$.

2.6 Sparse Gaussian Process Regression

The curse of dimensionality means that with increasing input dimensions, the amount of training data required to accurately model a system grows exponentially. For large training data sets, the matrix inversion $(\mathbf{K} + \sigma_\eta^2\mathbf{I})^{-1}$ in GP regression becomes a computational bottleneck. The computational complexity of GP training scales as $\mathcal{O}(N^3)$. Computing the predictive mean and variance (in Equation (2.3.1) and Equation (2.3.2)) scales as $\mathcal{O}(N)$ and $\mathcal{O}(N^2)$, respectively (with the term $(\mathbf{K} + \sigma_\eta^2\mathbf{I})^{-1}(\boldsymbol{y} - \boldsymbol{m})$ pre-computed).

Various methods have been proposed to reduce this computational bottleneck by sparsifying the GP regression. Sparse GP regression methods approximate the predictive distributions by selecting a smaller set of P inducing points. These inducing points can either be a subset of the original training data set [Smola and Bartlett, 2001; Seeger et al., 2003] or pseduo-inputs [Snelson and Ghahramani, 2006; Titsias, 2009]. Sparse GP regression scales as $\mathcal{O}(NP^2)$ at training, and $\mathcal{O}(P)$ and $\mathcal{O}(P^2)$ for computing the predictive mean and variance, respectively. The number P is chosen as a trade-off between predictive accuracy and computational complexity.

Other methods of reducing the computational bottleneck of GP regression look at e.g. the spectral representation of GPs [Hensman et al., 2018] or exploiting algebraic properties of the Kronecker and Khatri-Rao tensor products on a grid of inducing inputs [Evans and Nair, 2018]. A review of sparse and variational GP regression methods—some of which can be used for training sets with billions of data points—can be found in Liu et al. [2018].

2.7 Prediction with Uncertain Inputs

In Section 2.3, the predictive distribution for $g(\boldsymbol{x})$ is given for a deterministic test point $\boldsymbol{x} \in \mathbb{R}^{D_x}$ using GP regression. However, there are many cases where the input $\boldsymbol{x}$ may be uncertain, e.g. in control applications where $\boldsymbol{x}$ is a state estimate. For these cases, with an unknown function g and unknown input $\boldsymbol{x}$, we may still wish to compute the predictive distribution of $g(\boldsymbol{x})$.

Consider the general case of (possibly multi-output) GP regression, i.e. $g : \mathbb{R}^{D_x} \to \mathbb{R}^{D_y}$, $D_y \geq 1$,

$$p(g(\boldsymbol{x}) \,|\, \boldsymbol{x}, \mathbf{X}, \mathbf{Y}) = \mathcal{N}(\mu(\boldsymbol{x}), \Sigma(\boldsymbol{x}))\,.$$

If the input $\boldsymbol{x}$ is uncertain, e.g. $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$, the posterior predictive distribution is given by marginalising out the input $\boldsymbol{x}$

$$p\left(g(\boldsymbol{x}) \,|\, \mathbf{X}, \mathbf{Y}, \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x\right) = \int p\left(g(\boldsymbol{x}) \,|\, \boldsymbol{x}, \mathbf{X}, \mathbf{Y}\right)\, p\left(\boldsymbol{x} \,|\, \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x\right) \mathrm{d}\boldsymbol{x}\,. \tag{2.7.1}$$

Note that propagating a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$ through a non-linear function g yields a non-Gaussian predictive distribution. Hence, the marginal posterior distribution in Equation (2.7.1) is generally analytically intractable. A common approximation of the marginal posterior distribution is a Gaussian distribution with mean and covariance given by the first two moments of the marginal posterior distribution (moment matching)

$$\begin{aligned} p(g(\boldsymbol{x}) \,|\, \mathbf{X}, \mathbf{Y}, \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) &\approx \mathcal{N}\big(\breve{\boldsymbol{\mu}}, \breve{\boldsymbol{\Sigma}}\big)\,, \\ \breve{\boldsymbol{\mu}} &= \mathbb{E}_{\boldsymbol{x}}\left[\mu(\boldsymbol{x})\right]\,, \\ \breve{\boldsymbol{\Sigma}} &= \mathbb{V}_{\boldsymbol{x}}\left[\mu(\boldsymbol{x})\right] + \mathbb{E}_{\boldsymbol{x}}\left[\Sigma(\boldsymbol{x})\right] \end{aligned} \tag{2.7.2}$$

Computing the mean $\breve{\boldsymbol{\mu}}$ and variance $\breve{\boldsymbol{\Sigma}}$ in Equation (2.7.2), the first two moments of the marginal posterior distribution, is itself often analytically intractable. They can be approximated by propagating the Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$ through Taylor expansions of the predictive mean $\mu(\boldsymbol{x})$ and variance $\Sigma(\boldsymbol{x})$ with respect to $\boldsymbol{x}$ [Girard et al., 2003]. If the GPs for all output dimensions $d = 1, \ldots, D_y$ of g have RBF covariance functions, a closed-form solution exists for $\breve{\boldsymbol{\mu}}$ and $\breve{\boldsymbol{\Sigma}}$ [Quiñonero-Candela et al., 2003; Deisenroth et al., 2009].

For notational simplicity, let $\nabla_{\boldsymbol{x}}\phi = \partial\phi(\boldsymbol{x})/\partial\boldsymbol{x}|_{\boldsymbol{x}=\boldsymbol{\mu}_x} \in \mathbb{R}^{D_x}$ denote the partial derivative of a function ϕ with respect to the input $\boldsymbol{x} \in \mathbb{R}^{D_x}$, evaluated at the mean $\boldsymbol{x} = \boldsymbol{\mu}_x$. Furthermore, let $\nabla_{\boldsymbol{x}}\boldsymbol{\phi} = [\nabla_{\boldsymbol{x}}\phi_{(1)}^\top, \ldots, \nabla_{\boldsymbol{x}}\phi_{(D_y)}^\top]^\top \in \mathbb{R}^{D_y \times D_x}$ denote the partial derivative of a vector function ϕ with D_y outputs, and $\nabla_{\boldsymbol{x}}^2\boldsymbol{\phi} \in \mathbb{R}^{D_y \times D_x \times D_x}$ its Hessian. Note that $\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x} - \boldsymbol{\mu})$ denotes the multiplication of $\nabla_{\boldsymbol{x}}\phi$ and $(\boldsymbol{x} - \boldsymbol{\mu})$, and not a function evaluation.

2.7.1 First-Order Taylor Approximation

The first-order Taylor expansion of $\mu_{(d)}(\boldsymbol{x})$ and $\sigma_{(d)}^2(\boldsymbol{x})$ with respect to $\boldsymbol{x}$, around the point $\boldsymbol{x} = \boldsymbol{\mu}_x$, is given by

$$\begin{aligned}\mu_{(d)}(\boldsymbol{x}) &\approx \mu_{(d)}(\boldsymbol{\mu}_x) + \nabla_{\boldsymbol{x}}\mu_{(d)}(\boldsymbol{x} - \boldsymbol{\mu}_x)\,,\\ \sigma_{(d)}^2(\boldsymbol{x}) &\approx \sigma_{(d)}^2(\boldsymbol{\mu}_x) + \nabla_{\boldsymbol{x}}\sigma_{(d)}^2(\boldsymbol{x} - \boldsymbol{\mu}_x)\,.\end{aligned}$$

With a Gaussian distributed input $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$, using the first-order Taylor expansions yields

$$\begin{aligned}\mathbb{E}_{\boldsymbol{x}}\left[\mu(\boldsymbol{x})\right] &\approx \mu(\boldsymbol{\mu}_x)\,,\\ \mathbb{E}_{\boldsymbol{x}}\left[\Sigma(\boldsymbol{x})\right] &\approx \Sigma(\boldsymbol{\mu}_x)\,,\\ \mathbb{V}_{\boldsymbol{x}}\left[\mu(\boldsymbol{x})\right] &\approx \nabla_{\boldsymbol{x}}\boldsymbol{\mu}\boldsymbol{\Sigma}_x\nabla_{\boldsymbol{x}}\boldsymbol{\mu}^\top\,.\end{aligned} \tag{2.7.3}$$

The expressions in Equation (2.7.3) are inserted in Equation (2.7.2) to yield the mean $\breve{\boldsymbol{\mu}}$ and covariance $\breve{\boldsymbol{\Sigma}}$ of the marginal posterior distribution.

2.7.2 Second-Order Taylor Approximation

The second-order Taylor expansion of $\mu_{(d)}(\boldsymbol{x})$ and $\sigma^2_{(d)}(\boldsymbol{x})$ with respect to $\boldsymbol{x}$, around the point $\boldsymbol{x} = \boldsymbol{\mu}_x$, is given by

$$\mu_{(d)}(\boldsymbol{x}) \approx \mu_{(d)}(\boldsymbol{\mu}_x) + \nabla_{\boldsymbol{x}}\mu_{(d)}(\boldsymbol{x} - \boldsymbol{\mu}_x) + \tfrac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_x)^\top \nabla^2_{\boldsymbol{x}}\mu_{(d)}(\boldsymbol{x} - \boldsymbol{\mu}_x)\,,$$
$$\sigma^2_{(d)}(\boldsymbol{x}) \approx \sigma^2_{(d)}(\boldsymbol{\mu}_x) + \nabla_{\boldsymbol{x}}\sigma^2_{(d)}(\boldsymbol{x} - \boldsymbol{\mu}_x) + \tfrac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_x)^\top \nabla^2_{\boldsymbol{x}}\sigma^2_{(d)}(\boldsymbol{x} - \boldsymbol{\mu}_x)\,.$$

With a Gaussian distributed input $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$, using the second-order Taylor expansions yields the marginal mean $\breve{\boldsymbol{\mu}} = \mathbb{E}_{\boldsymbol{x}}\left[\mu(\boldsymbol{x})\right]$ with elements approximated by

$$\breve{\mu}_{(d)} \approx \mu_{(d)}(\boldsymbol{\mu}_x) + \tfrac{1}{2}\operatorname{tr}\left(\nabla^2_{\boldsymbol{x}}\mu_{(d)}\boldsymbol{\Sigma}_x\right)\,, \tag{2.7.4}$$

where we use the matrix identity $\boldsymbol{a}^\top\mathbf{B}\boldsymbol{c} = \operatorname{tr}(\mathbf{B}\boldsymbol{c}\boldsymbol{a}^\top)$ and $\mathbb{E}_{\boldsymbol{x}}[(\boldsymbol{x} - \boldsymbol{\mu}_x)(\boldsymbol{x} - \boldsymbol{\mu}_x)^\top] = \boldsymbol{\Sigma}_x$.

Similarly, approximating the expected value of the GP variance $\Sigma(\boldsymbol{x})$ using the second-order Taylor expansions yields a diagonal matrix $\mathbb{E}_{\boldsymbol{x}}\left[\Sigma(\boldsymbol{x})\right] \approx \operatorname{diag}(q_{(1)}, \ldots, q_{(D_y)})$ with elements $q_{(d)}$ given by

$$q_{(d)} = \sigma^2_{(d)}(\boldsymbol{\mu}_x) + \tfrac{1}{2}\operatorname{tr}\left(\nabla^2_{\boldsymbol{x}}\sigma^2_{(d)}\boldsymbol{\Sigma}_x\right)\,, \tag{2.7.5}$$

using the same matrix identity trick as in Equation (2.7.4).

The variance $\mathbb{V}_{\boldsymbol{x}}\left[\mu(\boldsymbol{x})\right] \approx \mathbf{Q}$ of the GP mean is a full matrix with elements

$$[\mathbf{Q}]_{(d_1),(d_2)} = \nabla_{\boldsymbol{x}}\mu_{(d_1)}\boldsymbol{\Sigma}_x\nabla_{\boldsymbol{x}}\mu^\top_{(d_2)} + \tfrac{1}{2}\operatorname{tr}\left(\nabla^2_{\boldsymbol{x}}\mu_{(d_1)}\boldsymbol{\Sigma}_x\nabla^2_{\boldsymbol{x}}\mu_{(d_2)}\boldsymbol{\Sigma}_x\right)\,, \tag{2.7.6}$$

for $d_1, d_2 = 1, \ldots, D_y$. Figure 2.3 illustrates the difference between the first- and second-order Taylor approximation; the marginal predictive distribution is Gaussian in both cases, but in this case the mean is shifted slightly downwards using the second-order approach to reflect the fact that the test point μ_x is almost at the location of the curve $g(x)$ peak.

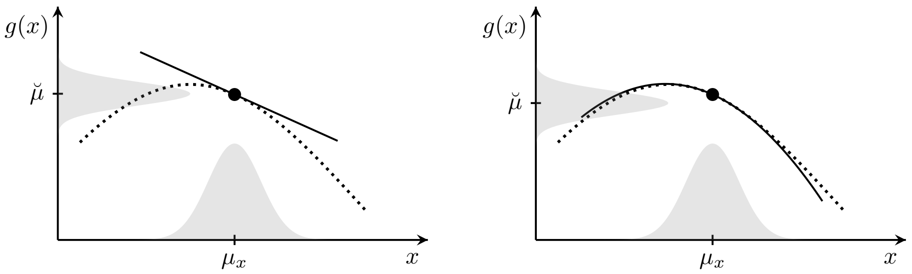


Figure 2.3: First- (left) and second-order (right) Taylor approximations of the marginal predictive distribution (grey area on y-axis), given an input distribution (grey area on x-axis) and Taylor approximation (solid line) of the function (dotted line). The marginal predictive distribution is shifted slightly downwards for the second-order approximation compared to the first-order one.

2.7.3 Exact Marginalisation

There are closed-form expressions for $\breve{\boldsymbol{\mu}}$ and $\breve{\boldsymbol{\Sigma}}$ for the case where the GPs for all output dimensions $d = 1, \ldots, D_y$ of g have RBF covariance functions [Quiñonero-Candela et al., 2003; Deisenroth et al., 2009]. Let $\rho^2_{(d)}$, $\boldsymbol{\Lambda}_{(d)}$ and $\sigma^2_{\eta,(d)}$ denote the RBF covariance function's signal variance, diagonal matrix of squared lengthscales and noise variance for the GP prior $g_{(d)} \sim \mathcal{GP}(m_{(d)}(\cdot), k_{(d)}(\cdot, \cdot))$. Additionally, let $\mathbf{K}_{(d)}$, $\boldsymbol{y}_{(d)}$ and $\boldsymbol{m}_{(d)} = m_{(d)}(\mathbf{X})$ denote the corresponding covariance matrix, training targets and prior mean. Then the marginal mean $\breve{\boldsymbol{\mu}}$ has elements $\breve{\mu}_{(d)} = \boldsymbol{\beta}^\top_{(d)} \boldsymbol{q}_{(d)}$, where $\boldsymbol{\beta}_{(d)} = (\mathbf{K}_{(d)} + \sigma^2_{\eta,(d)} \mathbf{I})^{-1} (\boldsymbol{y}_{(d)} - \boldsymbol{m}_{(d)})$ and $\boldsymbol{q}_{(d)}$ has elements given by

$$q_{(d),i} = \rho^2_{(d)} |\boldsymbol{\Sigma}_x \boldsymbol{\Lambda}^{-1}_{(d)} + \mathbf{I}|^{-\frac{1}{2}} \exp\left(-\tfrac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu}_x)^\top (\boldsymbol{\Sigma}_x + \boldsymbol{\Lambda}_{(d)})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_x)\right) ,$$

for $i = 1, \ldots, N$ the training input index. The marginal covariance $\breve{\boldsymbol{\Sigma}}$ has elements given by

$$[\breve{\boldsymbol{\Sigma}}]_{(d_1),(d_2)} = \begin{cases} \boldsymbol{\beta}_{(d_1)} \mathbf{Q} \boldsymbol{\beta}_{(d_2)} - \breve{\mu}_{(d_1)} \breve{\mu}_{(d_2)} \, , & e_1 \neq e_2 \\ \boldsymbol{\beta}_{(d_1)} \mathbf{Q} \boldsymbol{\beta}_{(d_1)} - \breve{\mu}^2_{(d_1)} + \rho^2_{(d_1)} - \operatorname{tr}\Big((\mathbf{K}_{(d_1)} + \sigma^2_{\eta,(d_1)} \mathbf{I})^{-1} \mathbf{Q} \Big) \, , & d_1 = d_2 \, , \end{cases}$$

where the matrix $\mathbf{Q}$ has elements

$$[\mathbf{Q}]_{i,j} = \frac{k_{(d_1)}(\boldsymbol{x}_i, \boldsymbol{\mu}_x)k_{(d_2)}(\boldsymbol{x}_j, \boldsymbol{\mu}_x)}{\sqrt{|\mathbf{R}|}} \exp\left(\tfrac{1}{2}\chi_{ij}^\top \mathbf{R}^{-1}\boldsymbol{\Sigma}_x\chi_{ij}\right)$$

for $i, j = 1, \dots, N$, with $\mathbf{R} = \boldsymbol{\Sigma}_x(\boldsymbol{\Lambda}_{(d_1)}^{-1} + \boldsymbol{\Lambda}_{(d_2)}^{-1}) + \mathbf{I}$ and $\chi_{ij} = \boldsymbol{\Lambda}_{(d_1)}^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_x) + \boldsymbol{\Lambda}_{(d_2)}^{-1}(\boldsymbol{x}_j - \boldsymbol{\mu}_x)$.

These closed-form expressions for $\breve{\boldsymbol{\mu}}$ and $\breve{\boldsymbol{\Sigma}}$ can be computed without inversion of $\boldsymbol{\Sigma}_x$, which means that they can be used even for deterministic inputs, i.e. when $\boldsymbol{\Sigma}_x = \mathbf{0}$ [Deisenroth, 2010].

The absolute values of the elements in $\boldsymbol{\beta}_{(d)} \in \mathbb{R}^N$ and $\mathbf{Q} \in \mathbb{R}^{N\times N}$ may be large, resulting in a loss of precision when computing $\boldsymbol{\beta}_{(d_1)}\mathbf{Q}\boldsymbol{\beta}_{(d_2)}$. If the noise variances $\sigma_{\eta,(d)}^2$ are very small, this loss of precision may result in non-positive definite marginal covariance matrices $\breve{\boldsymbol{\Sigma}}$. This ties back to the observation in Section 2.5 that we may need to lower-bound the noise variance σ_η^2 in order to avoid numerical issues.

3 Optimal Design of Experiments

Design of experiments is the task of finding experimental conditions (inputs) that yield observations (outputs) helpful in advancing the knowledge about the system being studied. Fisher [1926, 1971] (the latter originally published in 1935) did foundational work in the design of experiments field. But simple design of experiments had been used e.g. in agricultural field trials prior to this, with some success [Yates, 1964]. Classically, design of experiments has considered coming up with a design space-filling set of experiments for analysis of variance. The simplest method is to change one input at a time and observe the effect on the system's output. Fisher [1926] showed that factorial design, changing multiple experimental conditions at a time, was shown to provide a stronger basis for drawing conclusions about the input-output relationship, since it also allows the interaction between inputs to be studied.

Classical design of experiments has a number of limitations. Perhaps the biggest limitation is that the full set of experimental conditions is decided *a priori*. Not all data are equally useful [Mc Mahon et al., 2014], and many experiments risk adding little or nothing to what is already known. Experiments chosen *a priori* do not exploit knowledge gained from preceding experiments, which can result in the execution of an unnecessarily large number of expensive experiments. Classical design of experiments also do not account for constrained design spaces, where combinations of experimental conditions may be infeasible (e.g. for safety reasons).

Optimal design of experiments overcomes the limitations of classical design of experiments. Optimal design of experiments finds experimental conditions by optimising some statisti-

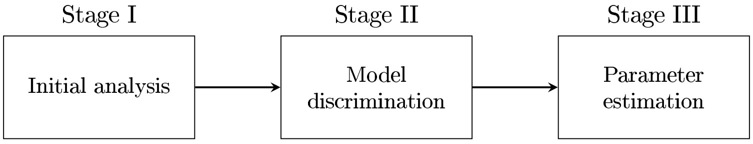


Figure 3.1: Asprey and Macchietto's [2000] model building process. After an initial analysis in Stage I, a set of likely rival model candidates are passed to Stage II, where experiments are designed to discriminate between them. The best model candidate found in Stage II is passed to Stage III where additional experiments are designed to improve the model parameter estimate.

cal design criterion. New experiments can be designed sequentially, exploiting information gained from previous experiments. Design space constraints are easily handled through bounds on the variables or constraints during optimisation.

This manuscript is concerned with optimal design of experiments for model discrimination, where additional data is required to discriminate between a set of rival mathematical models. This is Stage II in Asprey and Macchietto's [2000] model building process (see Figure 3.1). We can break down Stage II into the following steps (illustrated in Figure 3.2):

1. We start with an initial set of likely parametric model candidates, corresponding to rival hypotheses about the underlying system mechanism(s) that we wish to understand.

2. The models' parameters are estimated from existing data.

3. We perform model discrimination based on how well the models can explain existing data.

4. If more than one model remains a likely candidate, we design a new experiment to try to discriminate between them.

5. After collecting observations from the new experiment, we return to step 2. to update the model parameter estimates. This cycle continues while more than one model remains a likely candidate and the experimental budget is not exhausted.

Note that parameter estimation is also carried out in Stage II, but only to provide enough accuracy in the models to discriminate between them. In Stage III of Asprey and Macchietto's [2000] model building process, experiments are designed specifically to improve on the

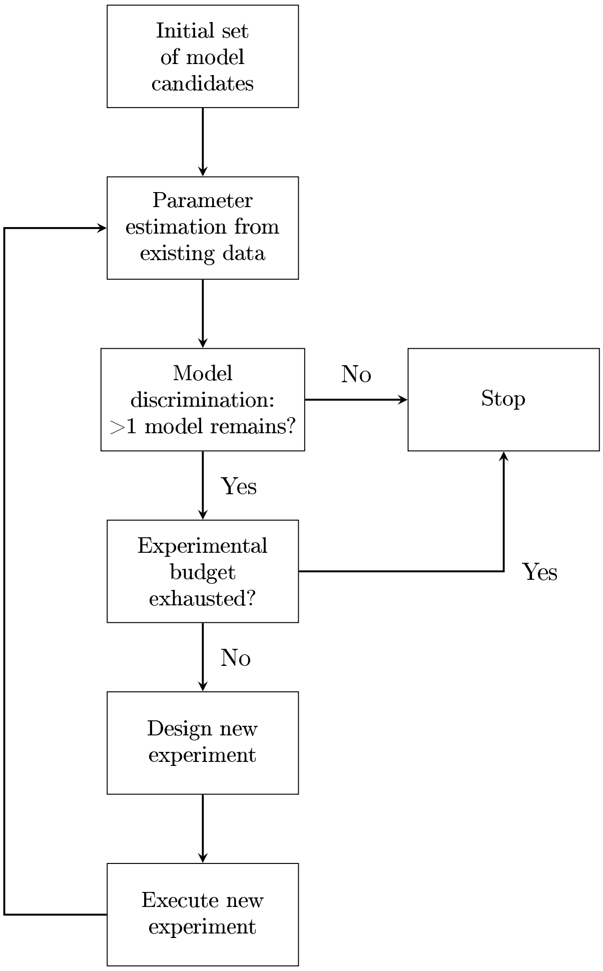

Initial set of model candidates
Parameter estimation from existing data
Model discrimination: >1 model remains?
No
Stop
Yes
Experimental budget exhausted?
Yes
No
Design new experiment
Execute new experiment

Figure 3.2: The model discrimination process.

parameter estimates.

Henceforth, the term “design of experiments” will refer to *optimal* design of experiments. Design of experiments for parameter estimation (Stage III in Figure 3.1) has received more attention in literature than design of experiments for model discrimination [Ryan et al., 2016]. Atkinson [2008], Tommasi [2009] and Waterhouse et al. [2009] investigate combined or hybrid design criteria for parameter estimation and model discrimination. Galvanin et al. [2016] use multi-objective optimisation to find designs with good trade-off between model

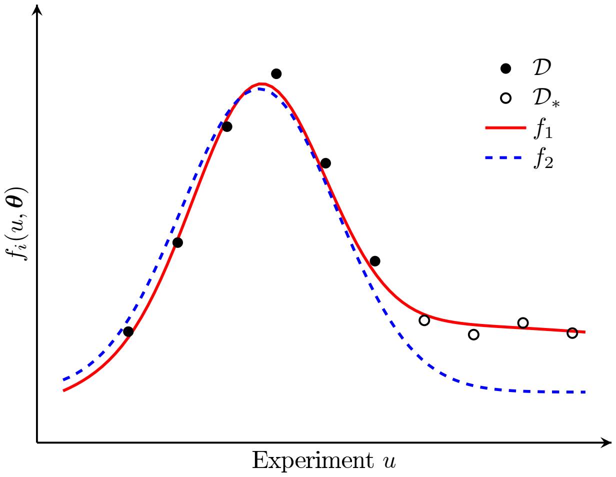


Figure 3.3: Existing experimental data $\mathcal{D}$ and two rival models f_1 and f_2 predicting the outcome of possible future experiments. Additional $\mathcal{D}_*$ reveal that model f_1 yields more accurate predictions than model f_2.

discrimination design criteria and parameter estimation design criteria. However, experimental design for both model discrimination and parameter estimation tends to perform sub-optimally in terms of model discrimination [Ryan et al., 2016]. This work mainly considers other aspects of the experimental design problem than the precise form of the objective function, hence we will focus solely on experimental design for model discrimination. But much of the work presented here is applicable to other problems in the optimal experimental design area.

3.1 Design of Experiments for Model Discrimination

The fundamental principle of design of experiments for model discrimination is selecting the next experimental point where the model predictions differ most [Hunter and Reiner, 1965]. The task is to find an appropriate measure of this difference in model predictions as a function of the model inputs that can be maximised. Figure 3.3 shows an example of two rival models f_1 and f_2. They both fit the existing experimental data $\mathcal{D}$ equally well. Additional experimental observations $\mathcal{D}_*$ reveal that model f_1 is a better candidate.

The measure of difference between model predictions should incorporate the level of confidence in said model predictions. Uncertainty in the model predictions comes mainly from uncertainty in the model parameters, which are tuned by fitting the model to noisy observations. The measurement noise is commonly assumed to be zero-mean Gaussian distributed with known (or upper-bounded) covariance $\mathbf{\Sigma}_y$ [Steimer et al., 1984; Ette and Williams, 2004; Tsimring, 2014]. Skews in the noise distribution that make the Gaussian distribution a poor approximation can often be handled, e.g. through a power transformation of the data [Box and Cox, 1964].

Let each model f_i, $i = 1, \dots, M$, assume experimental observations $\boldsymbol{y}$ given by

$$\boldsymbol{y} = f_i(\boldsymbol{u}, \boldsymbol{\theta}_i) + \boldsymbol{v} \, ,$$

for input $\boldsymbol{u}$, where $\boldsymbol{\theta}_i$ are the model parameters and $\boldsymbol{v}$ are assumed to be independent and identically (e.g. Gaussian) distributed experimental measurement noise terms. Let $\boldsymbol{u}_{1:N} = \{\boldsymbol{u}_1, \dots, \boldsymbol{u}_N\}$ and $\boldsymbol{y}_{1:N} = \{\boldsymbol{y}_1, \dots, \boldsymbol{y}_N\}$ denote the experimental conditions (inputs) and observations (outputs) from N experiments. Due to measurement noise and inherent system stochasticity, there will be uncertainty in the best-fit model parameter estimates $\hat{\boldsymbol{\theta}}_i$. This model parameter uncertainty can be accounted for at prediction by computing the marginal predictive distributions

$$p(f_i(\boldsymbol{u}) \,|\, \boldsymbol{u}, \boldsymbol{u}_{1:N}, \boldsymbol{y}_{1:N}) = \int p(f_i(\boldsymbol{u}, \boldsymbol{\theta}_i))\, p(\boldsymbol{\theta}_i \,|\, \boldsymbol{u}_{1:N}, \boldsymbol{y}_{1:N})\, \mathrm{d}\boldsymbol{\theta}_i \, . \tag{3.1.1}$$

Design of experiments for model discrimination is carried out by accounting for both uncertainty due to measurement noise $\boldsymbol{v}$ and model parameter uncertainty. The optimal next experiment is found by solving

$$\boldsymbol{u}^* = \arg\max_{\boldsymbol{u}} \mathbb{E}_{\boldsymbol{v}, \boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_M \,|\, \boldsymbol{u}_{1:N}, \boldsymbol{y}_{1:N}} \left[U(\boldsymbol{u}, f_1, \dots, f_M) \right] \, , \tag{3.1.2}$$

where $U(\cdot)$ is an appropriate statistical design criterion. However, the marginal predictive distributions in Equation (3.1.1), and hence $\mathbb{E}[U(\cdot)]$, are generally analytically intractable. In literature there are essentially two approaches for solving this problem:

- The analytical approach, where analytical and Gaussian approximations are used to find closed-form solutions for $\boldsymbol{u}^*$.
- The data-driven approach, where samples are drawn from $p(\boldsymbol{\theta}_i \,|\, \boldsymbol{u}_{1:N}, \boldsymbol{y}_{1:N})$ and used to solve for $\boldsymbol{u}^*$ using Monte Carlo techniques.

This section describes the difference between the analytical and data-driven approaches in more detail.

3.2 The Analytical Approach

Methods for tackling the design of experiments for discriminating simple, analytical models have been around for over 50 years. The term *analytical*, as used here, refers to the ability to find closed-form expressions for the gradients $\partial\xi(\boldsymbol{z})/\partial\boldsymbol{z}$ of a function ξ with respect to its input $\boldsymbol{z}$. The analytical approach to design of experiments uses a combination of linear and Gaussian approximations to find closed-form expressions for the experimental design objective function with respect to the input $\boldsymbol{u}$.

The marginal predictive distributions in Equation (3.1.1) are assumed to be approximately Gaussian

$$p(f_i(\boldsymbol{u}) \,|\, \boldsymbol{u}, \boldsymbol{u}_{1:N}, \boldsymbol{y}_{1:N}) \approx \mathcal{N}\left(f_i(\boldsymbol{u}, \hat{\boldsymbol{\theta}}_i), \breve{\Sigma}_i(\boldsymbol{u})\right) ,$$

where $\hat{\boldsymbol{\theta}}_i$ denotes the maximum *a posteriori* parameter estimate. Similarly, the measurement noise is assumed to be Gaussian distribution $\boldsymbol{v} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_y)$. In order to compute the approximate marginal covariance $\breve{\Sigma}_i(\boldsymbol{u})$, the analytical approach [Prasad and Someswara Rao, 1977; Buzzi-Ferraris et al., 1984] approximates the model parameters as being Gaussian distributed $\mathcal{N}(\hat{\boldsymbol{\theta}}_i, \boldsymbol{\Sigma}_{\theta,i})$ around $\hat{\boldsymbol{\theta}}_i$. The covariance $\boldsymbol{\Sigma}_\theta$ is given by a Laplace approximation

$$\boldsymbol{\Sigma}_{\theta,i}^{-1} \approx \sum_{n=1}^{N} \left.\frac{\partial f_i(\boldsymbol{u}_n, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}_i}\right|_{\boldsymbol{\theta}_i = \hat{\boldsymbol{\theta}}_i}^{\top} \boldsymbol{\Sigma}_y^{-1} \left.\frac{\partial f_i(\boldsymbol{u}_n, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}_i}\right|_{\boldsymbol{\theta}_i = \hat{\boldsymbol{\theta}}_i} . \tag{3.2.1}$$

Note that $\boldsymbol{\Sigma}_\theta^{-1}$ is the Fisher information matrix. If the Fisher information matrix is singular it means the model parameters poorly estimable or unidentifiable [Jacquez and Perry, 1990].

In such situations, the Laplace approximation of the model parameter uncertainty is not useful. Given an invertible Fisher information matrix, or different estimate of the model parameter covariance $\mathbf{\Sigma}_\theta$, the marginal covariance $\breve{\Sigma}_i(\boldsymbol{u})$ at input $\boldsymbol{u}$ is given by

$$\breve{\Sigma}_i(\boldsymbol{u}) = \nabla_{\boldsymbol{\theta}} f_i^\top(\boldsymbol{u}) \mathbf{\Sigma}_{\theta,i} \nabla_{\boldsymbol{\theta}} f_i(\boldsymbol{u}) \,. \tag{3.2.2}$$

Hence, model f_i predicts an experimental observation $\boldsymbol{y}$ for input $\boldsymbol{u}$ to follow the Gaussian distribution

$$\boldsymbol{y} \sim \mathcal{N}\left(f_i(\boldsymbol{u}, \hat{\boldsymbol{\theta}}_i),\, \breve{\Sigma}_i(\boldsymbol{u}) + \mathbf{\Sigma}_y \right) \,. \tag{3.2.3}$$

To simplify notation in this section, we let $\boldsymbol{f}_i = f_i(\boldsymbol{u}, \hat{\boldsymbol{\theta}}_i)$ denote the mean model prediction, and $\mathbf{\Sigma}_i = \breve{\Sigma}_i(\boldsymbol{u}) + \mathbf{\Sigma}_y$ the um of the model prediction covariance and measurement noise covariance. The dependence on the input $\boldsymbol{u}$ is implicit.

3.2.1 Model Discrimination Criteria

Given M rival models, we are interested in discarding inaccurate models. The goal is to find the "true" model that correctly explains the system we are studying. The analytical approach to design of experiments for model discrimination uses Equation (3.2.3) to compute a closed-form Gaussian approximation for the likelihood $p(\boldsymbol{y} \,|\, \boldsymbol{u})$. Given $n = 1, \dots, N$ input and observation pairs $(\boldsymbol{u}_n, \boldsymbol{y}_n)$, we wish to compute an accuracy score for each model to guide the model discrimination.

Box and Hill [1967] propose a version of the Bayes factor, i.e. the likelihood ratio of two models, for M models. Assuming sequential experiments, after the N^{th} observation is made, Box and Hill [1967] compute the normalised model posterior likelihood for model i as

$$\pi_{i,N} = \frac{\mathcal{N}(\boldsymbol{y}_N \,|\, \boldsymbol{f}_{i,N}, \mathbf{\Sigma}_{i,N}) \pi_{i,N-1}}{\sum_j \mathcal{N}(\boldsymbol{y}_N \,|\, \boldsymbol{f}_{j,N}, \mathbf{\Sigma}_{j,N}) \pi_{j,N-1}} \,, \tag{3.2.4}$$

with $\pi_{i,0} = p(f_i)$ model f_i's prior probability. Box and Hill [1967] use the normalised model posterior likelihoods $\pi_{i,N}$ to rank models against each other. The closer a model's normalised posterior likelihood is to 1, the more likely that it is the "true" model.

Buzzi-Ferraris and Forzatti [1983] criticise the normalised model posteriors $\pi_{i,N}$ by pointing out that observing the same values $\boldsymbol{y}_{1:N}$ in different order will yield different normalised model posteriors $\pi_{i,N}$, which contradicts common statistical sense. Buzzi-Ferraris and Forzatti [1983] and Buzzi-Ferraris and Manenti [2009] note that given Equation (3.2.3), under the null hypothesis ("the model is correct"), model f_i's prediction errors $\epsilon_{\text{pred.},i,n} = \boldsymbol{f}_{i,n} - \boldsymbol{y}_n$ should be zero-mean Gaussian distributed with variance $\boldsymbol{\Sigma}_y$. This means the sum of squared *normalised* prediction errors $E_{\text{pred.},i}$ should follow a $\chi^2(\ell)$-distribution

$$E_{\text{pred.},i} = \sum_{n=1}^{N} \left(\boldsymbol{f}_{i,n} - \boldsymbol{y}_n\right)^{\top} \boldsymbol{\Sigma}_y^{-1} \left(\boldsymbol{f}_{i,n} - \boldsymbol{y}_n\right) \sim \chi^2(\ell)\,,$$

with $\ell = N \times D_y - D_{\theta,i}$ degrees of freedom, with D_y the number of output dimensions and $D_{\theta,i}$ the number of model parameters for model f_i. Let χ_i^2 denote model f_i's χ^2 score

$$\chi_i^2 = \int_0^{E_{\text{pred.},i}} \chi^2(\ell, \gamma)\,\mathrm{d}\gamma\,, \tag{3.2.5}$$

where $\chi^2(\ell, \gamma)$ denotes the probability density function of the $\chi^2(\ell)$ distribution, evaluated at γ. Buzzi-Ferraris and Forzatti [1983] propose finding inaccurate models using a χ^2 test, where models are discarded if their score χ^2 score is below a given threshold, e.g. $\chi_i^2 < 10^{-3}$.

A well-known quality metric for ranking models against each other is the Akaike information criterion [De Leeuw, 1992]. For Gaussian predictive distributions, the Akaike information criterion for model i is given by

$$\text{AIC}_i = 2D_{\theta,i} - 2\sum_{n=1}^{N} \log \mathcal{N}\left(\boldsymbol{y}_n \,|\, \boldsymbol{f}_{i,n}, \boldsymbol{\Sigma}_{i,n}\right)\,.$$

The model with the lowest Akaike information criterion has the best goodness-of-fit. Akaike information criterion rewards goodness-of-fit and penalises model complexity. Using the Akaike information criterion, Michalik et al. [2010] define the relative Akaike weights ω_i for a set of M rival models as

$$\omega_i = \frac{\text{AIC}_i}{\sum_{j=1}^{M} \text{AIC}_j}\,. \tag{3.2.6}$$

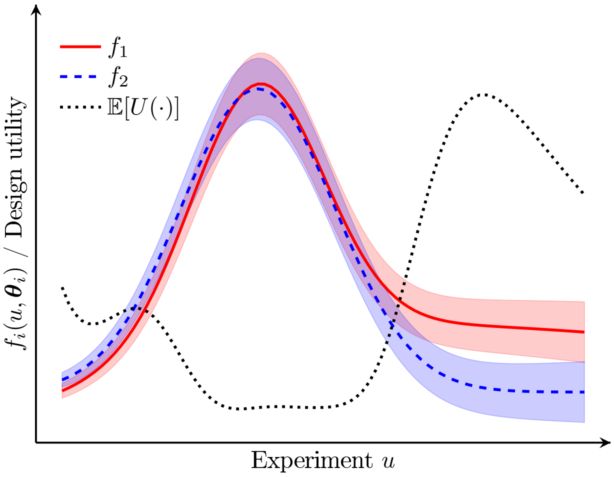


Figure 3.4: Analytical marginal predictive distribution approximations computed as in Equation (3.2.3) for two rival models f_1, f_2. The solid and dashed lines are the mean predictions, and the shaded areas denote two standard deviations. The marginal distributions are used to compute the design criterion $\mathbb{E}[U(\cdot)]$. In this example, the design criterion is D_{BH}.

The closer a model's Akaike weight ω_i is to 1, the more likely that it is the "true" model.

3.2.2 Analytical Design Criteria

To solve for $\boldsymbol{u}^*$ in Equation (3.1.2), the analytical approach proceeds to finding a closed-form design criterion $D_{**}(\cdot)$ utilising the rival models' marginal predictive means $\boldsymbol{f}_i$ and covariances $\boldsymbol{\Sigma}_i$. Figure 3.4 shows the predictive distributions $\mathcal{N}(f_i(\boldsymbol{u}, \hat{\boldsymbol{\theta}}_i), \breve{\Sigma}_i(\boldsymbol{u}) + \boldsymbol{\Sigma}_y)$ for two rival models f_1, f_2, and a corresponding design criterion $\mathbb{E}[U(\cdot)]$. The design criterion is maximised where the divergence between the marginal predictive distributions is largest.

Possibly the earliest recorded design criterion is the Mahalanobis distance between the models' predictive means, proposed by Hunter and Reiner [1965]. Let D_{HR} denote the Hunter and Reiner [1965] design criterion, which has been extended to multiple target dimensions and $M > 2$ rival models, e.g. by Espie and Macchietto [1989],

$$D_{\mathrm{HR}}(\boldsymbol{u}) = \sum_{i=1}^{M-1} \sum_{j=i+1}^{M} (\boldsymbol{f}_i - \boldsymbol{f}_j)^\top \mathbf{Q} (\boldsymbol{f}_i - \boldsymbol{f}_j) \, , \tag{3.2.7}$$

where $\mathbf{Q}$ is a diagonal scaling matrix. This design criterion is popular in many practical applications, mainly because of its simplicity. Atkinson and Fedorov [1975] define a T-optimality criterion and argue that the D_{HR} is the only design criterion that can realise T-optimal experimental designs.

Box and Hill [1967] criticise the Equation (3.2.7) design criterion for not considering parameter uncertainty and experimental noise, i.e. maximising the difference between model predictions without regard for the level of confidence in the predictions' accuracy. Instead, Box and Hill [1967] propose measuring the information gain of an additional experimental observation $\boldsymbol{y}_{N+1}$ through the change in Shannon entropy $H_{\mathrm{S},N+1} = \sum_{i=1}^{M} \pi_{i,N+1} \log \pi_{i,N+1}$, where $\pi_{i,N+1}$ is model i's normalised posterior likelihoods defined in Equation (3.2.4). From this, Box and Hill [1967] derive a new design criterion, extended to $D_y \geq 2$ output dimensions by Prasad and Someswara Rao [1977],

$$\begin{aligned} D_{\mathrm{BH}}(\boldsymbol{u}) = \sum_{i=1}^{M-1} \sum_{j=i+1}^{M} \pi_{i,N}\pi_{j,N} \Big\{ & \operatorname{tr}(\boldsymbol{\Sigma}_i \boldsymbol{\Sigma}_j^{-1} + \boldsymbol{\Sigma}_j \boldsymbol{\Sigma}_i^{-1} - 2\mathbf{I}) \\ & + (\boldsymbol{f}_i - \boldsymbol{f}_j)^\top (\boldsymbol{\Sigma}_i^{-1} + \boldsymbol{\Sigma}_j^{-1})(\boldsymbol{f}_i - \boldsymbol{f}_j) \Big\} . \end{aligned} \tag{3.2.8}$$

The design criterion in Equation (3.2.8) is the upper bound on the expected Shannon entropy change $\mathbb{E}_{\boldsymbol{y}_N}[H_{\mathrm{S},N+1}] - H_{\mathrm{S},N}$ from the next observation $\boldsymbol{y}_{N+1}$. Experiments are conducted until $\pi_{i,N} \approx 1$ for some model i, or until the experimental budget is exhausted.

