Óbuda University
Doctoral School of Applied Informatics and Applied Mathematics
Budapest, Hungary

ESTIMATION AND ROBUST CONTROL OF NONLINEAR DIABETES MODEL

NEMLINEÁRIS DIABÉTESZ MODELL BECSLÉSE ÉS ROBUSZTUS SZABÁLYOZÁSA

Thesis by

Péter Szalay

In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy

Supervisor:

Prof. Dr. habil. Levente Kovács
Óbuda University
Budapest, Hungary

Budapest, 2023

Declaration

Undersigned, Péter Szalay, hereby state that this Ph. D. thesis is my own work wherein I have used only the sources listed in the Bibliography. All parts taken from other works, either in a word for word citation or rewritten keeping the original contents, have been unambiguously marked by a reference to the source.

Nyilatkozat

Alulírott Szalay Péter kijelentem, hogy ezt a doktori értekezést magam készítettem és abban csak a megadott forrásokat használtam fel. Minden olyan részt, amelyet szó szerint, vagy azonos tartalomban, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Budapest, 2023.06.20.

.....
Szalay Péter

Az értekezésről készült bírálatok és a jegyzőkönyv a későbbiekben az Óbudai Egyetem Élettani Szabályozások Kutatóközpontjában elérhetőek.

The reviews of this Ph. D. thesis and the record of defense will be available later in the Physiological Controls Research Center of the Óbuda University.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my supervisors, Prof. Levente Kovács and the late Prof. Zoltán Benyó, for their invaluable advice, continuous support, and patience during my Ph. D. study. Furthermore, I am incredibly thankful to Prof. Levente Kovács for enduring the long hiatus in my studies. Likewise, I am deeply grateful to Dr. Dániel András Drexler for his immense support and guidance, especially over the last few years.

I would like to thank my teachers, Prof. Béla Lantos, Prof. József Bokor, Dr. Bálint Kiss, and Dr. István Harmati, who inspired me to pursue studies in the field of robust and nonlinear control.

I would also like to thank my current and former colleagues at Biomedical Engineering Group and Physiological Controls Research Center for a cherished time spent together in the lab and social settings. Moreover, I would like to offer my special thanks to Dr. Tamás Ferenci and József Homlok for the inspiring discussions over the years.

I would like to express my sincere gratitude to Shigeru Sato-san, for his guidance during my stay in Japan and for making me realize the importance of Kalman filters. I would also like to extend my sincere thanks to Dr. Mahsa Nemati for inspiring me to start Ph. D. studies. I would like to extend my gratitude to all my friends and colleagues, especially Imre Gaál, who contributed by proofreading the dissertation.

Finally, I would like to thank my entire family for their unwavering support and belief in me. Without their tremendous understanding and encouragement, it would be impossible for me to complete my study. I would like to sincerely thank my partner, Anne-Mari Ahola-Olli DVM, and her family for their support and enduring patience.

I dedicate this work to my dear mother and grandmother in loving memory.

KIVONAT

Az 1-es típusú diabétesz a szervezet cukor-háztartásának krónikus zavara, amelynek előfordulása világszerte növekvő tendenciát mutat. Ez egy jelenleg még gyógyíthatatlan, de kezelhető betegség, amelynek a szövődményei jelentősen befolyásolják élettartamot és életminőségét. A mesterséges hasnyálmirigy terápia kidolgozására irányuló fejlesztések célja, hogy a hagyományos manuális inzulin terápiánál hatékonyabb vércukorszint-szabályozást tegyenek lehetővé. Az utóbbi évek eredményes klinikai vizsgálatai és a félautomata inzulin pumpák sikeres alkalmazásának ellenére a mesterséges vércukorszint-szabályozás számos gyakorlati problémát vet fel.

A disszertáció a mesterséges hasnyálmirigy funkciói közül az alábbi háromra fókuszál: közvetlenül nem mérhető mennyiségek becslése, a vércukorszint hosszú távú előrejelzése, valamint a szabályozó algoritmus. Az alkalmazott módszereknek figyelembe kell vennie a vércukorszint szabályozás kihívásait: a humán anyagcsere-folyamatok összetett és nemlineáris dinamikáját, a korlátozott mérési lehetőségeket, a vezérlőjel aszimmetriáját, valamint az ételbevitelre vonatkozó információk hiányát.

Az elmúlt évek során a hangsúly a biztonság és megbízhatóság irányába tolódott el a mesterséges hasnyálmirigy kutatásában. Mivel az eszköz robusztusságának növelése elengedhetetlen a biztonságos működés biztosításához, ezért a disszertáció elsősorban ebből a szempontból vizsgálja a becslés, predikció és szabályozás kérdéskörét.

A bemutatásra kerülő módszerek mindegyike modellalapú. A szabályozó a nemlineáris modellek egy speciális fajtáját, lineáris változó paraméterű rendszert használ a vércukor háztartás modellezéséhez. A szabályozótervezés a modell súlyfüggvényekkel kibővített változatát alkalmazza. Ezek a súlyfüggvények többek között a rendszer bizonyta-

lanságát és a vezérlőjel korlátait képviselik. A végeredmény egy olyan nemlineáris állapot-visszacsatoló szabályozó, amely H-végtelen és H2 normákon alapuló robusztus vércukorszint szabályozást valósít meg.

Az állapot-visszacsatoláshoz szükséges állapotvektor becslését szigma pont szűrőn alapuló állapotmegfigyelő biztosítja. A szigma pont szűrők hatékonyak nemlineáris, sztochasztikus zajokkal és zavarásokkal terhelt rendszerek állapotainak becslésére. Továbbá a disszertációban bemutatott szigma pont szűrők kifejezetten úgy lettek módosítva, hogy figyelembe vegyék a vércukorszint háztartás modelljének bizonytalanságát, és az állapotváltozók nemnegatív voltát.

A bemutatott állapotmegfigyelő használható predikcióra is. Azonban ahhoz, hogy a predikció zárt szabályozási körben is hatékony maradjon, szükséges az állapotbecslés hibájának figyelembe vétele. Ez biztosítható abban az esetben, ha a prediktort együttesen alkalmazzuk egy olyan sztochasztikus becslővel, amely az állapotbecslés hibájából származó eltérésekről szolgáltat információt. A prediktor többek között használható bejelentett ételbevitel validálására és potenciálisan veszélyes vércukorszintek előrejelzésére.

A bemutatott szabályozó és állapotbecslő algoritmusok kiértékelése szimulációk segítségével történt. Az eredmények alapján az élettani változók nemnegatív jellegének figyelembevétele javíthatja a szigma pont szűrők pontosságát bizonyos feltételek teljesülése mellett. Ezenfelül megállapítást nyert, hogy a szigma pontok számának növelése egy adott ponton túl nem javítja érdemben a becslés minőségét. A szabályozó algoritmus egy a szakirodalomban rendszeresített vizsgálati módszer alapján kielégítő módon tudja szabályozni a vércukorszintet be nem jelentett ételbevitelek mellett.

A szimulációs eredmények alapján valószínűsíthető, hogy a bemutatott módszerek alkalmasak a mesterséges hasnyálmirigyben való használatra. Azonban figyelembe véve a módszerek korlátait, valamint a könnyű implementálhatóságot és kis számítási igényt, használatuk elsősorban támogató és diagnosztikai funkciókra javasolt.

ABSTRACT

Type 1 diabetes mellitus is a chronic condition of glucose metabolism, which has an increasing prevalence worldwide. It is currently incurable but treatable, with severe potential long-term outcomes. Artificial pancreas is an emerging automated treatment with the aim to manage glucose levels more efficiently than manual insulin injections. However, despite the promising clinical trials and the success of commercially available semi-automated solutions, glucose control has numerous practical challenges.

This dissertation focuses on three key aspects of Artificial Pancreas: estimation of unmeasured signals, long-term prediction of glucose levels, and control algorithm. The goal is to address the following challenges: the complex, nonlinear and uncertain dynamics of the impaired human metabolism; the limited measurement capabilities; the asymmetric nature of the control signal, which can only decrease but not increase the glucose concentration; and finally, unannounced meal intakes. In recent years, the focus of the corresponding research shifted toward safety. Robustness is one of the enablers of safe and reliable operation. Hence, this work puts robustness at the forefront, approaching all three aspects mentioned above from this perspective.

All presented methods are model-based. The control algorithm uses a linear parameter varying model to represent the nonlinear behavior. The model is extended with weighting functions to impose additional constraints and capture modeling uncertainties. The controller synthesis ensures robust stability and optimizes nominal performance for the closed-loop system. The resulting control algorithm is a robust quasi linear parameter varying controller optimized for particularities of glucose control.