Meeter et al. [1970] note that it seems strange to maximise the *upper* bound on the expected change in Shannon entropy rather than the *lower* bound. In a series of papers [Buzzi-Ferraris and Forzatti, 1983; Buzzi-Ferraris et al., 1984, 1990], another design criterion is proposed

$$\begin{aligned} D_{\mathrm{BF}}(\boldsymbol{u}) = \sum_{i=1}^{M-1} \sum_{j=i+1}^{M} \Big\{ & \operatorname{tr}\left(2\boldsymbol{\Sigma}_y (\boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j)^{-1}\right) \\ & + (\boldsymbol{f}_i - \boldsymbol{f}_j)^\top (\boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j)^{-1} (\boldsymbol{f}_i - \boldsymbol{f}_j) \Big\} . \end{aligned} \tag{3.2.9}$$

The D_{BF} design criterion is a heuristic based on the cross-covariance of different models' prediction errors. It can also be seen as a generalisation of D_{HR} that incorporates parameter uncertainty [Hoffmann, 2017, p. 5].

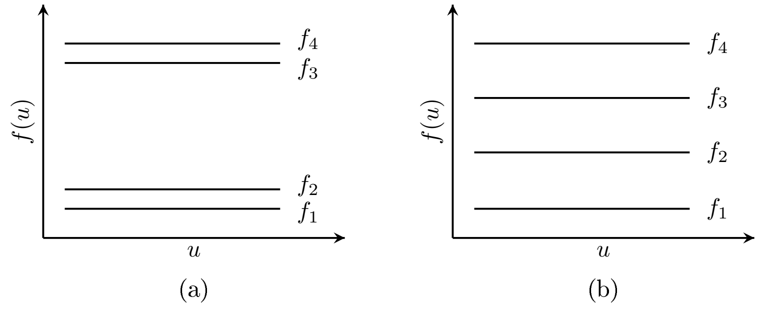


Figure 3.5: (a) An example of model aggregation, where $\{f_1,\ f_2\}$ and $\{f_3,\ f_4\}$ pairwise yield similar predictions. Given data, we can discriminate between groups of model pairs $\{f_1,\ f_2\}$ and $\{f_3,\ f_4\}$ but may be unable to discriminate between models intra-pair. (b) No model aggregation. Model aggregation may affect model discrimination [Michalik et al., 2010].

[Buzzi-Ferraris and Forzatti, 1983; Buzzi-Ferraris et al., 1984, 1990] use the χ^2 test for model discrimination. Experiments are conducted until only one model remains, all models have been discarded, or the experimental budget is exhausted. For comparing two models ($M = 2$), the D_{BF} design criterion also has the interpretation that if $D_{\mathrm{BF}}(\boldsymbol{u})$ is not "sufficiently larger than 1", the models cannot be discriminated [Buzzi-Ferraris and Forzatti, 1983; Buzzi-Ferraris, 2010], which can be used as a stopping criterion. Schwaab et al. [2006] derive a similar design criterion to D_{BF} consisting of the second term inside the sums in Equation (3.2.9) weighted by each model's prediction error χ^2 probability.

Michalik et al. [2010] argue that the design criteria D_{BH} in Equation (3.2.8) and D_{BF} in Equation (3.2.9) reward model *lumping*, i.e. model aggregation, where the predictions of some models are similar but far apart from predictions of other model aggregations (see Figure 3.5). An observation at a point with model aggregation may determine whether a group of models are more accurate than another group of models, but does not discriminate between models within each of those groups. This is an interesting engineering trade-off: the user may wish to identify the single best model, but significantly discriminating between model groups may be more practical than partially discriminating between many models [Buzzi-Ferraris, 2010].

To avoid sampling at model aggregation points, Michalik et al. [2010] draw on the Akaike information criterion weights ω_i defined in Equation (3.2.6) to construct the heuristic design criterion

$$D_{\mathrm{AW}}(\boldsymbol{u}) = \sum_{i=1}^{M} \frac{p(f_i)}{\sum_{j=1}^{M} \exp\left(-\frac{1}{2}(\boldsymbol{f}_i - \boldsymbol{f}_j)^\top \boldsymbol{\Sigma}_i^{-1}(\boldsymbol{f}_i - \boldsymbol{f}_j) + D_{\theta,i} - D_{\theta,j}\right)} , \tag{3.2.10}$$

with $p(f_i)$ model f_i's prior probability. Experiments are conducted until $\omega_i \approx 1$ for some model i or until the experimental budget is exhausted.

3.2.3 Limitations of the Analytical Approach

Most industrially relevant models are not analytical. They are often, from a practical point-of-view, complex black boxes of legacy code, e.g. large systems of ordinary or partial differential equations. We can evaluate the models f_i, but derivatives with respect to the input $\boldsymbol{u}$ and model parameters $\boldsymbol{\theta}_i$ are not readily available. When the model is noisy, e.g. due to discretisation or sampling, gradients may be meaningless.

Specialist software can retrieve the gradients from systems of ordinary or partial differential equations, but this requires implementing and maintaining the models in said specialist software packages. The number of function evaluations needed for finite-difference gradient approximation may also be computationally prohibitive. Automatic differentiation [Neidinger, 2010; Farrell et al., 2013; Baydin et al., 2018] can be used to retrieve gradient information from some models, but will not work e.g. for models with non-smoothness and discontinuities [Conn et al., 2009, pp. 3–5; Martelli and Amaldi, 2014; Boukouvala et al., 2016]. These may be due to switches (`if`/`else` statements) or mathematical models that require solving an optimisation problem. For model discrimination, it is desirable to be agnostic with regards to the software implementation or model type, since this flexibly (i) allows faster model prototyping and development, and (ii) satisfies the personal preferences of researchers and engineers.

3.3 The Data-Driven Approach

We use the term *data-driven* to refer to approaches that do not rely on closed-form approximations for the models' marginal predictive distributions. Typically the data-driven methods rely on Monte Carlo techniques, and attempt to solve the design of experiments objective function in Equation (3.1.2) directly, without resorting to Gaussian approximations for the distributions involved. Data-driven methods to accommodate non-analytical models have developed in parallel with increasing computer speed. These methods are typically closer to fully Bayesian than the classical analytical methods, e.g. Liepe et al. [2013] and Dony et al. [2017]. Statisticians have often focused on design of experiments (solving the optimisation problem) and model discrimination (the form of the optimisation problem) separately [Chaloner and Verdinelli, 1995]. Criteria for model discrimination are handled separately, usually under the name of model selection or hypothesis testing.

Vanlier et al. [2014] approximate the marginal predictive distributions of M models and their Jensen-Shannon divergence using Markov chain Monte Carlo (MCMC) sampling and a k-nearest neighbours density estimate. This density estimates become less accurate as the number of experimental observations $\boldsymbol{y}_{1:N}$ increases [Vanlier et al., 2014] and the method is computationally intensive [Ryan et al., 2016].

Ryan et al. [2015] use a Laplace approximation of the posterior model parameter distribution $p(\boldsymbol{\theta} \,|\, \mathcal{D},\, f)$ combined with importance sampling. Drovandi et al. [2014] develop a method based on sequential Monte Carlo (SMC) that is faster than using MCMC. Woods et al. [2017] use a Monte Carlo approximation of $\mathbb{E}[U(\cdot)]$ on which they place a Gaussian process prior, and maximise $\mathbb{E}[U(\cdot)]$ using Bayesian optimisation.

Figure 3.6 shows an example of data-driven design of experiments. Samples $\tilde{\boldsymbol{\theta}}_{i,j}$, $j = 1, \dots, N_{\text{sample}}$ are drawn from the model parameter distributions $p(\boldsymbol{\theta}_i \,|\, \mathcal{D},\, f_i)$ of two rival models f_1 and f_2. The models are evaluated for the sampled model parameter values $\tilde{\boldsymbol{\theta}}_{i,j}$ and inputs u_j, and the corresponding predictions $y_{i,j}$ used to approximate the design

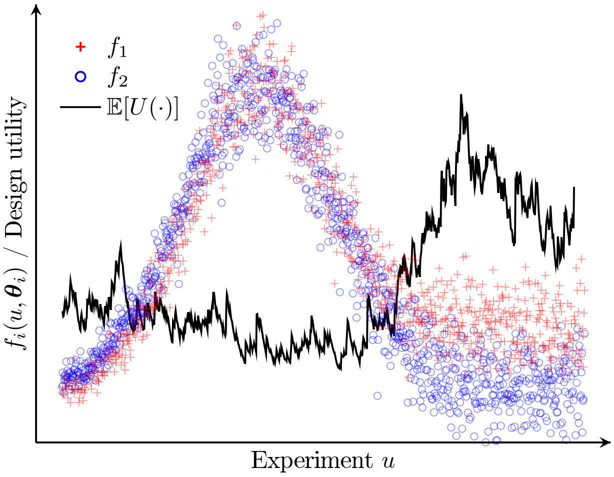


Figure 3.6: Samples are drawn from the model parameter distributions of two rival models f_1 and f_2. The models are evaluated for the sampled model parameter values and inputs u, and the predictions used to approximate the design criterion $\mathbb{E}[U(\cdot)]$ in Equation (3.3.1).

criterion $\mathbb{E}[U(\cdot)]$. In this toy example the design criterion is computed as a moving average

$$\mathbb{E}[U(u_j)] = \frac{1}{2n} \sum_{k=j-n}^{j+n} \frac{(y_{1,j} - y_{2,j})^2}{\sigma_{1,j}^2 + \sigma_{2,j}^2}, \tag{3.3.1}$$

with approximated variances $\sigma_{i,j}^2 = \mathbb{V}[y_{i,j-n}, \dots, y_{i,j+n}]$ and window size $n = 10$. The inputs u_j are assumed to be ordered, such that $u_{j-1} \leq u_j \leq u_{j+1}$. As the sample size N_{sample} increases, the approximated design criterion is expected to converge on the true design criterion.

3.3.1 Limitations of the Data-Driven Approach

These methods use a Monte Carlo-approach to agnostically accommodate non-analytical models but require exhaustive model sampling in the model parameter space. On a case study with four models, each with ten model parameters, and two design variables, the Vanlier et al. [2014] MCMC method requires six days of wall-clock time to compute two sets of four marginal predictive distributions. This cost is impractical for pharmacokinetic applications, where a physical experiment takes approximately 1–4 days. SMC methods can converge faster than MCMC methods [Woods et al., 2017]. But SMC methods can suffer from sample degeneracy, where only a few particles receive the vast majority of the probability weight [Li et al., 2014]. Also, convergence analysis for MCMC and SMC methods is difficult.

Furthermore, these methods are only for design of experiments. Once an experiment is executed, the model discrimination issue remains. In this case the marginal predictive distributions $p(f_i(\boldsymbol{u}) \,|\, \boldsymbol{u}, \boldsymbol{u}_{1:N}, \boldsymbol{y}_{1:N})$ would enable calculating the model likelihoods.

3.4 Discussion

The optimal design of experiments for model discrimination literature has focused on either classical analytical approaches or Monte Carlo-based approaches. The former is computationally cheap but limited in the model structures and approximations it can accommodate, whereas the latter is flexible and accurate but may be computationally expensive. Hence, there is a trade-off between flexibility, accuracy and computational speed. The same goes for model discrimination, where the analytical approach uses closed-form model discrimination criteria and the data-driven approach has to rely on e.g. Monte Carlo approximations.

Three different model discrimination criteria were introduced in this chapter. We choose to refer to the process of (ideally) finding the “true” model as *model discrimination*. Another common term is *model selection*. Some researchers distinguish between these two for an important reason: The Box and Hill [1967] normalised model posterior likelihoods π_i and the Michalik et al. [2010] Akaike weights ω_i ranks models against each other and *selects* the model with the highest score—implicitly they are rejecting all but the least inaccurate model; The Buzzi-Ferraris and Forzatti [1983] method of using the χ^2 test *discriminates* between models by discarding models that inadequately describe the observed data. The normalised model likelihoods and Akaike weights can vary significantly between consecutive experiments, especially when there are few available observations. Using the χ^2 increases robustness against inaccurate models, usually at the cost of having to run more experiments. This will be demonstrated in later chapters. The χ^2 is the most commonly used test in practice.

Some research, e.g. Quaglio et al. [2018], in the area of design of experiments for *parameter estimation* has considered structural model uncertainty. In the parameter estimation setting the model structure is fixed, but an incorrect model structure assumption may negatively impact the accuracy of the model parameters (assuming some "true" parameter values exist, e.g. a stoichiometric coefficient or reaction rate). However, for model discrimination the goal is to identify the most accurate model structure. Hence, work in this area does not consider structural uncertainty.

4 Jensen-Rényi Design Criterion

This chapter is based on Sections 3, 6 and 7 of Olofsson et al. [2019a].

The fundamental principle of sequential experimental design for model discrimination says to select the next experimental point where the model predictions differ the most [Hunter and Reiner, 1965]. The measure of how much the models differ is the *design criterion* and the design optimisation problem maximises this design criterion.

The D_{HR}, D_{BH}, D_{BF} and D_{AW} design criteria were introduced in Section 3.2.

- The D_{HR} criterion is the sum of pairwise squared distances between the means of the rival marginal predictive distributions.
- The D_{BH} criterion is an upper bound on the change in Jensen-Shannon entropy from making an additional observation.
- The D_{BF} criterion is a heuristic based on the cross-covariance between the models' prediction errors.
- The D_{AW} criterion is a heuristic based on Akaike's information criterion.

All four design criteria D_{BH}, D_{BF} and D_{AW} implicitly reward divergent model predictive distributions. This chapter introduces a novel design criterion D_{JR} based on the quadratic Jensen-Rényi divergence. The D_{JR} design criterion explicitly maximises the divergence between model predictive distributions.

4.1 Jensen-Rényi Divergence

The general expression for the divergence between $i = 1, \ldots, M$ predictive distributions $g_i(\boldsymbol{u})$ for design $\boldsymbol{u}$ is

$$\mathrm{Div}[H](\boldsymbol{u}) = H\left(\sum_{i=1}^{M} \varpi_i g_i(\boldsymbol{u})\right) - \sum_{i=1}^{M} \varpi_i H(g_i(\boldsymbol{u})) \,, \tag{4.1.1}$$

where ϖ_i are weights associated with the corresponding models, and H is some entropy measure. Vanlier et al. [2014] propose a design criterion based on the Jensen-Shannon divergence between model predictive distributions. The Jensen-Shannon divergence $\mathrm{Div}[H_\mathrm{S}]$ is the divergence measure corresponding to the continuous Shannon entropy H_S, or differential entropy. For a distribution $G(\cdot)$, the Jensen-Shannon divergence is defined by

$$H_\mathrm{S}(G) = -\int G(\boldsymbol{\gamma}) \log G(\boldsymbol{\gamma}) \mathrm{d}\boldsymbol{\gamma} \,,$$

with information measured in natural units (logarithm base e). Computing $\mathrm{Div}[H_\mathrm{S}](\boldsymbol{u})$ as in Equation (4.1.1) is intractable, even for the case of Gaussian distributions $g_i(\boldsymbol{u}) = \mathcal{N}(\boldsymbol{f}_i, \boldsymbol{\Sigma}_i)$. Hence, we can either approximate the divergence measure using Monte Carlo techniques [Vanlier et al., 2014], or find a different entropy measure that yields a closed-form solution for the divergence of Gaussian distributions. Given the computational complexity of the former, we choose the latter option and turn to the Rényi [1965] entropy measure, a generalisation of the Shannon entropy. Specifically, we look at the quadratic Rényi entropy H_2 defined as

$$H_2(G) = -\log \int G(\boldsymbol{\gamma})^2 \mathrm{d}\boldsymbol{\gamma} \,.$$

For a Gaussian distribution $g_i(\boldsymbol{u}) = \mathcal{N}(\boldsymbol{f}_i, \boldsymbol{\Sigma}_i)$, $\boldsymbol{f}_i \in \mathbb{R}^{D_y}$,, the quadratic Rényi entropy is given by

$$H_2(g_i(\boldsymbol{u})) = \tfrac{D_y}{2} \log(4\varpi) + \tfrac{1}{2} \log |\boldsymbol{\Sigma}_i| \,.$$

For a mixture of Gaussian distributions $\hat{G} = \sum_i \varpi_i g_i(\boldsymbol{u})$, the quadratic Rényi entropy is given by [Wang et al., 2009; Nielsen, 2012]

$$H_2(\hat{G}) = -\log \sum_{i=1}^{M} \sum_{j=1}^{M} \frac{\varpi_i \varpi_j}{(2\varpi)^{D_y/2}} \exp\left(-\tfrac{1}{2}\phi_{ij}(\boldsymbol{u})\right) \tag{4.1.2}$$

where, using $\tilde{\boldsymbol{f}}_{ij} = \boldsymbol{\Sigma}_i^{-1}\boldsymbol{f}_i + \boldsymbol{\Sigma}_j^{-1}\boldsymbol{f}_j$, the function $\phi_{ij}(\boldsymbol{u})$ is given by

$$\begin{aligned}\phi_{ij}(\boldsymbol{u}) = \boldsymbol{f}_i^\top \boldsymbol{\Sigma}_i^{-1}\boldsymbol{f}_i + \boldsymbol{f}_j^\top \boldsymbol{\Sigma}_j^{-1}\boldsymbol{f}_j - \tilde{\boldsymbol{f}}_{ij}^\top \left(\boldsymbol{\Sigma}_i^{-1} + \boldsymbol{\Sigma}_j^{-1}\right)^{-1} \tilde{\boldsymbol{f}}_{ij} \\ + \log|\boldsymbol{\Sigma}_i| + \log|\boldsymbol{\Sigma}_j| + \log|\boldsymbol{\Sigma}_i^{-1} + \boldsymbol{\Sigma}_j^{-1}| \,.\end{aligned}$$

By noting that $\phi_{ii}(\boldsymbol{u}) = \log|2\boldsymbol{\Sigma}_i|$, we rewrite the expression in Equation (4.1.2) as

$$H_2(\hat{G}) = \tfrac{D_y}{2}\log(2\varpi) - \log \sum_{i=1}^{M} \left[\frac{\varpi_i^2}{2^{D_y/2}|\boldsymbol{\Sigma}_i|^{\frac{1}{2}}} + 2\sum_{j=1}^{i-1} \varpi_i \varpi_j \exp\left(-\tfrac{1}{2}\phi_{ij}(\boldsymbol{u})\right) \right] ,$$

which reduces computational effort. After these reformulations, the quadratic Jensen-Rényi divergence $\mathrm{Div}[H_2](\boldsymbol{u})$ has a closed-form expression. We let

$$D_{\mathrm{JR}}(\boldsymbol{u}) = \mathrm{Div}[H_2](\boldsymbol{u}) \,, \tag{4.1.3}$$

denote the quadratic Jensen-Rényi divergence design criterion.

The weights ϖ_i are not uniquely defined in D_{JR}. A natural way to define the weights would be as the model likelihoods $\varpi_i = p(f_i \,|\, \mathrm{data})$. This is similar to the normalised model likelihoods $\pi_{i,N}$ of Box and Hill [1967] or the Akaike weights ω_i of Michalik et al. [2010]. Another possibility is to let the weights ϖ_i act as switches that determine whether models are still considered likely candidates. This is the approach of Buzzi-Ferraris and Forzatti [1983], setting $\varpi_i = 1$ if a model f_i passes the χ^2 test and $\varpi_i = 0$ otherwise. Schwaab et al. [2006] use the χ_i^2 scores in Equation (3.2.5) as weights. The D_{JR} design criterion is not derived from a specific method of model discrimination, but from the fundamental principle of maximising model predictive divergence. Thus, it is agnostic to the methods used for model weighting and discrimination.

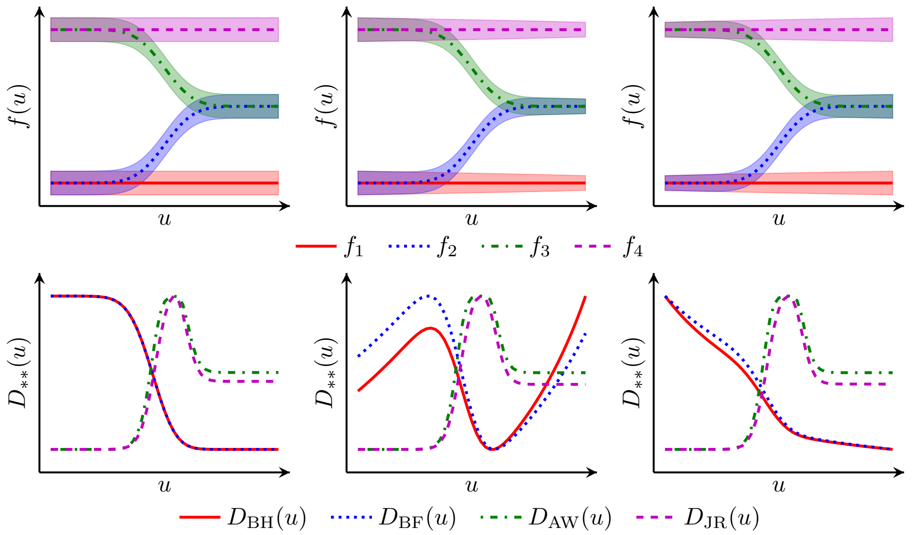


Figure 4.1: Comparison of discrimination criteria (bottom row) for three different sets of predictive distributions (top row): constant variance (left), linearly decreasing variance (centre) and linearly increasing variance (right).

4.2 Design Criteria Trade-Offs

Figure 4.1 compares the different design criteria (bottom row) for three sets of different predictive distributions (top row) for four equally weighted models ($\forall\, i : \pi_i = 1/4$). The means $f_i(u)$ of the predictive distributions are the same in each plot, but the variance $\sigma_i^2(u)$ changes. The design criteria have all been normalised to lie in the range [0, 1].

In the left-most Figure 4.1 column, the variance is constant. The top plot shows a model aggregation example. The D_{BH} and D_{BH} criteria prefer a small u, where $\{f_1, f_2\}$ and $\{f_3, f_4\}$ pairwise yield identical predictions, but the divergence between e.g. f_1 and f_3 is large. For constant covariance, i.e. where the predictive covariance is effectively independent of u, as in the left-most example in Figure 4.1, the D_{BH} and D_{BF} design criteria are equal to D_{HR} multiplied by a constant. The D_{AW} and D_{JR} criteria, on the other hand, are maximised for medium u, where all model predictive distributions are divergent. In the middle Figure 4.1 column, the variance decreases linearly with u. In the right-most column, the variance increases linearly. In the middle and right-most columns, the change in the

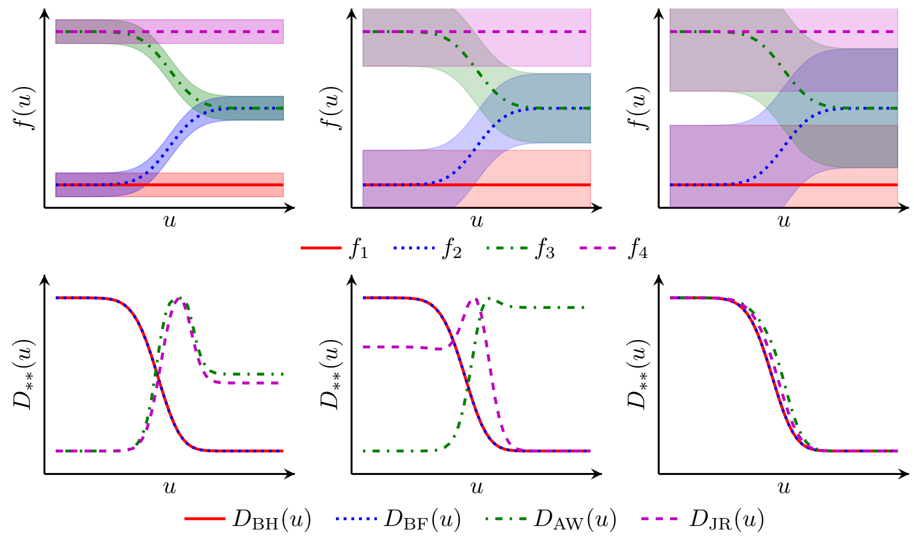


Figure 4.2: Comparison of discrimination criteria (bottom row) for three different sets of predictive distributions (top row) with constant covariance increasing from left to right.

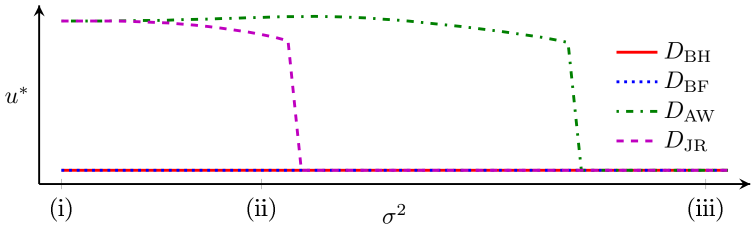


Figure 4.3: Optimal design $u^* = \arg\max_u D_{**}(u)$ when variance σ^2 increases for rival models in Figure 4.2. Variances corresponding to the (i) left-most, (ii) centre and (iii) right-most plots of Figure 4.2 are marked on the σ^2 axis. D_{BH} and D_{BF} are on top of each other.

variance significantly impacts the D_{BH} and D_{BF} criteria maxima. The D_{AW} and D_{JR} criteria consistently favour a medium u, i.e. aim for complete model discrimination.

In all three cases in Figure 4.1, none of the predictions $f_i(u) \pm \sigma_i(u)$ overlap for medium-sized u. Using Michalik et al.'s (2010) model aggregation-based reasoning, placing the next experiment at moderate u yields complete model disaggregation. However, with increasing model prediction uncertainty, the attractiveness of an experiment in the centre decreases, as shown in Figure 4.2 for the constant variance scenario. With increasing uncertainty, the D_{JR} criterion peak shifts to smaller u before the D_{AW} criterion peak (see Figure 4.3).

We note some similarity between the D_{AW} and D_{JR} design criteria. The D_{AW} and D_{JR} design criteria contain inverse exponential terms with the weighted squared differences between model predictions. These terms likely dominate in the small-variance scenario in Figure 4.1. However, the D_{JR} design criterion also contains pure covariance terms, similar to the D_{BH} and D_{BF} criteria. The D_{AW} criterion does not have these terms. This is why, as the covariance grows large, the peak of the D_{JR} criterion shifts to smaller u before D_{AW} in Figure 4.3.

Buzzi-Ferraris [2010] argue that complete discrimination between groups of models is preferable to partial discrimination between all models. Figure 4.1 and 4.2 show that the design criteria represent different trade-offs between the risk of partial discrimination, and the reward of complete discrimination.

The design criteria are normalised to lie in the $[0, 1]$ range for each plot independently. Hence, the maximum value of the design criteria in the left-most and right-most plots in Figure 4.2 are not the same, i.e. all design criteria yield higher values for small u in the left-most plot over small u in the right-most plot.

4.3 Comparison of Design Criteria Performance

In this section, the performance of the new design criterion D_{JR} is compared to the existing design criteria in literature for an ammonia synthesis case study. We ran a large number of simulations to gather relevant performance statistics. Every simulation has a different noise realisation. In each simulation, we follow the model discrimination process in Figure 3.2, repeated here as Figure 4.4 for convenience.

4.3.1 Performance Metrics

Let DC and MD denote the chosen design criterion and method of model discrimination, respectively. Section 3.2.1 describes three different methods of model discrimination:

- Normalised Gaussian posteriors $\pi_{i,N}$ with updates [Box and Hill, 1967]. The procedure terminates when $\exists i : \pi_{i,N} \geq 0.999$.

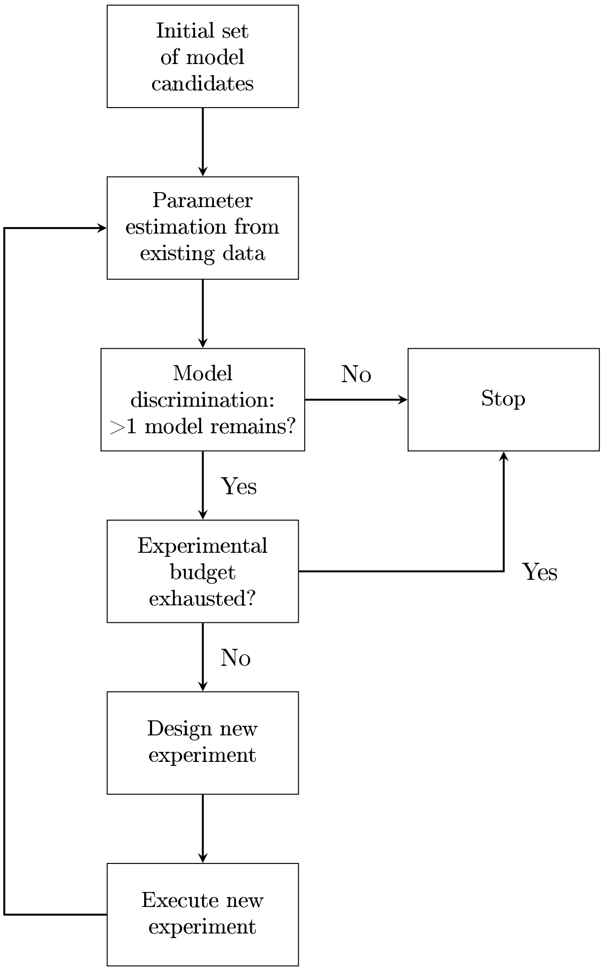

Initial set of model candidates
Parameter estimation from existing data
Model discrimination: >1 model remains?
No
Stop
Yes
Experimental budget exhausted?
Yes
No
Design new experiment
Execute new experiment

Figure 4.4: The model discrimination process (repeated Figure 3.2).

- χ^2 test [Buzzi-Ferraris and Forzatti, 1983], where a model i is deemed inadequate and discarded if $\chi_i^2 \leq 0.01$ for $N \cdot D_y - D_{\theta,i}$ degrees of freedom, with $N = |\mathcal{D}|$ the number of available data points.

- Akaike weights ω_i [Michalik et al., 2010]. The procedure terminates when $\exists i : \omega_i \geq 0.999$.

Alternatively, simulation terminates after reaching the maximum number of additional experiments.

For a given case study and combination of design criterion and method of model discrimination, let $\hat{a}_\ell$ denote the number of experiments executed before simulation ℓ terminated. A simulation terminates either when only one model candidate remains, all models have been discarded, or the experimental budget is exhausted. Let f_{true} denote the data-generating ("true") models, and let $\hat{f}_\ell \in \{f_1, \dots, f_M, \emptyset\}$ denote the result of the simulation, with $\hat{f}_\ell = \emptyset$ if all models have been discarded or the experimental budget is exhausted. Using $\hat{a}_\ell$ and $\hat{f}_\ell$ we define the set of *successful* simulations $\mathcal{S}$

$$\mathcal{S} = \{\ell \,:\, \hat{f}_\ell = f_{\text{true}}\} ,$$

the set of *failed* simulations $\mathcal{F}$

$$\mathcal{F} = \{\ell \,:\, \hat{f}_\ell \neq f_{\text{true}} \,\wedge\, \hat{f}_\ell \neq \emptyset\} ,$$

and the set of *inconclusive* simulations $\mathcal{I}$

$$\mathcal{I} = \{\ell \,:\, \hat{f}_\ell = \emptyset\} .$$

From this we define the statistics in Table 4.1: the average number of additional experiments required for *successful* model discrimination

$$\mathrm{A} = \frac{1}{|\mathcal{S}|} \sum_{\ell \in \mathcal{S}} \hat{a}_\ell , \tag{4.3.1}$$

the standard error of A

$$\mathrm{SE} = \sqrt{\frac{1}{|\mathcal{S}|(|\mathcal{S}|-1)} \sum_{\ell \in \mathcal{S}} (\hat{a}_\ell - \mathrm{A})^2} , \tag{4.3.2}$$

and the success (S), failure (F) and inconclusive (I) rates

$$\mathrm{S} = \frac{|\mathcal{S}|}{|\mathcal{S}| + |\mathcal{F}| + |\mathcal{I}|} , \quad \mathrm{F} = \frac{|\mathcal{F}|}{|\mathcal{S}| + |\mathcal{F}| + |\mathcal{I}|} , \quad \mathrm{I} = \frac{|\mathcal{I}|}{|\mathcal{S}| + |\mathcal{F}| + |\mathcal{I}|} . \tag{4.3.3}$$

For good performance, the average A should be as low as possible. A small SE value indicates

A	the average number of additional experiments required for *successful* model discrimination (Equation (4.3.1)), i.e. identifying the correct model as the data-generating model.
SE	the standard error of the average number A of additional experiments (Equation (4.3.2)).
S	the success rate, i.e. the percentage of simulations in which the correct model was identified as the data-generating model (Equation (4.3.3)).
F	the failure rate, i.e. the percentage of simulations in which a model other than the correct model was identified as the data-generating model (Equation (4.3.3)).
I	the rate of inconclusive simulations, i.e. the percentage of simulations in which more than one model–or no models in the case of the χ^2 test–remain when the experimental budget has been exhausted (Equation (4.3.3)).

Table 4.1: Statistics collected in the simulations.

the estimated A is close to the "true" average. The success rate S should be close to 100 %. An inconclusive result (true or false negative) is preferable to a failed result (false positive), since selecting an inaccurate model can incur a large cost at a later stage.

4.3.2 Case Study 1: Ammonia Synthesis

The first case study considers four different models for synthesis of ammonia (NH_3) from hydrogen (H_2) and nitrogen (N_2) [Buzzi-Ferraris et al., 1990]. There are $D_{\theta,i} \in \{2,4,6\}$ parameters per model, $D_y = 1$ observable output, and $D_x = 3$ design variables: pressure $P \in [300\,\mathrm{atm},\ 350\,\mathrm{atm}]$, temperature $T \in [703\,\mathrm{K},\ 753\,\mathrm{K}]$ and inlet ammonia mole fraction $\chi_{\mathrm{NH_3}} \in [0.1,\ 0.2]$. Each simulation has $N_0 = 5$ initial measurements and a maximum budget of 40 new experiments. The models are given by

$$\text{Model 1}: \quad f_1 = \frac{\phi_{\mathrm{N_2}} - \phi_{\mathrm{NH_3}}/(\phi_{\mathrm{H_2}}^3 K_{\mathrm{eq}}^2)}{C_1 \phi_{\mathrm{NH_3}}/\phi_{\mathrm{H_2}}^{3/2}}\,,$$

$$\text{Model 2}: \quad f_2 = \frac{\phi_{\mathrm{N_2}}\phi_{\mathrm{H_2}} - \phi_{\mathrm{NH_3}}/(\phi_{\mathrm{H_2}} K_{\mathrm{eq}})^2}{C_1 \phi_{\mathrm{NH_3}}}\,,$$

$$\text{Model 3}: \quad f_3 = \frac{\phi_{\mathrm{N_2}}^{1/2}\phi_{\mathrm{H_2}}^{3/2} - \phi_{\mathrm{NH_3}}/K_{\mathrm{eq}}}{C_1 \phi_{\mathrm{NH_3}} + C_2(\phi_{\mathrm{N_2}}/\phi_{\mathrm{H_2}})^{1/2}}\,,$$

$$\text{Model 4 :} \quad f_4 = \frac{\phi_{\mathrm{N_2}}^{1/2}\phi_{\mathrm{H_2}}^{3/2} - \phi_{\mathrm{NH_3}}/K_{\mathrm{eq}}}{C_1\phi_{\mathrm{NH_3}} + C_2\phi_{\mathrm{N_2}} + C_3\phi_{\mathrm{NH_3}}/\phi_{\mathrm{N_2}}} \,,$$

where the fugacities are given by $\phi_s = P\chi_s\gamma_s$ for $s \in \{\mathrm{H_2}, \mathrm{N_2}, \mathrm{NH_3}\}$. We assume inert-free, stoichiometric reaction, which gives the mole fractions $\chi_{\mathrm{N_2}} = \frac{1}{4}(1 - \chi_{\mathrm{NH_3}})$ and $\chi_{\mathrm{H_2}} = 3\chi_{\mathrm{N_2}}$. The activity coefficients γ_s for the reaction are given by [Dyson and Simon, 1968]:

$$\begin{aligned}\gamma_{\mathrm{H_2}} = \exp \Bigg[& P \exp\left(0.541 - 3.8402 \cdot T^{0.125}\right) \\ & - P^2 \exp\left(-15.98 - 0.1263 \cdot T^{0.5}\right) \\ & + \frac{300\,(\exp(-P/300) - 1)}{\exp\,(5.941 + 0.011901 \cdot T)} \Bigg] \,,\end{aligned} \tag{4.3.4}$$

$$\begin{aligned}\gamma_{\mathrm{N_2}} = {} & 0.93431737 + 3.101804\text{E-}4 \cdot T + 2.958960\text{E-}4 \cdot P \\ & - 2.707279\text{E-}7 \cdot T^2 + 4.775207\text{E-}7 \cdot P^2 \,,\end{aligned} \tag{4.3.5}$$

$$\begin{aligned}\gamma_{\mathrm{NH_3}} = {} & 0.14389960 + 2.028538\text{E-}3 \cdot T - 4.487672\text{E-}4 \cdot P \\ & - 1.142945\text{E-}6 \cdot T^2 + 2.761216\text{E-}7 \cdot P^2 \,.\end{aligned} \tag{4.3.6}$$

The thermodynamic equilibrium constant K_{eq} is given by [Gillespie and Beattie, 1930]:

$$\begin{aligned}\log_{10} K_{\mathrm{eq}} = {} & 2.6899 - 2.691122 \log_{10} T - 5.519265\text{E-}5 \cdot T \\ & + 1.848863\text{E-}7 \cdot T^2 + 2001.6/T \,.\end{aligned} \tag{4.3.7}$$

The model parameters appear in the coefficients $C_j = \exp\left(\theta_{j1} - \theta_{j2}\frac{T-700}{T}\right)$. The bounds on the model parameters are $\theta_{j1} \in [0.1,\, 10]$ and $\theta_{j2} \in [0.1,\, 100]$. We follow [Buzzi-Ferraris et al., 1990] by generating experimental data from model 1 with $\boldsymbol{\theta} = [3.68,\, 11.8]$ and experimental noise variance $\boldsymbol{\Sigma}_y = \sigma_y^2 = 90$.

Table 4.2 compares the novel design criterion D_{JR} to the classical design criteria D_{BH}, D_{BF} and D_{AW} for the ammonia synthesis case study. The comparison also includes the alternative of not optimising the design, but randomly and uniformly sampling the next experimental design, denoted U. We use the analytical expressions in Section 3.2 to approximate the models' marginal predictive distributions $\mathcal{N}(\boldsymbol{f}_i, \boldsymbol{\Sigma}_i)$. Table 4.2 shows the simulation performance statistics from 100 sets of random initial measurements.

MD	$\pi_{i,N}$			χ_i^2			ω_i		
DC	D_{BH}	D_{JR}	U	D_{BF}	D_{JR}	U	D_{AW}	D_{JR}	U
A	20.85	22.24	34.50	20.56	21.12	14.50	7.11	6.61	21.25
SE	0.82	0.72	1.77	1.43	1.22	3.02	0.47	0.49	1.08
S [%]	81	87	2	81	84	10	100	100	73
F [%]	0	0	0	1	1	1	0	0	2
I [%]	19	13	98	18	15	89	0	0	25

Table 4.2: Comparison of design criteria performance for the ammonia synthesis case study. The D_{JR} design criterion is compared to the classical design criteria D_{BH}, D_{BF} and D_{AW} for their corresponding model discrimination methods (see Section 3.2). The columns U uniformly sample the next experimental design rather than optimising the criterion.

For this case study, the new design criterion D_{JR} performs similarly to the classical criteria: D_{JR} has a higher average number of additional experiments (A) than D_{BH} and D_{BF}, but also a higher success rate (S). Compared to D_{AW}, D_{JR} has a lower average A. In all cases, the difference between the criteria's averages A is less than the sum of their standard errors SE.

The random design selection U results are a sanity check: the success rate is significantly lower while the inconclusive rate is higher. For the $\pi_{i,N}$ model discrimination method, random design selection succeeded in only 2 simulations. For the χ^2 model discrimination method, the average A is lower for the random design selection than for D_{BF} and D_{JR}, due to a low success rate: random design selection only succeeded for the easier simulations. Making an informed decision for the next experimental design is obviously beneficial to reduce the number of extra experiments needed for model discrimination.

4.4 Discussion

We have derived the novel design criterion D_{JR} and discussed trade-offs between different experimental designs. In our experiments, the different design criteria sometimes perform similarly and (in those cases) the model discrimination method has a large impact on the success rate or the number of additional experiments required. Another important consideration is parameter estimation: We may accidentally discard the data-generating model due to poor parameter estimation.

Model indiscriminability is a major hurdle for model discrimination. Parametric models may be very flexible, spanning a large part of the target space depending on the specific model parameter values. Useful stopping criteria for design of experiments, e.g. Buzzi-Ferraris and Forzatti's (1983) criterion $D_{\mathrm{BF}}(\boldsymbol{u}) > 1$ mentioned in Section 3.2, are difficult to come by. In practice, we may need to rethink the experimental set-up to reduce measurement noise or add new system inputs or target dimensions. Another option is to analyse the physical meaning of the model parameters with the goal of tightening the bounds on the allowed model parameter values. A smaller model parameter space should reduce model flexibility, i.e. the size of the target space spanned by the model. This may alleviate model indiscriminability.

This chapter has addressed design criteria for sequential design of experiments, but engineers and researchers may want to design several new experiments to run in parallel. This is called *batch optimisation* in the Bayesian optimisation literature [Gonzalez et al., 2016]. Galvanin et al. [2006, 2007] and Bazil et al. [2012] have studied design of parallel experiments for parameter estimation, but there have been fewer contributions on design of parallel experiments for model discrimination. A simple heuristic is to design new experiments in a sequential fashion while penalising new experiments in the vicinity of experiments already added to the next batch.

5 Design of Experiments for Black-Box Model Discrimination

This chapter is based on Olofsson et al. [2018b], Olofsson et al. [2018a], and Sections 4, 5, 6 and 7 of Olofsson et al. [2019a].

We now consider designing experiments and discriminating between black-box models, i.e. models where we can observe the output given an input, but where gradient information is not readily available. This gradient information is needed in the analytical approach to design of experiments and model discrimination for approximating the model prediction covariance (see Section 3.2). The data-driven approach in Section 3.3 makes fewer approximations and does not rely on gradient information, but may be computationally prohibitive. Section 3.4 discusses this trade-off between accuracy and computational cost.

We wish to bridge the gap between the analytical approach and the data-driven approach. We propose a hybridisation between the two: we sample the design and parameter spaces to learn surrogate models that can be incorporated into existing analytical design and model discrimination criteria. The surrogate model method presents a different trade-off between accuracy and computational cost, where we may benefit from the results of the analytical approach but also extend the analytical approach to cases of black-box models.

Surrogate models are common in applications where the original model does not easily lend itself to optimisation, e.g. Palmer and Realff [2002], Caballero and Grossmann [2008], Fahmi and Cremaschi [2012], Boukouvala et al. [2017], Beykal et al. [2018], Jones et al.

[2018], Carpio et al. [2018] and Yang et al. [2019]. Common surrogate models include e.g. support vector machines [Cortes and Vapnik, 1995]. Our surrogate models are GPs, flexible regression tools common in statistical machine learning [Rasmussen and Williams, 2006], e.g. for Bayesian black-box optimisation [Shahriari et al., 2016; Ulmasov et al., 2016; Mehrian et al., 2018; Olofsson et al., 2019b; Babutzka et al., 2019]. GPs provide model prediction confidence bounds, and their analytical nature allows us to extend the classical analytical methods to non-analytical models.

In addition to presenting the GP surrogate modelling method that will also be used in Chapter 6, this chapter introduces the *GPdoemd* open-source software package.

5.1 Gaussian Process Surrogate Model

Let us begin by studying a single model $f = f_i$, with parameters $\boldsymbol{\theta} = \boldsymbol{\theta}_i$. The function f may be a vector function, with predictions $f(\boldsymbol{u}, \boldsymbol{\theta}) \in \mathbb{R}^{D_y}$ where $D_y \geq 2$. Following Section 2.3.2, we place *independent* GP priors on each output (or target dimension) $d = 1, \dots, D_y$ of f:

$$f_{(d)} \sim \mathcal{GP}\left(0,\, k_{\boldsymbol{u},(d)}(\cdot,\cdot) k_{\boldsymbol{\theta},(d)}(\cdot,\cdot)\right) . \tag{5.1.1}$$

Training data is required for learning the GP hyperparameters and performing GP regression. The training data is acquired in the following way:

controls We assume the control space $\mathcal{U}$ is known and bounded. Training data control values $\boldsymbol{u}_\ell$ are sampled from $\mathcal{U}$. The training data should reasonable cover regions of $\mathcal{U}$ where optimal solutions to the experimental design problem may lie. The control training data sampling can be done using different strategies. In the Section 5.3 experiments, the control space $\mathcal{U}$ is a known hyperrectangle and the training data control values $\boldsymbol{u}_\ell$ placed on a uniform grid. The density of the grid depends on the dimensionality of $\mathcal{U}$ and how large training data set can be accommodated. Regular GP regression normally scales to a few thousand training data points. Sparse GP regression can scale to larger training data sets.

parameters Training data model parameter values $\boldsymbol{\theta}_\ell$ are sampled from $\boldsymbol{\theta}_\ell \sim \mathcal{N}(\hat{\boldsymbol{\theta}}, \epsilon \mathbf{I})$, for some small ϵ, where $\hat{\boldsymbol{\theta}}$ denotes the maximum *a posteriori* model parameter estimate. The model parameters are optimised during parameter estimation and then remain fixed when solving the experimental design problem. The training data explores a small region around $\hat{\boldsymbol{\theta}}$ such that we can approximate the gradients $\partial f / \partial \boldsymbol{\theta}$.

inputs Let $\hat{\boldsymbol{z}}_\ell = (\boldsymbol{u}_\ell, \boldsymbol{\theta}_\ell)$ denote the combined training inputs. Let $\hat{\mathbf{Z}}$ denote the set of combined training data inputs, with $[\hat{\mathbf{Z}}]_\ell = \hat{\boldsymbol{z}}_\ell$.

targets Training data targets $\hat{\boldsymbol{f}}_\ell$ are computed as $\hat{\boldsymbol{f}}_\ell = f(\boldsymbol{z}_\ell)$. Let $\hat{\mathbf{F}}$ denote the set of training data targets, with $[\hat{\mathbf{F}}]_\ell = \hat{\boldsymbol{f}}_\ell$, and let $\hat{\boldsymbol{f}}_{(d)}$ denote the vector of values for the d^{th} output dimension of $\hat{\mathbf{F}}$.

The (artificial) GP training data should not be confused with the experimental data set $\mathcal{D}$ that is used for parameter estimation and model discrimination. Similarly, the artificial GP regression noise $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \sigma_\eta^2 \mathbf{I})$ in Section 2.3.1 should not be confused with the experimental measurement noise $\boldsymbol{v} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_y)$ in Section 3.2.

The hyperparameters, e.g. signal variances $\rho_{(d)}^2$, length scales $\boldsymbol{\Lambda}_{(d)}$ and noise variance $\sigma_{\eta,(d)}^2$, are learnt by maximising the log-marginal likelihood $\log p(\hat{\boldsymbol{f}}_{(d)} \mid \rho_{(d)}^2, \boldsymbol{\Lambda}_{(d)}, \sigma_{\eta,(d)}^2)$. As discussed in Section 2.5, since f is a deterministic function the noise variance hyperparameter $\sigma_{\eta,(d)}^2$ should be lower-bounded (e.g. $\sigma_{\eta,(d)}^2 \geq 1\text{E-}6$) to avoid numerical issues.

The predictive distribution at a test location $\boldsymbol{z} = (\boldsymbol{u}, \boldsymbol{\theta})$ is $f(\boldsymbol{z}) \sim \mathcal{N}(\mu(\boldsymbol{z}), \Sigma_f(\boldsymbol{z}))$ with

$$\begin{aligned} \mu(\boldsymbol{z}) &= \left[\mu_{(1)}(\boldsymbol{z}), \dots, \mu_{(D_y)}(\boldsymbol{z})\right]^\top , \\ \Sigma_f(\boldsymbol{z}) &= \operatorname{diag}\left(\sigma_{(1)}^2(\boldsymbol{z}), \dots, \sigma_{(D_y)}^2(\boldsymbol{z})\right) , \end{aligned} \tag{5.1.2}$$

where the mean and variance elements $\mu_{(d)}(\boldsymbol{z})$ and $\sigma_{(d)}^2(\boldsymbol{z})$ are given by the adapted Equation (2.3.2) expressions

$$\begin{aligned} \mu_{(d)}(\boldsymbol{z}) &= \boldsymbol{k}_{(d)}^\top (\mathbf{K}_{(d)} + \sigma_{\eta,(d)}^2 \mathbf{I})^{-1} \hat{\boldsymbol{f}}_{(d)} , \\ \sigma_{(d)}^2(\boldsymbol{z}) &= k_{(d)}(\boldsymbol{z}, \boldsymbol{z}) - \boldsymbol{k}_{(d)}^\top (\mathbf{K}_{(d)} + \sigma_{\eta,(d)}^2 \mathbf{I})^{-1} \boldsymbol{k}_{(d)} . \end{aligned}$$

The vector $\boldsymbol{k}_{(d)}$ and matrix $\mathbf{K}_{(d)}$ have elements $[\boldsymbol{k}_{(d)}]_\ell = k_{(d)}(\boldsymbol{z}, \hat{\boldsymbol{z}}_\ell)$ and $[\mathbf{K}_{(d)}]_{\ell_1, \ell_2} = k_{(d)}(\hat{\boldsymbol{z}}_{\ell_1}, \hat{\boldsymbol{z}}_{\ell_2})$, respectively. Note that we use the simplified notation $\mu(\boldsymbol{z}) = \mu(\boldsymbol{u}, \boldsymbol{\theta})$, and that the covariance function $k_{(d)}$ for each target dimension is

$$k_{(d)}(\boldsymbol{z}, \boldsymbol{z}') = k_{\boldsymbol{u},(d)}(\boldsymbol{u}, \boldsymbol{u}') k_{\boldsymbol{\theta},(d)}(\boldsymbol{\theta}, \boldsymbol{\theta}') \,,$$

as given in Equation (5.1.1).

The model parameter distribution $p(\boldsymbol{\theta} \,|\, \mathcal{D})$, where $\mathcal{D}$ is real experimental data, is approximated as Gaussian distributed, with mean $\hat{\boldsymbol{\theta}}$ and covariance $\boldsymbol{\Sigma}_\theta$. The approximate model parameter covariance $\boldsymbol{\Sigma}_\theta$ is computed using the first-order Laplace approximation

$$\boldsymbol{\Sigma}_\theta^{-1} \approx \sum_{\boldsymbol{u} \in \mathcal{D}} \left. \frac{\partial \mu(\boldsymbol{u}, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}^{\top} \boldsymbol{\Sigma}_y^{-1} \left. \frac{\partial \mu(\boldsymbol{u}, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \,,$$

as in Section 3.2, with the gradients $\nabla_{\boldsymbol{\theta}} f$ replaced by the corresponding gradients $\nabla_{\boldsymbol{\theta}} \mu$.

Using the Gaussian model parameter distribution $p(\boldsymbol{\theta} \,|\, \mathcal{D})$ and the first- and second-order Taylor approximations described in Section 2.7, we can compute the approximate marginal predictive distribution

$$p\left(f(\boldsymbol{u}) \,|\, \mathcal{D}\right) \approx \mathcal{N}\left(\breve{\mu}(\boldsymbol{u}), \breve{\Sigma}(\boldsymbol{u})\right) \,. \tag{5.1.3}$$

The first-order Taylor approximation yields the marginal mean and variance

$$\begin{aligned} \breve{\mu}(\boldsymbol{u}) &= \mu(\boldsymbol{u}, \hat{\boldsymbol{\theta}} \,, \\ \breve{\Sigma}(\boldsymbol{u}) &= \Sigma_f(\boldsymbol{u}, \hat{\boldsymbol{\theta}}) + \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu} \boldsymbol{\Sigma}_\theta \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}^\top \,, \end{aligned}$$

where $\nabla_{\boldsymbol{\theta}} \boldsymbol{\mu} = \partial \mu(\boldsymbol{u}, \boldsymbol{\theta}) / \partial \boldsymbol{\theta} |_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}$. The second-order Taylor approximation yields the marginal mean $\breve{\mu}(\boldsymbol{u})$ with elements

$$\breve{\mu}_{(d)}(\boldsymbol{u}) = \mu_{(d)}(\boldsymbol{u}, \hat{\boldsymbol{\theta}}) + \tfrac{1}{2} \operatorname{tr}\left(\nabla_{\boldsymbol{\theta}}^2 \mu_{(d)} \boldsymbol{\Sigma}_\theta\right) \,,$$

and marginal covariance $\breve{\Sigma}(\boldsymbol{u})$

$$\breve{\Sigma}(\boldsymbol{u}) = \operatorname{diag}(q_{(1)}, \ldots, q_{(D_y)}) + \mathbf{Q} \,,$$

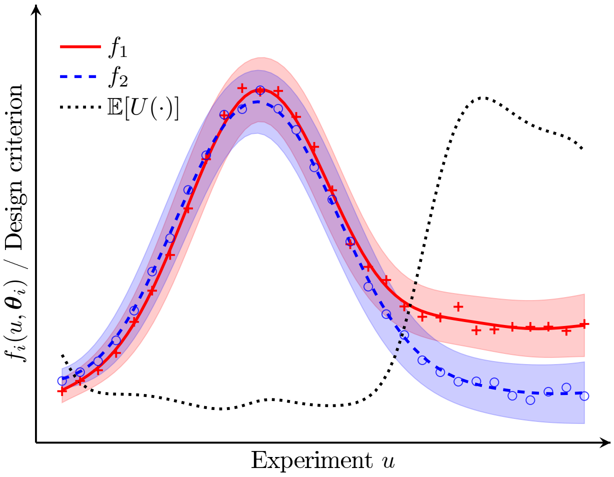


Figure 5.1: GP surrogates replace the original rival models f_1 and f_2 and are trained on a small number of sampled designs and model parameter values (crosses and circles). The GP surrogates are then used to compute analytical approximations of the marginal predictive distributions. These marginal predictive distribution approximations can be used with existing analytical design criteria. Compare to Figure 3.4 and Figure 3.6.

$$q_{(d)} = \sigma^2_{(d)}(\boldsymbol{u}, \hat{\boldsymbol{\theta}}) + \tfrac{1}{2}\,\mathrm{tr}\left(\nabla^2_{\boldsymbol{\theta}} \sigma^2_{(d)} \boldsymbol{\Sigma}_\theta\right) ,$$

$$[\mathbf{Q}]_{(d_1),(d_2)} = \nabla_{\boldsymbol{\theta}} \mu_{(d_1)} \boldsymbol{\Sigma}_\theta \nabla_{\boldsymbol{\theta}} \mu_{(d_2)}^\top + \tfrac{1}{2}\,\mathrm{tr}\left(\nabla^2_{\boldsymbol{\theta}} \mu_{(d_1)} \boldsymbol{\Sigma}_\theta \nabla^2_{\boldsymbol{\theta}} \mu_{(d_2)} \boldsymbol{\Sigma}_\theta\right) ,$$

for $d, d_1, d_2 = 1, \dots, D_y$.

We make two observations:

- Unlike in Section 2.7, only a subset of the input dimensions are marginalised out; $\breve{\boldsymbol{\mu}} = \breve{\mu}(\boldsymbol{u})$ and $\breve{\boldsymbol{\Sigma}} = \breve{\Sigma}(\boldsymbol{u})$ are functions of the experimental design $\boldsymbol{u}$.

- Replacing the original model f with a GP surrogate model introduces added uncertainty to the marginal predictive covariance $\breve{\Sigma}(\boldsymbol{u})$: The marginal predictive covariance of the GP surrogate model contains an extra term $\mathbb{E}[\Sigma_f(\cdot)]$ of uncertainty, compared to the marginal predictive covariance of the analytical models in Section 3.2.

Figure 5.1 shows an example of the marginal predictive distributions of two rival models approximated using GP surrogate models. These marginal predictive distributions can be used with existing analytical design criteria.

When using the second-order Taylor approximation of the marginal predictive distribution, it is still best to use the first-order Laplace approximation of the model parameter covariance since a second-order approximation more easily results in a singular Fisher information matrix $\mathbf{\Sigma}_\theta^{-1}$. The first-order Laplace approximation yields a positive semi-definite $\mathbf{\Sigma}_\theta^{-1}$ by construction, but it can still be ill-conditioned. A singular Fisher information matrix indicates non-identifiable model parameters, a common problem in mechanistic modelling. As discussed in Section 3.2, a (near-)singular Fisher information matrix means the Laplace approximation of the model parameter uncertainty is not useful, and an alternative approximation for $\mathbf{\Sigma}_\theta$ is recommended.

The approximate marginalisation using first- or second-order Taylor approximations can also be used for surrogate models with inducing input sparse GP regression (see Section 2.6). The gradient expressions for using sparse GP regression with inducing inputs are directly equivalent to the gradient expressions for the full GP model.

5.2 The GPdoemd Software Package

GPdoemd[1] is an open-source Python package implementing the GP surrogate method to design of experiments presented in Section 5.1. This section describes the package. Additional documentation for installing and using GPdoemd is available in the form of a PDF document and Jupyter notebook demonstrations in the GPdoemd GitHub repository. GPdoemd uses functionality from the GPy [since 2012] Python package for GP training and inference. Other dependencies are the standard numpy (v1.7-v1.15) and scipy (v0.17-v1.1) packages. GPdoemd is tested for Python version 3.4, 3.5 and 3.6. There are several Python interfaces to query models written in other languages, e.g. R or MATLAB. GPdoemd only requires point sampling of the original models in order to construct the GP surrogates.

[1] Available online at: `https://github.com/cog-imperial/GPdoemd`

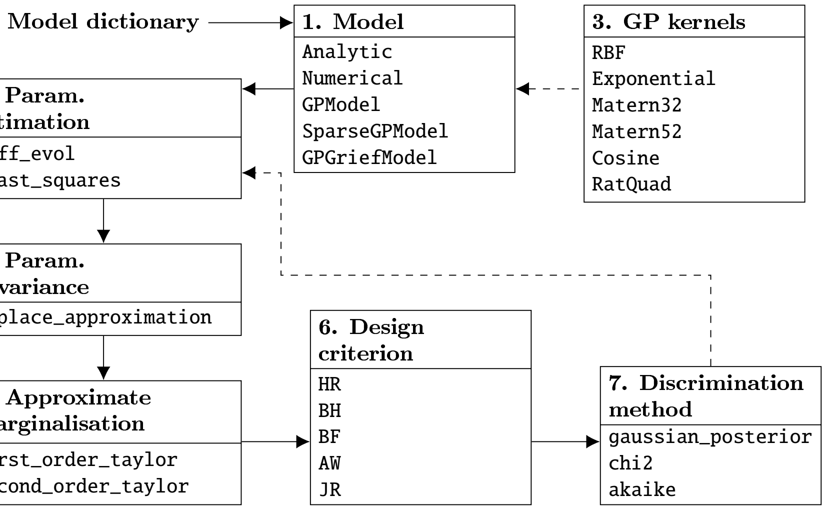


Figure 5.2: The modular structure of the GPdoemd open-source software package.

5.2.1 Implementation

GPdoemd consists of several modules, illustrated in Figure 5.2, that offer a choice between different GP kernel functions, inference methods, methods to approximate the marginal predictive distributions, design criteria and model discrimination methods. The modules can easily be extended and new functions implemented and added to the GPdoemd toolbox.

GPdoemd currently comes with the Table 5.1 case studies. Researchers may try the GP surrogate method and compare the performance to competing methods for design of experiments for model discrimination. The case study `mixing` was developed for GPdoemd and considers different order micro- and macrofluid models. Section 5.3.3 describes the `mixing` case study.

5.2.2 Syntax and Supported Features

Assuming the rival models $f_i(\boldsymbol{u}, \boldsymbol{\theta}_i)$ have been proposed, GPdoemd assists in model discrimination. Figure 5.2 illustrates the process.

Name	Reference	D_u	$D_{\theta,i}$	D_y	M	f_i
`bff1983`	Buzzi-Ferraris and Forzatti [1983]	3	5	1	5	A
`bffeh1984`	Buzzi-Ferraris et al. [1984]	2	4	2	4	A
`bffc1990a`	Buzzi-Ferraris et al. [1990]	3	2–6	1	4	A
`mixing`	Olofsson et al. [2019a]	3, (1)	1	1	5	A
`msm2010`	Michalik et al. [2010]	3	1	1	10	A
`vthr2014linear`	Vanlier et al. [2014]	1	2–4	1	4	A
`vthr2014ode`	Vanlier et al. [2014]	3, (2)	14	1	4	BB
`tandogan2017`	Tandogan et al. [2017]	4	8–14	2	3	BB

Table 5.1: GPdoemd case studies, with the number of design variables D_u (number of discrete variables in parenthesis), model parameters $D_{\theta,i}$, target dimensions D_y, rival models M. The last column states whether the models are (A) analytical, i.e. function gradients are provided, or (BB) black boxes, i.e. gradients are *not* provided.

Model type A model object is initialised using a Python dictionary containing the model name (`name`), the model function $f_i(\boldsymbol{u}, \boldsymbol{\theta}_i)$ handle (`call`), the design variable and model parameter dimensions D_x and $D_{\theta,i}$ (`dim_x` and `dim_p`), the number of target dimensions D_y (`num_outputs`), model parameter bounds (`p_bounds`), experimental noise (co)variance $\boldsymbol{\Sigma}_y$ (`meas_noise_var`), and a list of the dimensions for binary design variables (`binary_variables`). Binary design variables are handled by creating separate GP surrogates for each binary combination. This dictionary is passed to one of the implemented model types (Box 1 in Figure 5.2). GPdoemd uses the GPy implementation of sparse GP regression, with variational learning of the inducing inputs [Titsias, 2009].

Parameter estimation Given experimental data `Ydata` for designs `Xdata`, GPdoemd helps find the optimal model parameter values $\boldsymbol{\theta}^*$ using prediction error minimisation (Box 2 in Figure 5.2): differential evolution (`diff_evol`) or least squares with finite difference gradient approximation (`least_squares`). Both `diff_evol` and `least_squares` are wrappers for `scipy` functions.

GP kernels The GP surrogate models require a choice of GP kernel functions k_x and k_θ for the GP prior $\mathcal{GP}(0, k_x k_\theta)$. GPdoemd currently supports 6 kernel functions (Box 3 in Figure 5.2) from the `GPy` package, with minor extensions.

Model parameter covariance GPdoemd assumes a Gaussian approximation $\mathcal{N}(\boldsymbol{\theta}^*, \boldsymbol{\Sigma}_\theta)$ of the model parameter distribution. GPdoemd currently implements a Laplacian approximation of $\boldsymbol{\Sigma}_\theta$.

Approximating marginal predictive distributions The hybrid approach approximates the marginal predictive distribution in with a Gaussian distribution. GPdoemd implements the first- and second-order Taylor approximations (Box 5 of Figure 5.2) of the first two moments of the models' predictive distributions in Equation (5.1.3).

Design criterion GPdoemd provides five different criteria (Box 6 in Figure 5.2) for designing the next experiment: `HR` [Hunter and Reiner, 1965], `BH` [Box and Hill, 1967], `BF` [Buzzi-Ferraris et al., 1990] and `AW` [Michalik et al., 2010] and `JR` [Olofsson et al., 2019a].

Discrimination criterion GPdoemd provides three different criteria (Box 7 in Figure 5.2) for model discrimination: normalised Gaussian posteriors $\pi_{i,N}$ [Box and Hill, 1967], χ^2 test [Buzzi-Ferraris and Forzatti, 1983], and the Akaike information criterion weights [Michalik et al., 2010].

5.2.3 Example

Assume we have a list `dlist` of model dictionaries, experimental data `Xdata`, `Ydata` with experimental noise variance `measvar`, and lists `X`, `P` and `Y` of surrogate model training data (design, model parameters and predictions, respectively). We wish to select the optimal next experiment from candidates `Xnew`.

```
N = Xnew.shape[0]       # Number of test points
M = len( dlist )        # Number of rival models
E = Ydata.shape[1]      # Number of target dimensions
mu, s2 = np.zeros(( N, M, E )), np.zeros(( N, M, E, E ))
for i,d in enumerate( dlist ):
    # Initialise surrogate model
    m = GPdoemd.models.GPModel(d)
    # Estimate model parameter values
    opt_method = GPdoemd.param_estim.least_squares
    m.param_estim(Xdata, Ydata, opt_method, m.p_bounds)
    # Set—up surrogate model
    RBF = GPdoemd.kernels.RBF
    Z = np.c_[ X[i], P[i] ]
    m.gp_surrogate(Z=Z, Y=Y[i], kern_x=RBF, kern_p=RBF)
    m.gp_optimise()
    # Approximate model parameter covariance
    m.Sigma = GPdoemd.param_covar.laplace_approximation( m, Xdata )
    # Approximate marginal predictive distribution at test points
    mu[:,i], s2[:,i] = GPdoemd.marginal.taylor_first_order( m, Xnew )
# Design criterion at test points
dc = GPdoemd.design_criteria.JR(mu, s2, measvar)
# Optimal next experiment
xnext = Xnew[ np.argmax(dc) ]
```

The newly designed experiment is executed, and `xnext` and the new observation added to `Xdata` and `Ydata`, respectively. If model discrimination fails, the process above is repeated in order to find the optimal next experiment.

5.3 Performance of Gaussian Process Surrogate Method

We next compare the GP surrogate approach in Section 5.1 of approximating the marginal predictive distribution to the classical analytical approach in Section 3.2. We will show that the GP surrogate method is not significantly worse at model discrimination than the analytical method, since otherwise the GP surrogates would be ineffective for extending the analytical method to black-box models. For notational convenience, we call the GP surrogate method with first- and second-order Taylor approximations of the marginal predictive distribution GP-T1 and GP-T2, respectively. In each case study, training data is generated from a grid in input space to ensure some level of space-filling.

We study the GP surrogate method using four case studies:

Case study 1: *Ammonia synthesis* [Buzzi-Ferraris et al., 1990]

Case study 2: *Chemical kinetic models* [Buzzi-Ferraris et al., 1984]

Case study 3: *Mixing* [Olofsson et al., 2019a]

Case study 4: *Biochemical networks* [Vanlier et al., 2014; Olofsson et al., 2018a]

Case study 1 considers four different models for synthesis of ammonia (NH_3) from hydrogen (H_2) and nitrogen (N_2). There are $D_u = 3$ design variables, $D_{\theta,i} \in \{2, 4, 6\}$ parameters per model, and $D_y = 1$ observable output. Each simulation has $N_0 = 5$ initial measurements and a maximum budget of 40 new experiments. Section 4.3.2 further describes the case study.

Case study 2, further described in Section 5.3.2, has four different chemical kinetic models. There are $D_u = 2$ design variables, $D_{\theta,i} = 4$ parameters per model, and $D_y = 2$ observable outputs. Each simulation has $N_0 = 5$ initial measurements and a maximum budget of 40 additional experiments.

Case study 3 is new. It studies conversion of a reactant under mixing of a fluid. There are $D_u = 3$ design variables (one of which is binary), $D_{\theta,i} = 1$ parameter per model, and $D_y = 1$ observable output. Each simulation has $N_0 = 2$ initial measurements and a maximum budget of 20 additional experiments. Section 5.3.3 further describes the case study.

Case study 4 is a version of the Vanlier et al. [2014] biochemical networks case study, with models consisting of systems of ordinary differential equations [Olofsson et al., 2018a]. There are $D_u = 3$ design variables, $D_{\theta,i} = 10$ parameters per model, and $D_y = 2$ observable outputs. Each simulation has $N_0 = 20$ initial measurements and a maximum budget of 100 additional experiments. Section 5.3.5 further describes the case study.

5.3.1 Case Study 1: Ammonia Synthesis

First, we compare the GP surrogate method to the classical analytical method on the ammonia synthesis case study [Buzzi-Ferraris et al., 1990]. Case study 1 is further described in

MD	$\pi_{i,N}$			χ_i^2			ω_i		
DC	D_{BH}	D_{JR}	U	D_{BF}	D_{JR}	U	D_{AW}	D_{JR}	U
A	20.85	22.24	34.50	20.56	21.12	14.50	7.11	6.61	21.25
SE	0.82	0.72	1.77	1.43	1.22	3.02	0.47	0.49	1.08
S [%]	81	87	2	81	84	10	100	100	73
F [%]	0	0	0	1	1	1	0	0	2
I [%]	19	13	98	18	15	89	0	0	25

(a) Analytic method results from Table 4.2.

MD	$\pi_{i,N}$		χ_i^2		ω_i	
DC	D_{BH}	D_{JR}	D_{BF}	D_{JR}	D_{AW}	D_{JR}
A	19.72	21.73	17.28	17.55	6.73	6.53
SE	0.68	0.70	1.30	1.12	0.39	0.41
S [%]	86	84	79	82	100	100
F [%]	0	0	1	2	0	0
I [%]	14	16	20	16	0	0

(b) GP-T1 (first-order Taylor)

MD	$\pi_{i,N}$		χ_i^2		ω_i	
DC	D_{BH}	D_{JR}	D_{BF}	D_{JR}	D_{AW}	D_{JR}
A	6.31	6.13	17.29	13.64	3.03	2.94
SE	0.21	0.24	1.55	1.73	0.14	0.12
S [%]	96	95	63	22	100	99
F [%]	0	0	0	2	0	1
I [%]	4	5	37	76	0	0

(c) GP-T2 (second-order Taylor)

Table 5.2: Performance statistics of GP surrogate method with (b) first- (GP-T1) and (c) second-order (GP-T2) Taylor approximation of the marginal predictive distribution for case study 1 (ammonia synthesis). Compare to statistics for analytical method in Table 4.2, reproduced here in (a) for the reader's convenience.

Section 4.3.2. Table 5.2 shows the performance statistics of GP-T1 and GP-T2 from simulations from 100 sets of random initial measurements. Table 5.2 also shows the Table 4.2 performance statistics of the analytic method for comparison.

Table 5.2b shows that GP-T1 performs similarly to the analytical method. The averages A are similar, taking the sometimes relatively large standard errors SE into account. In Table 5.2c, GP-T2 largely produces better simulation statistics than both the analytical method and GP-T1. GP-T2 appears to produce marginal predictive distributions more beneficial for model discrimination in this case study. It may be that the second-order characteristics of GP-T2 improve the marginal predictive distribution accuracy. It may also be that the models' structure (all model parameters appear in exponents in the denominator) advantage models with the fewest model parameters (f_1 and f_2). In this case study, we generate data from model f_1.

5.3.2 Case Study 2: Chemical Kinetic Models

This case study looks at four chemical kinetic models [Buzzi-Ferraris et al., 1984]. There are two observable outputs y_1, y_2 and two design variables $u_1, u_2 \in [5, 55]$. Each chemical kinetic model i has four model parameters $\theta_{i,j} \in [0, 1]$. The model functions are given by:

$$\begin{aligned}
&\text{Model 1}: \quad f_{1,(1)} = \theta_{1,1} u_1 u_2 / g_1 \,, \quad f_{1,(2)} = \theta_{1,2} u_1 u_2 / g_1 \,, \\
&\text{Model 2}: \quad f_{2,(1)} = \theta_{2,1} u_1 u_2 / g_2^2 \,, \quad f_{2,(2)} = \theta_{2,2} u_1 u_2 / h_{2,1}^2 \,, \\
&\text{Model 3}: \quad f_{3,(1)} = \theta_{3,1} u_1 u_2 / h_{3,1}^2 \,, f_{3,(2)} = \theta_{3,2} u_1 u_2 / h_{3,2}^2 \,, \\
&\text{Model 4}: \quad f_{4,(1)} = \theta_{4,1} u_1 u_2 / g_4 \,, \quad f_{4,(2)} = \theta_{4,2} u_1 u_2 / h_{4,1} \,,
\end{aligned}$$

where $g_i = 1 + \theta_{i,3} u_1 + \theta_{i,4} u_2$ and $h_{i,j} = 1 + \theta_{i,2+j} u_j$. We follow [Buzzi-Ferraris et al., 1984] by generating experimental data from model 1 with $\theta_{1,1} = \theta_{1,3} = 0.1$ and $\theta_{1,2} = \theta_{1,4} = 0.01$ and experimental noise covariance $\mathbf{\Sigma}_y = \mathrm{diag}(0.35,\, 2.3\text{E-}3)$. We start each test with 5 randomly sampled experimental observations, and set a maximum budget of 40 additional experiments.

Table 5.3 shows the case study 2 (chemical kinetic models) performance statistics of GP-T1, GP-T2 and the analytical method from simulations from 500 sets of random initial measurements. In this table, GP-T1 and GP-T2 perform similarly. For the $\pi_{i,N}$ and ω_i model discrimination methods, the GP surrogate method has higher averages A than the analytical method, but also higher success rates and lower failure rate. For the χ^2 model discrimination method, the situation reverses, though the difference in average A is smaller and for GP-T1 the failure rate is still lower.

	Analytical			GP-T1			GP-T2		
MD	$\pi_{i,N}$	χ_i^2	ω_i	$\pi_{i,N}$	χ_i^2	ω_i	$\pi_{i,N}$	χ_i^2	ω_i
DC	D_{BH}	D_{BF}	D_{AW}	D_{BH}	D_{BF}	D_{AW}	D_{BH}	D_{BF}	D_{AW}
A	2.60	2.87	2.08	4.31	2.23	2.72	4.14	2.29	2.64
SE	0.04	0.12	0.04	0.09	0.06	0.08	0.09	0.07	0.06
S [%]	86.4	64.2	62.4	95.6	47.4	88.6	96.9	46.6	90.1
F [%]	13.6	5.0	37.6	4.4	4.8	11.4	3.1	9.9	9.9
I [%]	0.0	30.8	0.0	0.0	47.8	0.0	0.0	43.5	0.0

Table 5.3: Performance comparison between the GP surrogate method with first- (GP-T1) and second-order (GP-T2) Taylor approximation of the marginal predictive distribution, and the analytical methods, for case study 2 (chemical kinetic models).

Order	***R***	**Mixing**	**PFR**	**CSTR**
0th	$\theta u_1/u_2$	Micro	$f_1 = f_2 = 1 - R$	$f_1 = 1 - R$ ($f_1 = 0$ if $R \geq 1$)
		Macro	($f_1 = f_2 = 0$ if $R \geq 1$)	$f_2 = 1 - R + R\exp(-1/R)$
1st	θu_1	Micro	$f_3 = \exp(-R)$	$f_3 = 1/(1+R)$
		Macro		
2nd	$\theta u_1 u_2$	Micro	$f_4 = f_5 = 1/(1+R)$	$f_4 = \frac{1}{2R}\left(-1 + \sqrt{1+4R}\right)$
		Macro		$f_5 = \frac{1}{R}\exp(1/R)\,\mathrm{Ei}(1/R)$

Table 5.4: Conversion rate models for a reaction in a micro- or macrofluid during mixing in an ideal PFR or CSTR reactor [Levenspiel, 1999, p. 356]

Overall, Table 5.3 indicates that the GP surrogate method is more conservative than the classical analytical method. This conservatism may arise from the added surrogate uncertainty term Σ_f in Equation (5.1.2) yielding a larger predicted variance for the surrogate than the original model.

5.3.3 Case Study 3: Mixing

This case study considers a single fluid containing a reactant mixing during reaction. We wish to learn (i) whether it is a zero-, first- or second-order reaction, and (ii) whether mixing occurs on the microscopic or macroscopic level [Levenspiel, 1999, ch. 16]. We can run experiments in a plug flow reactor (PFR) or continuous stirred-tank reactor (CSTR) and observe the reactant conversion rate. We assume ideal reactors. The rival models f_i are given in Table 5.4, where $\mathrm{Ei}(x) = \int_{-x}^{\infty} t^{-1}\exp(-t)\mathrm{d}t$ is the exponential integral.

The design variables are the residence time $u_1 \in [1, 100]$, the initial concentration $u_2 \in$

θ_1	θ_2	θ_3	θ_4	θ_5
6E-3	6E-3	0.015	0.025	0.025

Table 5.5: Recommended model parameter values for the data-generating model in case study 3 (mixing). Case study 3.1 generates data from model f_3 with true parameter value θ_3, and Case study 3.2 generates data from model f_5 with true parameter value θ_5.

$[0.01, 1]$ and the reactor type $u_3 \in \{\mathrm{PFR}, \mathrm{CSTR}\}$. The model parameter is the reaction rate $\theta_i \in$ [1E-6, 0.1].

Note that the expression for the conversion rate in the ideal PFR reactor is the same for micro- and macrofluids. This is also true for the CSTR reactor 1^{st}-order reaction expression. All expression are differentiable with respect to the model parameter, with the exception of the f_1 and f_2 at $R = 1$.

Experimental data can be generated from any of the models; Proposed model parameter values for the data-generating model can be found in Table 5.5. For the experimental evaluation in this paper, we generate data from model f_3 and f_5 with experimental noise variance $\sigma_y^2 =$ 2.5E-3. Generating data from model f_5 produces a significantly more difficult problem of model discrimination, since models f_4 and f_5 yield predictions difficult to differentiate. Let case study 3.1 and 3.2 refer to case study 3 with data generated from model f_3 and f_5, respectively.

Table 5.6 shows the performance statistics for GP-T1, GP-T2 and the analytical method from simulations from 100 sets of random initial measurements for case study 3.1, with experimental data generated from model f_3. This case study has a high success rate and low number of additional experiments required for all discrimination criteria and methods of model discrimination.

Table 5.7 shows the performance statistics for GP-T1, GP-T2 and the analytical method from simulating 100 sets of random initial measurements for case study 3.2, with experimental data generated from model f_5. Case study 3.2 is more difficult than case study 3.1, so we increase the maximum number of additional experiments to 100. Table 5.7 shows that GP-T1 and the analytical method perform similarly. GP-T2 performs poorly for this case study: the failure rates for the $\pi_{i,N}$ and ω_i methods of model discrimination are often com-

MD	$\pi_{i,N}$		χ_i^2		ω_i	
DC	D_{BH}	D_{JR}	D_{BF}	D_{JR}	D_{AW}	D_{JR}
A	4.42	4.25	2.09	1.30	2.47	2.38
SE	0.09	0.07	0.04	0.05	0.06	0.09
S [%]	100	100	100	100	100	100
F [%]	0	0	0	0	0	0
I [%]	0	0	0	0	0	0

(a) Analytical

MD	$\pi_{i,N}$		χ_i^2		ω_i	
DC	D_{BH}	D_{JR}	D_{BF}	D_{JR}	D_{AW}	D_{JR}
A	4.65	4.48	1.78	1.24	2.61	2.19
SE	0.16	0.13	0.08	0.09	0.06	0.09
S [%]	99	100	99	100	100	100
F [%]	0	0	0	0	0	0
I [%]	1	0	1	0	0	0

(b) GP-T1 (first-order Taylor)

MD	$\pi_{i,N}$		χ_i^2		ω_i	
DC	D_{BH}	D_{JR}	D_{BF}	D_{JR}	D_{AW}	D_{JR}
A	5.75	4.45	1.21	1.15	2.27	2.30
SE	0.16	0.23	0.06	0.11	0.05	0.10
S [%]	99	95	100	100	100	100
F [%]	0	0	0	0	0	0
I [%]	1	5	0	0	0	0

(c) GP-T2 (second-order Taylor)

Table 5.6: Performance comparison between the GP surrogate method with first- (GP-T1) and second-order (GP-T2) Taylor approximation of the marginal predictive distribution, and the analytical methods, for case study 3.1 (mixing). Noisy observed data is generated from model f_3.

parable to the corresponding success rates. The averages A are higher and the success rates lower for case study 3.2 than for case study 3.1. This is due to indiscriminability between models f_4 and f_5 (further discussed in Section 5.3.5).

MD	$\pi_{i,N}$		χ_i^2		ω_i	
DC	D_{BH}	D_{JR}	D_{BF}	D_{JR}	D_{AW}	D_{JR}
A	54.71	50.79	50.97	45.00	26.07	27.70
SE	2.25	2.08	4.88	3.96	1.56	1.68
S [%]	75	72	34	49	98	97
F [%]	0	0	0	1	2	2
I [%]	25	28	66	50	0	1

(a) Analytical

MD	$\pi_{i,N}$		χ_i^2		ω_i	
DC	D_{BH}	D_{JR}	D_{BF}	D_{JR}	D_{AW}	D_{JR}
A	53.13	50.88	49.03	44.23	26.91	28.05
SE	1.95	1.96	4.84	4.23	1.52	1.79
S [%]	71	74	33	43	100	98
F [%]	0	0	2	1	0	2
I [%]	29	26	65	56	0	0

(b) GP-T1 (first-order Taylor)

MD	$\pi_{i,N}$		χ_i^2		ω_i	
DC	D_{BH}	D_{JR}	D_{BF}	D_{JR}	D_{AW}	D_{JR}
A	75.69	83.05	60.70	52.86	12.32	22.96
SE	5.82	2.25	4.13	11.89	0.82	2.65
S [%]	16	39	33	7	59	45
F [%]	17	2	0	3	41	55
I [%]	67	59	67	90	0	0

(c) GP-T2 (second-order Taylor)

Table 5.7: Performance comparison between the GP surrogate method with first- (GP-T1) and second-order (GP-T2) Taylor approximation of the marginal predictive distribution, and the analytical methods, for case study 3.2 (mixing). Noisy observed data is generated from model f_5.

5.3.4 Joint Comparison

The Table 5.2 results for case study 1 were generated with 3 combinations of design criteria and model discrimination methods (excluding random design). The Table 5.3 results for case study 2 were generated for 6 combinations. The Table 5.6 and Table 5.7 results for case study 3.1 and 3.2, respectively, were generated for 6 combinations each. Hence there are $j = 1, \dots, 21$ combinations of case studies, design criteria and model discrimination methods. In each case study, $\ell = 1, \dots, 100$ random initial data sets were generated. Thus, a total of 2100 simulations were run for each of GP-T1, GP-T2 and the analytical method. Section 4.3.1 defines $\hat{a}_{j,\ell}$ as the number of additional experiments needed for *successful* model discrimination in combination j with initial data set ℓ (using GP-T1, GP-T2 or the analytical method). Figure 5.3 shows the GP-T1 distribution of $\hat{a}_{j,\ell}$ for all combinations of case studies, model discrimination methods and design criteria. The figure illustrates the variance in the results, possibly due to the independent noise realisations in the different simulations, but that the spread of analytical and GP-T1 results tends to be symmetric around the central line. Figure 5.4 shows the $\hat{a}_{j,\ell}$ distribution for GP-T2. In this figure some $\hat{a}_{j,\ell}$ distributions are not symmetric around the central line.