A sigma point filter aids the controller by estimating both state vari-

ables and glucose flux from meal ingestion. Sigma point filters are effective on nonlinear models. Furthermore, they can handle stochastic disturbances and measurement noise. The proposed state observer algorithm also considers model uncertainties and that the state variables of a physiological system are nonnegative.

The presented state observer can also serve as a predictor. Combined with a modified state observer, it can provide long-term prediction in a closed-loop setting, considering the effect of estimation error. Furthermore, if a meal intake model is available, the predictor can validate meal announcements.

The controller and state observer have been evaluated via simulations. The results showed that taking the nonnegativity of the system into consideration increases state estimation accuracy. However, increasing the number of sigma points beyond a particular value provides little benefit. Control variability grid analysis of the control algorithm shows satisfactory disturbance rejection without hypoglycemic episodes.

The presented algorithms could contribute to an artificial pancreas system as primary components but more likely in a supporting role.

CONTENTS

NOMENCLATURE

$\mathbf{P}^{(xx)}$	: Estimated covariance matrix of variable vector x.
$\mathbf{P}^{(xy)}$	: Estimated cross-covariance matrix of variable vectors x and y.
$\Sigma, \Sigma_{(e)}$	: Covariance matrix of estimation error.
$\Sigma_{(p)}$	: Covariance matrix of prediction error.
ξ_i	: ith sigma point of a sigma point set.
φ_i	: Vector defining the spread of the ith sigma point.
ω_i	: Weight associated with the ith sigma point.
$\mathcal{X}$	: Sigma point set in Chapter 3 and Lyapunov function in Chapter 4.
$\mathbf{M}^{\frac{1}{2}}$	: Cholesky factor of matrix M.
ρ_i	: ith scheduling variable.
ρ_c	: Convex polytope spanned by a set of ρ vectors.
Δ	: Unstructured uncertainty.
$\|\mathrm{x}\|_p$	: p-norm of vector x.
$\|\mathbf{M}\|_p$	: induced p-norm of matrix M.
ϑ	: Scaling factor associated with output multiplicative uncertainty.
Θ	: Diagonal matrix for which the non-zero values are either 1 or ϑ.
ψ	: Scaling factor associated with performance output.
Ψ	: Diagonal matrix for which the non-zero values are either 1 or ψ.
$\mathcal{T}$	: Linear mapping/transformation.
$\mathcal{T}(.)$	: Function representing nonlinear transformation.
$\overline{x}$	: Average or weighted average of a set of x variables.
$\overline{\sigma}(\mathbf{M})$	: Largest singular value of the matrix M
$\underline{\sigma}(\mathbf{M})$	: Smallest singular value of the matrix M

LIST OF ABBREVIATIONS

Abbreviation	Definition
AP	Artificial pancreas
CDKF	Central difference Kalman filter
CGM	Continuous glucose monitoring
CGMS	Continuous glucose monitoring sensor
CHO	Carbohydrate
CKF	Cubature Kalman filter
CSII	Continuous subcutaneous insulin infusion
DM	Diabetes mellitus
EKF	Extended Kalman filter
FDA	United States Food and Drug Administration
GHQF	Gauss-Hermite quadrature filter
GUI	Graphical user interface
KF	Kalman filter
LPV	Linear parameter-varying
LMI	Linear matrix inequality
MPC	Model predictive control
NP	Nominal performance
OGTT	Oral glucose tolerance test
PID	Proportional integral derivative
qLPV	Quasi linear parameter-varying
RP	Robust performance
RS	Robust stability
RSPF	Reduced sigma point filter
SGQF	Sparse grid quadrature filter
SMBG	Self-monitoring of blood glucose
SPF	Sigma point filter
T1DM	Type 1 diabetes mellitus
T2DM	Type 2 diabetes mellitus
UKF	Unscented Kalman filter

1

INTRODUCTION

1.1 Diabetes Mellitus

Diabetes Mellitus (DM) is a collective term referring to several chronic metabolic diseases. They are characterized by elevated glucose levels (hyperglycemia) over a prolonged period of time. The World Health Organization reported that the number of people with diabetes increased from 108 million in 1980 to 422 million in 2014 [1]. In 2019, diabetes was the direct cause of 1.5 million deaths [2].

The three main types of diabetes mellitus are the following [3]:

- *Type 1 Diabetes Mellitus* (T1DM) is an autoimmune disease in which the pancreas cannot produce insulin due to the loss of β-islet cells. Insulin is a peptide hormone that plays a crucial role in glucose utilization, decreasing plasma glucose concentration. Currently, neither its cause nor means of prevention are known, but genetic susceptibility and environmental factors are suspected [4].

- *Type 2 Diabetes Mellitus* (T2DM) is the most common form of diabetes. More than 95% of all diabetes cases belong to type 2 diabetes. Insulin resistance, which is the inefficient use of the insulin produced in the body, characterizes this condition. This type of diabetes is primarily the result of excess body weight and physical inactivity. Symptoms may be similar to those of T1DM but are often less marked.

- *Gestational diabetes* occurs during pregnancy, with blood glucose values above normal but below those diagnostic of diabetes. The patients have no previous history of diabetes, and glucose values usually return to normal soon after delivery. However, women

with gestational diabetes are at an increased risk of complications during pregnancy and delivery.

The most common symptoms include excessive excretion of urine, increased appetite and thirst, weight loss, vision changes, and fatigue. If left untreated, it can lead to severe acute and long-term complications. Acute complications include diabetic ketoacidosis and hyperglycemic hyperosmolar state, both of which can be life-threatening in severe cases. Long-term complications include:

- Cardiovascular diseases: Adults with diabetes have a two- to threefold increased risk of heart attacks and strokes.
- Nephropathy: diabetes is among the leading causes of kidney failure.
- Neuropathy: various forms of nerve damage. Combined with the reduced blood flow, it increases the chance of foot ulcers.
- Retinopathy: the long-term damage to the retina's small blood vessels can lead to blindness [Énzsöly et al., 2014].
- People with diabetes are more likely to have poor outcomes for several infectious disease.

Despite its smaller prevalence compared to T2DM, this work focuses exclusively on Type 1 Diabetes Mellitus.

The treatment of T1DM mainly consists of regular insulin injections. All insulin delivery methods are invasive and administered via syringe or pump. Different types of insulin analogs are available in clinical practice, ranging from fast to long-acting versions. However, insulin treatment has its challenges. An insufficient amount does not decrease glucose levels enough to avoid long-term complications while administering too much can lead to hypoglycemia. Hypoglycemia is defined as blood glucose concentration below 3.9 mmol/L or 70 mg/dL. Severe cases can result in seizure, coma, or death.

Recent decades saw extensive research in the automation of insulin delivery, commonly referred to as Artificial Pancreas (AP) [5, 6, 7] [KSF$^+$11, KSF$^+$12, KSF$^+$13]. Artificial Pancreas can potentially lessen both the severity and time spent in hyperglycemia while completely avoiding dangerous hypoglycemic episodes. Keeping a T1DM patient consistently in the normal range of blood glucose concentration decreases the chance of developing acute and long-term complications of diabetes.

1.2 Artificial Pancreas

From the perspective of hardware, Artificial Pancreas is a portable medical device with two main components:

1. Continuous Glucose Monitoring (CGM) system provides up-to-date readings of the subcutaneous glucose concentration of the patient.

2. Continuous subcutaneous insulin infusion (CSII) pump.

The most common AP systems are insulin-only types, which achieve a target glucose level by automatically increasing or decreasing the amount of insulin infused based on the CGM values. Their functionality is limited to decreasing glucose levels, with no means to increase it. A bi-hormonal control system contains an additional pump delivering glucagon to increase blood glucose levels if necessary. Therefore, the bi-hormonal system mimics the functionality of a healthy pancreas more closely. However, bi-hormonal systems have practical limitations [8]. The unstable nature of the peptide limits the pharmaceutical use of glucagon. Furthermore, glucagon has undesirable side effects, such as nausea and vomiting.

Artificial Pancreas can be combined with existing wearable medical devices to get additional information about the patient, especially regarding physical activity. Physical activity has high significance for insulin-only systems since it can decrease glucose levels, which they have no means to compensate for.

From the perspective of control theory, Artificial Pancreas is a fully automated regulation of blood glucose levels. Currently, only hybrid closed-loop systems are available in clinical practice, with the U.S. Food and Drug Administration (FDA) approving the first device in 2016 [9]. Although a hybrid system delivers basal insulin via closed-loop control, the user still needs to inject insulin manually after meal intakes. However, there are several fully automated systems undergoing clinical trials [10, 11, 12, 13].

From the perspective of software, AP consists of several key components, some of which are listed below:

- Control algorithm: use the data acquired from the sensors of AP to deliver insulin via CSII whenever needed.

- State observer: it can reduce sensor noise, estimate signals necessary for the control algorithm, or even enable state feedback control.