The average A and standard error SE in Table 4.1 are defined for combination j as

$$\begin{aligned} \mathrm{A}_j &= \mathrm{mean}\{\hat{a}_{j,\ell}\}_\ell \,, \\ \mathrm{SE}_j &= \frac{1}{\sqrt{\sum_\ell 1}} \cdot \mathrm{std}\{\hat{a}_{j,\ell}\}_\ell \,. \end{aligned}$$

Figure 5.5 shows the outcomes of all case study simulations and compares the averages with standard errors for GP-T1, GP-T2 and the analytical method. GP-T1 performs very similarly to the analytic method on average. The similar average performance indicates that GP-T1 successfully emulates the analytical method, which also uses a first-order Taylor approximation. GP-T2 apparently performs better on average than the analytical method (in terms of average number of additional experiments required in successful simulations) for case study 1, but the analytical method possibly performs better for case study 2 and 3.2. In most case studies, GP-T2 performs as well as—or better than—GP-T1. For case studies where the GP surrogate model makes accurate predictions, a second-order approximation of the marginal predictive distributions may be more accurate than a first-order approximation. However, for case study 3.2, with data generated from model f_5, GP-T2 performed worse.

5.3.5 Case Study 4: Biochemical Networks

This case study is an adapted version of a Vanlier et al. [2014] case study. There are nine chemical components with concentrations C_i, $i = 1, \dots, 9$. The concentrations are given by

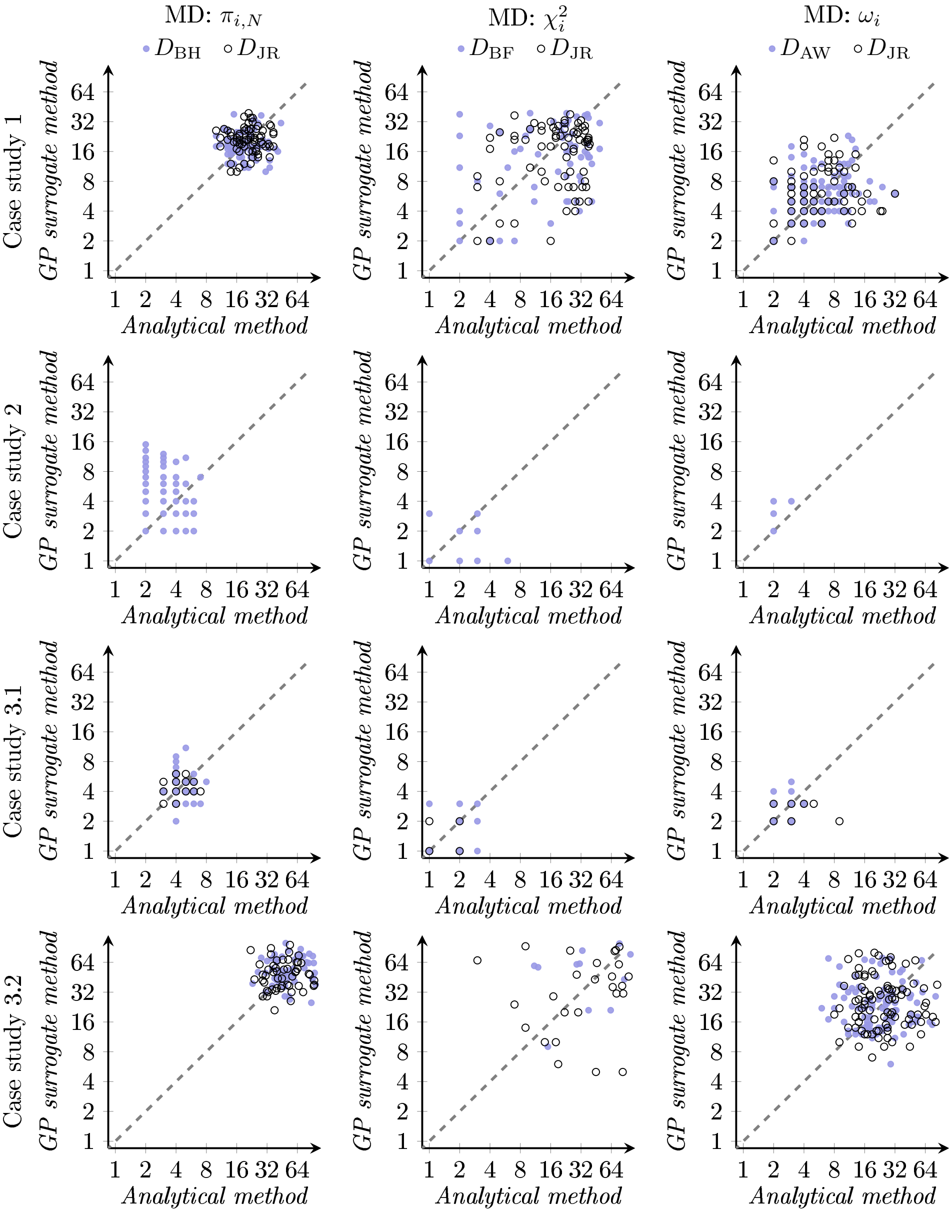


Figure 5.3: Comparison of the number of additional experiments $a_{j,\ell}$ needed for successful model discrimination using the analytical method or GP-T1. Each subplot shows the outcomes of all simulation with successful model discrimination for both the analytical and GP-T1, 1828 simulations in total, for a given case study, method of model discrimination and design criterion. Note that case study 2 does not compare the performance of D_{JR} to the other design criteria.

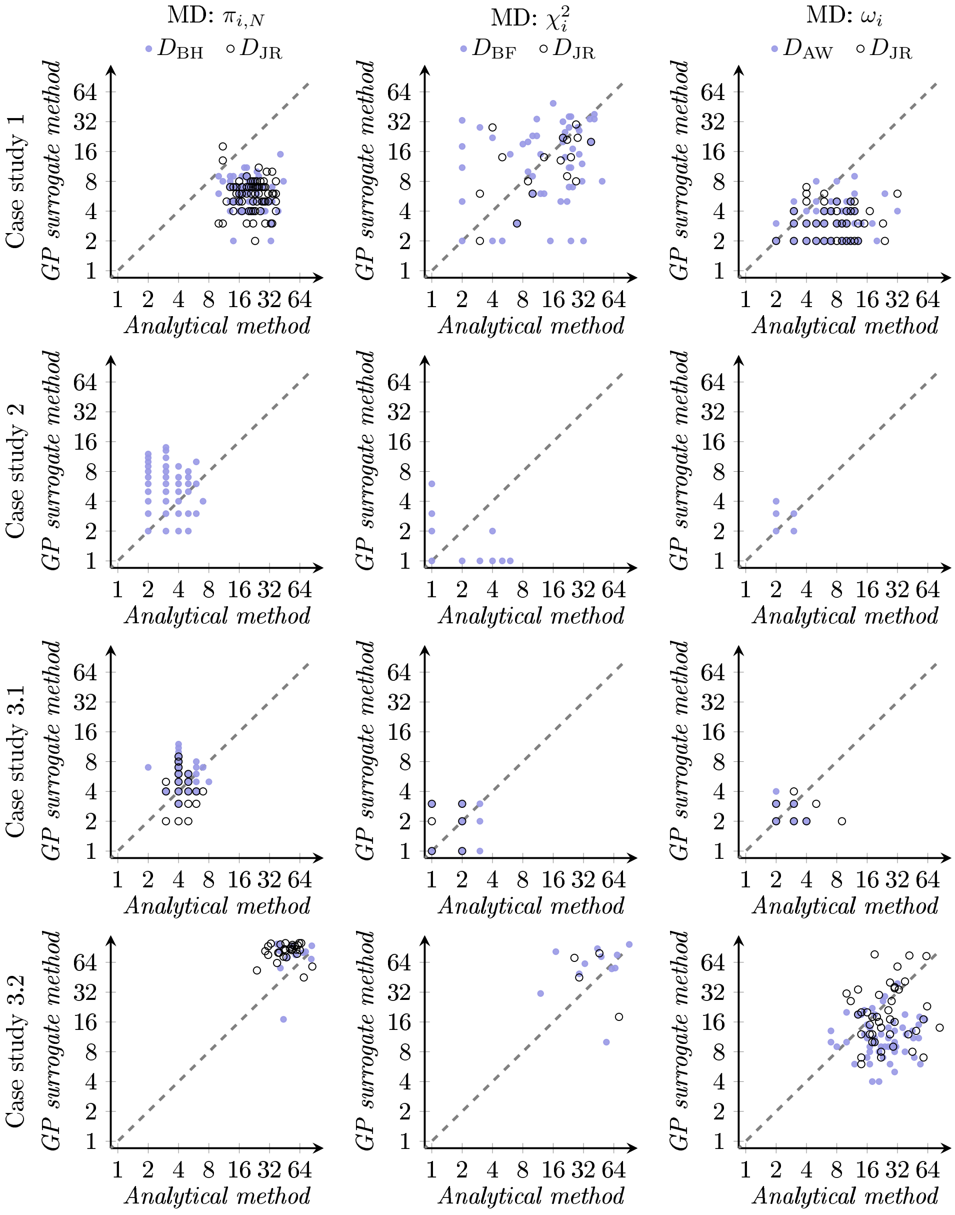


Figure 5.4: Comparison of the number of additional experiments $a_{j,\ell}$ needed for successful model discrimination using the analytical method or GP-T2. Each subplot shows the outcomes of all simulation with successful model discrimination for both the analytical and GP-T2, 1615 simulations in total, for a given case study, method of model discrimination and design criterion. Note that case study 2 does not compare the performance of D_{JR} to the other design criteria.

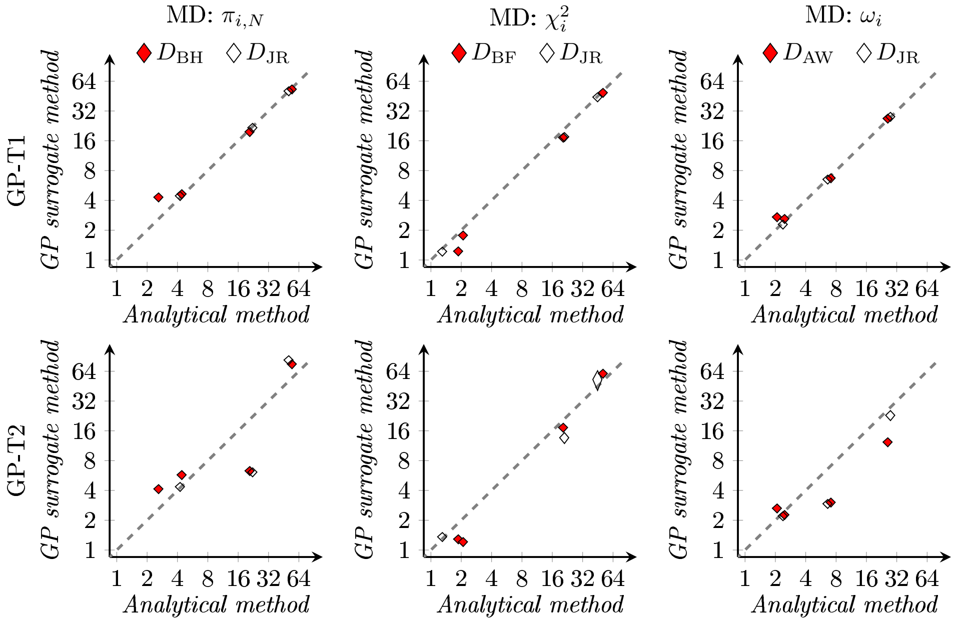


Figure 5.5: Comparison of the average number additional experiments A (with standard error SE) needed for successful model discrimination using the analytical method or (top row) GP-T1 or (bottom row) GP-T1. The average A decides the centre of each marker, and the length along each axis is the SE (with a minimum value enforced to make the markers visible). Each subplot shows the average number of additional experiments for both the analytical and GP-T1 or GP-T1, for each case study and a specified method of model discrimination and design criterion. Note that one case study does not compare the performance of D_{JR} to the other design criteria, so there are three D_{JR} averages and four averages for other design criteria. We note that on average, GP-T1 performs very similarly to the analytical method, whereas GP-T2's comparative performance varies more.

the solution to the system of ordinary differential equations

$$\begin{aligned}
\mathrm{d}C_1/\mathrm{d}t &= \theta_{i,2}C_2 - g_1\,, \\
\mathrm{d}C_2/\mathrm{d}t &= g_1 - \theta_{i,2}C_2\,, \\
\mathrm{d}C_3/\mathrm{d}t &= \theta_{i,4}C_4 - g_2\,, \\
\mathrm{d}C_4/\mathrm{d}t &= g_2 - \theta_{i,4}C_4 - \theta_{i,5}C_4C_6 + \theta_{i,6}C_7\,, \\
\mathrm{d}C_5/\mathrm{d}t &= \theta_{i,4}C_6 - \theta_{i,10}C_5\,, \\
\mathrm{d}C_6/\mathrm{d}t &= \theta_{i,6}C_7 - \theta_{i,5}C_4C_6 + \theta_{i,10}C_5 - \theta_{i,4}C_6\,,
\end{aligned}$$

$$\begin{aligned}
\mathrm{d}C_7/\mathrm{d}t &= \theta_{i,5}C_4C_6 - \theta_{i,6}C_7\,, \\
\mathrm{d}C_8/\mathrm{d}t &= \theta_{i,8}C_9 - \theta_{i,7}C_8\,, \\
\mathrm{d}C_9/\mathrm{d}t &= \theta_{i,7}C_8 - \theta_{i,8}C_9\,,
\end{aligned}$$

i.e. the stoichiometry is the same for all models. But the fluxes g_1 and g_2 differ for the different models. For flux g_1 the models differ in the following way:

$$\begin{aligned}
\text{Model } i \in \{1,3,4\}: \quad g_1 &= \theta_{i,1}C_1\,, \\
\text{Model } 2: \quad g_1 &= \frac{\theta_{2,1}C_1}{\theta_{2,9}+C_7}\,.
\end{aligned}$$

For flux g_2 the models differ in the following way:

$$\begin{aligned}
\text{Model } 1: \quad g_2 &= \frac{\theta_{1,3}C_2C_3}{\theta_{1,9}+C_7}\,, \\
\text{Model } 2: \quad g_2 &= \theta_{2,3}C_2C_3\,, \\
\text{Model } 3: \quad g_2 &= \frac{\theta_{3,3}C_2C_3}{\theta_{3,9}+C_9}\,, \\
\text{Model } 4: \quad g_2 &= \frac{\theta_{4,3}C_2C_3}{\theta_{4,9}+C_8}\,.
\end{aligned}$$

We assume that the only measured states are the concentrations C_4 and C_9, because these are the states from which Vanlier et al. [2014] collect their initial data. Similarly, we use the initial concentrations $C_4(t=0)$ and $C_9(t=0)$ as two of our design variables, the third design variable being the time point t at which to measure the concentrations.

Vanlier et al. [2014] look at times points in the range $t \in [0, 20]$, which we also adopt. We assume the initial concentrations $C_4(t=0), C_9(t=0) \in [0,1]$ and fix all other initial concentrations to

$$\begin{aligned}
C_1(t=0) = C_3(t=0) = C_5(t=0) = C_8(t=0) &= 1\,, \\
C_2(t=0) = C_6(t=0) = C_7(t=0) &= 0.1\,.
\end{aligned}$$

We assume the model parameter space $\boldsymbol{\theta} \in [0,1]^{10}$. Simulations show that sampling from

	4 models			*3 models*		
MD	$\pi_{i,N}$	χ_i^2	ω_i	π	χ_i^2	ω_i
DC	D_{BH}	D_{BF}	D_{AW}	D_{BH}	D_{BF}	D_{AW}
A	20.10	39.83	29.62	15.80	21.91	9.74
SE	3.72	12.09	7.72	2.05	2.52	1.70
S [%]	15.9	9.5	33.3	89.5	77.2	95.6
F [%]	7.9	0.0	7.9	6.1	0.9	1.8
I [%]	76.2	90.5	58.7	4.4	21.9	2.6

Table 5.8: Results from case study 4 (biochemical networks). With four models, we encounter model indiscriminability: Two of the models make predictions too similar to successfully discriminate between them in a majority of simulations. Experimental data is generated from one of the two models. If we remove the other model, we find that we are able to successfully perform model discrimination.

this parameter space gives a wide range of model realisations.

With reference to models 1 and 2 being similar, we see that the only difference between them is that the term $\theta_{i,9} + C_7$ divides g_1 and g_3 for models 1 and 2, respectively. If C_7 is small compared to $\theta_{i,9}$, then the models are nearly identical.

We wish to verify that the GP surrogate method successfully extends the classical, analytical method for design of experiment for model discrimination to situations with black-box models. The Table 5.8 results for 4 models show that the the success rates are significantly lower for this case study than for previous case studies. For the $\pi_{i,N}$ model discrimination method, the success rate is only twice as high as the failure rate. Rates of inconclusive results are high, despite allowing 100 additional experiments with averages A all below 50. The reason is that, in this case study, the prediction difference between models f_1 and f_2 is often smaller than the experimental noise.

For a simulation, we can examine the evolving model discrimination criterion ($\pi_{i,N}$, χ^2 or ω_i) while adding measurements. For example, Figure 5.6 shows the evolution of the Akaike weights ω_i for all simulations. Figure 5.6a suggests that models f_1 and f_2 cannot be discriminated in many simulations. To verify this, we remove model f_2 from the set of rival models. Table 5.8 (3 models) and Figure 5.6b show that removing model f_2 enables correct identification of model f_1 as the data-generating model in most simulations.

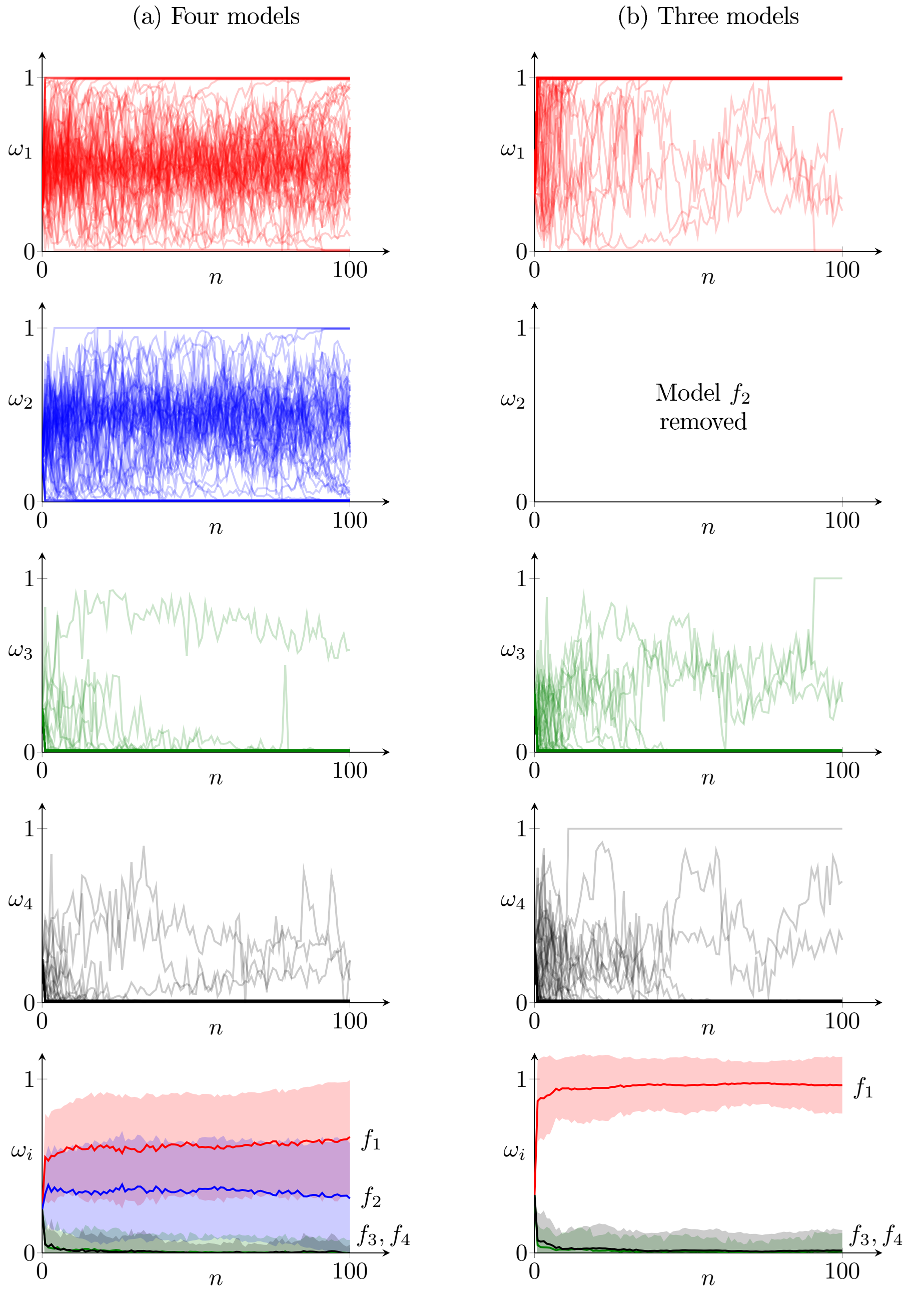


Figure 5.6: Results from case study 4 (biochemical networks) with (a) four rival models and (b) three rival models. The first four plots from the top show the evolution of the Akaike weights ω_i for all simulations. The plots on the bottom row show the averages (with one standard deviation). The plots in column (a) and statistics in Table 5.8 indicate that model f_1 and f_2 are almost indiscriminable. When we remove model f_2 from the set of models, as in column (b), the GP surrogate method successfully finds that f_1 is the data-generating model.

5.4 Discussion

The Section 5.3 experiments are not in every way representative of model discrimination in a real setting. The large-scale tests of the D_{JR} design criterion and GP surrogate method require fixed thresholds for when model discrimination (using $\pi_{i,N}$, χ_i^2 or ω_i) rejects or selects a model. Our experiments set the "winning" threshold for $\pi_{i,N}$ and ω_i to 99.9%, which is arguably high. But a high threshold also decreases the failure risk. In practice, an engineer may select a model maintaining a probability score, e.g. of $\omega_i = 0.98$ over multiple experiments. Regardless of the precise threshold value, the results in Section 5.3 illustrate the different discrimination methods' relative performance. As described in Section 3.4, using the χ^2 test is significantly less likely to result in selecting the incorrect model. Typically, this results in a higher number of required experiments to confidently discard inaccurate models. In our results we do not see a higher average number of additional required experiments. However, this average is only computed for *successful* simulations, where the correct model was identified. We see that the rate of inconclusive experiments is higher for the χ^2 test method than when using the normalised model likelihoods $\pi_{i,N}$ or Akaike weights ω_i. For some case studies the failure rate is unacceptably high when using the $\pi_{i,N}$ or ω_i discrimination criteria, but significantly lower when using the χ^2 test. These are some of the reasons why the χ^2 test is the most widely used discrimination criterion in practice.

The results comparing GP-T1 (first-order approximation) and GP-T2 (second-order approximation) paint a mixed picture. In case study GP-T2 arguably performs significantly better than GP-T1 and the analytical method (using the $\pi_{i,N}$ and ω_i tests for model discrimination). In case study 1 the data-generating model has 2 model parameters, compared to 2, 4 and 6 parameters, respectively, for the three rival models. It is possible the result would have been different if data had been generated from the models with 4 or 6 model parameters, since this makes accurate approximations of the model prediction covariance more difficult. It is interesting to note that GP-T2 does not perform better than GP-T1 and the analytical method (in terms of average A and success rate S) when the χ^2 test is used for model discrimination. This indicates that the GP-T2 predictive distributions are not necessarily more accurate than the GP-T1 predictive distributions. For case study 3.2, the

GP-T2 failure rate is also much higher than the GP-T1 failure rate. We would recommend using GP-T1 for designing experiments as GP-T1 has results that are more consistently in line with the baseline results of the analytical method. Chapter 6 will consider a first-order approximation for design of dynamic experiments.

6 Design of Dynamic Experiments for Model Discrimination

Chapter 5 considered static models $\boldsymbol{y} = f(\boldsymbol{u}, \boldsymbol{\theta}) + \boldsymbol{v}$, i.e. models that do not explicitly depend on time. Static models are sufficient, e.g. for experiments where (i) the design variable $\boldsymbol{u}$ only controls the initial conditions of an experiment and (ii) a single set of measurements y are taken. Static models are relevant, e.g. in very quick chemical reactions, where the time from beginning to end of an experiment is too short to take intermediate measurements. However, many industrially relevant processes are slow-moving and dynamic: the control signals can be varied during an experiment, and measurements taken (almost) continuously during an experiment. Examples include growing organisms and fermentation processes. Dynamic (explicitly time-dependent) models best describe such processes [Bar-Joseph et al., 2012].

This chapter considers state space formulations of dynamic models. The state space formulation stems from the fact that many dynamic processes are most easily expressed mathematically using differential or difference equations. Dynamic models introduce some added complexity to the design of experiments, in particular for the case of dynamic black-box models. This chapter builds and expands on the Chapter 5 methodology to address this.

Assume the system at time τ is in a state $x(\tau) \in \mathbb{R}^{D_x}$. The rate of change to the system's latent state is a function of the current latent state $x(\tau)$, a control input $u(\tau) \in \mathbb{R}^{D_u}$, and some process noise $w(\tau)$. In general, we cannot observe all dimensions of the state $x(\tau)$, hence $x(\tau)$ is referred to as a *latent state*. Let $z(\tau) \in \mathbb{R}^{D_z}$ denote the observed state. We take measurements $\boldsymbol{y}_t \in \mathbb{R}^{D_z}$ of the observed states $z(\tau)$ at discrete time-points $\tau_t \in \mathcal{T}_{\text{meas}}$.

Measurement noise $\boldsymbol{v}_t$ corrupts the measurements of the observed states.

The state space model $\mathcal{M}$ describes the dynamic system's behaviour

$$\mathcal{M}: \quad \begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}\tau} x(\tau) = f(x(\tau), u(\tau), \boldsymbol{\theta}) + w(\tau)\,, \\ \quad z(\tau) = \mathbf{H} x(\tau)\,, \\ \quad\quad \boldsymbol{y}_t = z(\tau_t) + \boldsymbol{v}_t\,, \quad \tau_t \in \mathcal{T}_{\mathrm{meas}}\,, \end{cases} \tag{6.0.1}$$

where f is called the transition function and parameterised by $\boldsymbol{\theta}$, and the observed state $z(\tau)$ is a subset (or linear combination) $\mathbf{H}x(\tau)$ of the latent state. Rival models $\mathcal{M}_1, \dots, \mathcal{M}_M$ may have different dimensionality $D_{x,i}$ of the latent state $x(\tau)$, but the dimensionality D_z of the observed state $z(\tau)$ and measurements $\boldsymbol{y}_t$ is the same for all models.

The model in Equation (6.0.1) is an example of a *continuous-time model*, where the latent state $x(\tau)$ is the solution to a system of ordinary differential equations. However, we may choose to express the change in latent state as discrete transitions between time steps. Let $\mathcal{T} = \{t\Delta\tau\}_{t=1}^{T}$ denote a set of equidistant time points, and let $\boldsymbol{x}_t$ and $\boldsymbol{z}_t$ denote the latent state and observed state at time step t, respectively. Piece-wise constant control inputs $\boldsymbol{u}_t$ form the control signal. We can write a *discrete-time model* as

$$\mathcal{M}: \quad \begin{cases} \boldsymbol{x}_t = f(\boldsymbol{x}_{t-1}, \boldsymbol{u}_{t-1}, \boldsymbol{\theta}) + \boldsymbol{w}_{t-1}\,, \\ \boldsymbol{z}_t = \mathbf{H}\boldsymbol{x}_t\,, \\ \boldsymbol{y}_t = \boldsymbol{z}_t + \boldsymbol{v}_t\,. \end{cases} \tag{6.0.2}$$

Another form of discrete-time model, which follows from an Euler discretisation of the continuous-time model, is the discrete-time model with a Δ-transition

$$\mathcal{M}: \quad \begin{cases} \boldsymbol{x}_t = \boldsymbol{x}_{t-1} + f(\boldsymbol{x}_{t-1}, \boldsymbol{u}_{t-1}, \boldsymbol{\theta}) + \boldsymbol{w}_{t-1}\,, \\ \boldsymbol{z}_t = \mathbf{H}\boldsymbol{x}_t\,, \\ \boldsymbol{y}_t = \boldsymbol{z}_t + \boldsymbol{v}_t\,, \end{cases} \tag{6.0.3}$$

where the latent state at time step t is given by a perturbation to the latent state at time step $t-1$. Note that the transition functions f are different for the continuous-time model,

	Mapping	Input dimensionality	Output dimensionality
Static case	$\boldsymbol{x}_0, \boldsymbol{u}_{0:T-1}, \boldsymbol{\theta} \mapsto \boldsymbol{z}_{1:T}$	$D_x + D_u \times T + D_\theta$	$D_z \times T$
Dynamic case	$\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\theta} \mapsto \boldsymbol{x}_{t+1}$	$D_x + D_u + D_\theta$	D_x

Table 6.1: Comparison of the input and output dimensionality of the function f mapping in the static case (Chapter 5) and dynamic case (this chapter).

discrete-time model, and discrete-time model with Δ-transition.

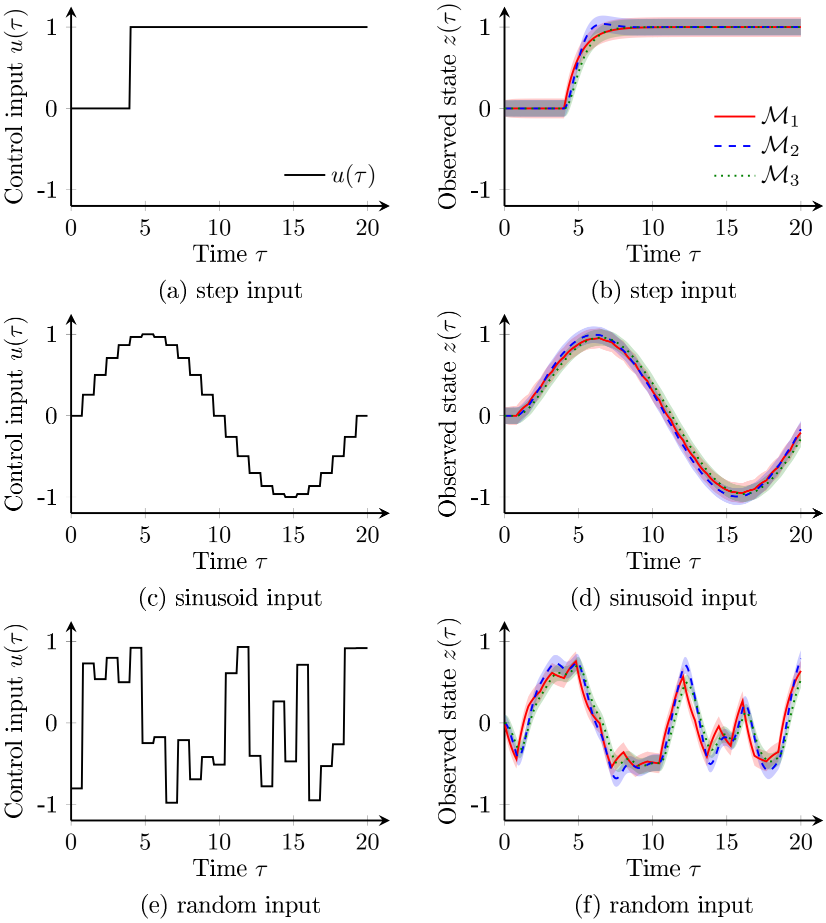
Let $\boldsymbol{x}_{0:T}$, $\boldsymbol{z}_{1:T}$ and $\boldsymbol{u}_{0:T-1}$ denote the sequences of latent states, observed states and control inputs, respectively, in the discrete-time models. The initial latent state $\boldsymbol{x}_0$ and control inputs $\boldsymbol{u}_{0:T-1}$ are controlled by the user, and may be optimised. In order to make optimisation feasible for systems described by ***continuous-time models***, it is common practice to discretise the control signal $u(\tau)$ and let it be piece-wise constant (e.g. Espie and Macchietto [1989] and Asprey and Macchietto [2000]). Hence, the inputs to the system models have total dimensionality $D_x + D_u \times T + D_\theta$, for both discrete- and continuous-time models, and the outputs (measurements) have dimensionality $D_z \times T$. Hence the input and output dimensionality are equivalent for discrete- and continuous-time models. It is clear that dynamic experiment design problems typically have high-dimensional input and output spaces. However, the state space model formulation in this chapter reduces the input-output dimensionality of the function f, compared to the equivalent input-output dimensionality of the equivalent static formulation in Chapter 5 (see Table 6.1). But this mapping is carried out T times in the dynamic case, instead of once for the static case.

Figure 6.1 shows an example with three rival models $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$ from Bania [2019]. The models are linear continuous-time models of the form

$$\mathcal{M}_i : \quad \begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}\tau} x(\tau) = \mathbf{A}_i x(\tau) + \mathbf{B}_i u(\tau) + \mathbf{C}_i w(\tau) \,, \\ \quad z(\tau) = [1,\, 0,\, \ldots,\, 0]\, x(\tau) \,, \\ \quad y_t = z(\tau_t) + v_t \,, \end{cases} \tag{6.0.4}$$

with the matrices $\mathbf{A}_i$, $\mathbf{B}_i$ and $\mathbf{C}_i$ defined by [Bania, 2019]

$$\mathcal{M}_1 : \quad \mathbf{A}_1 = -1 \,, \quad \mathbf{B}_1 = 1 \,, \quad \mathbf{C}_1 = 1 \,, \tag{6.0.5a}$$



(a) step input

(b) step input

(c) sinusoid input

(d) sinusoid input

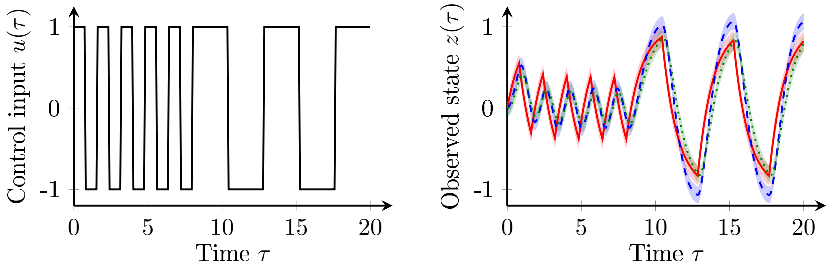
(e) random input

(f) random input

Figure 6.1: Example from Bania [2019], with three models $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$ defined in Equation (6.0.4) and Equation (6.0.5). (a), (c) and (e) show three different piece-wise constant control input as function of time τ, and (b), (d) and (e) show the corresponding observed state $z(\tau)$, plotted with two standard deviations of measurement noise.

$$\mathcal{M}_2: \quad \mathbf{A}_2 = \begin{bmatrix} 0 & 1 \\ -3 & -2.5 \end{bmatrix}, \quad \mathbf{B}_2 = \begin{bmatrix} 0 \\ 3 \end{bmatrix}, \quad \mathbf{C}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \tag{6.0.5b}$$

$$\mathcal{M}_3: \quad \mathbf{A}_3 = \begin{bmatrix} 0 & 1 & 0 \\ -3 & -3.5 & 1 \\ 0 & 0 & -10 \end{bmatrix}, \quad \mathbf{B}_3 = \begin{bmatrix} 0 \\ 0 \\ 30 \end{bmatrix}, \quad \mathbf{C}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{6.0.5c}$$



(a) optimised input

(b) optimised input

Figure 6.2: Example from Bania [2019], with three models $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$ defined in Equation (6.0.4) and Equation (6.0.5). (a) shows an optimised piece-wise constant control input, and (b) the corresponding observed state $z(\tau)$.

The initial latent state is $\boldsymbol{x}_0^{(i)} = [0, \ldots, 0]^\top$, and the process noise $w(\tau)$ and the measurement noise v_t are both Gaussian distributed with variances $\sigma_x^2 = \sigma_y^2 = 2.5\text{E-}3$. Let the piece-wise constant control inputs u_t lie in the range $[-1, 1]$. In Figure 6.1 there are three different piece-wise constant control signals: (a) a step input, (c) a scaled sinusoid input, and (e) uniformly distributed random inputs. The models $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$ are all different, and yield different interpretations of mechanisms in the underlying system. Despite this, as shown in Figure 6.1, it is non-trivial to find control inputs that yield different enough model predictions to allow for model discrimination under reasonable levels of process and measurement noise.

Naively applying a step or random control input is not guaranteed to produce data that will help us solve the model discrimination problem. A naive control input may also result in violations of system safety constraints. Figure 6.2 shows an example of a different control signal, optimised to yield sufficiently large differences in the model predictions that it may allow us to discriminate between the models. Often, more than one experiment is required to discriminate between models. This chapter is concerned with finding optimal control inputs (and other experimental conditions) to aid us with discrimination of dynamic models under uncertainty and subject to constraints.

6.1 Existing Work

Extensions of the methods in Section 3.2 exist for design of dynamic experiments. Espie and Macchietto [1989] and Asprey and Macchietto [2000] consider discrimination between multiple analytical continuous-time models and formulate an optimal control problem. Both works consider the Hunter and Reiner [1965] criterion, i.e. the Mahalanobis distance between model predictions, for design of experiments. Espie and Macchietto [1989] compare the results of using an optimal constant control inputs *versus* an optimal dynamic control input. Asprey and Macchietto [2000] discuss accounting for model parameter uncertainty by taking the expected value over the design criterion, or by considering worst-case parameter realisations. Chen and Asprey [2003] also consider continuous-time models and use a Laplace approximation for the model parameter covariance and linear propagation of the Gaussian model parameter uncertainty to approximate the marginal predictive distributions.

Skanda and Lebiedz [2010] assume Gaussian measurement noise to derive an expression for the Kullback-Leibler (KL) divergence between the predictive distributions of two rival models (with the same number of states). They include the measurement time points $\mathcal{T}_{\text{meas}}$ as variables in the optimisation problem, together with the initial state $x(0) = \boldsymbol{x}_0$ and control inputs $\boldsymbol{u}_{0:T-1}$. The control inputs Skanda and Lebiedz [2010] consider are additive perturbations to the state, and they assume the states cannot be measured and perturbed in the same time step. Skanda and Lebiedz [2013] extend the setup of Skanda and Lebiedz [2010] by considering model parameter uncertainty. They propose a robust optimisation formulation

$$\operatorname*{arg\,max}_{\substack{\mathcal{T}_{\text{meas}} \\ \boldsymbol{x}_0 \in \mathcal{X} \\ \boldsymbol{u}_{0:T-1} \in \mathcal{U}}} \min_{\substack{i,j \in \{1,\dots,M\} \\ i \neq j}} \min_{\substack{\boldsymbol{\theta}_i \in \boldsymbol{\Theta}_i \\ \boldsymbol{\theta}_j \in \boldsymbol{\Theta}_j}} \sum_{t \in \mathcal{T}_{\text{meas}}} \mathrm{KL}\left[p(\boldsymbol{y}_t \,|\, \boldsymbol{\theta}_i) \,\|\, p(\boldsymbol{y}_t \,|\, \boldsymbol{\theta}_j)\right]$$

subject to constraints, with model parameter spaces $\boldsymbol{\Theta}_i$ and $\boldsymbol{\Theta}_j$ and $p(\boldsymbol{y}_t \,|\, \boldsymbol{\theta}_i)$ denoting the predictive distribution at time step t given model i with parameter values $\boldsymbol{\theta}_i$.

None of Espie and Macchietto [1989], Asprey and Macchietto [2000], Chen and Asprey [2003], or Skanda and Lebiedz [2010, 2013] consider process noise or uncertainty in the initial state

$x(0) = \boldsymbol{x}_0$ or control signal $u(\tau)$. Nor do any of them, when solving the optimisation problem subject to path constraints on the observed states $\boldsymbol{z}_{1:T}$, account for the uncertainty in the observed states $\boldsymbol{z}_{1:T}$ predictions.

Cheong and Manchester [2014a] consider non-parametric linear discrete-time systems with process noise (but no separate measurement noise) and uncertainty in the initial states $\boldsymbol{x}_0$. For optimising the control signal they consider design criteria based on the pairwise difference in models' score in the χ^2 goodness-of-fit test. Cheong and Manchester [2014b] extend this approach by deriving a model discrimination control law for model predictive control. Though Cheong and Manchester [2014a,b] consider path constraints in the observed states $\boldsymbol{z}_{1:T}$, they do not account for the uncertainty in the observed states $\boldsymbol{z}_{1:T}$ predictions.

Streif et al. [2014] and Mesbah et al. [2014a] look at cases of two rival non-linear models $\mathcal{M}_1$ and $\mathcal{M}_2$ with multiplicative measurement noise

$$\mathcal{M}_i : \quad \begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}\tau} x(\tau) = f_i(x(\tau), u(\tau), \boldsymbol{\theta}_i) \,, \\ z(\tau) = g_i(x(\tau), u(\tau), \boldsymbol{\theta}_i) \,, \\ \boldsymbol{y}_t = \mathrm{diag}(\boldsymbol{1} + \boldsymbol{w}_t)\boldsymbol{z}_t \end{cases}$$

where f_i and g_i, $i \in \{1, 2\}$, are polynomial functions. They consider uncertainty in the initial states and model parameters using polynomial chaos expansions, from which they compute higher moments of the predictive distributions—"a computationally formidable task" according to Streif et al. [2014]. They discretise the control signal and solve the design problem by minimising the norm of the control signal such that the divergence between the predictive distributions is greater than or equal to some threshold value. The divergence can be computed using the predictive distributions' higher moments [Streif et al., 2014] or through Markov Chain Monte Carlo integration [Mesbah et al., 2014a].

Keesman and Walter [2014] look at continuous-time models of the kind $\frac{\mathrm{d}}{\mathrm{d}\tau} y(\tau) = f(y(\tau)) + bu(\tau)$. They define the Hamiltonian and from this derive an optimal control law in closed form for two rival models. This requires gradient information of *at least* the first order. They do not account for parametric uncertainty, process noise or measurement noise.