- Parameter identification: necessary for adaptive control strategies.

- Event and fault detection: the human metabolism is affected by various factors, some of which occur in an event-like manner, such as meal intake or physical activity. Faults can occur in any of the hardware or software of the device. Detecting them in time is of paramount importance to ensure safe operation.
- Prediction: while event detection concerns itself with the present, prediction allows the device to make decisions based on potential future occurrences.

The list is not exhaustive, and items are tightly connected. For example, some state observer algorithms can be used for parameter identification, while events can be detected by comparing past predictions with current readings. Furthermore, future AP devices might use a fusion of multiple different instances in each category for redundancy, safety, and robustness. For example, multiple parameter identification algorithms with different limitations or a specialized but sensitive control algorithm backed up by a less efficient but robust counterpart.

1.2.1 Control Algorithms

Maintaining normal glucose concentration (normoglycemia) is a challenging control problem.

- The human metabolism is a highly complex dynamic system. The most popular models that can capture the glucose-insulin interaction in the body of a T1DM patient to a satisfying degree are severely nonlinear [14]. Furthermore, the dynamics tend to change significantly over time.
- The dynamics of the human metabolism concerning glucose are slower when the glucose levels are lower than when they are high [15].
- The metabolic glucose process may be affected by various factors. These factors are usually difficult to measure, detect or even quantify. Notable examples are physical activity, dietary changes, stress, infections and medical conditions, menstrual cycle, or even circadian rhythm [16].
- Hypoglycemia is a more severe acute complication than hyperglycemia, even though reducing the latter in severity and frequency is the main goal of AP.
- Even rapid-acting insulin - if injected subcutaneously - has a significantly slower effect on the plasma glucose concentration than meal intake or physical activity [17, 18].

- The commercially available continuous glucose monitoring (CGM) sensors have significant noise and drift [19, 20, 21].
- An insulin-only device has no means to elevate glucose levels. The control signal should always be nonnegative.

A wide range of control algorithms is proposed in the literature to overcome these challenges [22, 23].

Proportional Integral Derivative (PID) control is one of the most well-known control methods; hence, it is also applied for blood glucose control. Barnes and Jones [24] explored the potential of intraperitoneal insulin infusion by comparing a PID controller with a CSII-based AP that uses Model Predictive Control (MPC). Alshalalfah et al. [25] used a combination of two PID controllers to handle varying meal conditions. Finally, Calupiña et al. [26] showed how a nonlinear PID controller could outperform a dynamic sliding mode controller in some instances.

Despite that, there are promising attempts at sliding mode control as well. Beneyto et al. [27] combined sliding mode control with a carbohydrate recommender system to avoid hypoglycemia due to unannounced physical activity. Leyva et al. [28] compared positive sliding mode control with control Lyapunov function theory to ensure nonnegative insulin signal.

There are attempts to address the nonlinearity of the models directly [Kovács et al., 2012, Kovács et al., 2011] [29], using nonlinear state feedback. However, physiological systems are highly complex, with numerous unknowns that are difficult to model, which data-driven and soft computing methods can handle efficiently. Furthermore, they are more commonly used for prediction and fault detection. For example, Zhu et al. [30] used Deep Reinforcement Learning for glucose control, Peiró et al. [31] improved existing hybrid control schemes with machine learning, and Mohammadzadeh et al. [32] combined fractional-order modeling with general type-2 fuzzy predictive control.

Adaptive controllers are good candidates to address the time-variant dynamics of a T1DM patient [33]. Shi et al. [34] used a multivariate Bayesian optimization-assisted parameter adaptation framework, while Kovács et al. [35] proposed an adaptive controller using a novel robust fixed point transformation.

However, one of the most widely accepted approaches is model predictive control (MPC). Various MPC-based AP prototypes have been proposed over the years [36, 37, 38], some of which are undergoing clinical trials [39, 13]. Some recent examples combine MPC with soft computing methods such as feed-forward neural networks and fuzzy logic controllers [40], while others include circadian insulin sensitivity scheme [41], employing several layers of adaptivity [42], or aiding MPC with meal estimation [43].

MPC usually relies heavily on the accuracy of the model, which is rarely available in clinical practice. Furthermore, the more complex the model is, the more computational power MPC algorithms require. Hence, increasing the robustness of the MPC controllers gained popularity [44, 45, 46]. Alternatively, some proposals make robustness the core feature of the control algorithm [47] [48]. Linear Parameter Varying (LPV) systems are a class of nonlinear models that are easy to combine with robust control methods. Although signals enabling LPV control are usually not measured in clinical practice, quasi LPV (qLPV) methods relying on estimated signals are applicable. Robust qLPV control addresses both the nonlinearity and uncertainty of the impaired metabolism of the T1DM patient [49, 50] [Kovács et al., 2011b, SEK14, SDKew].

One of the contributions of this dissertation is a control algorithm that addresses the above-listed challenges of blood glucose control using a robust control technique based on an uncertain qLPV model while satisfying constraints on the control signal under unannounced meal intakes.

1.2.2 Estimation

AP control algorithms, especially model-based techniques, can significantly benefit from state and disturbance estimation. Furthermore, they can aid in fault detection and parameter tuning as well. Due to the stochastic nature of the disturbances present, Kalman Filter (KF) is a popular choice [51]. KF is a linear model-based method. However, there are several ways to address the nonlinearity of the T1DM model. One of the earliest approaches is the Extended Kalman Filter (EKF). For example, Fushimi et al. [52] used EKF to detect unannounced meal intakes, while Mahmoudi et al. [53] used it for fault detection in CGMS. Finally, in the work of Boiroux et al. [54], EKF supports parameter estimation using Maximum Likelihood method.

Despite its popularity in nonlinear state estimation, EKF relies on linearization, which has limitations in severely nonlinear models. Sigma point filters can provide a more practical alternative [SMM+14] at the cost of computational power. Unscented Kalman Filter (UKF) is a sigma point filter frequently used for various roles in AP [55, 56, 57].

Combining the Kalman filter with Linear Parameter Varying modeling can address the nonlinearity of the model without the need for linearization and excessive computation power [58].

Despite the prevalence of KF, there have been other types of state observers applied for AP. For example, Alam et al. [59] aided a sliding mode control algorithm with a gain-scheduled Luenberger observer. Chen et al. [60] detected and estimated unannounced meal intake with

Moving Horizon estimation, while Sanz et al. [61] used an extended state observer for the same task. Sliding mode observers are also good contenders for KF in nonlinear state and disturbance estimation [58, 62].

Another contribution of this dissertation is exploring sigma point filtering for AP use, assessing the effectiveness of some of the popular techniques, and providing a novel algorithm for effective state and disturbance estimation.

1.2.3 Prediction

Sufficiently long-term prediction of glucose concentration is helpful for an AP in various ways:

- Prediction is the core concept behind MPC.
- An overseer logic can assess the expected quality of the blood glucose control and tune controller parameters accordingly.
- It is possible to intervene if the prediction indicates a potentially hazardous hypoglycemic episode.
- A significant deviation between predicted and measured behavior can indicate fault or an unannounced event (e.g., meal intake).
- Multiple predictions based on different hypotheses can identify unannounced events or validate announced ones.

Some of the estimation algorithms referenced earlier can be re-purposed for prediction [SBK16] [63]. However, the more commonly applied techniques come from Machine Learning. Kushner et al. [64] used a non-deterministic data-driven model to predict up to 5 hours how a patient would perform under a PID-based closed-loop system. Meneghetti et al. [65] proposed an unsupervised model-free approach to detect insulin pump malfunction, while Güemes et al. [66] and Eigner et al. [67] used algorithms based on Supervised Learning to predict overnight glycaemic control quality and to support decision-making in conservative diabetes therapy, respectively.

There are numerous examples of neural network-based prediction algorithms. Some recent examples come from Aliberti et al. [68], Li et al. [69], and Zhu et al. [70].

The third contribution of this dissertation is the extension of the previously presented state estimation algorithm into a predictor that can support meal announcement validation and detection of unannounced meal intakes.

1.3 Motivation

The motivation behind the presented theses is to address some challenges of automated blood glucose regulation with AP. The focus is on insulin-only setup and only on the software level. Within the numerous features an AP can have, this work addresses three key areas: estimation, prediction, and control. It is possible to contribute to these features even if one has no access to medical data due to the existence of validated models and simulation environments.

The first goal is to provide a control algorithm that has the following properties:

- A model-based approach that is adaptable to the most popular T1DM models. It shall address the nonlinearity of these models and the difference in dynamics between control signal and disturbances.
- The algorithm shall be able to deal with glucose increase due to meal intake, as it is the most impactful disturbance from blood glucose control perspective.
- There should be no reliance on meal intake announcements.
- The controller shall be robust against the intra-patient variability and uncertainty of the human metabolism.
- The controlled system shall completely avoid hypoglycemia while reducing time in hyperglycemia as much as possible.

Although asymptotic tracking via nonlinear state feedback has been extensively examined for this role [Szalay et al., 2012, Kovács et al., 2011, Szalay and Kovács, 2012, Kovács et al., 2012, Kovács et al., 2011b] [Kovács et al., 2011a], the corresponding research has been discontinued in favor of a robust qLPV control scheme. While asymptotic tracking is a powerful method for nonlinear systems, it is less suited to deal with uncertainties or satisfy inequality-type constraints. Furthermore, sensitivity to disturbances and parameter or state estimation inaccuracies can be difficult to analyze.

The second goal is to provide a state observer that can support the control algorithm with accurate estimations derived from the readings of a single continuous glucose sensor affected by measurement noise. Additionally, the proposed state observer should be extendable into a predictor. Sigma point filters are a good choice since they have good estimation capabilities for severely nonlinear models, can deal with stochastic noises and disturbances, and strike a good balance between accuracy and required computation power.

The third goal is to validate the proposed control and state observer methods via simulations.

1.4 Outline of the Thesis

Chapter 2 briefly introduces the T1DM model used throughout this work. Furthermore, it presents the modeling methods used to capture the uncertainty of the chosen T1DM model.

Chapter 3 presents sigma point filter algorithms and updates toward more accurate state estimation in an AP setting. Moreover, state estimation in the presence of modeling uncertainties is discussed. Finally, the chapter concludes with the means to use sigma point filters for the long-term prediction of glucose concentration. Chapter 4 consists of two main parts. The first section proposes an LPV representation of a commonly used T1DM model, while the second half of the chapter focuses on robust control methodologies. Simulation results are presented in Chapter 5. The chapter starts with state observer estimation, followed by a control variability grid analysis of control algorithms.

The new results are summarized in three Thesis groups in Chapter 6. Finally, Chapter 7 concludes the dissertation.

2

MODELING THE METABOLISM OF A TYPE 1 DIABETES PATIENT

A wide range of mathematical models has been introduced over the last four decades for capturing the metabolism of a T1DM patient. Bergman et al. published the first of these models [71], focusing on the core processes with two nonlinear continuous differential equations:

$$\begin{aligned} \dot{G}(t) &= (p_1 - X(t))G(t) + p_4 \\ \dot{X}(t) &= p_2 X(t) + p_3 I(t) \end{aligned} \tag{2.1}$$

The two compartments in this model are $G(t)$ plasma glucose concentration, and $X(t)$ is proportional to insulin in the remote compartment. $I(t)$ represents injected insulin. The model parameters $p_1 - p_4$ are derived from various transfer and disappearance rates. Please note that (2.1) preserves the original notation presented in [71]. Hence, parameters p_1 and p_2 have negative values, which can be misleading since parameters in all other models presented in this chapter are strictly positive.

Further research led to the extension to a third-order model [72], which inspired the models most widely used to develop automated and semi-automated treatments for T1DM patients in the intensive care unit (ICU). Chase et al. at the University of Cambridge developed a series of models with three compartments [73, 74, 75], where the additional state variable $Q(t)$ represents insulin bounded to interstitial sites. The model is defined by the differential equations (2.2). The model is further extended with parameters representing other physiological

processes, such as insulin losses to the liver and kidneys. Furthermore, their nonlinearity is not limited to the product of state variables as in (2.1), but also employs Michaelis-Menten kinetics for saturation dynamics, resulting in

$$\begin{aligned}\dot{G}(t) &= -p_G G(t) - S_I(t)(G(t)+G_E)\frac{Q(t)}{1+\alpha_G Q(t)} + P(t)\\ \dot{Q}(t) &= k(I(t)-Q(t))\\ \dot{I}(t) &= -\frac{nI(t)}{1+\alpha_I I(t)} + \frac{u_{ex}(t)}{V}\end{aligned} \tag{2.2}$$

On the other end of the complexity spectrum lies the T1DM model presented by Sorensen et al. [76], which has up to 19 state variables. Besides the potential accuracy and precision, the state variables themselves are more closely linked to actual compartments in the human body than other models presented in this chapter. However, the large number of parameters makes fitting the model to an actual T1DM patient challenging. Hence, using the Sorensen model for either simulation or controller design is impractical.

The following is a non-exhaustive list of other T1DM models:

- Extension of the third order Bergman model with fractional derivative [77];
- Capturing the glucose-insulin interaction with delay differential equations [78, 79];
- Including hormones other than insulin, such as glucagon, epinephrine, and incretins [80, 81];
- Representing the intestinal glucose absorption with partial differential equations [82];

2.1 T1DM models used for Artificial Pancreas development

The models used for AP development need to strike a delicate balance between several different aspects:

- The model shall capture the core dynamics of the impaired metabolism of a T1DM patient.
- State variables shall at least loosely correlate to actual physiological quantities.

- The fitting of the model parameters to an actual T1DM patient shall be feasible and a numerically well-conditioned problem. The parameter fitting should satisfy accuracy and precision expectations using the measurement data available in clinical practice.

Magni et al. [18] proposed a model consisting of 13 differential equations. Three of them represent the intestinal glucose absorption model first published in [83]. Even though the complete model is less detailed than the Sorensen model, it is more sophisticated in capturing glucose absorption from meal intake, which was not a primary focus in the latter. The Magni model is the core of a simulation tool developed by researchers at the University of Virginia, which is the first approved by the U.S. Food and Drug Administration (FDA) as an alternative to animal testing of Type 1 diabetes control strategies [84]. Section A.1.1 presents details of the model.

2.1.1 Cambridge model

Hovorka et al. introduced [17] and later updated [85] an 11th-order model, including glucose absorption from meal intake. Similarly to the Magni model, it is part of a simulator environment developed in the Cambridge University Metabolic Research Laboratories for in silico testing and the development of automated diabetes treatment algorithms. However, this simulator is yet to achieve approval from the FDA. The state variables of the model are as follows:

- $C(t)$ [mmol/L] glucose concentration in the subcutaneous tissue;
- $Q_1(t)$ [mmol] mass of glucose in accessible compartments;
- $Q_2(t)$ [mmol] mass of glucose in non-accessible compartments;
- $x_1(t)$ [1/min] remote effects of insulin on glucose distribution;
- $x_2(t)$ [1/min] remote effects of insulin on glucose disposal;
- $x_3(t)$ [-] remote effects of insulin on endogenous glucose production;
- $I(t)$ [mU/L] insulin concentration in plasma;
- $S_1(t)$ [mU] insulin mass in the accessible compartments;
- $S_2(t)$ [mU] insulin mass in the non-accessible compartments;
- $G_1(t)$ [mmol/kg] mass of ingested glucose in the stomach;
- $G_2(t)$ [mmol/kg] mass of ingested glucose in the gut.

The following differential equations describe the Cambridge model itself:

$$\begin{aligned}
\dot{C}(t) = & -k_{a,int}C(t) + \frac{k_{a,int}}{V_G}Q_1(t) \\
\dot{Q}_1(t) = & -\left(\frac{F_{01}}{Q_1(t)+V_G} + x_1(t)\right)Q_1(t) + k_{12}Q_2(t) - \\
& -R_{cl}\max\{0, Q_1(t) - R_{thr}V_G\} - Phy(t) + \\
& +EGP_0\max\{0, 1 - x_3(t)\} + \min\left\{U_{G,ceil}, \frac{G_2(t)}{t_{max}}\right\} \\
\dot{Q}_2(t) = & \; x_1(t)Q_1(t) - \Big(k_{12} + x_2(t)\Big)Q_2(t) \\
\dot{x}_1(t) = & -k_{b1}x_1(t) + S_{IT}k_{b1}I(t) \\
\dot{x}_2(t) = & -k_{b2}x_2(t) + S_{ID}k_{b2}I(t) \\
\dot{x}_3(t) = & -k_{b3}x_3(t) + S_{IE}k_{b3}I(t) \\
\dot{I}(t) = & \; \frac{k_a}{V_I}S_2(t) - k_eI(t) \\
\dot{S}_2(t) = & -k_aS_2(t) + k_aS_1(t) \\
\dot{S}_1(t) = & -k_aS_1(t) + u(t) \\
\dot{G}_2(t) = & \; \frac{G_1(t) - G_2(t)}{\max\left\{t_{max}, \frac{G_2(t)}{U_{G,ceil}}\right\}} \\
\dot{G}_1(t) = & -\frac{G_1(t)}{\max\left\{t_{max}, \frac{G_2(t)}{U_{G,ceil}}\right\}} + D(t)
\end{aligned} \tag{2.3}$$

$u(t)$ is the injected insulin flow of rapid-acting insulin [mU/min], which serves as the input of the system. $D(t)$ is the amount of ingested carbohydrates [mmol/min], and $Phy(t)$ is the effect of physical activity [mmol/min]. Both of the latter are considered as disturbances.