Bania [2019] consider non-parametric linear discrete-time models with process and measure-

Ref.	(a)	(b)	(c)	(d)	(e)	(f)
Non-linear f	✓	✓		✓	(✓)	✓
Discrete-time models			✓		✓	✓
Continuous-time models	✓	✓		✓		✓
Black-box models						✓
Measurement noise	✓	✓	(✓)	✓	✓	✓
Process noise			✓		✓	✓
Uncertain $\boldsymbol{x}_0$			✓	✓		✓
Uncertain $\boldsymbol{u}_t$						✓
Uncertain $\boldsymbol{\theta}$	✓	(✓)		✓		✓
Optimise $\boldsymbol{x}_0$	✓	✓				✓
Optimise $\boldsymbol{u}_{0:T-1}$	✓	✓	✓	✓	✓	✓
Optimise $\mathcal{T}_{\text{meas}}$	✓	✓				✓
Path constraints	(✓)	(✓)	(✓)			✓

Table 6.2: References: (a) Chen and Asprey [2003], (b) Skanda and Lebiedz [2013], (c) Cheong and Manchester [2014a,b], (d) Streif et al. [2014], (e) Bania [2019], (f) This work. Bracketed check marks: Bania [2019] discuss how their approach can be extended to non-linear transition functions f; Cheong and Manchester [2014a,b] has one noise signal that affects both latent states and measurements; Skanda and Lebiedz [2013] use a robust problem formulation instead of marginalising out the model parameters; Chen and Asprey [2003], Skanda and Lebiedz [2013] and Cheong and Manchester [2014a,b] solve the design of experiments optimisation problem subject to path constraints that do not account for the uncertainty in the predicted states.

ment noise. By looking at the mutual information between choice of model and observed output, they derive an optimisation formulation based on minimising the probability of selecting the wrong model. They mention how to extend their approach to non-linear models.

Table 6.2 summarises the different approaches in literature in columns (a)–(e). Column (f) shows the novelty of the approach proposed in this chapter compared to existing literature, e.g. accounting for more possible types of uncertainty as well as accommodating black-box models. As described in Section 3.2.3, many industrially relevant models may effectively be black boxes, i.e. non-analytical in the sense that gradients of the function f are not readily available. This may be due to legacy code, switches (`if`/`else` statements) or models that require solving an optimisation problem (e.g. minimising the Gibbs free energy). Accommodating black-box models lets us be agnostic with regards to the model software implementation, which is desirable since it flexibly (i) allows faster model prototyping and development, and (ii) satisfies the personal preferences of researchers and engineers.

To the best extent of our knowledge, there is no existing work using a data-driven approach to tackle design of dynamic experiments for model discrimination. As mentioned, the dimensionality of the input and output spaces for the rival models can grow large when we discretise in time. The computational cost associated with solving design of dynamic experiments problems using the data-driven approaches in Section 3.3 would likely become insurmountable. Likewise, naively applying the Chapter 5 approach is infeasible, since it would operate directly on the mapping from $(\boldsymbol{x}_0, \boldsymbol{u}_{0:T-1}, \mathcal{T}_{\text{meas}})$ to $\boldsymbol{z}_{1:T}$; both the input and output dimensionality would grow linearly with the number of time steps. The dimensionality of the input space would be too high to perform accurate GP inference, and the number of GP surrogates would have to equal the output dimensionality, which would be very expensive memory-wise. Therefore, we propose an alternative approach to Chapter 5 for solving design of dynamic experiments for discrimination of black-box models.

6.2 A Generalised Formulation

This section describes a generalised formulation for design of dynamic experiments for model discrimination. This formulation makes it possible to create a common optimal experimental design framework for discrimination between rival dynamic models with:

- Analytical or black-box transition functions f_i.
- Continuous- or discrete-time models.
- Multiple types of uncertainty.

Consider an ordered set of discrete time instances $\mathcal{T} = \{\tau_0, \dots, \tau_T\}$. These are indexed as $t = 0, \dots, T$, with $t = 0$ the starting time step of an experiment, and $t = T$ the final time step. For simplicity, we abuse notation by writing $t \in \mathcal{T}$. For discrete-time models, assume that the time points in $\mathcal{T}$ are equidistant, such that $\tau_t = \tau_0 + t\Delta\tau$ for some $\Delta\tau > 0$. Each time step t has a corresponding control input $\boldsymbol{u}_t$ to the system, and the control signal $u(\tau)$ is piece-wise constant, with $u(\tau') \equiv \boldsymbol{u}_t$ for $\tau_t \leq \tau' < \tau_{t+1}$. Measurements $\boldsymbol{y}_t$ are taken at times $\tau_t \in \mathcal{T}_{\text{meas}} \subseteq \mathcal{T}$.

Assume we are given M rival state space models $\mathcal{M}_1, \dots, \mathcal{M}_M$. For a latent state $\boldsymbol{x}_t \in \mathbb{R}^{D_{x,i}}$ and control input $\boldsymbol{u}_t \in \mathbb{R}^{D_u}$ at time step t, and model parameter vector $\boldsymbol{\theta}_i \in \mathbb{R}^{D_{\theta,i}}$, model $\mathcal{M}_i$ specifies a latent state transition, observation and measurement model

$$\mathcal{M}_i : \begin{cases} \boldsymbol{x}_{t+1}^{(i)} = \phi_i \left[\boldsymbol{x}_{t-1}^{(i)}, \, \boldsymbol{u}_{t-1}, \, \boldsymbol{\theta}_i \,\middle|\, f_i \right] + \boldsymbol{w}_t^{(i)} \,, & \text{(Latent state transition)} \\ \boldsymbol{z}_t^{(i)} = \mathbf{H}_i \boldsymbol{x}_t^{(i)} \,, & \text{(Observed states)} \\ \boldsymbol{y}_t^{(i)} = \boldsymbol{z}_t^{(i)} + \boldsymbol{v}_t^{(i)} \,, & \text{(Noisy measurement)} \end{cases} \tag{6.2.1}$$

where $\mathbf{H}_i$ is a matrix selecting the observed states. The number of latent states $D_{x,i}$ may differ between models $\mathcal{M}_i$, but the number of observed states $D_{z,i} = D_z$ for all models. The operator ϕ_i is defined as

$$\phi_i \left[\boldsymbol{x}_{t-1}^{(i)}, \, \boldsymbol{u}_{t-1}, \, \boldsymbol{\theta}_i \,\middle|\, f_i \right] = \boldsymbol{x}_{t-1}^{(i)} + \int_{\tau_{t-1}}^{\tau_t} f_i(x^{(i)}(\tau), \, \boldsymbol{u}_{t-1}, \, \boldsymbol{\theta}_i) \mathrm{d}\tau \,, \tag{6.2.2a}$$

for continuous-time models,

$$\phi_i \left[\boldsymbol{x}_{t-1}^{(i)}, \, \boldsymbol{u}_{t-1}, \, \boldsymbol{\theta}_i \,\middle|\, f_i \right] = f_i(\boldsymbol{x}_{t-1}^{(i)}, \, \boldsymbol{u}_{t-1}, \, \boldsymbol{\theta}_i) \,, \tag{6.2.2b}$$

for discrete-time models, and

$$\phi_i \left[\boldsymbol{x}_{t-1}^{(i)}, \, \boldsymbol{u}_{t-1}, \, \boldsymbol{\theta}_i \,\middle|\, f_i \right] = \boldsymbol{x}_{t-1}^{(i)} + f_i(\boldsymbol{x}_{t-1}^{(i)}, \, \boldsymbol{u}_{t-1}, \, \boldsymbol{\theta}_i) \,, \tag{6.2.2c}$$

for discrete-time models with a Δ-transition.

6.2.1 Problem Uncertainty

In Equation (6.2.1), $\boldsymbol{w}_t^{(i)}$ is zero-mean Gaussian distributed process noise, and $\boldsymbol{v}_t^{(i)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_y)$ is independent and identically distributed measurement noise. The process noise $\boldsymbol{w}_t^{(i)}$ has known covariance $\boldsymbol{\Sigma}_{x,i}$ if $\mathcal{M}_i$ is a discrete-time model, or covariance $(\tau_{t+1} - \tau_t)\boldsymbol{\Sigma}_{x,i}$ if $\mathcal{M}_i$ is a continuous-time model. We assume the measurement noise covariance $\boldsymbol{\Sigma}_y$ is known but may be different ($\boldsymbol{\Sigma}_y = \boldsymbol{\Sigma}_y^{(i)}$) for different models $\mathcal{M}_i$[1]. Apart from uncertainty due to

[1] Note that model-specific measurement noise covariances interferes with some design criterion definitions.

process noise and measurement noise, we may have uncertainty in the control input, initial latent state and model parameters. Uncertainty in the control input[2] and initial latent state is a reasonable assumption since we cannot control any physical system to infinite precision.

The control input $\boldsymbol{u}_t \sim \mathcal{N}(\hat{\boldsymbol{u}}_t, \boldsymbol{\Sigma}_{u,t})$ at time step t is Gaussian distributed with mean given by a user-specified desired control input $\hat{\boldsymbol{u}}_t$ and covariance $\boldsymbol{\Sigma}_{u,t}$. A noisy control signal may be for example (i) the temperature in a computer-filled laboratory hit by direct sunlight, (ii) controlling the inlet velocity in a reactor where a valve-opening mechanism has finite precision, or (iii) the amount of substrate added in a fermentation experiment. Some properties, e.g. temperature, may be specified by an engineer but controlled by sensors and PID controllers. The control covariance would generally be assumed to be relatively small, or the experiment would not be controllable. The control covariance $\boldsymbol{\Sigma}_{u,t} = \boldsymbol{\Sigma}_{u,t}^{(i)}$ may be model-dependent. For simplicity, let the control inputs $\boldsymbol{u}_t$ be piece-wise constant and the control covariance constant $\boldsymbol{\Sigma}_{u,t} = \boldsymbol{\Sigma}_u$. Simple extensions of the framework could accommodate control inputs described e.g. by piece-wise polynomials or time-dependent control covariance. Let $\hat{\boldsymbol{u}}_{0:T-1} = \{\hat{\boldsymbol{u}}_0, \dots, \hat{\boldsymbol{u}}_{T-1}\}$ denote the sequence of user-specified control inputs.

The initial latent state $\boldsymbol{x}_0^{(i)} \sim \mathcal{N}(\boldsymbol{\mu}_0^{(i)}(\hat{\boldsymbol{x}}_0), \boldsymbol{\Sigma}_0^{(i)})$ is dependent on some user-specified initial state settings $\hat{\boldsymbol{x}}_0 \in \mathbb{R}^{D_x}$ common for all models. Different models may have different number of latent states, but the number of variables to optimise is fixed. The initial state setting $\hat{\boldsymbol{x}}_0$ is the variable to optimise, and we assume there is a known mapping $h_{\hat{\boldsymbol{x}},\boldsymbol{x}}^{(i)} : \mathbb{R}^{D_x} \to \mathbb{R}^{D_{x,i}}$ from $\hat{\boldsymbol{x}}_0$ to each model's initial latent state space. The initial state setting $\hat{\boldsymbol{x}}_0$ may also in this manuscript be referred to as the *desired initial state*.

The model parameters $\boldsymbol{\theta}_i \sim \mathcal{N}(\hat{\boldsymbol{\theta}}_i, \boldsymbol{\Sigma}_{\theta,i})$ are Gaussian distributed with mean given by the maximum *a posteriori* parameter estimate $\hat{\boldsymbol{\theta}}_i$. The model parameter covariance $\boldsymbol{\Sigma}_{\theta,i}$ is computed using a Laplace approximation [Chen and Asprey, 2003], as described in Section 3.2.

6.2.2 Problem Formulation

The initial state settings $\hat{\boldsymbol{x}}_0$ and the control inputs $\hat{\boldsymbol{u}}_{0:T-1}$ are the main means by which the experimental outcomes are controlled. For continuous-time models $\mathcal{M}_i$ we may also want to

[2] Also known as *input disturbances*

optimise the measurement time points $\mathcal{T}_{\text{meas}}$. Hence, we formulate the optimisation problem of design of dynamic experiments for model discrimination as

$$\begin{aligned} \underset{\substack{\hat{\boldsymbol{x}}_0,\, \hat{\boldsymbol{u}}_{0:T-1} \\ \mathcal{T}_{\text{meas}}}}{\arg\max} \; & \sum_{t \in \mathcal{T}_{\text{meas}}} D_{**}\left(\boldsymbol{y}_t^{(1)}, \dots, \boldsymbol{y}_t^{(M)}\right) \\ \text{s.t.} \quad & \forall t \in \{1, \dots, T\}\,,\, \forall i \in \{1, \dots, M\}: \\ & \mathcal{M}_i : \begin{cases} \boldsymbol{x}_t^{(i)} = \phi_i \left[\boldsymbol{x}_{t-1}^{(i)},\, \boldsymbol{u}_{t-1},\, \boldsymbol{\theta}_i \,\middle|\, f_i\right] + \boldsymbol{w}_{t-1}^{(i)}\,, \\ \boldsymbol{z}_t^{(i)} = \mathbf{H}_i \boldsymbol{x}_t^{(i)}\,, \\ \boldsymbol{y}_t^{(i)} = \boldsymbol{z}_t^{(i)} + \boldsymbol{v}_t\,, \end{cases} \\ & \begin{array}{r|r|r} C_{x_0}(\hat{\boldsymbol{x}}_0) \geq \mathbf{0}\,, & C_x(\boldsymbol{x}_t^{(i)}) \geq \mathbf{0}\,, & C_{\mathcal{T}}(\mathcal{T}_{\text{meas}}) \geq \mathbf{0}\,, \\ C_u(\hat{\boldsymbol{u}}_t) \geq \mathbf{0}\,, & C_z(\boldsymbol{z}_t^{(i)}) \geq \mathbf{0}\,, & \end{array} \end{aligned} \tag{6.2.3}$$

where D_{**} is the design criterion, e.g. one of the design criteria described in Section 3.2 and Section 4.1, and C_{x_0}, C_u, C_x, C_z and $C_{\mathcal{T}}$ are constraints on the corresponding variables (see Section 6.5). The operators ϕ_i are defined as in Equation (6.2.2).

6.3 Latent State Transition

Let us study a single model $\mathcal{M} = \mathcal{M}_i$ with corresponding transition operator $\phi = \phi_i$. For a latent state distribution $\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$, the predicted observed state and measurement distributions are given by

$$\begin{aligned} \boldsymbol{z}_t &\sim \mathcal{N}(\mathbf{H}\boldsymbol{\mu}_t,\, \mathbf{H}\boldsymbol{\Sigma}_t\mathbf{H}^\top)\,, \\ \boldsymbol{y}_t &\sim \mathcal{N}(\mathbf{H}\boldsymbol{\mu}_t,\, \mathbf{H}\boldsymbol{\Sigma}_t\mathbf{H}^\top + \boldsymbol{\Sigma}_y)\,. \end{aligned}$$

Solving the optimisation problem in Equation (6.2.3) requires the predictive distribution of the latent state $\boldsymbol{x}_t$. This means propagating the uncertainty in the inputs to the transition function f to its outputs. We assume we know *a priori* whether f is an analytical function or a black box, i.e. whether we *do* (analytical) or *do not* (black box) have derivative information of f with respect to its inputs. The derivative information is required for closed-form uncertainty propagation from inputs to outputs using Taylor approximations, as in Section 3.2 and Section 5.1.

To obtain derivative information from a black-box transition function f, we place independent GP priors $f_{(d)} \sim \mathcal{GP}(m_{(d)}(\cdot),\, k_{x,(d)}(\cdot,\cdot)k_{u,(d)}(\cdot,\cdot)k_{\theta,(d)}(\cdot,\cdot))$ on output dimensions $d = 1, \dots, D_x$ of f. To simplify notation, let $f(\tilde{\boldsymbol{x}}_t)$ denote the transition function f evaluated at the concatenated latent state, control input and model parameters $\tilde{\boldsymbol{x}}_t = [\boldsymbol{x}_t^\top, \boldsymbol{u}_t^\top, \boldsymbol{\theta}^\top]^\top$, $\tilde{\boldsymbol{x}}_t \in \mathbb{R}^{D_x + D_u + D_\theta}$. Additionally, let $\mu_f(\tilde{\boldsymbol{x}}_t) = \mathbb{E}_f[f(\tilde{\boldsymbol{x}}_t)]$ and $\Sigma_f(\tilde{\boldsymbol{x}}_t) = \mathbb{V}_f[f(\tilde{\boldsymbol{x}}_t)]$, such that

$$f(\tilde{\boldsymbol{x}}_t) \sim \mathcal{N}(\mu_f(\tilde{\boldsymbol{x}}_t), \Sigma_f(\tilde{\boldsymbol{x}}_t)) \equiv \begin{cases} \mathcal{N}(f(\tilde{\boldsymbol{x}}_t), \boldsymbol{0}) \,, & f \text{ analytical} \\ \mathcal{N}(\mu(\tilde{\boldsymbol{x}}_t), \Sigma(\tilde{\boldsymbol{x}}_t)) \,, & f \text{ black box} \end{cases} \tag{6.3.1}$$

where the posterior GP mean $\mu(\cdot)$ and covariance $\Sigma(\cdot)$ are computed as in Equation (2.3.3) on page 43.

Given an initial latent state estimate $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$, a sequence of control inputs $\boldsymbol{u}_t \sim \mathcal{N}(\hat{\boldsymbol{u}}_t, \boldsymbol{\Sigma}_u)$, $t = 0, \dots, T-1$, a model parameter posterior $\boldsymbol{\theta} \sim \mathcal{N}(\hat{\boldsymbol{\theta}}, \boldsymbol{\Sigma}_\theta)$, and the latent state transition described by Equation (6.2.1), we wish to find the approximate latent state distribution $\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$ at any time step $1 \leq t \leq T$, with mean and covariance given by the moments

$$\boldsymbol{\mu}_t = \mathbb{E}_{f, \boldsymbol{x}_0, \boldsymbol{u}_{0:t-1}, \boldsymbol{\theta}, \boldsymbol{w}_{0:t-1}} \left[\boldsymbol{x}_t \right] \,, \tag{6.3.2a}$$

$$\boldsymbol{\Sigma}_t = \mathbb{V}_{f, \boldsymbol{x}_0, \boldsymbol{u}_{0:t-1}, \boldsymbol{\theta}, \boldsymbol{w}_{0:t-1}} \left[\boldsymbol{x}_t \right] \,. \tag{6.3.2b}$$

We assume that the control covariance $\boldsymbol{\Sigma}_u$, model parameter covariance $\boldsymbol{\Sigma}_\theta$ and process noise covariance $\boldsymbol{\Sigma}_x$ are all constant and independent of $\boldsymbol{x}_t$ and $\hat{\boldsymbol{u}}_t$. The latent state mean $\boldsymbol{\mu}_t$ and covariance $\boldsymbol{\Sigma}_t$ depend on the form of the transition operator ϕ. Three different types of transitions are considered here: discrete-time steps, discrete-time Δ-transition steps, and continuous transitions.

Assuming the latent state, control input and model parameters are Gaussian distribution, the concatenated vector has Gaussian distribution $\tilde{\boldsymbol{x}}_t \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}_t, \tilde{\boldsymbol{\Sigma}}_t)$ with

$$\tilde{\boldsymbol{\mu}}_t = \begin{bmatrix} \boldsymbol{\mu}_t \\ \hat{\boldsymbol{u}}_t \\ \hat{\boldsymbol{\theta}} \end{bmatrix} , \quad \tilde{\boldsymbol{\Sigma}}_t = \begin{bmatrix} \boldsymbol{\Sigma}_t & \boldsymbol{0} & \operatorname{cov}(\boldsymbol{x}_t, \boldsymbol{\theta}) \\ \boldsymbol{0} & \boldsymbol{\Sigma}_u & \boldsymbol{0} \\ \operatorname{cov}(\boldsymbol{x}_t, \boldsymbol{\theta})^\top & \boldsymbol{0} & \boldsymbol{\Sigma}_\theta \end{bmatrix} , \tag{6.3.3}$$

where $\operatorname{cov}(\boldsymbol{x}_0, \boldsymbol{\theta}) \equiv \boldsymbol{0}$, and $\operatorname{cov}(\boldsymbol{x}_t, \boldsymbol{u}_t) \equiv \boldsymbol{0}$ (assuming $\boldsymbol{u}_t \neq \boldsymbol{u}_{t-1}$) since the latent state cannot depend on future control inputs.

To simplify notation, let $\nabla_{\boldsymbol{\xi}} g$, with $g \in \{f, \mu_f, \Sigma_f, \dots\}$, denote the partial derivative of $g(\boldsymbol{\xi})$ with respect to a variable $\boldsymbol{\xi}$, evaluated at the point $\mathbb{E}[\boldsymbol{\xi}]$.

6.3.1 Discrete-Time State Space Models

The discrete-time state space model assumes the latent state transition is described by

$$\mathcal{M}: \quad \boldsymbol{x}_{t+1} = f(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\theta}) + \boldsymbol{w}_t \, ,$$

with process noise $\boldsymbol{w}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_x)$. Using a first-order Taylor expansion of $\mu_f(\tilde{\boldsymbol{x}})$ around $\tilde{\boldsymbol{x}}_{t-1} = \tilde{\boldsymbol{\mu}}_{t-1}$ (see Section 2.7.1), the mean and variance of the latent state at time step $t \geq 1$ in Equation (6.3.2a) are approximately given by

$$\begin{aligned} \boldsymbol{\mu}_t &\approx \mu_f(\tilde{\boldsymbol{\mu}}_{t-1}) \, , \\ \boldsymbol{\Sigma}_t &\approx \nabla_{\tilde{\boldsymbol{x}}_{t-1}} \boldsymbol{\mu}_t \tilde{\boldsymbol{\Sigma}}_{t-1} \left(\nabla_{\tilde{\boldsymbol{x}}_{t-1}} \boldsymbol{\mu}_t \right)^\top + \boldsymbol{\Sigma}_x + \Sigma_f(\tilde{\boldsymbol{\mu}}_t) \, , \\ \operatorname{cov}(\boldsymbol{x}_t, \boldsymbol{\theta}) &\approx \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_t \operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta})^\top + \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_t \boldsymbol{\Sigma}_\theta \, . \end{aligned} \tag{6.3.4}$$

Note that $\nabla_{\tilde{\boldsymbol{x}}_{t-1}} \boldsymbol{\mu}_t \in \mathbb{R}^{D_x \times (D_x + D_u + D_\theta)}$. Derivatives of $\boldsymbol{\mu}_t$ and $\boldsymbol{\Sigma}_t$ with respect to $\boldsymbol{\mu}_{t-1}$, $\boldsymbol{\Sigma}_{t-1}$ and $\hat{\boldsymbol{u}}_{t-1}$ are calculated following the standard rules of matrix calculus, and requires second-order derivative information of f or the GP prediction (see Appendix A.2).

6.3.2 Discrete-Time Model with Δ-Transition

The discrete-time state space model with a Δ-transition assumes the latent state transition is described by

$$\mathcal{M}: \quad \boldsymbol{x}_{t+1} = \boldsymbol{x}_t + f(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\theta}) + \boldsymbol{w}_t \,,$$

with process noise $\boldsymbol{w}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_x)$. Using a first-order Taylor expansion of $\mu_f(\tilde{\boldsymbol{x}})$ around $\tilde{\boldsymbol{x}}_{t-1} = \tilde{\boldsymbol{\mu}}_{t-1}$ (see Section 2.7.1), the mean and variance of the latent state at time step $t \geq 1$ in Equation (6.3.2a) are approximately given by

$$\begin{aligned} \boldsymbol{\mu}_t &\approx \boldsymbol{\mu}_{t-1} + \mu_f(\tilde{\boldsymbol{\mu}}_{t-1}) \,, \\ \boldsymbol{\Sigma}_t &\approx \nabla_{\tilde{\boldsymbol{x}}_{t-1}} \boldsymbol{\mu}_t \tilde{\boldsymbol{\Sigma}}_{t-1} \left(\nabla_{\tilde{\boldsymbol{x}}_{t-1}} \boldsymbol{\mu}_t \right)^\top + \boldsymbol{\Sigma}_x + \Sigma_f(\tilde{\boldsymbol{\mu}}_t) \,, \\ \operatorname{cov}(\boldsymbol{x}_t, \boldsymbol{\theta}) &\approx \operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_t \operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta})^\top + \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_t \boldsymbol{\Sigma}_\theta \,. \end{aligned} \tag{6.3.5}$$

Note that $\nabla_{\boldsymbol{x}_{t-1}} \boldsymbol{\mu}_t = \mathbf{I} + \nabla_{\boldsymbol{x}_{t-1}} \mu_f$ with the Δ-transition model, and that $\nabla_{\tilde{\boldsymbol{x}}_{t-1}} \boldsymbol{\mu}_t \in \mathbb{R}^{D_x \times (D_x + D_u + D_\theta)}$. Derivatives of $\boldsymbol{\mu}_t$ and $\boldsymbol{\Sigma}_t$ with respect to $\boldsymbol{\mu}_{t-1}$, $\boldsymbol{\Sigma}_{t-1}$ and $\hat{\boldsymbol{u}}_{t-1}$ are calculated following the standard rules of matrix calculus, and requires second-order derivative information of f or the GP prediction (see Appendix A.3).

It is common in GP regression to use zero-mean GP priors ($m_{(d)}(\cdot) \equiv 0$) to simplify calculations. The zero-mean prior suitable is suitable for the Δ-transition state space model formulation [Ko et al., 2007; Deisenroth and Rasmussen, 2011].

6.3.3 Continuous-Time State Space Models

For continuous-time state space models, the state transition between time steps is described by the solution to the system of ordinary differential equations

$$\mathcal{M}: \quad \begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}\tau} x(\tau) = f_i(x(\tau), u(\tau), \boldsymbol{\theta}) + w(\tau) \,, \\ x(\tau_0) \equiv \boldsymbol{x}_0 \,, \end{cases} \tag{6.3.6}$$

with process noise distribution $w(\tau) \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_x)$. The control input $u(\tau_t)$ may be a continuous function of the time τ, but we will assume it is piece-wise constant. Model $\mathcal{M}$'s state prediction at time step t is given by the solution $\boldsymbol{x}_t = x(\tau_t)$ to Equation (6.3.6) at time τ_t.

Let $\tilde{x}(\tau)$ denote the continuous concatenated latent state, control input and model parameters $\tilde{x}(\tau) = [x(\tau)^\top, u(\tau)^\top, \boldsymbol{\theta}^\top]^\top$, with Gaussian distribution $\tilde{x}(\tau) \sim \mathcal{N}(\tilde{\mu}(\tau), \tilde{\Sigma}(\tau))$, and let $\tilde{\mu}_f$ denote the concatenated transition function

$$\tilde{\mu}_f(\tau) = \left[\mu_f(\tilde{\mu}(\tau))^\top,\, \mathbf{0},\, \mathbf{0}\right]^\top, \quad \tilde{\mu}_f(\tau) \in \mathbb{R}^{D_x+D_u+D_\theta}\,.$$

We find the latent state prediction $\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{\Sigma}_t)$ at time step t by extracting the corresponding elements from $\tilde{\mu}(\tau_t)$ and $\tilde{\Sigma}(\tau)$ (see Equation (6.3.3)) which we compute by solving the following system of ordinary differential equations

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}\tau}\tilde{\mu}(\tau) = \tilde{\mu}_f(\tau)\,, \\ \dfrac{\mathrm{d}}{\mathrm{d}\tau}\tilde{\Sigma}(\tau) = \nabla_{\tilde{\mu}(\tau)}\tilde{\mu}_f\tilde{\Sigma}(\tau) + \tilde{\Sigma}(\tau)\left(\nabla_{\tilde{\mu}(\tau)}\tilde{\mu}_f\right)^\top + \mathrm{diag}(\Sigma_f(\tilde{\mu}(\tau)) + \mathbf{\Sigma}_x,\, \mathbf{0},\, \mathbf{0})\,, \\ \tilde{\mu}(\tau_{t-1}) \equiv \tilde{\boldsymbol{\mu}}_{t-1}\,, \\ \tilde{\Sigma}(\tau_{t-1}) \equiv \tilde{\mathbf{\Sigma}}_{t-1}\,. \end{cases} \tag{6.3.7}$$

Derivatives of $\boldsymbol{\mu}_{t+1}$ and $\mathbf{\Sigma}_{t+1}$ with respect to $\boldsymbol{\mu}_t$, $\mathbf{\Sigma}_t$ and $\hat{\boldsymbol{u}}_t$ are calculated by integrating over the chain rule, and require second-order derivative information of f or the GP prediction (see Appendix A.4).

6.3.4 Combining Original Transition Function and GP Surrogate

Equation (6.3.1) places a GP prior on the transition function f if f is black-box. The reason for replacing black-box transition functions f with GP surrogates is that they allow for approximating the derivatives of f, as $\nabla f \approx \nabla \mu$, without resorting to finite difference approximations. Finite difference approximations can be expensive if f is expensive to evaluate or the input space is large, especially for second-order (and higher) derivatives.

Equation (6.3.1) presents a binary choice: either f is analytical, in which case $\mu_f(\cdot) = f(\cdot)$ and $\nabla\mu_f(\cdot) = \nabla f(\cdot)$, or f is a black box, in which case it is replaced entirely with a GP

Fully analytical approach	Fully black-box approach	Third approach
$\mu_f(\cdot) = f(\cdot)\,,$ $\Sigma_f(\cdot) = \mathbf{0}\,,$ $\nabla\mu_f(\cdot) = \nabla f(\cdot)$	$\mu_f(\cdot) = \mu(\cdot)\,,$ $\Sigma_f(\cdot) = \Sigma(\cdot)\,,$ $\nabla\mu_f(\cdot) = \nabla\mu(\cdot)$	$\mu_f(\cdot) = f(\cdot)\,,$ $\Sigma_f(\cdot) = \mathbf{0}\,,$ $\nabla\mu_f(\cdot) \approx \nabla\mu(\cdot)$

Table 6.3: When f is analytical we use the fully analytical approach in Equation (6.3.1). When f is a black box we may choose to replace it in our computations with a GP surrogate for a fully black-box approach. The fully black-box approach has been used so far. A third approach is to use the original black-box transition function f for computing the mean $\mu_f(\cdot)$, but use the GP surrogate to approximate its gradients $\nabla\mu_f(\cdot)$.

surrogate during prediction and $\mu_f(\cdot) = \mu(\cdot)$ and $\nabla\mu_f(\cdot) = \nabla\mu(\cdot)$. This is the approach described so far, and the approach used in Chapter 5. The idea is that the approximation $f(\cdot) \approx \mu(\cdot)$ is accurate enough that it allows us to design optimal experiments. However, the approximation may not be accurate enough for robust model discrimination.

There is a third possible approach besides the fully analytical and fully black-box approaches. In the third approach both the original transition function f and the GP surrogate are used. Consider the following approximations

$$\begin{aligned}
\mathbb{E}_{f,\boldsymbol{x}_t,\boldsymbol{u}_t,\boldsymbol{\theta}}[f(\boldsymbol{x}_t,\boldsymbol{u}_t,\boldsymbol{\theta})] &\approx f(\boldsymbol{\mu}_t,\hat{\boldsymbol{u}}_t,\hat{\boldsymbol{\theta}})\,, \\
\mathbb{E}_{f,\boldsymbol{x}_t,\boldsymbol{u}_t,\boldsymbol{\theta}}[\nabla f(\boldsymbol{x}_t,\boldsymbol{u}_t,\boldsymbol{\theta})] &\approx \nabla\mu(\boldsymbol{\mu}_t,\hat{\boldsymbol{u}}_t,\hat{\boldsymbol{\theta}})\,,
\end{aligned}$$

i.e. use the original black-box transition function f for computing the predictive mean, and the GP surrogate only for approximating the derivatives of f. Table 6.3 compactly shows the difference between the fully analytical approach (where f is analytical), the fully black-box approach (where only the GP surrogate is used) and the proposed third approach (where both the original transfer function f and the GP surrogate are used). The third approach limits the use of the GP surrogates to the purpose for which they were introduced: approximating the gradients of f. Note that the third approach does not yield the additional uncertainty term associated with the GP surrogates, since $\Sigma_f(\cdot) \equiv \mathbf{0}$ when the exact transition function mean is used.

This third approach is appropriate to use during model discrimination, i.e. when analysing agreement between model predictions and experimental observations, even for expensive-

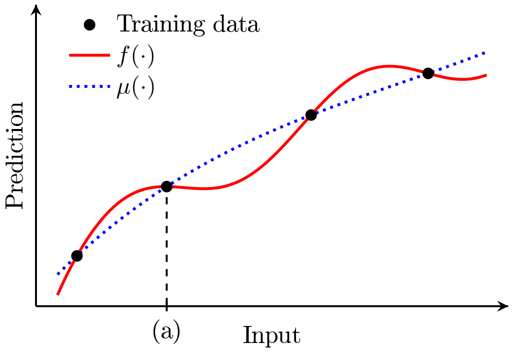


Figure 6.3: GP mean prediction $\mu(\cdot)$ of a function f given four training data points. Even if the approximation $f(\cdot) \approx \mu(\cdot)$ is sufficiently accurate, the approximation $\nabla f(\cdot) \approx \nabla \mu(\cdot)$ may cause numerical issues. In the region around the training data point denoted (a), the true gradient ∇f goes from positive to negative whereas the GP approximation $\nabla \mu$ remains constantly positive. This discrepancy between the true gradient and the gradient approximation may cause a numerical solver to have issues converging, or even throw an error.

to-evaluate transition functions during model discrimination. This reduces the risk that a model is discarded because of poor accuracy in the GP surrogate prediction.

However, there are also disadvantages associated with the third approach. If f is expensive to evaluate, we may still choose to use the fully black-box approach to speed up design of experiments, i.e. when solving the optimisation problem in Equation (6.2.3).

Table 6.3 shows that for the fully analytical and black-box approaches, the gradient $\nabla \mu_f$ is exact, whereas for the third approach the gradient $\nabla \mu_f(\cdot) \approx \nabla \mu(\cdot)$ is an approximation. This may cause numerical issues when the GP mean $\mu(\cdot)$—or rather the gradient $\nabla \mu(\cdot)$—does not capture the behaviour in the transition function f with sufficient accuracy, e.g. as in Figure 6.3. If a numerical solver is provided with inaccurate gradients it may e.g. converge slowly to a solution, time-out before reaching any solution, or even throw an error if a line search in the direction provided by a gradient fails to find a step in input space that will improve the value of the objective function. Therefore, from an optimisation point-of-view, it may be better to use the fully black-box approach when solving the optimisation problem in Equation (6.2.3), even if the transition function f is cheap to evaluate. On the one hand this means solving an approximation of the optimisation problem that we would ideally like to solve, but on the other hand we may be more likely to find a solution.

6.4 Latent State Filter Update

We use the notation $\boldsymbol{\mu}_{t'|t}$ and $\boldsymbol{\Sigma}_{t'|t}$ to denote the predicted latent state mean and covariance at time t' given measurements $\boldsymbol{y}_{1:t} = \{\boldsymbol{y}_1, \dots, \boldsymbol{y}_t\}$. Given latent state estimate $\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t})$ at time t, let $\boldsymbol{x}_{t+1} \,|\, \boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t} \sim \mathcal{N}(\boldsymbol{\mu}_{t+1|t}, \boldsymbol{\Sigma}_{t+1|t})$ denote the predicted latent state distribution at time $t+1$, computed as in Section 6.3.

Since we assume a linear observation model $\boldsymbol{z}_t = \mathbf{H}\boldsymbol{x}_t$, the posterior latent state distribution $\boldsymbol{x}_{t+1} \,|\, \boldsymbol{y}_{1:t+1} \sim \mathcal{N}(\boldsymbol{\mu}_{t+1|t+1}, \boldsymbol{\Sigma}_{t+1|t+1})$ given an observation at time $t+1$ can be computed using standard Kalman [1960] filter updates

$$\boldsymbol{\mu}_{t+1|t+1} = \boldsymbol{\mu}_{t+1|t} + \boldsymbol{\Sigma}_{t+1|t}\mathbf{H}^\top(\mathbf{H}\boldsymbol{\Sigma}_{t+1|t}\mathbf{H}^\top + \boldsymbol{\Sigma}_y)^{-1}(\boldsymbol{y}_{t+1} - \mathbf{H}\boldsymbol{\mu}_{t+1|t}) \,, \tag{6.4.1a}$$

$$\boldsymbol{\Sigma}_{t+1|t+1} = \boldsymbol{\Sigma}_{t+1|t} - \boldsymbol{\Sigma}_{t+1|t}\mathbf{H}^\top(\mathbf{H}\boldsymbol{\Sigma}_{t+1|t}\mathbf{H}^\top + \boldsymbol{\Sigma}_y)^{-1}\mathbf{H}\boldsymbol{\Sigma}_{t+1|t} \,. \tag{6.4.1b}$$

The posterior latent state distribution $p(\boldsymbol{x}_t \,|\, \boldsymbol{y}_{1:t})$ is useful for discrimination between rival models given observations.

The latent state filter update in Equation (6.4.1) using the first-order Taylor approximation in Section 6.3 for the prediction is equivalent to the GP-EKF of Ko and Fox [2009], extended with uncertainty in the controls and model parameters.

6.5 Constraints

The optimisation problem in Equation (6.2.3) is solved subject to constraints on the initial state settings $\hat{\boldsymbol{x}}_0$, the sequence of control inputs $\hat{\boldsymbol{u}}_{0:T-1}$, the latent states $\boldsymbol{x}_{1:T}$, the observed states $\boldsymbol{z}_{1:T}$, and the measurement time points $\mathcal{T}_{\text{meas}}$. Constraints are common in optimisation problems due to physical or safety constraints in real systems, e.g. the maximum allowed electrical current in a machine or drug dose given to a patient. However, we may also need to apply constraints on the latent states $\boldsymbol{x}_{1:T}$ if the transition function f is replaced with a data-driven surrogate, e.g. a GP surrogate (see Section 6.5.3).

The initial state settings $\hat{\boldsymbol{x}}_0$, the control signal $\hat{\boldsymbol{u}}_{1:T}$ and the measurement time points $\mathcal{T}_{\text{meas}}$ are independent, deterministic variables set directly by the user. The latent states $\boldsymbol{x}_{1:T}$ and observed states $\boldsymbol{z}_{1:T}$ are dependent, stochastic variables. Constraints on independent and dependent variables are handled differently.

This section considers two types of constraints: The first type are linear constraints

$$\mathbf{C}\boldsymbol{\xi} - \bar{\boldsymbol{\xi}} \geq \mathbf{0}\,, \tag{6.5.1}$$

where $\boldsymbol{\xi} \in \{\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{u}}_t, \boldsymbol{x}_t, \boldsymbol{z}_t, \tau_t\}$, $\boldsymbol{\xi} \in \mathbb{R}^{D_\xi}$ is an independent or a dependent variable, $\mathbf{C} \in \mathbb{R}^{D_C \times D_\xi}$ and $\bar{\boldsymbol{\xi}} \in \mathbb{R}^{D_C}$, and the inequality is applied element-wise; The second type of constraints are constraints on the absolute difference, e.g. the rate of change, between independent variables. We will not consider constraints on the absolute difference between stochastic, dependent variables.

6.5.1 Independent Variable Constraints

The independent variables in the optimisation problem in Equation (6.2.3) are the initial state settings $\hat{\boldsymbol{x}}_0$, the sequence of desired control inputs $\hat{\boldsymbol{u}}_{0:T-1}$ and the measurement time points $\mathcal{T}_{\text{meas}}$ (for continuous-time models). Through Section 6.5.1, let $\boldsymbol{\xi} \in \{\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{u}}_t, \tau_t\}$ denote an independent variable.

Linear constraints such as in Equation (6.5.1) on independent, deterministic variables are straight-forward to handle. Note that the constraint in Equation (6.5.1) is written in the format of constraints in the Equation (6.2.3) optimisation problem.

Constraints on the absolute difference between independent variables are useful for two reasons: Firstly, there may be limitations (for physical or safety reasons) to how quickly the control input $\hat{\boldsymbol{u}}_t$ can be varied; Secondly, a minimum amount of time between measurements need to be enforced during optimisation. Let the absolute difference in dimension $d =$

$1, \ldots, D_u$ of the control input between consecutive time steps be upper-bounded by $\Delta_{u,(d)}$

$$\left|\hat{u}_{t+1,(d)} - \hat{u}_{t,(d)}\right| \leq \Delta_{u,(d)} .$$

Using a standard reformulation, this constraint can equivalently be written as

$$\hat{u}_{t+1,(d)} - \hat{u}_{t,(d)} + \Delta_{u,(d)} \geq 0 \quad \wedge \quad \hat{u}_{t,(d)} - \hat{u}_{t+1,(d)} + \Delta_{u,(d)} \geq 0 ,$$

using the constraints format in Equation (6.2.3). Additionally, let the absolute difference in time between two measurement time points τ_t and $\tau_{t'}$ be lower-bounded by $\Delta_\tau \geq 0$

$$|\tau_t - \tau_{t'}| \geq \Delta_\tau , \quad \forall \tau_t, \tau_{t'} \in \mathcal{T}_{\text{meas}} . \tag{6.5.2}$$

This constraint is non-convex. To simplify the problem formulation, we introduce additional constraints to maintain a fixed order of the measurement time points, and reformulate Equation (6.5.2) in convex form as

$$\forall \tau_t, \tau_{t'} \in \mathcal{T}_{\text{meas}} : \quad \begin{cases} \tau_t - \tau_{t'} - \Delta_\tau \geq 0 , & t \geq t' , \\ \tau_{t'} - \tau_t - \Delta_\tau \geq 0 , & t < t' . \end{cases}$$

using the constraints format in Equation (6.2.3).