Table 2.1 provides details on model parameters. Parameters $k_{a,int}$, F_{01}^s, k_{12}, EGP_0, k_{b1}, k_{b2}, k_{b3}, S_{IT}, S_{ID}, S_{IE}, k_a and k_e are time-varying with $\pm 5\%$ deviation. In the in silico tests, this is represented by sinusoidal oscillations superimposed on the nominal values with three hour period and a randomly generated phase.

2.2 Model reduction

A high-order model is more challenging to handle than one with fewer state variables, let it be analysis, identification, observer, or controller design, regardless of the method used. Limited measurement capabilities also encourage focusing on the dominant components of the model dynamics. Hence, it can be advantageous to reduce the model to one that retains the most characteristic properties, as long as the error resulting from this simplification is manageable by the controller. The

Table 2.1. Cambridge model parameters

Name	Unit	Description
$k_{a,int}$	1/min	transfer rate constant between the plasma and the subcutaneous compartment
V_G	L	distribution volume of glucose in the accessible compartment
F_{01}^s	mmol/min	parameter of the total non-insulin dependent glucose flux
k_{12}	1/min	transfer rate constant from the non-accessible to the accessible compartment
R_{cl}	1/min	renal clearance constant
R_{thr}	mmol/L	glucose threshold for renal clearance
EGP_0	mmol/min	endogenous glucose production extrapolated to the zero insulin concentration
t_{max}	min	time-to-maximum appearance rate of glucose in the accessible compartment
$U_{G,ceil}$	mmol/min/kg	maximum glucose flux from the gut
k_{b1}, k_{b2}	1/min	deactivation rate constants
k_{b3}	1/min	deactivation rate constant for the insulin effect on endogenous glucose production
S_{IT}	L/mU/min	insulin sensitivity for transport
S_{ID}	L/mU/min	insulin sensitivity for distribution
S_{IE}	L/mU	insulin sensitivity for endogenous glucose production
k_a	1/min	insulin absorption rate constant
V_I	L	volume of distribution of rapid-acting insulin
k_e	1/min	fractional elimination rate from plasma

controller design must consider the errors resulting from model reduction, especially in the case of robust control.

The Cambridge model has more than ten state variables. For the ones that belong to meal absorption (considered disturbance), it is easy to choose a linear first-order substitution representing worst-case meal intake. For this subsystem, let us use the notation $W_{meal}(s)$.

$$W_{meal,H}(s) = \frac{U_{G,ceil}}{t_{max}s + 1} \tag{2.4}$$

The Cambridge model contains a second-order nonlinear component. The corresponding state variables are denoted as $Q_1(t)$ and $Q_2(t)$. The rest of the model, excluding the meal intake, consists of linear differential equations. This linear part can be further divided into two main categories: subcutaneous glucose transfer and insulin dynamics. C represents the former in the Cambridge model. Subcutaneous glucose transfer is negligible if an adequate state observer can accurately estimate plasma glucose concentration. Alternatively, it can be grouped into a CGMS model. The rest incorporates the transfer of the fast-acting insulin from the subcutaneous regions to the plasma, insulin degradation, and insulin effect. This linear system with a single input and multiple outputs can be simplified depending on the patient-specific parameters. In the Cambridge model they are x_1, x_2, x_3, I, S_1, and S_2.

Depending on the patient, the speed of transfer between specific compartments is comparable to the sampling time of the CGM. Hence, the states associated with them can be eliminated. The resulting reduced model is as follows:

$$\begin{aligned}
\dot{Q}_1(t) = \; & -\left(\tfrac{F_{01}}{Q_1(t)+V_G} + x_1(t)\right) Q_1(t) + k_{12}Q_2(t) - \\
& -R_{cl}\max\{0, Q_1(t) - R_{thr}V_G\} - Phy(t) + \\
& +EGP_0 \max\left\{0, 1 - \tfrac{k_a S_{IE}}{V_I k_e} S_2(t)\right\} + \min\left\{U_{G,ceil}, \tfrac{G_2(t)}{t_{max}}\right\} \\
\dot{Q}_2(t) = \; & x_1(t)Q_1(t) - \left(k_{12} + \tfrac{k_a S_{ID}}{V_I k_e} S_2(t)\right) Q_2(t) \\
\dot{x}_1(t) = \; & k_{b1}\left(\tfrac{k_a S_{IT}}{V_I k_e} S_2(t) - x_1(t)\right) \\
\dot{S}_2(t) = \; & -k_a S_2(t) + k_a S_1(t) \\
\dot{S}_1(t) = \; & -k_a S_1(t) + u(t) \\
\dot{G}_2(t) = \; & \frac{G_1(t) - G_2(t)}{\max\left\{t_{max}, \frac{G_2(t)}{U_{G,ceil}}\right\}} \\
\dot{G}_1(t) = \; & -\frac{G_1(t)}{\max\left\{t_{max}, \frac{G_2(t)}{U_{G,ceil}}\right\}} + D(t)
\end{aligned} \tag{2.5}$$

The output of this reduced modell, $C(t)$, is approximated from the state variable Q_1 as follows: $C(t) \approx Q_1(t)/V_G$. This simplification essentially ignores the delay of subcutaneous transfer of glucose to sensor.

The equations for state variables $S_1(t)$, $S_2(t)$, and $x_1(t)$ can be regarded as a third-order linear system of equations with injected insulin $u(t)$ as the only input and three outputs: $x_1(t)$, $(k_a S_{ID} S_2(t))/(V_I k_e)$ and $(k_a S_{IE} S_2(t))/(V_I k_e)$. The latter two are approximations of state variables $x_2(t)$ and $x_3(t)$ of the original model respectively. Figure 2.1 presents a representation of state elimination.

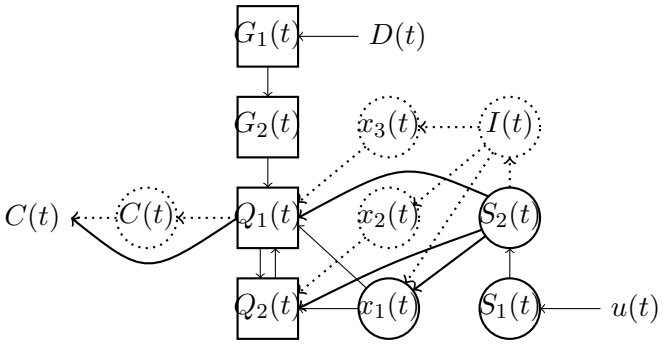

Figure 2.1. The elimination of state variables. Circle nodes represent linear; square nodes represent nonlinear equations. Disturbance input $Phy(t)$ is not displayed. Dotted lines indicate the eliminated state variables and connections.

Further reduction of this third-order system is possible using frequency-weighted balanced reduction presented in [86] or [87]. The nonlinear glucose subsystem of the model, linearized at a chosen working point, can serve as the frequency-depended weighting function if necessary. However, this may not be the most effective choice when using the system in other working points. Furthermore, the frequency-weighted balanced reduction does not guarantee that the reduced model remains stable and minimum-phase.

Finally, considering the range of parameter k_{12} as presented in [85], if $1/k_{12}$ is comparable to the CGMS sample time, then there is a need to apply reduction in the nonlinear dynamics as well. Section 4.3 presents an effective approach for a specific type of nonlinear control.

2.3 Modeling Uncertainties

All T1DM models are abstractions of the underlying physiological processes. They aim to capture essential aspects of the glucose-insulin interaction but are inherently inaccurate to a lesser or greater degree. Not even the Sorensen model can fully capture a system as complex as human metabolism. When developing a model-based control algorithm for AP, reductions, linearizations and other types of approximations of the

original T1DM are commonplace, usually to remove certain theoretical or practical limitations of using a chosen control methodology. However, these techniques make the inaccuracy of the model even more severe. Furthermore, human physiology tends to change and adapt over time. Hence, several T1DM models use time-varying parameters. Finally, human metabolism is affected by various external factors that are difficult to quantify, let alone measure [88] [89]. Therefore, regardless of which T1DM model one chooses for a model-based blood glucose control, the deviation of the model from the actual dynamics and the presence of disturbances must be addressed in the controller design.

2.3.1 Uncertainty functions

Let P denote a nominal nonlinear T1DM model, with a single output $y(t)$, a single controlled input $u(t)$, and a set of external disturbances $\mathbf{d}(t)$. Figure 2.2 presents a simple system that uses linear output uncertainty weighting functions.

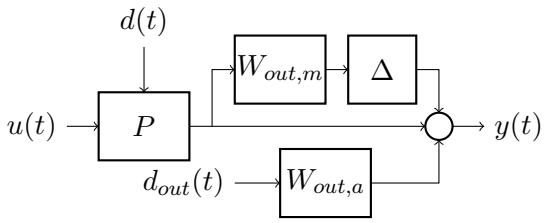

Figure 2.2. Simple system with output uncertainty weighting functions.

$W_{out,a}$ and $W_{out,m}$ are linear minimum-phase systems representing additive and multiplicative output uncertainty, respectively. Δ is an unstructured uncertainty, which is an unknown linear system with the following properties:

- minimum-phase;
- the $\mathcal{H}_\infty$-norm is less or equal to 1;

$d_{in}(t)$ and $d_{out}(t)$ are considered stochastic disturbance signals, that are usually modelled with distribution $\mathcal{N}(0,1)$ or $\mathcal{U}(-1,1)$. Section 4.4 provides more information regarding $\mathcal{H}_\infty$-norm.

There are other ways to model uncertainties, but this work only uses output uncertainties. Since most T1DM models have a single measured output, output uncertainties are sufficient to capture the deviation of the nominal model from the actual system. For example, the parameters

of $W_{out,a}$ and $W_{out,m}$ may come from the residual error of fitting T1DM parameters or from a priori knowledge from clinical practice.

Extending the nominal model with uncertainties is essential for both controller and state observer, as it can ensure robustness and stability for both control and state estimation [KS16]. However, it also poses a challenge if the control law is realized with state feedback. In this particular case, the state variables of the uncertainty models shall also be estimated.

2.4 Discrete-Time Modeling

All physiological models presented in this chapter are defined in continuous time. However, the glucose measurements are available only in discrete time instances. Furthermore, the controllers used in practice work in discrete time as well. Therefore, it is crucial to address the problem of defining a discrete-time nonlinear model for a given continuous-time T1DM model.

Most T1DM models presented or referenced in Chapter 2 belong to a subset of continuous-time nonlinear models described by (2.6):

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathrm{f}(\mathbf{x}(t)) + \mathbf{B}_1\mathbf{d}(t) + \mathbf{B}_2 u(t) \\ y(t) &= \mathbf{C}\mathbf{x}(t), \end{aligned} \tag{2.6}$$

where $\mathbf{x}(t)$ is the vector of state variables with n elements, $\mathrm{f}(\mathbf{x}(t)) : \mathbb{R}^n \to \mathbb{R}^n$ is a vector field, while $\mathbf{B}_2$ and $\mathbf{C}$ are $n \times 1$ and $1 \times n$ vectors respectively. $\mathbf{B}_1$ is a $n \times m$ matrix. $u(t)$ denotes the single controllable output of the system, which is the external insulin injected intravenously or subcutaneously. The system has only one input denoted as $y(t)$ and usually refers to the glucose concentration in the plasma or the subcutaneous regions. The model is affected by m disturbances $\mathbf{d}(t)$.

For a continuous-time T1DM model, one can define a discrete-time nonlinear model, usually by approximation with numerical methods. The following equations represent the discrete-time nonlinear model (2.7):

$$\begin{aligned} \mathbf{x}[k+1] &= \mathrm{f}_d(\mathbf{x}[k], \mathbf{u}[k]) \\ y[k] &= g_d(\mathbf{x}[k]) \end{aligned} \tag{2.7}$$

The square bracket represents the value of a time-dependent variable at a discrete time instance, e.g., $\mathbf{x}[k] = \mathbf{x}(kT_s)$. The new functions introduced in (2.7) are $\mathrm{f}_d(\mathbf{x}[k], \mathbf{d}[k], u[k]) : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^n$, and $g_d(\mathbf{x}[k]) : \mathbb{R}^n \to \mathbb{R}$.

The most straightforward technique for deriving a discrete-time model is the Euler method [90], which provides first order approximation for a defined sampling time T_s:

$$\mathrm{f}_d(\mathbf{x}[k], \mathbf{d}[k], u[k]) \approx \mathbf{x}[k] + T_s\Big(\mathrm{f}(\mathbf{x}[k]) + \mathbf{B}_1\mathbf{d}[k] + \mathbf{B}_2 u[k]\Big). \tag{2.8}$$

This method is relatively easy to implement. Furthermore, considering the continuous time model (2.6), the discrete-time model will also depend linearly on the input and disturbance signals.

The control algorithms in later chapters use the four-step Runge-Kutta method, presented by (2.9), equivalent to a fourth-order approximation, and hence offer higher accuracy than the Euler method. On the other hand, the four-step Runge-Kutta is relatively straightforward to implement and execute even on embedded hardware with limited computational power.

$$
\begin{aligned}
\chi_1[k] &= T_s\Big(\mathrm{f}\Big(\mathbf{x}[k]\Big) + \mathbf{B}_1\mathbf{d}[k] + \mathbf{B}_2 u[k]\Big) \\
\chi_2[k] &= T_s\left(\mathrm{f}\Big(\mathbf{x}[k] + \tfrac{\chi_1[k]}{2}\Big) + \mathbf{B}_1\mathbf{d}[k+\tfrac{1}{2}] + \mathbf{B}_2 u[k+\tfrac{1}{2}]\right) \\
\chi_3[k] &= T_s\left(\mathrm{f}\Big(\mathbf{x}[k] + \tfrac{\chi_2[k]}{2}\Big) + \mathbf{B}_1\mathbf{d}[k+\tfrac{1}{2}] + \mathbf{B}_2 u[k+\tfrac{1}{2}]\right) \\
\chi_4[k] &= T_s\Big(\mathrm{f}\Big(\mathbf{x}[k] + \chi_3[k]\Big) + \mathbf{B}_1\mathbf{d}[k+1] + \mathbf{B}_2 u[k+1]\Big) \\
\mathbf{x}[k+1] &= \mathbf{x}[k] + \tfrac{1}{6}\Big(\chi_1[k] + 2\chi_2[k] + 2\chi_3[k] + \chi_4[k]\Big)
\end{aligned}
\tag{2.9}
$$

2.5 Continuous Glucose Measurement

Reliable continuous plasma glucose measurement is vital in automated type 1 diabetes treatment. Self-monitoring of blood glucose (SMBG), an invasive measurement method performed manually by the patient or physician, is insufficient for closed-loop control. While measurements acquired through SMBG are usually highly accurate and precise, only around 4-6 samples are recorded during a single day, usually in an event-driven manner (e.g., postprandial) [91]. On the other hand, continuous glucose measurement sensors (CGMS) automatically update readings every 3-5 minutes. The CGMS has two main groups: invasive and non- or minimally invasive.

Invasive sensors commonly use enzyme-based detection, realized with a needle covered by a glucose permeable membrane [92]. The enzymes catalyze reduction-oxidation reactions and, in doing so, produce a concentration-dependent current or voltage. This current or voltage is measurable with electrodes. The enzyme commonly used in CGMS is glucose oxidase. However, due to protein and cell coating of the sensor, variable tissue oxygen tension, tissue interferents, and wound response, patients need to change the electrode of the sensor every 10-12 days. The same effects are also partially responsible for significant sensor drift. Hence, earlier models required frequent calibration, while later models could function without calibration. Some commercially available invasive CGMS are Dexcom G6 CGM System, Abbott FreeStyle Libre 2, and Medtronic Guardian™ Connect System.

Non-invasive or minimally invasive CGMS are less common in clinical practice but remain an actively researched field. There are several potential approaches to measuring glucose in the interstitial fluid in a non-invasive manner [92]. Section A.1.2 provides more details about these sensors.

3

OBSERVER DESIGN

This Chapter addresses the first thesis group: a new state observer framework for estimating the state variables of nonlinear T1DM models and the glucose flux resulting from meal intake. The state observer considers the measurement noise of CGM sensors, the nonlinearity, uncertainty, and nonnegativity of the model, as well as the glucose utilization resulting from physical activity. The contributions that belong to this thesis group are the following:

1. A method for a generic stochastic state observer algorithm to consider the nonnegativity of the model and the nonnegativity of the most significant disturbances: meal intake and physical activity.

2. A new state observer framework that considers and estimates the additive and multiplicative output uncertainty of a T1DM model.

3. A state observer framework that can estimate the glucose flux resulting from meal intake and ingestion.

4. A predictor algorithm that can provide long-term prediction for a nonlinear and uncertain T1DM model using planned meal intake announcements.

The presented observer and predictor were published in: [SEK14, SMM$^+$14, SSBK14, KS16, SBK16, SDKew].

When it comes to diabetes treatment, mainly the following measurements and devices support glucose control:

1. *Blood glucose meter*: This measurement provides the most accurate real-time blood glucose reading. Precisely, the glucose meter measures the capillary glucose concentration. It is an invasive

method performed by the patient or clinician 4-6 times per day. The results determine the need for insulin injection or glucose intake in classical Diabetes treatment. In the case of Artificial Pancreas, SMBG with a blood glucose meter can aid sensor calibration or enable safety and diagnostic features.