6.5.2 Dependent Variable Constraints

The dependent variables in the optimisation problem in Equation (6.2.3) are the latent states $\boldsymbol{x}_{1:T}$ and observed states $\boldsymbol{z}_{1:T}$. Constraints on the dependent variables are typically more difficult to satisfy [Pesch, 1989; Faust et al., 2016], because, as the name suggests, they are dependent on the initial state $\hat{\boldsymbol{x}}_0$ and the control sequence $\hat{\boldsymbol{u}}_{0:T-1}$. Constraints on the dependent state variables $\boldsymbol{x}_{1:T}$ and $\boldsymbol{z}_{1:T}$ are often referred to as *path constraints.*

Let model $\mathcal{M}$ predict the latent state distribution $\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$ and observed state distribution $\boldsymbol{z}_t \sim \mathcal{N}(\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$ at time step t, where $\boldsymbol{\mu}_z = \mathbf{H}\boldsymbol{\mu}_t$ and $\boldsymbol{\Sigma}_z = \mathbf{H}\boldsymbol{\Sigma}_t\mathbf{H}^\top$. Through Section 6.5.2, let $\boldsymbol{\xi}_t \sim \mathcal{N}(\boldsymbol{\mu}_\xi, \boldsymbol{\Sigma}_\xi) \in \{\boldsymbol{x}_t, \boldsymbol{z}_t\}$ denote a Gaussian-distributed dependent variable.

We will assume that the path constraint may be time-dependent, such that the expression in Equation (6.5.1) becomes

$$\mathbf{C}_t \boldsymbol{\xi}_t - \bar{\boldsymbol{\xi}}_t \geq 0 \,. \tag{6.5.3}$$

To simplify notation, let Ξ_t denote the space $\Xi_t \subset \mathbb{R}^{D_\xi}$ at time step t defined by

$$\Xi_t = \left\{ \boldsymbol{\xi} \mid \mathbf{C}_t \boldsymbol{\xi} - \bar{\boldsymbol{\xi}}_t \geq 0 \right\} \,,$$

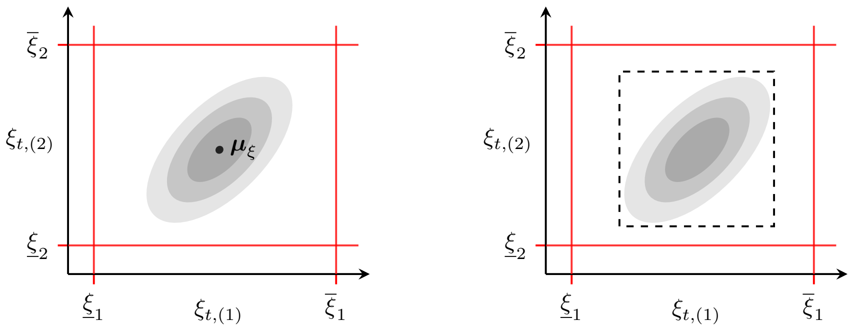
such that satisfying the linear constraint in Equation (6.5.3) at time t is equivalent to satisfying $\boldsymbol{\xi}_t \in \Xi_t$. Multiple sources of uncertainty affect the states $\boldsymbol{\xi}_t$ and need to be accounted for. Path constraints on variables with unbounded probability distributions (e.g. Gaussian distributions) are referred to as path *chance constraints.* The path chance constraint equivalent of $\boldsymbol{\xi}_t \in \Xi_t$ is

$$P(\boldsymbol{\xi}_t \in \Xi_t) \geq 1 - \gamma \,, \tag{6.5.4}$$

for some $\gamma \in (0, 1)$ that determines the chance constraint's confidence level requirement. Path chance constraints of the type in Equation (6.5.4) are typically analytically intractable [Prékopa, 1995, Ch. 11; Shapiro et al., 2009, Ch. 1]. There is a range of different tractable approximations for path chance constraint, e.g. the scenario approach [Calafiore and Campi, 2006], the sample average approximation [Pagnoncelli et al., 2009], and the convex second-order cone approximation [Shapiro et al., 2009, Ch. 1; Mesbah et al., 2014b]. In this work we will only consider the cone approximation, and compare it to a path constraint that does not account for the uncertainty in the state $\boldsymbol{\xi}_t$.

Mean Constraint

We let the term *mean constraint* denote the regular linear constraint in Equation (6.5.3) operating on the state mean, i.e. requiring $\boldsymbol{\mu}_\xi \in \Xi_t$. The mean constraint is used by existing literature on design of dynamic experiments (see Table 6.2). Figure 6.4a shows an example



(a) Mean constraint

(b) Cone constraint

Figure 6.4: Example with constants defined by Equation (6.5.5) with (a) mean constraint, where the mean $\boldsymbol{\mu}_\xi$ of the distribution must lie within the bounds, and (b) cone constraint, where the mean ± some number of standard deviations (the dashed box) must lie within the bounds.

defined by

$$\mathbf{C}_t = \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} , \quad \bar{\boldsymbol{\xi}}_t = \begin{bmatrix} \underline{\xi}_1 \\ -\bar{\xi}_1 \\ \underline{\xi}_2 \\ -\bar{\xi}_2 \end{bmatrix} . \tag{6.5.5}$$

The mean constraint has the advantage of simplicity, at the expense of not accounting for the uncertainty in $\boldsymbol{\xi}_t$, and is approximately equivalent to solving the optimisation problem subject to

$$P(\boldsymbol{\xi}_t \in \Xi_t) \geq \left(\tfrac{1}{2}\right)^{D_\xi} , \quad \forall \, t \in \mathcal{T} .$$

Hence the mean constraint provides a poor guarantee that the constraint $\boldsymbol{\xi}_t \in \Xi_t$ will be satisfied for all time steps t.

Cone Constraint

The convex second-order cone approximation [Shapiro et al., 2009, Ch. 1; Mesbah et al., 2014b] decomposes the linear constraint in Equation (6.5.3) into multiple constraints

$$\boldsymbol{c}_{t,(j)}^{\top}\boldsymbol{\xi}_t - \bar{\xi}_{t,(j)} \geq 0 \,, \tag{6.5.6}$$

for $\mathbf{C}_t = [\boldsymbol{c}_{t,(1)}, \dots, \boldsymbol{c}_{t,(D_C)}]^{\top}$ and $\bar{\boldsymbol{\xi}}_t = [\bar{\xi}_{t,(1)}, \dots, \bar{\xi}_{t,(D_C)}]^{\top}$. Each individual chance constraint $P(\boldsymbol{c}_{t,(j)}^{\top}\boldsymbol{\xi}_t - \bar{\xi}_{t,(j)} \geq 0) \geq 1 - \gamma$ can be satisfied by satisfying the constraints

$$\begin{bmatrix} \boldsymbol{c}_{t,(j)}^{\top} \\ \boldsymbol{c}_{t,(j)}^{\top} \end{bmatrix} \boldsymbol{\mu}_{\xi} + \alpha\sqrt{\boldsymbol{c}_{t,(j)}^{\top}\boldsymbol{\Sigma}_{\xi}\boldsymbol{c}_{t,(j)}} \begin{bmatrix} 1 \\ -1 \end{bmatrix} - \begin{bmatrix} \bar{\xi}_{t,(j)} \\ \bar{\xi}_{t,(j)} \end{bmatrix} \geq \mathbf{0} \,, \tag{6.5.7}$$

where $\alpha = \sqrt{2}\,\mathrm{erf}^{-1}(1-\gamma)$, with $\mathrm{erf}^{-1}(\cdot)$ the inverse error function. Figure 6.4b illustrates the chance constraint for the example defined in Equation (6.5.5).

If $\boldsymbol{\Sigma}_{\xi}$ is constant, then the cone constraint in Equation (6.5.7) is equivalent to the mean constraint for a smaller space $\hat{\Xi}_t \subset \Xi_t$. Whereas the mean constraint provides a poor guarantee that the chance constraint will be satisfied, the cone constraint may be overly conservative since it decomposes the full chance constraint into individual chance constraints.

6.5.3 Latent State Constraints for Data-Driven Surrogate Models

When solving Equation (6.2.3), the predicted latent state mean $\boldsymbol{\mu}_t$ may stray away from the latent state space region where there is latent state training data $\mathbf{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$. This can cause numerical issues in the solver as the GP predictive variance grows large, and we may have reasons not to trust a corresponding allegedly optimal solution $\hat{\boldsymbol{u}}_{1:T}$. Hence $\boldsymbol{\mu}_t$ should be appropriately constrained as $\boldsymbol{\mu}_t \in \mathcal{X}$.

The feasible control space $\mathcal{U}$ is assumed known, and control input training data can be sampled appropriately to fill the control space. We sample model parameter training data in a small region around the maximum *a posteriori* parameter estimate $\hat{\boldsymbol{\theta}}$. This allows approximation of the gradient of f with respect to θ, e.g. required for the Laplace approximation of $\mathbf{\Sigma}_\theta$. Assume the observed states $\boldsymbol{z}_t$ are subject to a constraint $\boldsymbol{z}_t \in \mathcal{Z}$, $\forall t \in \mathcal{T}$. Ideally, we would like to sample latent state training data from a domain $\mathcal{X}^*$

$$\mathcal{X}^* = \left\{ \boldsymbol{x} \;\middle|\; \exists\, \boldsymbol{u} \in \mathcal{U} \,:\, \mathbf{H}\boldsymbol{x} \in \mathcal{Z} \iff \mathbf{H}\phi[\boldsymbol{x}, \boldsymbol{u}, \hat{\boldsymbol{\theta}} \,|\, f] \in \mathcal{Z} \right\} ,$$

i.e. the space of latent states $\boldsymbol{x}$ (i) whose corresponding observed state $\boldsymbol{z}_t = \mathbf{H}\boldsymbol{x}_t$ satisfies $\boldsymbol{z}_t \in \mathcal{Z}$, and (ii) for which there exists a control input $\boldsymbol{u}$ that generates a transitioned state $\boldsymbol{x}_{t+1} = \phi[\boldsymbol{x}, \boldsymbol{u}, \hat{\boldsymbol{\theta}} \,|\, f]$ with corresponding observed state $\boldsymbol{z}_{t+1}$ that satisfies $\boldsymbol{z}_{t+1} \in \mathcal{Z}$.

Finding $\mathcal{X}^*$ is non-trivial even if the inverse transition function f^{-1} is known. $\mathcal{X}^*$ can be approximated through exhaustive sampling, but we may wish to limit the number of model evaluations, especially if f is expensive to evaluate. Proceeding, we will assume $\mathcal{X} \approx \mathcal{X}^*$ is a known hypercube

$$\mathcal{X} = \left\{ \boldsymbol{x} \;\middle|\; \forall\, d \in \{1, \ldots, D_x\} \,:\, \underline{x}_{(d)} \leq x_{(d)} \leq \overline{x}_{(d)} \right\} .$$

GP training data is sampled from $\mathcal{X}$. In our implementation, latent state training data is sampled from a grid (with samples on the borders of $\mathcal{X}$) and combined with grid-sampled control training data and model parameter samples drawn from a small uniform distribution centred around $\hat{\boldsymbol{\theta}}$. Additionally, the noise variance hyperparameters $\sigma^2_{\eta,(d)}$ is assumed greater than zero for all $d = 1, \ldots, D_x$ independent GP priors' covariance functions.

For numerical stability when solving Equation (6.2.3), especially for problems with initial latent states on the edge of $\mathcal{X}$, we propose relaxing $\mathcal{X}$, by using a latent state constraint $\boldsymbol{\mu}_t \in \tilde{\mathcal{X}}_{d_\text{out}}$ for the GP surrogate corresponding to output dimension d_out of f, $d_\text{out} = 1, \ldots, D_x$, with

$$\tilde{X}_{d_\text{out}} = \left\{ \boldsymbol{x} \;\middle|\; \forall\, d_{\mathbf{in}} \in \{1, \ldots, D_x\} \,:\, \underline{x}_{(d_\text{in})} - \chi_{d_\text{out}, d_\text{in}} \leq x_{(d_\text{in})} \leq \overline{x}_{(d_\text{in})} + \chi_{d_\text{out}, d_\text{in}} \right\} ,$$

where $\chi_{d_\text{out}, d_\text{in}} = |\overline{x}_{(d_\text{in})} - \underline{x}_{(d_\text{in})}| / \lambda_{(d_\text{out}),(d_\text{in})}$ and $\lambda_{(d_\text{out}),(d_\text{in})}$ denotes input dimension d_in's lengthscale hyperparameter of output dimension d_out's GP surrogate's latent state covariance function $k_{x,(d)}$. This ensures $\mathcal{X} \subset \tilde{X}_{d_\text{out}}$, $\forall d_\text{out} \in \{1, \ldots, D_x\}$.

6.6 Model Discrimination

Previous sections show how to approximate the models' marginal predictive distributions and incorporate the predictive uncertainty into the Equation (6.2.3) experimental design problem constraints. The optimisation problem can thus be solved to find the optimal dynamic experimental design for model discrimination. The optimal experimental design is given by the desired initial state $\hat{\boldsymbol{x}}_0^*$, the desired control signal $\hat{\boldsymbol{u}}_{0:T-1}^*$ and the measurement time points $\mathcal{T}_{\text{meas}}$. Once the optimal experimental design is found, an experiment may be launched and measurements $\mathbf{Y} = \{\boldsymbol{y}_t\}_{t \in \mathcal{T}_{\text{meas}}}$ collected. This section will discuss the subsequent model discrimination step.

Let us re-visit the notation of Section 6.4, with $\boldsymbol{\xi}_{t|t'}$ the estimate of some variable $\boldsymbol{\xi}$ at time step t given measurements up till time step t'. The rival models $\mathcal{M}_1, \dots, \mathcal{M}_M$ produce predictions $\{\boldsymbol{y}_{t|0}^{(i)}\}$, $t \in \mathcal{T}_{\text{meas}}$, with

$$\boldsymbol{y}_{t|0}^{(i)} \sim p\left(\boldsymbol{y}_{t|0}^{(i)} \,\middle|\, \hat{\boldsymbol{x}}_0^*,\, \hat{\boldsymbol{u}}_{0:t-1}^*\right) \approx \mathcal{N}\left(\mathbf{H}_i \boldsymbol{\mu}_{t|0}^{(i)},\, \mathbf{H}_i \boldsymbol{\Sigma}_{t|0}^{(i)} \mathbf{H}_i^\top + \boldsymbol{\Sigma}_y\right) .$$

Similarly, the rival models $\mathcal{M}_1, \dots, \mathcal{M}_M$ produce filtered predictions $\{\boldsymbol{y}_{t|t}^{(i)}\}$, $t \in \mathcal{T}_{\text{meas}}$, with

$$\boldsymbol{y}_{t|t}^{(i)} \sim p\left(\boldsymbol{y}_{t|t}^{(i)} \,\middle|\, \hat{\boldsymbol{x}}_0^*,\, \hat{\boldsymbol{u}}_{0:t-1}^*, \boldsymbol{y}_1, \dots, \boldsymbol{y}_t\right) \approx \mathcal{N}\left(\mathbf{H}_i \boldsymbol{\mu}_{t|t}^{(i)},\, \mathbf{H}_i \boldsymbol{\Sigma}_{t|t}^{(i)} \mathbf{H}_i^\top + \boldsymbol{\Sigma}_y\right) ,$$

with the filtered mean $\boldsymbol{\mu}_{t|t}^{(i)}$ and covariance $\boldsymbol{\Sigma}_{t|t}^{(i)}$ given measurements computed as in Section 6.4. For convenience, we define

$$\begin{aligned} \mathbf{L}_{t|t'}^{(i)} (\mathbf{L}_{t|t'}^{(i)})^\top &= \mathbf{H}_i \boldsymbol{\Sigma}_{t|t'}^{(i)} \mathbf{H}_i^\top + \boldsymbol{\Sigma}_y , \\ \boldsymbol{\delta}_{t|t'}^{(i)} &= \left(\mathbf{L}_{t|t'}^{(i)}\right)^{-1} \left(\boldsymbol{y}_t - \mathbf{H}_i \boldsymbol{\mu}_{t|t'}^{(i)}\right) , \end{aligned}$$

where $\mathbf{L}_{t|t'}^{(i)}$ is the lower-triangular Cholesky decomposition of the variance of $\boldsymbol{y}_{t|t'}^{(i)}$ and $\boldsymbol{\delta}_{t|t'}^{(i)}$ the weighted vector-difference between observation $\boldsymbol{y}_t$ and prediction $\mathbf{H}_i \boldsymbol{\mu}_{t|t'}^{(i)}$, with $t' \in \{0, t\}$ and $t \in \mathcal{T}_{\text{meas}}$. The weighted differences $\boldsymbol{\delta}_{t|t'}^{(i)}$ should have a distribution given by $\boldsymbol{\delta}_{t|t'}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ under the hypothesis that model $\mathcal{M}_i$ has generated the data $\mathbf{Y}$. The log-likelihood of an observation $\boldsymbol{y}_t$ given a model $\mathcal{M}_i$ is

$$\log \mathcal{N}\left(\boldsymbol{y}_t \,\middle|\, \mathbf{H}_i \boldsymbol{\mu}^{(i)}_{t|t'}, \mathbf{H}_i \boldsymbol{\Sigma}^{(i)}_{t|t'} \mathbf{H}_i^\top + \boldsymbol{\Sigma}_y\right) = -\log \left|\mathbf{L}^{(i)}_{t|t'}\right| - \frac{1}{2}\left(\boldsymbol{\delta}^{(i)}_{t|t'}\right)^\top \boldsymbol{\delta}^{(i)}_{t|t'} + \text{const.}.$$

Section 3.2 describes three discrimination criteria from literature: ranking based on normalised model likelihoods [Box and Hill, 1967], the χ^2 test [Buzzi-Ferraris and Forzatti, 1983], and ranking based on Akaike weights [Michalik et al., 2010]. This chapter will not compare the performance of these different discrimination criteria against each other; only the χ^2 test will be used for model discrimination. Following the reasoning of Buzzi-Ferraris and Forzatti [1983], the weighted squared residuals $(\boldsymbol{\delta}^{(i)}_{t|t'})^\top \boldsymbol{\delta}^{(i)}_{t|t'}$ should be χ^2-distributed. The χ^2 score is defined as 1 minus the χ^2 cumulative distribution at $\sum_{t \in \mathcal{T}_{\text{meas}}} (\boldsymbol{\delta}^{(i)}_{t|t'})^\top \boldsymbol{\delta}^{(i)}_{t|t'}$ with $|\mathcal{T}_{\text{meas}}| \times D_y - D_{\theta,i}$ degrees of freedom. Models are deemed inadequate if their corresponding χ^2 score is below some threshold, e.g. 1E-3. The χ^2 test is more conservative than ranking based on log-likelihood or Akaike weights, which may result in more experiments but also fewer false positives (i.e. choosing the wrong model).

Given the choice of using the χ^2 test for model discrimination, there are still two alternatives for computing the χ^2 score for each model: to use the weighted residuals $\delta^{(i)}_{t|0}$ from prediction alone, or the filtered versions $\delta^{(i)}_{t|t}$. Both alternatives have their advantages and disadvantages. The residuals $\delta^{(i)}_{t|0}$ from prediction shows the predictive power of a model. Predictive accuracy is typically one of the main goals of modelling. Someone in favour of using $\delta^{(i)}_{t|t}$ might argue that looking at predictions alone means we base the choice of model on the specific process noise realisations in each experiment. This leaves us vulnerable to outlier process noise early in the experiment, which may have a large impact on latent states and measurements for later time points. However, since we add the process noise covariance to the model predictive covariance at prediction, possible outlier cases should be covered as long as the Gaussian process noise approximation is valid. In many cases, models are also similar enough that even when model predictions differ significantly, the filtered predictions may be nearly indistinguishable, i.e. data generated from a model $\mathcal{M}_1$ may be explained well by a rival model $\mathcal{M}_2$ if model $\mathcal{M}_2$ is allowed to correct its prediction at each time step by blaming observed deviations solely on noise.

For the purpose of the simulations in Section 6.7, we will compute each model's χ^2 score using the residuals $\delta_{t|0}^{(i)}$ computed only from predictions. For models with black-box transitions functions f replaced with GP surrogates, we use the exact transition function mean for the purpose of model discrimination (see Section 6.3.4).

6.7 Results

This chapter makes two novel contributions to design of dynamic experiments for model discrimination: we allow for black-box transitions functions in the models, and we account for more sources of uncertainty in the optimisation problem. In our literature review we did not find any previous work accounting for uncertainty in the control signal, or the effect of path chance-constraints on design of experiments. This section presents experimental results for the following:

- A comparison of performance of mean and cone path constraints.
- Simulations with (i) correctly modelled, (ii) underestimated or (iii) overestimated control signal covariance.
- The performance of the GP surrogates as analytical emulators of black-box transition functions.

To this end, we have a case study from Espie and Macchietto [1989] that considers yeast fermentation. We also present a comparison of result using our methodology to results reported by Espie and Macchietto [1989] and Chen and Asprey [2003] for this case study.

6.7.1 Case Study: Yeast Fermentation

The yeast fermentation case study is taken from Espie and Macchietto [1989]. There are $D_x = 2$ latent states (biomass and substrate concentration, respectively) and $D_u = 2$ control inputs (feed velocity and feed substrate concentration). We observe both states, hence $D_z = D_y = 2$ and $\mathbf{H}_i = \mathbf{H} = \mathbf{I}$, with $D_{\theta,i} \in \{3, 4\}$ model parameters.

For simplicity, we omit the model index (except in $\mathcal{M}_i$) and time step index when writing out the models below. Model $\mathcal{M}_1$ assumes Monod kinetics with constant specific death rate

$$\mathcal{M}_1 \; : \; \begin{cases} \dfrac{\mathrm{d}x_1}{\mathrm{d}\tau} = (r - u_1 - \theta_4)x_1 \, , \\ \dfrac{\mathrm{d}x_2}{\mathrm{d}\tau} = -\dfrac{rx_1}{\theta_3} + u_1(u_2 - x_2) \, , \\ \quad r = \dfrac{\theta_1 x_2}{\theta_2 + x_2} \, . \end{cases}$$

Model $\mathcal{M}_2$ assumes Contois kinetics with constant specific death rate

$$\mathcal{M}_2 \; : \; \begin{cases} \dfrac{\mathrm{d}x_1}{\mathrm{d}\tau} = (r - u_1 - \theta_4)x_1 \, , \\ \dfrac{\mathrm{d}x_2}{\mathrm{d}\tau} = -\dfrac{rx_1}{\theta_3} + u_1(u_2 - x_2) \, , \\ \quad r = \dfrac{\theta_1 x_2}{\theta_2 x_1 + x_2} \, . \end{cases}$$

Model $\mathcal{M}_3$ assumes linear specific growth rate

$$\mathcal{M}_3 \; : \; \begin{cases} \dfrac{\mathrm{d}x_1}{\mathrm{d}\tau} = (r - u_1 - \theta_3)x_1 \, , \\ \dfrac{\mathrm{d}x_2}{\mathrm{d}\tau} = -\dfrac{rx_1}{\theta_2} + u_1(u_2 - x_2) \, , \\ \quad r = \theta_1 x_2 \, . \end{cases}$$

Model $\mathcal{M}_4$ assumes Monod kinetics with constant maintenance energy

$$\mathcal{M}_4 \; : \; \begin{cases} \dfrac{\mathrm{d}x_1}{\mathrm{d}\tau} = (r - u_1)x_1 \, , \\ \dfrac{\mathrm{d}x_2}{\mathrm{d}\tau} = -\dfrac{rx_1}{\theta_3} + u_1(u_2 - x_2) \, , \\ \quad r = \dfrac{\theta_1 x_2}{\theta_2 + x_2} \, . \end{cases}$$

Espie and Macchietto [1989] do not consider any variable uncertainty in their formulation of the experimental design problem. They simulate 72 hours of yeast fermentation, with measurements taken every 0.75 hour. The controls have bounds $u_1 \in [0.05\,\mathrm{h}^{-1},\, 0.2\,\mathrm{h}^{-1}]$ and $u_2 \in [5\,\mathrm{g/L},\, 35\,\mathrm{g/L}]$. Figure 6.5 shows an example of model predictions for models fitted to data generated from model $\mathcal{M}_1$ using a random uniformly sampled control signal $\hat{\boldsymbol{u}}_{0:T-1}$,

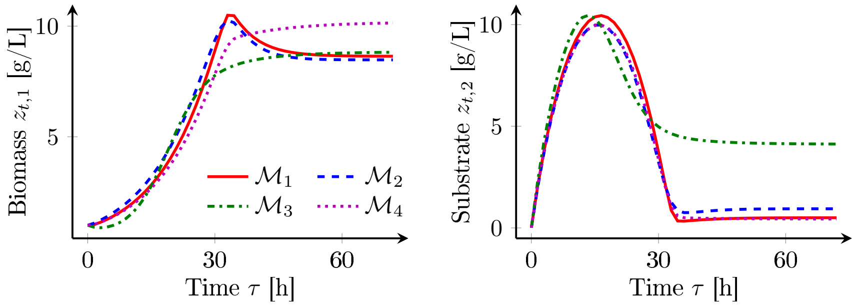


Figure 6.5: Example of model predictions for Espie and Macchietto [1989] yeast fermentation case study. The model predictions are generated from a random uniformly sampled control input $\hat{\boldsymbol{u}}_{0:T-1}$.

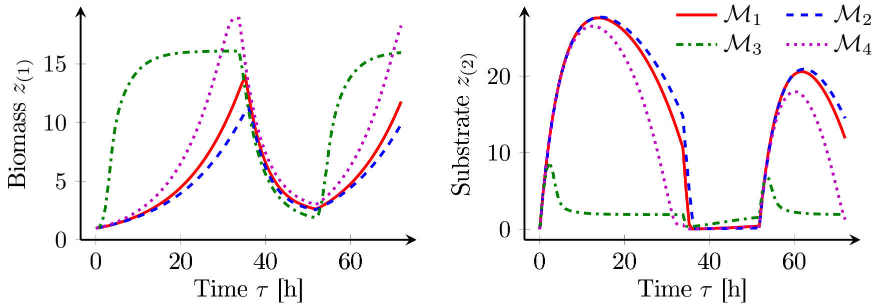


Figure 6.6: Yeast fermentation case study model predictions for Espie and Macchietto [1989] optimal control input $\mathbf{U}^*_{\mathrm{EM}}$.

with time steps $\tau_t = t \times 0.75\,\mathrm{h}$. Espie and Macchietto [1989] find the following control inputs optimal for model discrimination:

$$u_1(\tau) = 0.2\,\mathrm{h}^{-1}\,,$$

$$u_2(\tau) = \begin{cases} 5\,\mathrm{g/L} & \text{if } 33\,\mathrm{h} \leq \tau \leq 51\,\mathrm{h} \\ 35\,\mathrm{g/L} & \text{otherwise}\,. \end{cases}$$

Let $\mathbf{U}^*_{\mathrm{EM}}$ denote the discretised optimal control signal $\hat{\boldsymbol{u}}_{0:T-1}$ found by Espie and Macchietto [1989]. Figure 6.6 shows the corresponding model predictions. We let the model parameters $\boldsymbol{\theta}_i$ take the estimated values reported in Espie and Macchietto [1989]:

$$\begin{aligned}
\boldsymbol{\theta}_1 &= \begin{bmatrix} 0.30, & 0.25, & 0.56, & 0.02 \end{bmatrix} , \\
\boldsymbol{\theta}_2 &= \begin{bmatrix} 0.30, & 0.03, & 0.55, & 0.03 \end{bmatrix} , \\
\boldsymbol{\theta}_3 &= \begin{bmatrix} 0.12, & 0.56, & 0.03 \end{bmatrix} , \\
\boldsymbol{\theta}_4 &= \begin{bmatrix} 0.30, & 0.30, & 0.55 \end{bmatrix} ,
\end{aligned}$$

We find that if we perturb the elements of $\mathbf{U}^*_{\mathrm{EM}}$ by 1.5%, 5% or 10% and use the perturbed control inputs as an initial guess for optimiser in our experimental design formulation, we retrieve $\mathbf{U}^*_{\mathrm{EM}}$ again. We see this as a sanity test of our methodology; for the simplest formulation of the experimental design problem we can rediscover a known optimum.

Let us add measurement noise, which results in parametric uncertainty. Chen and Asprey [2003] assume the noise covariance $\boldsymbol{\Sigma}_y$ is known

$$\boldsymbol{\Sigma}_y = \begin{bmatrix} 0.06 & -0.01 \\ -0.01 & 0.04 \end{bmatrix} .$$

Chen and Asprey [2003] discriminate between the two most similar models $\mathcal{M}_1$ and $\mathcal{M}_2$. Data is generated from $\mathcal{M}_1$ with parameters $\boldsymbol{\theta} = [0.25, 0.25, 0.88, 0.09]$. For the experimental design they use the D_{HR} design criterion in Equation (3.2.7) weighted by the inverse noise covariance

$$D_{\mathrm{CA}}(\hat{\boldsymbol{u}}_{0:T-1}) = \sum_{t=1}^{T} (\boldsymbol{z}_{1,t} - \boldsymbol{z}_{2,t})^\top \boldsymbol{\Sigma}_y^{-1} (\boldsymbol{z}_{1,t} - \boldsymbol{z}_{2,t}) .$$

Chen and Asprey [2003] find 20 optimal measurement time instances $\mathcal{T}_{\mathrm{meas}}$ and an optimal control signal $\mathbf{U}^*_{\mathrm{CA}}$ consisting of 5 piece-wise constant sections. We find $D_{\mathrm{CA}}(\mathbf{U}^*_{\mathrm{CA}}) = 2601$. For the same measurement time instances and control switch time instances, we find a solution $\mathbf{U}^*_{\mathrm{new},1}$ that yields $D_{\mathrm{CA}}(\mathbf{U}^*_{\mathrm{new},1}) = 2805$, i.e. a 7.8% larger divergence. We assume this discrepancy is due to the improvement in optimisation solvers after 20 years. Figure 6.7 shows both control signals $\mathbf{U}^*_{\mathrm{CA}}$ and $\mathbf{U}^*_{\mathrm{new},1}$.

We add process noise with variance $\boldsymbol{\Sigma}_x \equiv 0.01 \cdot \mathbf{I}$, i.e. the same order of magnitude as measurement noise. With process noise, we find that Chen and Asprey's [2003] optimal

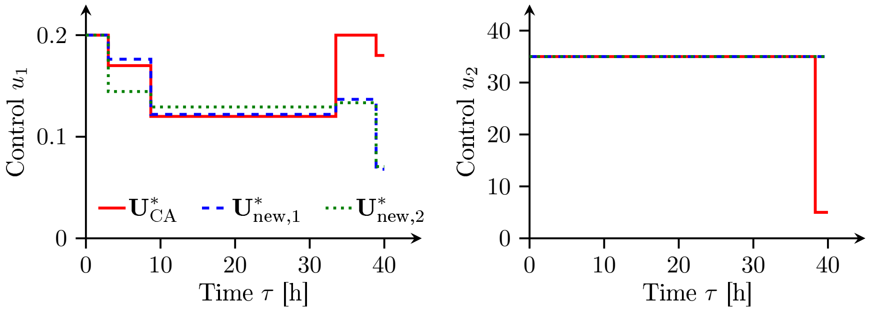


Figure 6.7: Optimal control input $\mathbf{U}^*_{\text{CA}}$ from Chen and Asprey [2003], our solution $\mathbf{U}^*_{\text{new},1}$, and our solution $\mathbf{U}^*_{\text{new},2}$ for the case of added process noise.

control signal yields $D_{\text{CA}}(\mathbf{U}^*_{\text{CA}}) = 244$. We find the optimal control signal $\mathbf{U}^*_{\text{new},2}$ that yields $D_{\text{CA}}(\mathbf{U}^*_{\text{new},2}) = 263$, i.e. an 8% larger divergence. Figure 6.7 shows that $\mathbf{U}^*_{\text{new},1} \neq \mathbf{U}^*_{\text{new},2}$, which means that adding process noise affects the optimal control signal.

For the remaining experiments we will look at all $M = 4$ models $\mathcal{M}_i$ and use true parameter values $\boldsymbol{\theta} = [0.25, 0.25, 0.88, 0.09]$ and measurement noise variance $\boldsymbol{\Sigma}_y$ from Chen and Asprey [2003]. We assume there is no process noise, hence $\boldsymbol{\Sigma}_x \equiv \mathbf{0}$. We start without any experimental data and initial model parameter estimates $\theta_{i,d} = 0.5$ and covariance $\boldsymbol{\Sigma}_{\theta,i} = 0.05 \cdot \mathbf{I}$ for all models $\mathcal{M}_i$ and $d \in \{1, \dots, D_{\theta,i}\}$. The reason for this is the difficulty in finding initial experimental conditions that do not immediately render one or more models obviously inadequate. The initial states are given by $x_1 = 1$ and $x_2 = 0.01$, with initial latent state covariance $\boldsymbol{\Sigma}_0 = \text{diag}(10^{-3}, 10^{-6})$. We let the control inputs have covariance given by $\boldsymbol{\Sigma}_u = \text{diag}(10^{-6}, 10^{-3})$. We simulate 72 hours of fermentation, with measurements and changes in control signal every 1.5 hours (i.e. 48 measurements $\boldsymbol{y}_t$ and control inputs $\hat{\boldsymbol{u}}_t$ in total). The controls have bounds $u_1 \in [0.05\,\text{h}^{-1}, 0.2\,\text{h}^{-1}]$ and $u_2 \in [5\,\text{g/L}, 35\,\text{g/L}]$.

6.7.2 Comparing Path Constraints

We compare the results of solving the design of experiments optimisation problem in Equation (6.2.3) using (i) the mean path constraint in Section 6.5.2, or (ii) the cone path constraint in Section 6.5.2 with $\alpha = 2$ standard deviations margin. We are interested in comparing the constraints' performance in terms of the number of violations of the path constraints,

		Experiments required		Avg. num. of models remaining after #n exp.		
Bound	**Constraint**	Mean	Std	#1	#2	#3
7	Mean	2.04	0.20	2.00	1.04	1
	Cone	2.44	0.51	2.64	1.44	1
10	Mean	2.00	0	2.00	1	–
	Cone	2.16	0.37	2.12	1.16	1
15	Mean	2.08	0.28	2.04	1.08	1
	Cone	2.00	0	2.00	1	–

Table 6.4: Average number of experiments (with standard deviation) required for successful model discrimination in 25 simulations of the yeast fermentation case study, using a mean or cone path constraint (Section 6.5) with an upper bound on $z_{t,2}$. The right-most columns show the average number of models (out of four) that pass the χ^2 test after 1, 2 or 3 experiments.

how severe the violations are, and the number of experiments required for successful model discrimination.

Let $\bar{z}_2$ be a constraint upper bound on the substrate concentration, such that we wish to satisfy $z_{t,(2)} \leq \bar{z}_2$ for $t = \{1, \dots, T\}$. For the simulations, we let the upper bound take one of the values $\bar{z}_2 \in \{7, 10, 15\}$.

Results are collected through simulations. Each simulation follows the experimental design process laid out in Figure 3.2 on page 55: (1) parameter estimation, (2) model discrimination, (3) experimental design, (4) execution of experiment, and return to step (1). Measurements $\boldsymbol{y}_{1:T}$ are generated in each simulated experiment, and used for model discrimination using the χ^2 test, as described in Section 6.6. Models are deemed inadequate if their χ^2 score is below a given threshold (see Section 3.2.1). We use the same threshold, 1E–3, as in Section 4.3 and Section 5.3. As discussed in Section 5.4, this threshold is arbitrary.

Table 6.4 shows the performance in 25 simulations of the different path constraint types (mean constraint and cone constraint) in terms of the average number of experiments needed for successful model discrimination. Model $\mathcal{M}_1$ was correctly identified as the true data-generating model in all simulations.

We are interested in comparing the performance of the two different types of constraints. When comparing the performance, we are faced with the following problem: The measurements $\boldsymbol{y}_{1:T}$ in each experiment in each simulation are generated from one particular realisation of initial latent state $\boldsymbol{x}_0 \sim \mathcal{N}(\hat{\boldsymbol{x}}_0, \boldsymbol{\Sigma}_{x,0})$, control inputs $\boldsymbol{u}_t \sim \mathcal{N}(\hat{\boldsymbol{u}}_t, \boldsymbol{\Sigma}_u)$ and measurement noise $\boldsymbol{v} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_y)$. We can ask the following two questions:

1. If we run an experiment with control $\hat{\boldsymbol{u}}^*_{0:T-1}$, and do not observe any constraint violations, is the control input $\hat{\boldsymbol{u}}^*_{0:T-1}$ safe? If the constraints are not violated for a particular noise realisation, they may still be likely to be violated for other experiments generated using the same control $\hat{\boldsymbol{u}}^*_{0:T-1}$.

2. Are all constraint violations equally bad, or are some worse than others? We may think we have a safe control $\hat{\boldsymbol{u}}^*_{0:T-1}$, but due to noise we violate the constraint a tiny bit. Is this equally bad to completely overshooting the constraint?

Instead of relying on single noise realisations, we use Monte Carlo sampling to make a statistically more sound assessment of how safe a control signal really is.

Let $\mathcal{M}_0$ denote the data-generating model ($\mathcal{M}_1$ with "true" model parameter values). Assume we are studying a control sequence $\hat{\boldsymbol{u}}^*_{0:T-1}$. We generate $N_u = 100$ noisy control sequences $\boldsymbol{u}_{0:T-1,n}$, $n = 1, \ldots, N$ by drawing samples $\boldsymbol{u}_{t,n} \sim \mathcal{N}(\hat{\boldsymbol{u}}^*_t, \boldsymbol{\Sigma}_u)$. For each control sequence we sample a random initial state $\boldsymbol{x}_{0,n} \sim \mathcal{N}([1,\ 0.01]^\top, \boldsymbol{\Sigma}_0)$. The control sequences and initial states are used with the true model $\mathcal{M}_0$ to generate corresponding sequences of observed states $\boldsymbol{z}_{1:T,n}$. Let $\mathbf{Z} = \{\boldsymbol{z}_{1:T,1}, \ldots, \boldsymbol{z}_{1:T,N_u}\}$ denote the set of observed state sequences. These observed state sequences are different possible experimental outcomes generated by the same control sequence $\hat{\boldsymbol{u}}^*_{0:T-1}$. This helps answer question 1 above: The control sequence is completely safe if none of the observed state sequences in $\mathbf{Z}$ violate our constraints.

Next, let $\mathbf{Z}_{\text{viol}}$ define the set of observation sequences for which the constraint $z_{t,(2)} \leq \bar{z}_2$ is violated for at least one time step t:

$$\mathbf{Z}_{\text{viol}} = \{\boldsymbol{z}_{1:T} \,|\, \boldsymbol{z}_{1:T} \in \mathbf{Z} \wedge \exists t : z_{t,(2)} > \bar{z}_2\}\,.$$

The *violation level* $V(\cdot)$ for an experiment is the ratio of simulations that violate the constraints

$$V\left(\hat{\boldsymbol{u}}^*_{0:T-1}\right) = \frac{|\mathbf{Z}_{\text{viol}}|}{N_u}\,.$$

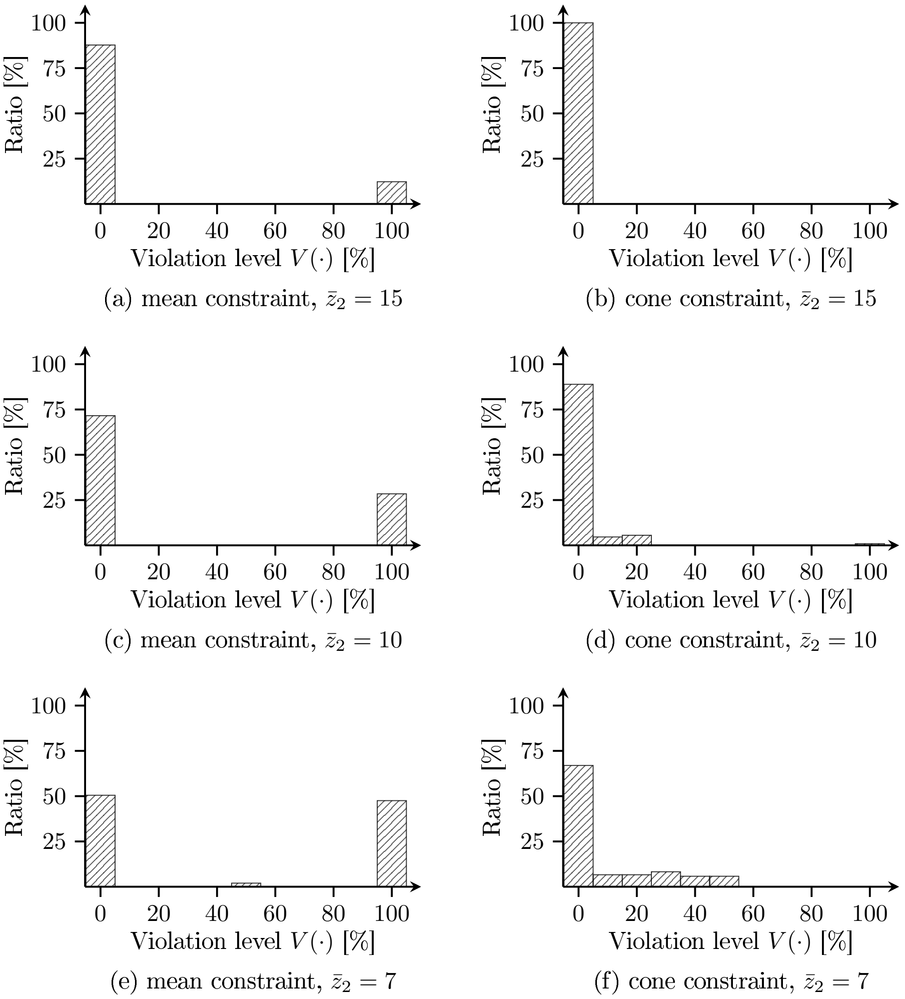
The violation level $V(\cdot)$ helps answer question 2 above: A violation level of $0\,\%$ means that a given control signal is apparently safe, whereas a violation level of $100\,\%$ means that a control signal is almost guaranteed to result in constraint violations. A control signal with a higher violation level is deemed worse than a control signal resulting in a lower violation level.