2. *Glycated hemoglobin test* (HbA1c): Most monosaccharides, including glucose, galactose, and fructose, can non-enzymatically bond with hemoglobin in the bloodstream. Hence, the formation of the sugar-hemoglobin indicates the presence of excessive glucose. HbA1c test provides an estimation of a three-month average blood sugar level. Therefore, it can assess the effectiveness of a Diabetes treatment regarding the long-term complications.

3. *Continuous Glucose Monitor* (CGM): The primary sensor in automated Diabetes treatment. The commercially available sensors have limited accuracy and precision due to significant noise and drift. See Section 2.5 for further details.

Moreover, some therapeutic and fitness devices may be re-purposed for AP, as they can provide information on physical activity of the patient [93]. This kind of measurement, however, is not in the scope of this work.

Aside from sensory input, announcements made by the patient can further the range of information available for the AP. For example, the patient can inform the AP about meal intake, insulin injection, planned physical activity, and visible damage to hardware components, among others. All these can improve the treatment, as long as they are reliable. Consequently, the AP must validate all announcements and use them to the point that invalid inputs cannot endanger the patient or have a detrimental effect on controlling glucose levels.

Despite all the previously presented sensors and inputs, multiple quantities cannot be measured directly, or the measurement is impractical in clinical practice. Nevertheless, they have high importance for AP. Notable examples are plasma glucose concentration, glucose flux from the gut, and plasma insulin concentration. Furthermore, human metabolism is affected by various outside factors that act as disturbances. These are difficult to measure or even quantify, especially in real time.

Since direct measurement is not an option, it is necessary to estimate the needed quantities. State observers are good candidates to provide this estimation if an appropriate state-space model is available. Not only can they estimate quantities not available for measurement, but they also enable control algorithms realized as linear or nonlinear state feedback, among others. Moreover, state observers can detect

unannounced events, such as meal intake or physical activity, or validate announcements of the patient.

Techniques derived from Kalman filter are some of the most frequently used approaches for this task. The Extended Kalman filter addresses the nonlinearity of the human metabolism with linearization. Notable examples include [53, 52, 54]. The applications are not limited to state estimation but include meal and fault detection and parameter estimation. However, linearization may lead to inaccuracies when the nonlinearity is severe. Unscented Kalman Filter, a type of sigma point filter, can provide a more efficient alternative in AP use [55, 56, 57]. The state observer presented in this Chapter continues this trend by choosing the sigma point filter as the core of the proposed algorithm. Other notable examples that do not rely on the Kalman filter include Moving Horizon estimation [60], Sliding mode observers [58, 62], Machine Learning [65, 66], and Neural Networks [68, 69, 70] with the latter two more prevalent in prediction.

3.1 Methods

3.1.1 Kalman Filter

Measurement noise and disturbances affect all real-life systems, and neither human metabolism nor AP is an exception. In this work, *noise* refers to the unwanted and unknown modifications that the measurement signals may suffer during capture. *Disturbances*, on the other hand, refer to unwanted or unknown inputs of the system. Usually, both the former and the latter have stochastic aspects. A state observer shall provide a sufficiently accurate and precise estimation of the state variables of a T1DM model despite the noises and disturbances affecting it.

The well-known and widely used Kalman filter [94], [95] gives a recursive optimal estimation of the state variables of a linear discrete-time dynamic system in the sense that the vector of estimation error has a zero mean and minimal covariance matrix (based on the partial ordering of positive definite matrices). There are various ways to derive the Kalman filter algorithm. This work uses the approach presented in [96].

Let us have the following linear discrete-time nonlinear system:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{w}_k \\ \mathbf{y}_k &= \mathbf{C}_k \mathbf{x}_k + \mathbf{D}_k \mathbf{v}_k, \end{aligned} \tag{3.1}$$

where $\mathbf{x}_k \in \mathbb{R}^{n_x}$ denotes the vector of state variables with n_x elements at time k. $\mathbf{y}_k \in \mathbb{R}^{n_y}$ is the vector of measured signals, $\mathbf{w}_k \in \mathbb{R}^{n_w}$

represents disturbances affecting the states while $\mathbf{v}_k \in \mathbb{R}^{n_v}$ is the measurement noise. The matrices $\mathbf{A}_k$, $\mathbf{B}_k$, $\mathbf{C}_k$, and $\mathbf{D}_k$, are $n_x \times n_x$, $n_x \times n_w$, $n_y \times n_x$, and $n_y \times n_v$ real matrices respectively, which define linear projection of $\mathbf{x}_k$ and $\mathbf{w}_k$ to $\mathbf{x}_{k+1}$, as well as $\mathbf{x}_k$ and $\mathbf{v}_k$ to $\mathbf{y}_k$. We have the following assumptions about the system, which is referred to as *stochastic hypothesis*:

1. The disturbance is white and has $\bar{\mathbf{w}}_k$ mean value (usually zero) with known covariance matrix:
   $\begin{cases} E(\mathbf{w}_k \mathbf{w}_k^T) = \mathbf{Q}_k \\ E(\mathbf{w}_i \mathbf{w}_j^T) = \mathbf{0} \qquad i \neq j \end{cases}$
2. The measurement noise is white and has $\bar{\mathbf{v}}$ mean value (usually zero) with known covariance matrix:
   $\begin{cases} E(\mathbf{v}_k \mathbf{v}_k^T) = \mathbf{R}_k \\ E(\mathbf{v}_i \mathbf{v}_j^T) = \mathbf{0} \qquad i \neq j \end{cases}$
3. The measurement noise and the process noise are independent $E(\mathbf{w}_i \mathbf{v}_j^T) = \mathbf{0} \quad \forall i, j$
4. $\mathbf{x}_0$ is independent from $\mathbf{w}_k$ and $\mathbf{v}_k$ for all k;
5. Given an initial estimation $\hat{\mathbf{x}}_0$ of $\mathbf{x}_0$ the initial estimation error has zero mean and the covariance matrix $\mathbf{\Sigma}_0 = E((\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T)$ is known;

For (3.1) we can define the Kalman filter in the following form:

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A}_k \hat{\mathbf{x}}_k + \mathbf{B}_k \bar{\mathbf{w}}_k + \mathbf{K}_k (\mathbf{y}_k - \mathbf{C}_k \hat{\mathbf{x}}_k - \mathbf{D}_k \bar{\mathbf{v}}_k) \tag{3.2}$$

where $\hat{\mathbf{x}}_k$ is the estimation of the state variables at time k and $\mathbf{K}_k$ is the Kalman gain. The state estimation algorithm is as follows:

0. *Initialization*:
   The initial estimation of the $\mathbf{x}_0$ state variables $\hat{\mathbf{x}}_0$ and the initial estimation error covariance matrix $\mathbf{\Sigma}_0$ are chosen based on a priori knowledge.

1. *Estimation*:
   The system matrices in equations (3.1) are known, $\hat{\mathbf{x}}_k$ and $\mathbf{\Sigma}_k$ are available from the previous iteration, while $\bar{\mathbf{w}}_k$, $\mathbf{Q}_k$, $\bar{\mathbf{v}}_{k+1}$ and $\mathbf{R}_{k+1}$ are derived from a priori information.

   (a) Calculate the initial estimation of the state vector using the state equations (3.1): $\bar{\mathbf{x}}_{k+1} = \mathbf{A}_k \hat{\mathbf{x}}_k + \mathbf{B}_k \bar{\mathbf{w}}_k$.

   (b) Calculate the covariance matrix:
   $\mathbf{P}_{k+1}^{(xx)} = \mathbf{A}_k \mathbf{\Sigma}_k \mathbf{A}_k^T + \mathbf{B}_k \mathbf{Q}_k \mathbf{B}_k^T$.

(c) Calculate the initial estimation of the output:
$\bar{\mathbf{y}}_{k+1} = \mathbf{C}_{k+1}\bar{\mathbf{x}}_{k+1} + \mathbf{D}_{k+1}\bar{\mathbf{v}}_{k+1}$.

(d) Calculate the covariance matrices: $\mathbf{P}_{k+1}^{(xy)} = \mathbf{P}_{k+1}^{(xx)}\mathbf{C}_{k+1}^T$ and $\mathbf{P}_{k+1}^{(yy)} = \mathbf{C}_{k+1}\mathbf{P}_{k+1}^{(xx)}\mathbf{C}_{k+1}^T + \mathbf{D}_{k+1}\mathbf{R}_{k+1}\mathbf{C}_{k+1}^T$.

2. *Update*:
The actual value of output $\mathbf{y}_{k+1}$ is available through measurement.