We compute the violation levels for all optimal control signals for the 25 simulations with each set of mean or cone constraint and upper bound $\bar{z}_2 \in \{7, 10, 15\}$ The histograms in Figure 6.8 show the ratio of experiments (i.e. optimised control signals) at each violation level. We see e.g. in Figure 6.8(a) that around $12\,\%$ of the control signals $\hat{\boldsymbol{u}}^*_{0:T-1}$ optimised with the mean constraint and upper bound $\bar{z}_2 = 15$ result in violation levels of $100\,\%$. We make three observations:

- The cone constraint results in *fewer* constraint violations than the mean constraint.
- The cone constraint results in *less severe* constraint violations (lower violation levels) than the mean constraint.
- The number of constraint violations increases as the upper bound decreases.

All three observations agree with expectations.

The experimental design process starts with parameter estimation. In this case study each simulation starts without any available experimental data, with parameter prior means $\theta_{i,d} = 0.5$ for all models $i = 1, \dots, M$ and $d = 1, \dots, D_{\theta,i}$. Data is generated from model $\mathcal{M}_1$ with "true" parameter values $\boldsymbol{\theta}$ = [0.25, 0.25, 0.88, 0.09]. We assume the Espie and Macchietto [1989] case study has identifiable model parameters. Figure 6.9 shows the model $\mathcal{M}_1$ parameter estimates in the 25 simulations with a cone constraint and $\bar{z}_2 = 15$. We see that θ_1 and θ_4 are well-estimated. The θ_2 and θ_3 estimates have higher variance, and the estimates do not seem to be directly correlated (the θ_2-θ_3 plot shows a cloud of estimate parameter values). This indicates θ_2 and θ_3 may suffer some level of unidentifiability. How-



(a) mean constraint, $\bar{z}_2 = 15$

(b) cone constraint, $\bar{z}_2 = 15$

(c) mean constraint, $\bar{z}_2 = 10$

(d) cone constraint, $\bar{z}_2 = 10$

(e) mean constraint, $\bar{z}_2 = 7$

(f) cone constraint, $\bar{z}_2 = 7$

Figure 6.8: The violation level is the ratio of Monte Carlo simulations using optimised control signals that result in constraint violations $z_{t,(2)} > \bar{z}_2$. The vertical axes show the ratio of optimised control signals for each violation level (horizontal axes) for a given constraint.

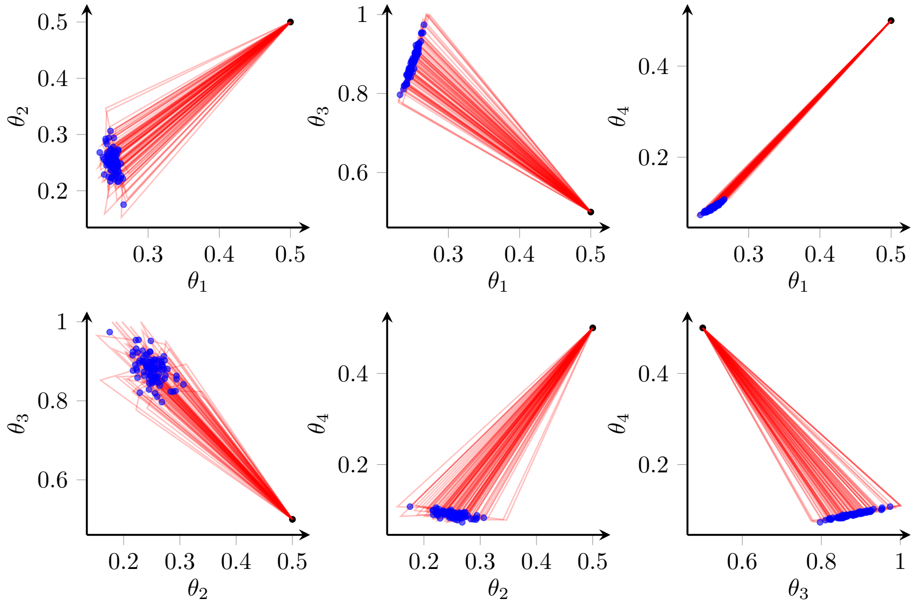
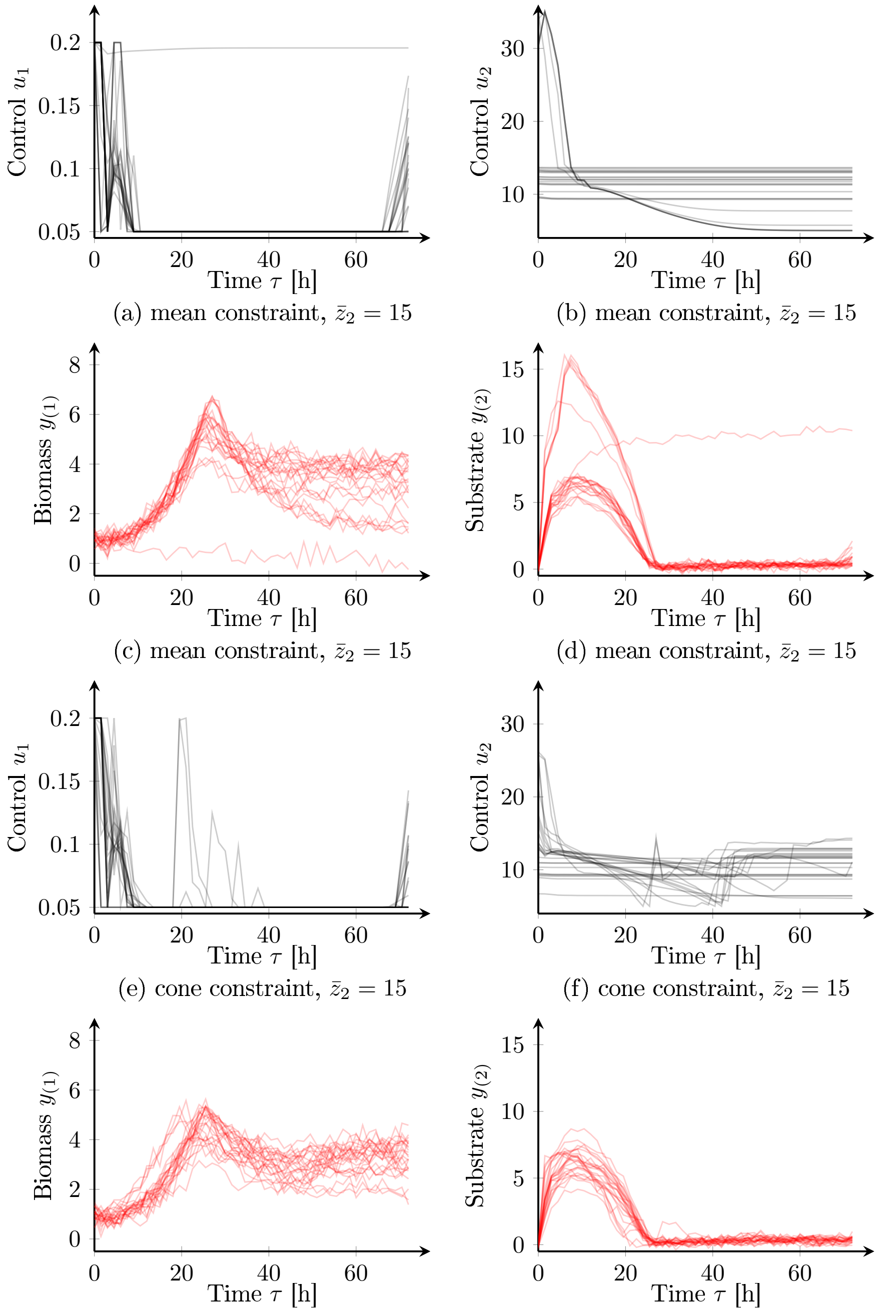


Figure 6.9: Model $\mathcal{M}_1$ parameter estimates from 25 simulations of the Espie and Macchietto [1989] case study with a cone constraint and $\bar{z}_2 = 15$. The black dot at [0.5, 0.5] in each plot denotes the parameter prior mean, the red lines show the parameter estimate paths, and the blue dots the final model $\mathcal{M}_1$ parameter estimates. θ_1 and θ_4 are well-estimated. θ_2 and θ_3 estimates have a higher variance and do not seem to be correlated, possibly indicating some level of parameter unidentifiability.

ever, since the θ_2 and θ_3 estimates are consistent, we do not believe this should affect the result of our experiments.

In Figure 6.10 and Figure 6.11 we compare the optimal control signals $\hat{\boldsymbol{u}}_{0:T-1}$ found using the mean and cone constraints, for the first and second experiments designed in each of the 25 simulations. Also plotted are the corresponding experimental measurements $\boldsymbol{y}_{1:T}$. We note large similarities between the control signals found in each simulation, and with the different constraints. In the first experiment, the optimal strategy is apparently to shock the system early by giving a high initial control input, and then maintaining a constant low input. In the second experiment the optimal strategy is apparently to keep u_2 constantly at its lowest value, while ramping up u_1 with constant small perturbations. The observed biomass $y_{(1)}$ and substrate $y_{(2)}$ curves are very different between the first and second experiment. In the first experiment the parameter estimates are more uncertain than in the second experiment,



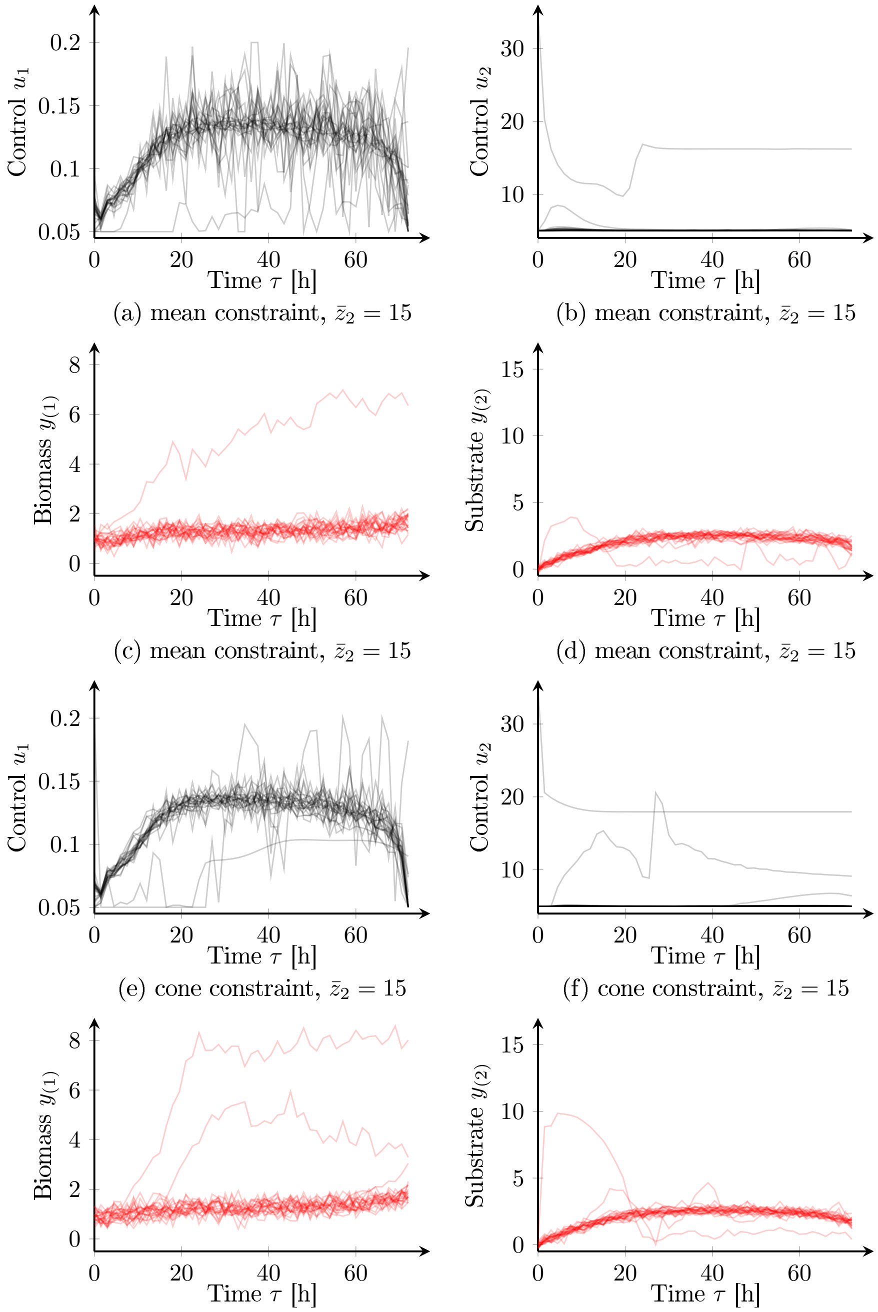
(a) mean constraint, $\bar{z}_2 = 15$ (b) mean constraint, $\bar{z}_2 = 15$

(c) mean constraint, $\bar{z}_2 = 15$ (d) mean constraint, $\bar{z}_2 = 15$

(e) cone constraint, $\bar{z}_2 = 15$ (f) cone constraint, $\bar{z}_2 = 15$

(g) cone constraint, $\bar{z}_2 = 15$ (h) cone constraint, $\bar{z}_2 = 15$

Figure 6.10: Plots showing optimal control signals and corresponding observations for the *first* experiment designed in 25 simulations of the Espie and Macchietto [1989] yeast fermentation case study. We compare the controls and observations found with mean and cone constraints, with $\bar{z}_2 = 15$. With the mean constraint there are constraint violations in (d).



(a) mean constraint, $\bar{z}_2 = 15$

(b) mean constraint, $\bar{z}_2 = 15$

(c) mean constraint, $\bar{z}_2 = 15$

(d) mean constraint, $\bar{z}_2 = 15$

(e) cone constraint, $\bar{z}_2 = 15$

(f) cone constraint, $\bar{z}_2 = 15$

(g) cone constraint, $\bar{z}_2 = 15$

(h) cone constraint, $\bar{z}_2 = 15$

Figure 6.11: Plots showing optimal control signals and corresponding observations for the *second* experiment designed in 25 simulations of the Espie and Macchietto [1989] yeast fermentation case study. We compare the controls and observations found with mean and cone constraints, with $\bar{z}_2 = 15$. We see no constraint violations.

so we wish for the optimal experiment to be more "cautious" in order not to yield constraint violations. However, the mean constraint does not incorporate this uncertainty, and we do see constraint violations in the first experiment. These are the constraint violations that appear in Figure 6.8(a). The second experiments are far from violating the constraint $\bar{z}_2 = 15$.

6.7.3 Effect of Control Input Uncertainty

In the optimisation problem in Equation (6.2.3) there are several covariance matrices that are assumed known. In most real-world applications, the exact covariances are not known and have to be approximated. We wish to study the effect of under- or overestimating the size of the covariance, compared to having the correct covariance. We choose to study this by varying the control input covariance. More specifically, let models $\mathcal{M}_i$ assume control input distribution $\boldsymbol{u}_t \sim \mathcal{N}(\hat{\boldsymbol{u}}_t, \hat{\boldsymbol{\Sigma}}_u)$. We generate experimental data by sampling control inputs $\boldsymbol{u}_t \sim \mathcal{N}(\hat{\boldsymbol{u}}_t, \boldsymbol{\Sigma}_u)$. In the simulations, the modelled covariance $\hat{\boldsymbol{\Sigma}}_u$ and the true covariance $\boldsymbol{\Sigma}_u$ are assigned values denoted "small" and "large", with

$$\text{small:} \quad \mathrm{diag}(1\text{E-}8,\ 1\text{E-}4)\,, \tag{6.7.1a}$$

$$\text{large:} \quad \mathrm{diag}(1\text{E-}4,\ 1\text{E-}2)\,. \tag{6.7.1b}$$

There are four combinations of small and large modelled and true control covariances. The resulting scenarios can be described as (i) correctly modelled uncertainty $\hat{\boldsymbol{\Sigma}}_u = \boldsymbol{\Sigma}_u$, (ii) underestimated uncertainty $|\hat{\boldsymbol{\Sigma}}_u| \leq |\boldsymbol{\Sigma}_u|$, and (iii) overestimated uncertainty $|\hat{\boldsymbol{\Sigma}}_u| \geq |\boldsymbol{\Sigma}_u|$.

Table 6.5 shows the result of 25 simulations of the yeast fermentation case study with the different modelled and true control covariances. A cone path constraint is enforced with an upper bound $\bar{z}_2 = 15\,\mathrm{g/L}$ on the substrate concentration (see Section 6.7.2). We see that, as expected, a correctly modelled small control covariance yields the best result in terms of average number of experiments required and the model discrimination success rate. A large modelled control covariance results in a higher average number of required experiments, and a marginally lower success rate. Model discrimination is deemed failed if an incorrect model is identified as the data-generating model, and inconclusive if the experimental budget

Control covariance		**Experiments required**		**Model discrimination**			**Cone constraint**
$\hat{\mathbf{\Sigma}}_u$	$\mathbf{\Sigma}_u$	Mean	Std	Succ.	Fail.	Inconcl.	**violations**
small	small	2.04	0.20	100 %	0 %	0 %	0 %
small	large	2.05	0.22	84 %	0 %	16 %	4 %
large	small	2.58	0.97	96 %	0 %	4 %	0 %
large	large	2.21	0.66	96 %	0 %	4 %	0 %

Table 6.5: The first set of columns shows the modelled control covariance $\hat{\mathbf{\Sigma}}_u$ and the true control covariance $\mathbf{\Sigma}_u$ used for generating experimental data defined in Equation (6.7.1). The second set of columns show the average number of experiments required for successful model discrimination in 25 simulations of the yeast fermentation case study. The third set of columns show the rate of successful, failed or inconclusive model discrimination. The last column shows the rate of simulations in which the cone path constraint was violated.

(maximum number of allowed experiments) is exhausted or the χ^2-test discards all models as inadequate. In these simulations the experimental budget was never exhausted, i.e. all instances of inconclusive model discrimination is due to all models being deemed inadequate. We see that the rate of inconclusive model discrimination is significantly higher when the true control covariance is underestimated, and we have a violation of the cone constraint. Hence, we are punished less for conservative estimates of the control covariance than overly optimistic estimates.

6.7.4 Black-Box Transition Functions

Next we study the performance of the GPs surrogate approach. The performance is compared to the analytical results in Section 6.7.2.

The GP surrogate predictions are used in the simulations during design of experiments. As described in Section 6.3.4, the original ("black-box") transition function f is used to compute the means $\boldsymbol{\mu}_{1:T}^{(i)}$ of the predictive distributions during model discrimination, with the derivatives $\nabla\mu(\cdot)$ of the GP surrogates' predictive mean used to compute the predictive covariances $\mathbf{\Sigma}_{1:T}^{(i)}$.

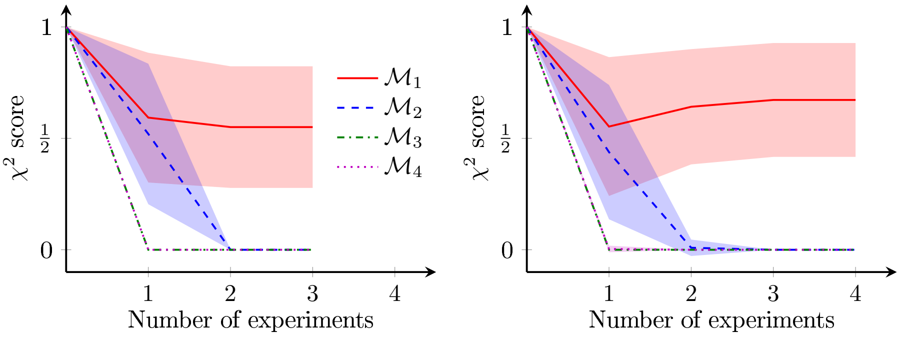
Each simulation starts with no initial data and a relatively uninformed model parameter distribution. This model parameter distribution is used to design the first experiment. Once an experiment has been executed, the model parameter distribution is updated before

			Experiments required		**Avg. num. of models remaining after #n exp.**		
	Bound	**Constraint**	Mean	Std	#1	#2	#3
Table 6.4	7	Mean	2.04	0.20	2.00	1.04	1
		Cone	2.44	0.51	2.64	1.44	1
	10	Mean	2.00	0	2.00	1	–
		Cone	2.16	0.37	2.12	1.16	1
	15	Mean	2.08	0.28	2.04	1.08	1
		Cone	2.00	0	2.00	1	–
	15	Cone	2.12	0.33	2.08	1.12	1

Table 6.6: Performance comparison of analytical and GP approaches. The bottom, highlighted row shows the performance of the GP approach, with average number of experiments (with standard deviation) required for successful model discrimination in 25 simulations of the yeast fermentation case study, using a cone path constraint (Section 6.5) with an upper bound on $z_{t,2}$. The right-most columns show the average number of models (out of four) that pass the χ^2 test after 1, 2 or 3 experiments. Compare to the analytical approach results in Table 6.4, reproduced here for convenience.

the models' χ^2 score is computed during model discrimination. For the case of analytical transition functions in the path constraint test in Section 6.7.2 and the control uncertainty test in Section 6.7.3, the model parameter prior described in Section 6.7.1 is used. However, for the case of GP surrogates, where uncertainty in the model predictions is added, the model parameter prior in Section 6.7.1 is too large—the uncertainty grows too large to be able to satisfy the path constraint $\bar{z}_2 = 15\,\mathrm{g/L}$. We solve this by reducing the variance in the model parameter prior to $\mathbf{\Sigma}_{\theta,i} = 1\textsc{e}\text{-}4 \cdot \mathbf{I}$ when using the GP surrogates. On the one hand this means the prior parameter estimate may be overly confident and there is a higher probability of path constraint violations in the GP surrogate tests, but on the other hand the optimisation is more likely to converge on a feasible first experiment in each simulation.

Table 6.6 shows the performance of the GP surrogate approach in terms of average number of experiments required for successful model discrimination in 25 simulations of the yeast fermentation case study. Model $\mathcal{M}_1$ was correctly identified as the true data-generating model in all simulations. Compare to the results in Table 6.4 for the analytical method. The GP surrogate approach has a marginally worse performance, but still very similar to the analytical method.



(a) Analytical

(b) GP surrogates

Figure 6.12: Evolution of the average χ^2 scores (with one standard deviation) for the four rival models in the yeast fermentation case study, with experiments design using (a) the analytical approach, and (b) the GP surrogates approach.

Figure 6.12 shows the evolution of the average χ^2 scores (with one standard deviation) for the four rival models in the yeast fermentation case study. Figure 6.12a shows the average χ^2 score for the analytical approach with a cone constraint, and Figure 6.12b the average χ^2 score using the GP surrogates approach.

6.7.5 Computational Cost

The GP surrogate method assumes the black-box transition function is sufficiently expensive to evaluate to make it computationally infeasible to solve the design of experiments problem using either (i) finite difference approximations of the function's gradients for approximate marginalisation, or (ii) Monte Carlo techniques. But the GP surrogate models also introduce computational overhead. The Espie and Macchietto [1989] yeast fermentation considers cheap-to-evaluate analytical models. Table 6.7 compares the computational time for designing one experiment in this continuous-time case study using the GP surrogate method or the analytical method. It is clear from the table that the GP surrogate computational overhead may be significant. Each GP surrogate model (one for each output $f_{i,(d)}$ of each model f_i) has one the order of $N = 1000$ training data points. As described in Section 2.6, GP regression scales as $\mathcal{O}(N)$ and $\mathcal{O}(N^2)$ for mean and variance computation, respectively. Computing the gradients $\nabla_{\boldsymbol{x},\boldsymbol{u},\boldsymbol{\theta}}\boldsymbol{\mu}$ required for the first-order Taylor approximation of the

Action	**Computational time (GP method)**	**Computational time (analytical method)**
Train models	8-15s	-
Laplace approx. of $\mathbf{\Sigma}_{\theta,i}$	2.5–6min	1–2s
Solve for $\hat{\boldsymbol{u}}_{1:T}^*$	6–10h	2–5min
Evaluate $p(\boldsymbol{y}_{1:T}^{(i)} \,\vert\, \boldsymbol{x}_0, \hat{\boldsymbol{u}}_{1:T}^*)$	58–59s	0.10–0.15s

Table 6.7: The numbers in this table illustrate the computational overhead introduced by using GP surrogates for continuous-time models. The numbers are taken from a single run, with added time logging, on a machine with an Intel Core i7 9700K 3.6GHz 8-core processor and 16GB RAM.

marginal predictive distribution $\mathcal{N}(\breve{\boldsymbol{\mu}}, \breve{\mathbf{\Sigma}})$ in Section 5.1 scales as $\mathcal{O}(D_y N(D_x + D_u + D_\theta))$. For continuous-time models the computational overhead of using the GP surrogate method for design of experiments is often an order of magnitude higher than for discrete-time models, due to the number of surrogate model evaluations required for the ODE solver to converge when evaluating the marginal predictive distribution.

6.8 Open-Source Software

The results in Section 6.7 were produced using *doepy*[3], an open-source Python package developed by us for design of dynamic experiments.

6.9 Discussion

We have extended the Chapter 5 methodology of replacing black-box models with GP surrogates to design of *dynamic* experiments. The assumption is that the black-box element of a model is the latent state transition function f. In our problem formulation all models are written as first-order models, i.e. for discrete-time models the state $\boldsymbol{x}_{t+1}$ depends only on the state $\boldsymbol{x}_t$, and for continuous-time models the first-order ordinary differential equation

[3] `https://github.com/scwolof/doepy`

Approach	Mapping	Input dimensionality	Output dimensionality
Chapter 5	$\boldsymbol{x}_0, \boldsymbol{u}_{0:T-1}, \boldsymbol{\theta} \mapsto \boldsymbol{z}_{1:T}$	$D_x + D_u \times T + D_\theta$	$D_z \times T$
Chapter 6	$\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\theta} \mapsto \boldsymbol{x}_{t+1}$	$D_x + D_u + D_\theta$	D_x

Table 6.8: Comparison of the input and output dimensionality different GP surrogate models have for different mappings.

system have only the zeroth-order latent states $x(\tau)$ on the right-hand side. All n^{th} order models can be re-written as first-order models by introducing additional latent states.

In Chapter 5, the GP prior is put on the mapping from control to observation. This approach would not work for design of dynamic experiments, since the input dimensionality would be too high for accurate GP inference. The output dimensionality would also be high and require a large number of GP surrogate models. This work instead proposes putting the GP prior on the latent state transition, thus reducing the input and output dimensionality. Table 6.8 lists the input and output dimensionalities of the two different approaches.

We have assumed the observed states $\boldsymbol{z}_t$ are a linear combination of the latent states $\boldsymbol{x}_t$. Since the transition function f can be any non-linear function, using $\boldsymbol{z}_t = \mathbf{H}\boldsymbol{x}_t$ is only a minor restriction to the types of models we consider for domains such as pharmaceuticals and chemical manufacturing. State space models are also used e.g. in pixels-to-torque problems, where the latent states may be coordinates and velocities and the observed states are images [Wahlström et al., 2015], in which case the mapping from latent state to observed state is highly non-linear. In the more general case of $\boldsymbol{z}_t = g(\boldsymbol{x}_t)$, with g a non-linear function, approximate moment matching can be applied to infer the predictive distribution for $\boldsymbol{z}_t$.

We have considered an open-loop control approach, where a designed experiment is run to completion before data is analysed. Some work in literature, e.g. Galvanin et al. [2009] and De-Luca et al. [2016], consider a closed-loop approach for parameter estimation, where the optimal sequence of control inputs is updated online at time step t using data collected up until time step $t-1$. A closed-loop approach means repeatedly solving the experimental design optimisation problem in Equation (6.2.3). Although possible in principle for the GP surrogate method of design of dynamic experiments for model discrimination, the computational cost of solving the optimisation problem would likely be too high in practice.

7 Discussion and Conclusions

Design of experiments for black-box model discrimination is a difficult but important problem. Our novel method, hybridising the classical analytical approach and the data-driven approach using GP surrogate models, performs similarly to the analytical approach on classical case studies and is often several orders of magnitude faster than existing black-box approaches. It allows flexibility with regards to the structure and software implementations of the underlying models.

The GP surrogate method has been implemented in the open-source Python packages GPdoemd (for design of static experiments) and doepy (for design of dynamic experiments) and made available on GitHub. These software packages allow researchers and engineers to implement design of experiments for model discrimination, and include methods for approximation of marginal predictive distributions, design of experiments, and model discrimination. For methodology experts wishing to develop new model discrimination approaches, both GPdoemd and doepy include standard sets of case studies. The software packages in their current form require users to be familiar with Python. A GUI or API allowing external software to connect to GPdoemd and doepy would be needed for full uptake of these packages in industry and academia.

This work has not sought to find the global optima when solving the design of experiments optimisation problems. The objective functions are expected to be highly non-convex, hence advanced methods and software (see e.g. Tawarmalani and Sahinidis [2005], Misener and Floudas [2014] or Bongartz et al. [2018]) would be required in order to solve the objective functions to global optimality.

The GP surrogate approach currently has four major limitations:

1. The GP surrogates do not scale well to the large training data sets required to accurately emulate models with high-dimensional design and parameter spaces.
2. The GP (and sparse GP) regression methods presented in Chapter 2 are sensitive to the choice of covariance function and hyperparameter settings, and it is difficult to model functions with very different behaviour in different regions of the design space.
3. The GP surrogate approach relies on the same linear and Gaussian approximations as the analytical approach.
4. Challenges surrounding the training of the GP surrogate models, which usually require manual intervention from the user.

Statistical machine learning research is constantly improving GP regression methods, in terms of the size of training data sets the GP models can accommodate and in terms of the expressiveness of the models. Deep GPs have been trained on very large data sets [Liu et al., 2018]. Switching from regular GP regression to using deep GPs may help tackle the first two limitations listed above. Deep GPs can also infer non-Gaussian predictive distributions [Dutordoir et al., 2018; Salimbeni et al., 2019]. Thus, combined with design criteria that do not rely on Gaussian predictive distributions, deep GPs could also help tackle the third limitation. Deep GPs are just one example of more advanced surrogate models that in the future may improve optimal experimental design and model discrimination performance.

One of the major limitations a user will experience when interacting with GPdoemd and doepy is the manual intervention usually required to train the GP surrogate models. This includes identifying the training data space, selecting the locations for training data inputs and setting hyperparameter bounds. Efficient automated surrogate model training would facilitate using GPdoemd and doepy for design of experiments for black-box model discrimination. The challenges of training accurate GP surrogate models are not unique to GPdoemd and doepy, given the popularity of GP surrogate models in different fields, e.g. global optimisation [Boukouvala et al., 2017], structural engineering [Su et al., 2017] and chemical engineering [Jones et al., 2018]. It is possible that future research and more advanced surrogate models, e.g. deep GP models, will alleviate these challenges.

We have presented the GP surrogate approach as a hybridisation of analytical and data-driven approaches to optimal design of experiments. Plotted on an axis, with the analytical approach at one end and the data-driven approach at the other, the GP surrogate approach lies closer to the analytical end than the fully data-driven end. It is possible to imagine a spectrum of different trade-offs between accuracy and computational complexity along this axis. At the moment, many of the more advanced GP models' posteriors do not have closed form expressions, e.g. Salimbeni and Deisenroth [2017], hence they require sampling to approximate the predictive distributions. Hybrid approaches for design of experiments using such advanced GP surrogates may therefore lie closer to the original data-driven approaches. Future research should explore the range of different hybrid approaches and try to find useful guidance to practitioners for when different approaches are most suitable.

Model discrimination is useful for finding mechanistic models that adequately describe and predict a system's behaviour. As discussed in Chapter 1, mechanistic models are often needed in industry, e.g. to satisfy regulatory requirements. In the future, more models may be completely data-driven, following improvements in machine learning techniques and the reliability and interpretability of machine learning models. However, data-driven models can only be as accurate as the training data and model priors allow them to be. Extrapolation in regions of the input space where there is little training data will likely continue to be risky. Design of experiments for parameter estimation may help improve reliability of data-driven models. New research is combining mechanistic modelling and data-driven learning (see e.g. Sæmundsson et al. [2019]) for data-driven models with physically meaningful embeddings. This is a case of *hybrid modelling*. The challenge for model discrimination becomes to discern the effect of the mechanistic model part from the data-driven model part. In the future we may see new methods developed for design of experiments for hybrid model discrimination.

Bibliography

M. A. Álvarez. Multiple-output Gaussian processes. URL: `http://gpss.cc/gpss17/slides/multipleOutputGPs.pdf`, September 2017. Gaussian Process Summer School.

S. P. Asprey and S. Macchietto. Statistical tools for optimal dynamic model building. *Comput Chem Eng*, 24(2):1261–1267, 2000.

A. C. Atkinson. DT-optimum designs for model discrimination and parameter estimation. *J Stat Plan Inference*, 138:56–64, 2008.

A. C. Atkinson and V. V. Fedorov. The design of experiments for discriminating between two rival models. *Biometrika*, 62(1):57–70, 1975.

J. Babutzka, M. Bortz, A. Dinges, G. Foltin, D. Hajnal, H. Schultze, and H. Weiss. Machine learning supporting experimental design for product development in the lab. *Chem Ing Tech*, 91:277–284, 2019.

P. Bania. Bayesian input design for linear dynamical model discrimination. *Entropy*, 21(4): 351, 2019.

Z. Bar-Joseph, A. Gitter, and I. Simon. Studying and modelling dynamic biological processes using time-series gene expression data. *Nat Rev Genet*, 13(8):552–564, 2012.

A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind. Automatic differentiation in machine learning: A survey. *J Mach Learn Res*, 18(153):1–43, 2018.

J. N. Bazil, G. T. Buzzard, and A. E. Rundell. A global parallel model based design of experiments method to minimize model output uncertainty. *Bull Math Biol*, 74(3):688–716, 2012.

B. Beykal, F. Boukouvala, C. A. Floudas, N. Sorek, H. Zalavadia, and E. Gildin. Global optimization of grey-box computational systems using surrogate functions and application to highly constrained oil-field operations. *Comput Chem Eng*, 114:99–110, 2018.

D. Bongartz, J. Najman, S. Sass, and A. Mitsos. MAiNGO – McCormick-based algorithm for mixed-integer nonlinear global optimization. Technical report, Process Systems Engineering (AVT.SVT), RWTH Aachen University, 2018.

F. Boukouvala, R. Misener, and C. A. Floudas. Global optimization advances in mixed-integer nonlinear programming, MINLP, and constrained derivative-free optimization, CDFO. *Eur J Oper Res*, 252(3):701–727, 2016.

F. Boukouvala, M. M. F. Hasan, and C. A. Floudas. Global optimization of general constrained grey-box models: New method and its application to constrained PDEs for pressure swing adsorption. *J Global Optim*, 67(1):3–42, 2017.

G. E. P. Box and D. R. Cox. An analysis of transformations. *J R Stat Soc Series B Methodol*, 26(2):211–252, 1964.

G. E. P. Box and W. J. Hill. Discrimination among mechanistic models. *Technometrics*, 9 (1):57–71, 1967.

G. Buzzi-Ferraris. Some observations on the paper "Optimal experimental design for discriminating numerous model candidates: The AWDC criterion". *Ind Eng Chem Res*, 49: 9561–9562, 2010.

G. Buzzi-Ferraris and P. Forzatti. A new sequential experimental design procedure for discriminating among rival models. *Chem Eng Sci*, 38(2):225–232, 1983.

G. Buzzi-Ferraris and F. Manenti. Kinetic models analysis. *Chem Eng Sci*, 64(5):1061–1074, 2009.

G. Buzzi-Ferraris, P. Forzatti, G. Emig, and H. Hofmann. Sequential experimental design for model discrimination in the case of multiple responses. *Chem Eng Sci*, 39(1):81–85, 1984.

G. Buzzi-Ferraris, P. Forzatti, and P. Canu. An improved version of a sequential design criterion for discriminating among rival multiresponse models. *Chem Eng Sci*, 45(2): 477–481, 1990.

J. A. Caballero and I. E. Grossmann. An algorithm for the use of surrogate models in modular flowsheet optimization. *AIChE J*, 54(10):2633–2650, 2008.

G. C. Calafiore and M. C. Campi. The scenario approach to robust control design. *IEEE Trans Autom Control*, 51(5):742–753, 2006.

R. R. Carpio, F. F. Furlan, R. C. Giordano, and A. R. Secchi. A Kriging-based approach for conjugating specific dynamic models into whole plant stationary simulations. *Comput Chem Eng*, 119:190–194, 2018.

K. Chaloner and I. Verdinelli. Bayesian experimental design: A review. *Statist Sci*, 10(3): 273–304, 1995.

B. H. Chen and S. P. Asprey. On the design of optimally informative dynamic experiments for model discrimination in multiresponse nonlinear situations. *Ind Eng Chem Res*, 42(7): 1379–1390, 2003.

S. Cheong and I. R. Manchester. Input design for model discrimination and fault detection via convex relaxation. In *American Control Conference (ACC)*, pages 684–690, Portland, OR, USA, 2014a.

S. Cheong and I. R. Manchester. Model predictive control combined with model discrimination and fault detection. In *Proceedings of the 19th World Congress of the International Federation of Automatic Control (IFAC)*, pages 1434–1439, Cape Town, South Africa, 2014b.

A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to derivative-free optimization.* MPS-SIAM Series on Optimization. SIAM, Philadelphia, 2009.

C. Cortes and V. Vapnik. Support-vector networks. *Mach Learn*, 20(3):273–297, 1995.

J. De Leeuw. Introduction to Akaike (1973) information theory and an extension of the maximum likelihood principle. In S. Kotx and N. L. Johnson, editors, *Breakthroughs in Statistics I*, pages 599–609. Springer, 1992.

R. De-Luca, F. Galvanin, and F. Bezzo. A methodology for direct exploitation of available information in the online model-based redesign of experiments. In J. K. Huusom, K. V. Gernaey, and R. Gani, editors, *Proceedings of the 12th International Symposium on Process Systems Engineering & 25th European Symposium of Computer Aided Process Engineering*, volume 91 of *Comp Chem Eng*, pages 195–205. Elsevier, Copenhagen, Denmark, 2016.

M. P. Deisenroth. *Efficient Reinforcement Learning using Gaussian Processes.* PhD thesis, The Karlsruhe Institute of Technology, 2010. ISBN 978-3-86644-569-7.

M. P. Deisenroth and C. E. Rasmussen. PILCO: A model-based and data-efficient approach to policy search. In *ICML '11: Proceedings of the 28th International Conference on Machine Learning*, pages 465–472, Bellevue, WA, USA, 2011.

M. P. Deisenroth, M. F. Huber, and U. D. Hanebeck. Analytic moment-based Gaussian process filtering. In *ICML '09: Proceedings of the International Conference on Machine Learning*, pages 225–232, Montreal, Canada, 2009.

R. P. Dickinson and R. J. Gelinas. Sensitivity analysis of ordinary differential equation systems–A direct method. *J Comput Phys*, 21(2):123–143, 1976.

J. A. DiMasi, H. G. Grabowski, and R. W. Hansen. Innovation in the pharmaceutical industry: New estimates of R&D costs. *J Health Econ*, 47:20–33, 2016.

L. Dony, J. Mackerodt, S. Ward, S. Filippi, M. P. H. Stumpf, and J. Liepe. PEITH(Θ): perfecting experiments with information theory in Python with GPU support. *Bioinformatics*, 34(7):1249–1250, 2017.

C. C. Drovandi, J. M. McGree, and A. N. Pettitt. A sequential Monte Carlo algorithm to incorporate model uncertainty in Bayesian sequential design. *J Comput Graph Stat*, 23 (1):3–24, 2014.

V. Dutordoir, H. Salimbeni, J. Hensman, and M. P. Deisenroth. Gaussian process conditional density estimation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 2385–2395. Curran Associates, Inc., 2018.

D. C. Dyson and J. M. Simon. Kinetic expression with diffusion correction for ammonia synthesis on industrial catalyst. *Ind Eng Chem Fundamen*, 7(4):605–610, 1968.

D. Espie and S. Macchietto. The optimal design of dynamic experiments. *AIChE Journal*, 35(2):223–229, 1989.

E. I. Ette and P. J. Williams. Population pharmacokinetics i: Background, concepts, and models. *Ann Pharmacother*, 38:1702–1706, 2004.

European Medicines Agency. Guideline on the use of pharmacogenetic methodologies in the pharmacokinetic evaluation of medicinal products, December 2011.

European Medicines Agency. Guideline on the qualification and reporting of physiologically based pharmacokinetic (PBPK) modelling and simulation, July 2016.

European Union. Regulation (EU) 2017/745, April 2017.

T. Evans and P. Nair. Scalable Gaussian processes with grid-structured eigenfunctions (GP-GRIEF). In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proc Mach Learn Res*, pages 1417–1426. PMLR, Stockholm, Sweden, 2018.

I. Fahmi and S. Cremaschi. Process synthesis of biodiesel production plant using artificial neural networks as the surrogate models. *Comput Chem Eng*, 46:105–123, 2012.

P. Farrell, D. Ham, S. Funke, and M. Rognes. Automated derivation of the adjoint of high-level transient finite element programs. *SIAM J Sci Comput*, 35(4):C369–C393, 2013.

J. M. M. Faust, J. Fu, B. Chachuat, and A. Mitsos. Optimization of dynamic systems with rigorous path constraint satisfaction. In Z. Kravanja and M. Bogataj, editors, *Proceedings of the 26th European Symposium on Computer Aided Process Engineering (ESCAPE)*, volume 38 of *Comput Aided Chem Eng*, pages 643–648. Elsevier, Portorož, Slovenia, 2016.

R. I. Field. Why is health care regulation so complex? *Pharm Ther*, 33(10):607–608, 2008.

R. A. Fisher. The arrangement of field experiments. *Journal of the Ministry of Agriculture of Great Britain*, 33:503–513, 1926.

R. A. Fisher. *The Design of Experiments*. Hafner Publishing Company, New York, NY, USA, 9th edition, 1971. (1st edition published in 1935).

F. Galvanin, M. Barolo, F. Bezzo, and S. Macchietto. A framework for model-based design of parallel experiments in dynamic systems. In W. Marquardt and C. Pantelides, editors, *16th European Symposium on Computer Aided Process Engineering and 9th International Symposium on Process Systems Engineering*, volume 21 of *Comput Aided Chem Eng*, pages 249–254. Elsevier, 2006.

F. Galvanin, S. Macchietto, and D. Bezzo. Model-based design of parallel experiments. *Ind Eng Chem Res*, 46(3):871–882, 2007.

F. Galvanin, M. Barolo, and F. Bezzo. Online model-based redesign of experiments for parameter estimation in dynamic systems. *Ind Eng Chem Res*, 48(9):4415–4427, 2009.

F. Galvanin, C. C. Ballan, M. Barolo, and F. Bezzo. A general model-based design of experiments approach to achieve practical identifiability of pharmacokinetic and pharmacodynamic models. *J Pharmacokinet Phar*, 40(4):451–467, 2013.

F. Galvanin, E. Cao, N. Al-Rifai, A. Gavriilidis, and V. Dua. A joint model-based experimental design approach for the identification of kinetic models in continuous flow laboratory reactors. *Comp Chem Eng*, 95:202–215, 2016.

L. J. Gillespie and J. A. Beattie. The thermodynamic treatment of chemical equilibria in systems composed of real gases. I. An approximate equation for the mass action function applied to the existing data on the Haber equilibrium. *Phys Rev*, 36:743–753, 1930.

A. Girard, C. Rasmussen, J. Quiñonero-Candela, and R. Murray-Smith. Gaussian process priors with uncertain inputs – application to multiple-step ahead time series forecasting. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems*, volume 15, pages 545–552. MIT Press, 2003.

J. Gonzalez, Z. Dai, P. Hennig, and N. Lawrence. Batch Bayesian optimization via local penalization. In A. Gretton and C. C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proc Mach Learn Res*, pages 648–657. PMLR, Cadiz, Spain, 2016.

GPy. GPy: A Gaussian process framework in python. `http://github.com/SheffieldML/GPy`, since 2012.

A. A. Heller, S. Y. Lockwood, T. M. Janes, and D. M. Spence. Technologies for measuring pharmacokinetic profiles. *Annu Rev Anal Chem*, 11(1), 2018.

J. D. Helterbrand and N. Cressie. Universial cokriging under intrinsic coregionalization. *Math Geol*, 26:205–226, 1994.

J. Hensman, N. Durrande, and A. Solin. Variational Fourier features for Gaussian processes. *J Mach Learn Res*, 18(151):1–52, 2018.

T. Hervey. The impacts of European Union law on the health care sector: Institutional overview. *Eurohealth*, 16(4):5–7, 2010.

C. Hoffmann. *Numerical aspects of uncertainty in the design of optimal experiments for model discrimination.* PhD thesis, Ruprecht-Karls-Universität Heidelberg, 2017.

W. G. Hunter and A. M. Reiner. Designs for discriminating between two rival models. *Technometrics*, 7(3):307–323, 1965.

J. A. Jacquez and T. Perry. Parameter estimation: local identifiability of parameters. *Am J Physiol Endocrinol Metab*, 258(4):E727–E736, 1990.

M. Jones, H. Forero-Hernandez, A. Zubov, B. Sarup, and G. Sin. Superstructure optimization of oleochemical processes with surrogate models. In M. R. Eden, M. G. Ierapetritou, and G. P. Towler, editors, *Proceedings of the 13th International Symposium on Process Systems Engineering*, volume 44 of *Comput Aided Chem Eng*, pages 277–282. Elsevier, San Diego, CA, USA, 2018.

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

K. J. Keesman and E. Walter. Optimal input design for model discrimination using Pontryagin's maximum principle: Application to kinetic model structures. *Automatica*, 50: 1535–1538, 2014.

J. Ko and D. Fox. GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models. *Auton Robot*, 27:75–90, 2009.

J. Ko, D. J. Klein, D. Fox, and D. Haehnel. Gaussian processes and reinforcement learning for identification and control of an autonomous blimp. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 742–747, Roma, Italy, 2007.

N. D. Lawrence. Fitting covariance and multioutput Gaussian processes. URL: `http://gpss.cc/gpss15/talks/gp_gpss15_session3.pdf`, September 2015. Gaussian Process Summer School.

O. Levenspiel. *Chemical Reaction Engineering*. Wiley, New York, NY, USA, 3rd edition, 1999.

T. Li, S. Sun, T. P. Sattar, and J. M. Corchado. Fight sample degeneracy and impoverishment in particle filters: A review of intelligent approaches. *Expert Syst Appl*, 41(8): 3944–3954, 2014.

J. Liepe, S. Filippi, M. Komorowski, and M. P. H. Stumpf. Maximizing the information content of experiments in systems biology. *PLOS Comput Biol*, 9(1):1–13, 2013.

H. Liu, Y.-S. Ong, X. Shen, and J. Cai. When Gaussian process meets big data: A review of scalable GPs. *ArXiv e-print 1807.01065*, 2018. URL `https://arxiv.org/abs/1807.01065`.

E. Martelli and E. Amaldi. PGS-COM: A hybrid method for constrained non-smooth black-box optimization problems: Brief review, novel algorithm and comparative evaluation. *Comput Chem Eng*, 63:108–139, 2014.

S. S. Mc Mahon, A. Sim, S. Filippi, R. Johnson, J. Liepe, D. Smith, and M. P. H. Stumpf. Information theory and signal transduction systems: From molecular information processing to network inference. *Semin Cell Dev Biol*, 35:98–108, 2014.

D. Meeter, W. Pirie, and W. Blot. A comparison of two model-discrimination criteria. *Technometrics*, 12(3):457–470, 1970.

M. Mehrian, Y. Guyot, I. Papantoniou, S. Olofsson, M. Sonnaert, R. Misener, and L. Geris. Maximizing neotissue growth kinetics in a perfusion bioreactor: an in silico strategy using model reduction and Bayesian optimization. *Biotechnol Bioeng*, 115(3):617–629, 2018.

A. Mesbah, S. Streif, , R. Findeisen, and R. D. Braatz. Active fault detection for nonlinear systems with probabilistic uncertainties. In *Proceedings of the 19th World Congress of the International Federation of Automatic Control (IFAC)*, pages 7079–7084, Cape Town, South Africa, 2014a.

A. Mesbah, S. Streif, , R. Findeisen, and R. D. Braatz. Stochastic nonlinear model predictive control with probabilistic constraints. In *American Control Conference (ACC)*, pages 2413–2419, Portland, OR, USA, 2014b.

C. Michalik, M. Stuckert, and W. Marquardt. Optimal experimental design for discriminating numerous model candidates: The AWDC criterion. *Ind Eng Chem Res*, 49:913–919, 2010.

R. Misener and C. A. Floudas. ANTIGONE: Algorithms for coNTinuous / Integer Global Optimization of Nonlinear Equations. *J Global Optim*, 59:503–526, 2014.

R. M. Neal. Regression and classification using gaussian process priors. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, editors, *Bayesian Statistics 6*, pages 475–501. Oxford University Press, 1999.

R. Neidinger. Introduction to automatic differentiation and MATLAB object-oriented programming. *SIAM Rev*, 52(3):545–563, 2010.

F. Nielsen. Closed-form information-theoretic divergences for statistical mixtures. In *Proceedings of the 21st International Conference on Pattern Recognition*, pages 1723–1726, 2012.

K. Ogungbenro and L. Aarons. How many subjects are necessary for population pharmacokinetic experiments? Confidence interval approach. *Eur J Clin Pharmacol*, 64(7):705, 2008.

S. Olofsson, M. P. Deisenroth, and R. Misener. Design of experiments for model discrimination hybridising analytical and data-driven approaches. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proc Mach Learn Res*, pages 3908–3917. PMLR, Stockholm, Sweden, 2018a.

S. Olofsson, M. P. Deisenroth, and R. Misener. Design of experiments for model discrimination using Gaussian process surrogate models. In M. R. Eden, M. G. Ierapetritou, and G. P. Towler, editors, *Proceedings of the 13th International Symposium on Process Systems Engineering*, volume 44 of *Comput Aided Chem Eng*, pages 847–852. Elsevier, San Diego, CA, USA, 2018b.

S. Olofsson, L. Hebing, S. Niedenführ, M. P. Deisenroth, and R. Misener. GPdoemd: A Python package for design of experiments for model discrimination. *Comp Chem Eng*, 125:54–70, 2019a.

S. Olofsson, M. Mehrian, R. Calandra, L. Geris, M. P. Deisenroth, and R. Misener. Bayesian multi-objective optimisation with mixed analytical and black-box functions: Application to tissue engineering. *IEEE Trans Biomed Eng*, 66:727–739, 2019b.

B. K. Pagnoncelli, S. Ahmed, and A. Shapiro. Sample average approximation method for chance constrained programming: Theory and applications. *J Optimiz Theory App*, 142 (2):399–416, 2009.

K. Palmer and M. Realff. Optimization and validation of steady-state flowsheet simulation metamodels. *Chem Eng Res Des*, 80(7):773–782, 2002.

H. J. Pesch. Real-time computation of feedback controls for constrained optimal control problems. part 1: Neighbouring extremals. *Optim Contr Appl Met*, 10(2):129–145, 1989.

K. B. Petersen and M. S. Pedersen. The Matrix Cookbook, 2012. Technical University of Denmark. Version 20121115.

R. M. Plenge. Disciplined approach to drug discovery and early development. *Sci Transl Med*, 8(349):349ps15, 2016.

K. B. S. Prasad and M. Someswara Rao. Use of expected likelihood in sequential model discrimination in multiresponse systems. *Chem Eng Sci*, 32:1411–1418, 1977.

A. Prékopa. *Stochastic Programming.* Springer Netherlands, Dordrecht, Netherlands, 1995.

M. Quaglio, E. S. Fraga, and F. Galvanin. Model-based design of experiments in the presence of structural model uncertainty: an extended information matrix approach. *Chem Eng Res Des*, 136:129–143, 2018.

J. Quiñonero-Candela, A. Girard, J. Larsen, and C. E. Rasmussen. Propagation of uncertainty in Bayesian kernel models - application to multiple-step ahead forecasting. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 701–704, Hong Kong, China, 2003.

H. Rabitz, M. Kramer, and D. Dacol. Sensitivity analysis in chemical kinetics. *Annu Rev Phys Chem*, 34(1):419–461, 1983.

C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning.* MIT Press, 2006.

A. Rényi. On the foundations of information theory. *Rev Inst Int Stat*, 33(1):1–14, 1965.

E. G. Ryan, C. C. Drovandi, and A. N. Pettitt. Fully Bayesian experimental design for pharmacokinetic studies. *Entropy*, 17:1063–1089, 2015.

E. G. Ryan, C. C. Drovandi, J. M. McGree, and A. N. Pettitt. A review of modern computational algorithms for Bayesian optimal design. *Int Stat Rev*, 84:128–154, 2016.

S. Sæmundsson, A. Terenin, K. Hofmann, and M. P. Deisenroth. Variational integrator networks for physically meaningful embeddings. *ArXiv e-print 1910.09349*, 2019. URL `https://arxiv.org/abs/1910.09349`.

H. Salimbeni and M. P. Deisenroth. Doubly stochastic variational inference for deep Gaussian processes. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4588–4599. Curran Associates, Inc., 2017.

H. Salimbeni, V. Dutordoir, J. Hensman, and M. P. Deisenroth. Deep Gaussian processes with importance-weighted variational inference. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proc Mach Learn Res*, pages 5589–5598, Long Beach, California, USA, 2019. PMLR.

J. W. Scannell and J. Bosley. When quality beats quantity: Decision theory, drug discovery, and the reproducibility crisis. *PLOS ONE*, 11(2):1–21, 2016.

M. Schwaab, F. M. Silva, C. A. Queipo, A. G. Barreto Jr., M. Nele, and J. C. Pinto. A new approach for sequential experimental design for model discrimination. *Chem Eng Sci*, 61: 5791–5806, 2006.

M. Seeger, C. K. I. Williams, and N. D. Lawrence. Fast forward selection to speed up sparse Gaussian process regression. In C. M. Bishop and B. J. Frey, editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics, AISTATS*, pages 1–8. Society for Artificial Intelligence and Statistics, Key West, Florida, USA, 2003.

M. Seeger, Y. Teh, and M. I. Jordan. Semiparametric latent factor models. Technical report, Workshop on Artificial Intelligence and Statistics 10, 2004.

B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proc IEEE*, 104(1):148–175, 2016.

A. Shapiro, A. P. Ruszczyński, and D. Dentcheva. *Lectures on Stochastic Programming: Modeling and Theory.* SIAM, Philadelphia, PA, USA, 2009.

D. Skanda and D. Lebiedz. An optimal experimental design approach to model discrimination in dynamic biochemical systems. *Bioinformatics*, 26(7):939–945, 2010.

D. Skanda and D. Lebiedz. A robust optimization approach to experimental design for model discrimination of dynamical systems. *Math Program*, 141:405–433, 2013.

A. J. Smola and P. L. Bartlett. Sparse greedy Gaussian process regression. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 619–625. MIT Press, 2001.

E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In Y. Weiss, B. Schölkopf, and J. C. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 1257–1264. MIT Press, 2006.

J.-L. Steimer, A. Mallet, J.-L. Golmard, and J.-F. Boisvieux. Alternative approaches to estimation of population pharmacokinetic parameters: Comparison with the nonlinear mixed-effect model. *Drug Metab Rev*, 15(1–2):265–292, 1984.

S. Streif, F. Petzke, A. Mesbah, R. Findeisen, and R. D. Braatz. Optimal experimental design for probabilistic model discrimination using polynomial chaos. In *Proceedings of the 19th World Congress of the International Federation of Automatic Control (IFAC)*, pages 4103–4109, Cape Town, South Africa, 2014.

G. Su, L. Peng, and L. Hu. A Gaussian process-based dynamic surrogate model for complex engineering structural reliability analysis. *Struct Saf*, 68:97–109, 2017.

N. Tandogan, S. García-Muñoz, M. Sen, T. M. Wilson, J. Y. Buser, S. P. Kolis, I. V. Borkar, and C. A. Alt. Use of model discrimination method in drug substance process developments. Presentation at the AIChE Annual Meeting 29 Oct–3 Nov, 2017, Minneapolis, MN, USA, 2017.

M. Tawarmalani and N. V. Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Math Program*, 103:225–249, 2005.

M. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In D. van Dyk and M. Welling, editors, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 5 of *Proc Mach Learn Res*, pages 567–574, Clearwater Beach, FL, USA, 2009. PMLR.

C. Tommasi. Optimal designs for both model discrimination and parameter estimation. *J Stat Plan Inference*, 139:4123–4132, 2009.

L. S. Tsimring. Noise in biology. *Rep Prog Phys*, 77(2):026601, 2014.

T. Tuntland, B. Ethell, T. Kosaka, F. Blasco, R. X. Zang, M. Jain, T. Gould, and K. Hoffmaster. Implementation of pharmacokinetic and pharmacodynamic strategies in early research phases of drug discovery and development at Novartis Institute of Biomedical Research. *Front Pharmacol*, 5:174, 2014.

D. Ulmasov, C. Baroukh, B. Chachuat, M. P. Deisenroth, and R. Misener. Bayesian optimization with dimension scheduling: Application to biological systems. In Z. Kravanja and M. Bogataj, editors, *Proceedings of the 26th European Symposium on Computer Aided Process Engineering*, volume 38 of *Comput Aided Chem Eng*, pages 1051–1056. Elsevier, 2016.

U.S. Food & Drug Administration. Food and drugs, 21 C.F.R. § 314, 25 January 2018.

J. Vanlier, C. A. Tiemann, P. A. J. Hilbers, and N. A. W. van Riel. Optimal experiment design for model selection in biochemical networks. *BMC Syst Biol*, 8(20), 2014.

N. Wahlström, T. B. Schön, and M. P. Deisenroth. Learning deep dynamical models from image pixels. *IFAC-PapersOnLine*, 48(28):1059–1064, 2015. 17th IFAC Symposium on System Identification SYSID 2015.

F. Wang, T. Syeda-Mahmood, B. C. Vemuri, D. Beymer, and A. A. Rangarajan. Closed-form Jensen-Renyi divergence for mixture of Gaussians and applications to group-wise shape registration. In G.-Z. Yang, D. Hawkes, D. Rueckert, A. Noble, and C. Taylor, editors, *Medical Image Computing and Computer-Assisted Intervention*, pages 648–655, London, United Kingdom, 2009. Springer Berlin Heidelberg.

T. H. Waterhouse, J. A. Eccleston, and S. B. Duffull. Optimal design criteria for discrimination and estimation in nonlinear models. *J Biopharm Stat*, 19:386–402, 2009.

D. C. Woods, A. M. Overstall, M. Adamou, and T. W. Waite. Bayesian design of experiments for generalized linear models and dimensional analysis with industrial and scientific application. *Qual Eng*, 29(1):91–103, 2017.

S. Yang, S. Kiang, P. Farzan, and M. Ierapetritou. Optimization of reaction selectivity using CFD-based compartmental modeling and surrogate-based optimization. *Processes*, 7(1): 9, 2019.

F. Yates. Sir Ronald Fisher and the design of experiments. *Biometrics*, 20(2):307–321, 1964.

A Latent State Derivatives

This appendix is concerned with the calculation of

$$\frac{\partial\{\boldsymbol{\mu}_t,\, \boldsymbol{\Sigma}_t\}}{\partial\{\boldsymbol{\mu}_{t-1},\, \boldsymbol{\Sigma}_{t-1},\, \hat{\boldsymbol{u}}_{t-1}\}}\,, \tag{A.0.1}$$

required for gradient-based optimisation using the Taylor approximation-based prediction of latent state distributions described in Section 6.3. The derivatives of $\boldsymbol{\mu}_t$ and $\boldsymbol{\Sigma}_t$ with respect to the latent state mean $\boldsymbol{\mu}_{t-n}$ and covariance $\boldsymbol{\Sigma}_{t-n}$, and control inputs $\hat{\boldsymbol{u}}_{t-n}$ at time steps $t-n$ for $n \geq 2$ follow from the chain rule, e.g.

$$\frac{\partial \boldsymbol{\mu}_t}{\partial \boldsymbol{\mu}_{t-2}} = \frac{\partial \boldsymbol{\mu}_t}{\partial \boldsymbol{\mu}_{t-1}} \frac{\partial \boldsymbol{\mu}_{t-1}}{\partial \boldsymbol{\mu}_{t-2}} + \frac{\partial \boldsymbol{\mu}_t}{\partial \boldsymbol{\Sigma}_{t-1}} \frac{\partial \boldsymbol{\Sigma}_{t-1}}{\partial \boldsymbol{\mu}_{t-2}}\,,$$

with some abuse of tensor product notation.

We assume that the partial first- and second-order derivatives of $\mu_f(\cdot)$ and $\Sigma_f(\cdot)$ are available with respect to the latent state $\boldsymbol{x}_t$, control input $\boldsymbol{u}_t$ and model parameters $\boldsymbol{\theta}$, either in the form of an analytical transition function f or with a GP surrogate replacing f, with $\mu_f(\cdot)$ and $\Sigma_f(\cdot)$ defined as in Equation (6.3.1) on page 119.

To simplify notation, let $\nabla_{\boldsymbol{\xi}} g$ denote

$$\nabla_{\boldsymbol{\xi}} g = \left.\frac{\partial g(\boldsymbol{\xi},\, \boldsymbol{\xi}',\, \dots,\, \boldsymbol{\xi}'')}{\partial \boldsymbol{\xi}}\right|_{(\boldsymbol{\xi},\, \boldsymbol{\xi}', \dots,\, \boldsymbol{\xi}'') = (\mathbb{E}[\boldsymbol{\xi}],\, \mathbb{E}[\boldsymbol{\xi}']), \dots,\, \mathbb{E}[\boldsymbol{\xi}''])}\,,$$

with $g(\cdot)$ a function or dependent variable.

A.1 Matrix Calculus Identities

To simplify the calculations of the derivatives in Equation (A.0.1) in the following sections, some recurring terms are described here using identities from The Matrix Cookbook [Petersen and Pedersen, 2012].

Identity 1: Let $\boldsymbol{g} \in \mathbb{R}^{D_g}$ depend on $\boldsymbol{\xi} \in \mathbb{R}^{D_\xi}$ and $\boldsymbol{\xi}' \in \mathbb{R}^{D_{\xi'}}$, and let $\mathbf{B} \in \mathbb{R}^{D_\xi \times D}$ be a constant matrix. Then

$$\begin{aligned} \partial\mathbf{A} &= \frac{\partial}{\partial\boldsymbol{\xi}'}\left[\nabla_{\boldsymbol{\xi}}\boldsymbol{g}\,\mathbf{B}\right], \quad \partial\mathbf{A} \in \mathbb{R}^{D_g \times D \times D_{\xi'}} \\ \Rightarrow \quad [\partial\mathbf{A}]_i &= \mathbf{B}^\top \frac{\partial^2[\boldsymbol{g}]_i}{\partial\boldsymbol{\xi}\partial\boldsymbol{\xi}'}, \quad [\partial\mathbf{A}]_i \in \mathbb{R}^{D \times D_{\xi'}}. \end{aligned}$$

Identity 2: Let $\boldsymbol{g} \in \mathbb{R}^{D_g}$ depend on $\boldsymbol{\xi} \in \mathbb{R}^{D_\xi}$, $\boldsymbol{\xi}' \in \mathbb{R}^{D_{\xi'}}$ and $\boldsymbol{\xi}'' \in \mathbb{R}^{D_{\xi''}}$, and let $\mathbf{B} \in \mathbb{R}^{D_\xi \times D_{\xi'}}$ be a constant matrix. Then

$$\begin{aligned} \partial\mathbf{A} &= \frac{\partial}{\partial\boldsymbol{\xi}''}\left[\nabla_{\boldsymbol{\xi}}\boldsymbol{g}\,\mathbf{B}\,\left(\nabla_{\boldsymbol{\xi}'}\boldsymbol{g}\right)^\top\right], \quad \partial\mathbf{A} \in \mathbb{R}^{D_g \times D_g \times D_{\xi''}} \\ \Rightarrow \quad [\partial\mathbf{A}]_{ij} &= \frac{\partial^2[\boldsymbol{g}]_i}{\partial\boldsymbol{\xi}''\partial\boldsymbol{\xi}}\,\mathbf{B}\,\frac{\partial[\boldsymbol{g}]_j}{\partial\boldsymbol{\xi}'} + \left(\frac{\partial[\boldsymbol{g}]_i}{\partial\boldsymbol{\xi}}\right)^\top \mathbf{B}\,\frac{\partial^2[\boldsymbol{g}]_j}{\partial\boldsymbol{\xi}'\partial\boldsymbol{\xi}''}, \quad [\partial\mathbf{A}]_{ij} \in \mathbb{R}^{D_{\xi''}}. \end{aligned}$$

Identity 3: Let $\boldsymbol{g} \in \mathbb{R}^{D_g}$ depend on $\boldsymbol{\xi} \in \mathbb{R}^{D_\xi}$, and let $\mathbf{B} \in \mathbb{R}^{D_\xi \times D}$. Then

$$\begin{aligned} \partial\mathbf{A} &= \frac{\partial}{\partial\mathbf{B}}\left[\nabla_{\boldsymbol{\xi}}\boldsymbol{g}\,\mathbf{B}\right], \quad \partial\mathbf{A} \in \mathbb{R}^{D_g \times D \times D_\xi \times D} \\ \Rightarrow \quad [\partial\mathbf{A}]_{ijmn} &= \begin{cases} \left[\frac{\partial\boldsymbol{g}}{\partial\boldsymbol{\xi}}\right]_{im}, & \text{if } j = n, \\ 0, & \text{otherwise}. \end{cases} \end{aligned}$$

Identity 4: Let $\boldsymbol{g} \in \mathbb{R}^{D_g}$ depend on $\boldsymbol{\xi} \in \mathbb{R}^{D_\xi}$ and $\boldsymbol{\xi}' \in \mathbb{R}^{D_{\xi'}}$, and let $\mathbf{B} \in \mathbb{R}^{D_\xi \times D_{\xi'}}$. Then

$$\begin{aligned} \partial\mathbf{A} &= \frac{\partial}{\partial\mathbf{B}}\left[\nabla_{\boldsymbol{\xi}}\boldsymbol{g}\,\mathbf{B}\,\left(\nabla_{\boldsymbol{\xi}'}\boldsymbol{g}\right)^\top\right], \quad \partial\mathbf{A} \in \mathbb{R}^{D_g \times D_g \times D_\xi \times D_{\xi'}} \\ \Rightarrow \quad [\partial\mathbf{A}]_{ijmn} &= \left[\frac{\partial\boldsymbol{g}}{\partial\boldsymbol{\xi}}\right]_{im} \times \left[\frac{\partial\boldsymbol{g}}{\partial\boldsymbol{\xi}'}\right]_{jn}. \end{aligned}$$

Identity 5: Let $\boldsymbol{g} \in \mathbb{R}^{D_g}$ depend on $\boldsymbol{\xi} \in \mathbb{R}^{D_\xi}$, and let $\mathbf{B} \in \mathbb{R}^{D_\xi \times D}$ depend on $\boldsymbol{\xi}' \in \mathbb{R}^{D_{\xi'}}$. Then

$$\begin{aligned} \partial\mathbf{A} &= \frac{\partial}{\partial\boldsymbol{\xi}'}\left[\nabla_{\boldsymbol{\xi}}\boldsymbol{g}\,\mathbf{B}\right], \quad \partial\mathbf{A} \in \mathbb{R}^{D_g \times D \times D_{\xi'}} \\ \Rightarrow \quad [\partial\mathbf{A}]_{ij} &= \sum_{n=1}^{D_\xi}\left(\left[\frac{\partial\boldsymbol{g}}{\partial\boldsymbol{\xi}}\right]_{in} \times \left[\frac{\partial\mathbf{B}}{\partial\boldsymbol{\xi}'}\right]_{nj}\right), \quad [\partial\mathbf{A}]_{ij} \in \mathbb{R}^{D_{\xi'}}. \end{aligned}$$

Note that these identities hold for all cases $\boldsymbol{\xi} \neq \boldsymbol{\xi}'$, $\boldsymbol{\xi} = \boldsymbol{\xi}'$, $\boldsymbol{\xi} = \boldsymbol{\xi}''$, *et cetera.*

A.2 Discrete-Time State Space Models

The latent state transition between consecutive time steps is described by a discrete-time model (see Section 6.3.1). The derivatives of the predicted latent state mean $\boldsymbol{\mu}_t$ at time step t with respect to the latent state mean $\boldsymbol{\mu}_{t-1}$ and the desired control input $\hat{\boldsymbol{u}}_{t-1}$ at time step $t-1$, and the model parameters $\hat{\boldsymbol{\theta}}$ are given by

$$\frac{\partial \boldsymbol{\mu}_t}{\partial \{\boldsymbol{\mu}_{t-1},\, \hat{\boldsymbol{u}}_{t-1}, \hat{\boldsymbol{\theta}}\}} = \left.\frac{\partial \mu_f(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\theta})}{\partial \{\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\theta}\}}\right|_{(\boldsymbol{x},\boldsymbol{u},\boldsymbol{\theta})=(\boldsymbol{\mu}_{t-1},\hat{\boldsymbol{u}}_{t-1},\hat{\boldsymbol{\theta}})} .$$

The Taylor approximation of $\boldsymbol{\mu}_t$ does not depend on $\boldsymbol{\Sigma}_{t-1}$, hence $\partial\boldsymbol{\mu}_t/\partial\boldsymbol{\Sigma}_{t-1} = \mathbf{0}$.

The expression for the predicted latent state covariance $\boldsymbol{\Sigma}_t$ in Equation (6.3.4) on page 120 can be expanded as

$$\begin{aligned}\boldsymbol{\Sigma}_t = {} & \nabla_{\boldsymbol{x}_{t-1}}\boldsymbol{\mu}_t\boldsymbol{\Sigma}_{t-1}\left(\nabla_{\boldsymbol{x}_{t-1}}\boldsymbol{\mu}_t\right)^\top + \nabla_{\boldsymbol{u}_{t-1}}\boldsymbol{\mu}_t\boldsymbol{\Sigma}_u\left(\nabla_{\boldsymbol{u}_{t-1}}\boldsymbol{\mu}_t\right)^\top + \nabla_{\boldsymbol{\theta}}\boldsymbol{\mu}_t\boldsymbol{\Sigma}_\theta\left(\nabla_{\boldsymbol{\theta}}\boldsymbol{\mu}_t\right)^\top \\ & + \nabla_{\boldsymbol{x}_{t-1}}\boldsymbol{\mu}_t \operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta})\left(\nabla_{\boldsymbol{\theta}}\boldsymbol{\mu}_t\right)^\top + \nabla_{\boldsymbol{\theta}}\boldsymbol{\mu}_t \operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta})^\top\left(\nabla_{\boldsymbol{x}_{t-1}}\boldsymbol{\mu}_t\right)^\top \\ & + \boldsymbol{\Sigma}_x + \Sigma_f(\tilde{\boldsymbol{\mu}}_t)\,.\end{aligned} \tag{A.2.1}$$

The covariance $\operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta})$ is given by

$$\operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta}) = \frac{\partial\boldsymbol{\mu}_{t-1}}{\partial\hat{\boldsymbol{\theta}}}\operatorname{cov}(\boldsymbol{x}_{t-2}, \boldsymbol{\theta}) + \frac{\partial\boldsymbol{\mu}_{t-1}}{\partial\hat{\boldsymbol{\theta}}}\boldsymbol{\Sigma}_\theta\,,$$

hence $\partial\operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta})/\partial\{\boldsymbol{\mu}_{t-1},\, \boldsymbol{\Sigma}_{t-1},\, \hat{\boldsymbol{u}}_{t-1}\} = \mathbf{0}$. The derivatives of the remaining terms on the right-hand side in Equation (A.2.1) are computed using identity 2 and 4 in Section A.1.

A.3 Discrete-Time Model with Δ-Transition

The latent state transition between consecutive time steps is described by a discrete-time model with Δ-transition (see Section 6.3.2). The derivative of the predicted latent state mean $\boldsymbol{\mu}_t$ at time step t with respect to the latent state mean $\boldsymbol{\mu}_{t-1}$ at time step $t-1$ is given by

$$\frac{\partial\boldsymbol{\mu}_t}{\partial\boldsymbol{\mu}_{t-1}} = \mathbf{I} + \left.\frac{\partial\mu_f(\boldsymbol{x}, \hat{\boldsymbol{u}}_{t-1}, \hat{\boldsymbol{\theta}})}{\partial\boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{\mu}_{t-1}} ,$$

and the derivatives with respect to the desired control input $\hat{\boldsymbol{u}}_{t-1}$ at time step $t-1$ and the model parameters $\hat{\boldsymbol{\theta}}$ are given by

$$\frac{\partial\boldsymbol{\mu}_t}{\partial\{\hat{\boldsymbol{u}}_{t-1}, \hat{\boldsymbol{\theta}}\}} = \left.\frac{\partial\mu_f(\boldsymbol{\mu}_{t-1}, \boldsymbol{u}, \boldsymbol{\theta})}{\partial\{\boldsymbol{u}, \boldsymbol{\theta}\}}\right|_{(\boldsymbol{u},\boldsymbol{\theta})=(\hat{\boldsymbol{u}}_{t-1},\hat{\boldsymbol{\theta}})} .$$

The Taylor approximation of $\boldsymbol{\mu}_t$ does not depend on $\boldsymbol{\Sigma}_{t-1}$, hence $\partial\boldsymbol{\mu}_t/\partial\tilde{\boldsymbol{\Sigma}}_{t-1} = \mathbf{0}$.

The expression for the predicted latent state covariance $\boldsymbol{\Sigma}_t$ in Equation (6.3.5) on page 121 can be expanded as

$$\begin{aligned}\boldsymbol{\Sigma}_t = \boldsymbol{\Sigma}_{t-1} &+ \boldsymbol{\Sigma}_{t-1} \left(\nabla_{\boldsymbol{x}_{t-1}} \boldsymbol{\mu}_t\right)^\top + \nabla_{\boldsymbol{x}_{t-1}} \boldsymbol{\mu}_t \boldsymbol{\Sigma}_{t-1} \\ &+ \nabla_{\boldsymbol{x}_{t-1}} \boldsymbol{\mu}_t \boldsymbol{\Sigma}_{t-1} \left(\nabla_{\boldsymbol{x}_{t-1}} \boldsymbol{\mu}_t\right)^\top + \nabla_{\boldsymbol{u}_{t-1}} \boldsymbol{\mu}_t \boldsymbol{\Sigma}_u \left(\nabla_{\boldsymbol{u}_{t-1}} \boldsymbol{\mu}_t\right)^\top + \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_t \boldsymbol{\Sigma}_\theta \left(\nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_t\right)^\top \\ &+ \nabla_{\boldsymbol{x}_{t-1}} \boldsymbol{\mu}_t \operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta}) \left(\nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_t\right)^\top + \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_t \operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta})^\top \left(\nabla_{\boldsymbol{x}_{t-1}} \boldsymbol{\mu}_t\right)^\top \\ &+ \boldsymbol{\Sigma}_x + \Sigma_f(\tilde{\boldsymbol{\mu}}_t)\,.\end{aligned} \tag{A.3.1}$$

The covariance $\operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta})$ is given by

$$\operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta}) = \operatorname{cov}(\boldsymbol{x}_{t-2}, \boldsymbol{\theta}) + \frac{\partial \boldsymbol{\mu}_{t-1}}{\partial \hat{\boldsymbol{\theta}}} \operatorname{cov}(\boldsymbol{x}_{t-2}, \boldsymbol{\theta}) + \frac{\partial \boldsymbol{\mu}_{t-1}}{\partial \hat{\boldsymbol{\theta}}} \boldsymbol{\Sigma}_\theta\,,$$

hence $\partial \operatorname{cov}(\boldsymbol{x}_{t-1}, \boldsymbol{\theta}) / \partial\{\boldsymbol{\mu}_{t-1}, \boldsymbol{\Sigma}_{t-1}, \hat{\boldsymbol{u}}_{t-1}\} = \mathbf{0}$. The derivatives of the remaining terms on the right-hand side in Equation (A.3.1) are computed using identity 1, 2, 3 and 4 in Section A.1.

A.4 Continuous-Time State Space Models

The latent state transition is described by a continuous-time model (see Section 6.3.3). For a dynamic variable $\boldsymbol{\xi} = \xi(\tau, \boldsymbol{p})$ parameterised by $\boldsymbol{p}$ and described by the differential equation

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}\tau} \xi(\tau, \boldsymbol{p}) = g(\xi(\tau, \boldsymbol{p}), \boldsymbol{p})\,, \\ \quad \xi(\tau_0, \boldsymbol{p}) = \boldsymbol{\xi}_0\,, \end{cases}$$

the partial derivative of $\boldsymbol{\xi}$ with respect to $\boldsymbol{p}$ is described by the differential equation [Dickinson and Gelinas, 1976; Rabitz et al., 1983]

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}\tau} \left[\dfrac{\partial \xi(\tau, \boldsymbol{p})}{\partial \boldsymbol{p}}\right] = \dfrac{\partial g(\xi(\tau, \boldsymbol{p}), \boldsymbol{p})}{\partial \boldsymbol{p}} + \dfrac{\partial g(\xi(\tau, \boldsymbol{p}), \boldsymbol{p})}{\partial \xi(\tau, \boldsymbol{p})} \dfrac{\partial \xi(\tau, \boldsymbol{p})}{\partial \boldsymbol{p}}\,, \\ \qquad \dfrac{\partial \xi(\tau_0, \boldsymbol{p})}{\partial \boldsymbol{p}} = \dfrac{\partial \boldsymbol{\xi}_0}{\partial \boldsymbol{p}}\,, \end{cases}$$

with some abuse of notation for the case of tensor products.

The latent state mean $\boldsymbol{\mu}_t$ and covariance $\boldsymbol{\Sigma}_t$ at time step t are found by extracting the corresponding elements from the concatenated mean $\tilde{\mu}(\tau_t)$ and covariance $\tilde{\Sigma}(\tau_t)$, found by solving the system of differential equations in Equation (6.3.7) on page 122. Similarly, the derivatives of $\boldsymbol{\mu}_t$ and $\boldsymbol{\Sigma}_t$ with respect to the latent state mean $\boldsymbol{\mu}_{t-1}$, latent state covariance $\boldsymbol{\Sigma}_{t-1}$ and the desired control input $\hat{\boldsymbol{u}}_{t-1}$ at time step $t-1$ are found by extracting the corresponding elements from the derivatives of $\tilde{\mu}(\tau_t)$ and $\tilde{\Sigma}(\tau_t)$ with respect to $\tilde{\boldsymbol{\mu}}_{t-1}$ and $\tilde{\boldsymbol{\Sigma}}_{t-1}$.

The derivative of $\tilde{\mu}(\tau_t)$ with respect to $\tilde{\boldsymbol{\mu}}_{t-1}$ is found by solving

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}\tau}\left[\dfrac{\partial\tilde{\mu}(\tau)}{\partial\tilde{\boldsymbol{\mu}}_{t-1}}\right] = \dfrac{\partial\tilde{\mu}_f(\tau)}{\partial\tilde{\boldsymbol{\mu}}_{t-1}} + \dfrac{\partial\tilde{\mu}_f(\tau)}{\partial\tilde{\mu}(\tau)}\,\dfrac{\partial\tilde{\mu}(\tau)}{\partial\tilde{\boldsymbol{\mu}}_{t-1}}\,, \\ \dfrac{\partial\tilde{\mu}(\tau_{t-1})}{\partial\tilde{\boldsymbol{\mu}}_{t-1}} = \left[\mathbf{I},\, \mathbf{0},\, \dfrac{\partial\mu(\tau_{t-1})}{\partial\hat{\boldsymbol{\theta}}}\right]\,, \\ \dfrac{\partial\mu(\tau_0)}{\partial\hat{\boldsymbol{\theta}}} \equiv \mathbf{0}\,, \end{cases} \tag{A.4.1}$$

in parallel with Equation (6.3.7) on page 122. Note that

$$\frac{\partial\tilde{\mu}_f(\tilde{\mu}(\tau))}{\partial\tilde{\boldsymbol{\mu}}_{t-1}} = \begin{bmatrix} \mathbf{0} & \dfrac{\partial\mu_f(\mu(\tau), \hat{\boldsymbol{u}}_{t-1}, \hat{\boldsymbol{\theta}})}{\partial\hat{\boldsymbol{u}}_{t-1}} & \dfrac{\partial\mu_f(\mu(\tau), \hat{\boldsymbol{u}}_{t-1}, \hat{\boldsymbol{\theta}})}{\partial\hat{\boldsymbol{\theta}}} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

The mean $\tilde{\mu}(\tau)$ does not depend on $\tilde{\Sigma}(\tau)$, hence $\partial\tilde{\boldsymbol{\mu}}_t/\partial\boldsymbol{\Sigma}_{t-1} = 0$.

Let $\tilde{\Sigma}_f$ denote the covariance transition function in Equation (6.3.7) on page 122, given by

$$\tilde{\Sigma}_f(\tau) = \nabla_{\tilde{\mu}(\tau)}\tilde{\mu}_f\tilde{\Sigma}(\tau) + \tilde{\Sigma}(\tau)\left(\nabla_{\tilde{\mu}(\tau)}\tilde{\mu}_f\right)^{\top} + \mathrm{diag}(\Sigma_f(\tilde{\mu}(\tau)) + \boldsymbol{\Sigma}_x,\, \mathbf{0},\, \mathbf{0})\,,$$

and let $\nu \in \mathbb{R}$ be an element in $\tilde{\boldsymbol{\mu}}_{t-1}$ or $\tilde{\boldsymbol{\Sigma}}_{t-1}$. The derivative of $\tilde{\boldsymbol{\Sigma}}_t$ with respect to ν is found by solving

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}\tau}\left[\dfrac{\partial\tilde{\Sigma}(\tau)}{\partial\nu}\right] = \dfrac{\partial\tilde{\Sigma}_f(\tau)}{\partial\nu} + \displaystyle\sum_{i=1}^{D}\dfrac{\partial\tilde{\Sigma}_f(\tau)}{\partial\left[\tilde{\mu}(\tau)\right]_i}\,\dfrac{\partial\left[\tilde{\mu}(\tau)\right]_i}{\partial\nu} + \displaystyle\sum_{i=1}^{D}\sum_{j=1}^{D}\dfrac{\partial\tilde{\Sigma}_f(\tau)}{\partial[\tilde{\Sigma}(\tau)]_{ij}}\,\dfrac{\partial[\tilde{\Sigma}(\tau)]_{ij}}{\partial\nu}\,, \\ \dfrac{\partial\tilde{\Sigma}(\tau_{t-1})}{\partial\{\boldsymbol{\mu}_{t-1},\, \hat{\boldsymbol{u}}_{t-1}\}} = \mathbf{0}\,, \\ \dfrac{\partial\tilde{\Sigma}(\tau_0)}{\partial\hat{\boldsymbol{\theta}}} \equiv \mathbf{0}\,, \\ \left[\dfrac{\partial\tilde{\Sigma}(\tau_{t-1})}{\partial\tilde{\boldsymbol{\Sigma}}_{t-1}}\right]_{ijmn} = \begin{cases} 1\,, & \text{if } i = m \wedge j = n\,, \\ 0\,, & \text{otherwise}\,, \end{cases} \end{cases}$$

in parallel with the system of differential equations in Equation (6.3.7) on page 122 and Equation (A.4.1), with $D = D_x + D_u + D_\theta$. The terms on the right-hand side are computed using identity 1, 3 and 5 in Section A.1 as well as the chain rule.