(a) Calculate the Kalman gain: $\mathbf{K}_{k+1} = \mathbf{P}_{k+1}^{(xy)}\left(\mathbf{P}_{k+1}^{(yy)}\right)^{-1}$.

(b) Update the initial estimation of the state variables:
$\hat{\mathbf{x}}_{k+1} = \bar{\mathbf{x}}_{k+1} + \mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{k+1})$.

(c) Update the estimation error covariance matrix:
$\mathbf{\Sigma}_{k+1} = \mathbf{P}_{k+1}^{(xx)} - \mathbf{K}_{k+1}\mathbf{P}_{k+1}^{(yy)}\mathbf{K}_{k+1}^T$, or
$\mathbf{\Sigma}_{k+1} = \mathbf{P}_{k+1}^{(xx)} - \mathbf{P}_{k+1}^{(xy)}\left(\mathbf{P}_{k+1}^{(yy)}\right)^{-1}\left(\mathbf{P}_{k+1}^{(xy)}\right)^T$.

3. Set $k = k + 1$ and continue from *Estimation*.

3.1.2 Extended Kalman filter

Kalman filter is a powerful tool for state estimation in the case of linear stochastic models. However, in numerous practical applications, a linear model is insufficient to capture the system's dynamics, so nonlinear representation is necessary. This limitation motivated the extension of the Kalman filter to nonlinear models. One of the most widely used nonlinear filters is still the Extended Kalman Filter (EKF), which is based on first-order linearization of the observed nonlinear model [96, 97].

Let us have the following nonlinear system:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathrm{f}(\mathbf{x}_k, \mathbf{w}_k, k) \\ \mathbf{y}_k &= \mathrm{h}(\mathbf{x}_k, \mathbf{v}_k, k) \end{aligned} \tag{3.3}$$

where $\mathrm{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_w} \times \mathbb{R} \to \mathbb{R}^{n_x}$ and $\mathrm{h} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_v} \times \mathbb{R} \to \mathbb{R}^{n_y}$ are piece-wise continuous nonlinear mappings.

In the EKF algorithm, the *Initialization* and *Update* steps remain identical. The *Estimation* however, has differences and is performed as follows:

1. Calculate $\mathbf{A}_k = \frac{\partial \mathrm{f}(\hat{\mathbf{x}}_k, 0, k)}{\partial \mathbf{x}}$, $\mathbf{B}_k = \frac{\partial \mathrm{f}(\hat{\mathbf{x}}_k, 0, k)}{\partial \mathbf{w}}$.

2. Use the nonlinear model (3.3) to calculate the initial estimation of the state vector: $\bar{\mathbf{x}}_{k+1} = \mathrm{f}(\hat{\mathbf{x}}_k, \bar{\mathbf{w}}_k, k)$.

3. Use the linearized model from step 1 to calculate the covariance matrix $\mathbf{P}_{k+1}^{(xx)}$ the same way as in the regular Kalman filter algorithm.

4. Calculate $\mathbf{C}_{k+1} = \frac{\partial \mathrm{h}(\bar{\mathbf{x}}_{k+1},0,k+1)}{\partial \mathbf{x}}$ and $\mathbf{D}_{k+1} = \frac{\partial \mathrm{h}(\bar{\mathbf{x}}_{k+1},0,k+1)}{\partial \mathbf{v}}$ using the initial estimation $\bar{\mathbf{x}}_{k+1}$.

5. Use the nonlinear model to calculate the initial estimation of the output: $\bar{\mathbf{y}}_{k+1} = \mathrm{h}(\bar{\mathbf{x}}_{k+1}, \bar{\mathbf{v}}_k, k+1)$.

6. Use the linearized model to calculate the covariance matrices: $\mathbf{P}_{k+1}^{(xy)}$ and $\mathbf{P}_{k+1}^{(yy)}$ the same way as in the regular Kalman filter algorithm.

Despite how versatile and practical EKF can be, it may fail when the system contains severe nonlinearities, significant disturbances, or initial estimation errors. The distributions will not be Gaussian anymore and are usually not symmetric. The linearization in the EKF cannot cope with this effect.

3.1.3 Sigma Point Filters

The need to overcome the limitations of the EKF gave birth to sigma point filters. They are a form of nonlinear Kalman filters which use a number of deterministic samples - called sigma points - to represent the probability distribution of the system state [98]. Several versions differ mainly on how these sigma points are selected. Cubature Kalman Filter (CKF) is based on the Cubature rule [99] and is one of the most straightforward approaches. The slightly more popular Unscented Kalman Filter (UKF) relies on the Unscented Transformation [98] with tunable parameters for each filtering problem to achieve better performance. On the other hand, the Central Difference Kalman filter (CDKF) works with the Stirling polynomial interpolation formula [100]. Gauss-Hermite Quadrature Filter (GHQF) offers even higher accuracy when Gaussian distribution is guaranteed [101]. However, it also requires a large number of sigma points and hence increased computational power. Sparse-grid quadrature nonlinear filtering (SGQF) primarily relies on the same assumptions and aims to achieve similar accuracy and precision as GHQF but with significantly fewer sigma points [102]. Particle filters [103] can offer the highest accuracy since it works with a statistically significant number of samples. Instead of focusing merely on the mean and the covariance, one can acquire more details from the distribution of the estimated state variables. However, particle filters require the most computational power from all the filters presented here. In contrast, the Reduced Sigma Point filter (RSPF) [104] uses the least

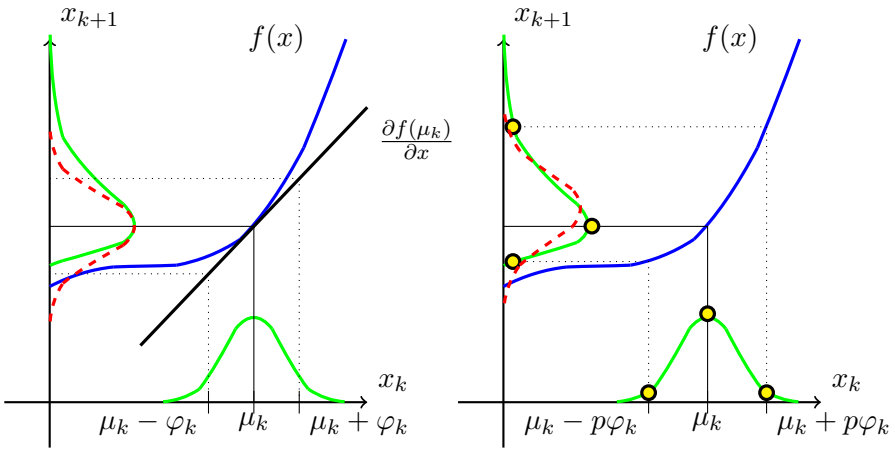

Figure 3.1. Illustration of the difference between estimating using linearization (left) and with sigma points (right). The solid green lines represent the real distribution of x_k and x_{k+1}. The distribution of x_k is $\mathcal{N}(\mu_k, \varphi_k)$. p denotes a parameter associated with sigma point selection. The solid blue line represents the nonlinear mapping $f(x)$. The estimated distribution of x_{k+1} is illustrated with red dashed line. The yellow dots represent sigma points before and after the mapping.

amount of sigma points, prioritizing computational efficiency over estimation accuracy.

Figure 3.1 illustrates the advantage of sigma point filters over EKF. Let us have a nonlinear system of the first order: $x_{k+1} = f(x_k)$ and assume, that x_k is a stochastic variable with Gaussian distribution with given mean and variance (μ_k and φ_k respectively). Linearization can lead to an incorrect estimation of $E\{x_{k+1}\}$, and therefore cannot ensure zero mean value for the estimation error. Sigma point filters can provide a better estimation as long as the sigma points are chosen to reflect the system's nonlinearity. Sigma point collapse [105] can occur when sigma points are chosen incorrectly (Figure 3.2).

3.1.4 Sigma point selection

Let us introduce the notation $\mathcal{X}$ for a sigma point set. This set contains N number of sigma points denoted as $\xi_i \in \mathbb{R}^L$, $i = 1, \ldots, N$. The sigma points that represent the stochastic variable $\mathbf{x} \sim \mathcal{N}(\mu, \mathbf{\Sigma})$ can be written in the following form:

$$\xi_i = \mathbf{\Sigma}^{\frac{1}{2}} \varphi_i + \mu \tag{3.4}$$

where φ_i is a vector corresponding to a sigma point selection strategy. $\mathbf{\Sigma}^{\frac{1}{2}}$ is the factor of $\mathbf{\Sigma}$ so that $\mathbf{\Sigma} = \mathbf{\Sigma}^{\frac{1}{2}} \mathbf{\Sigma}^{\frac{T}{2}}$. Cholesky decomposition is a

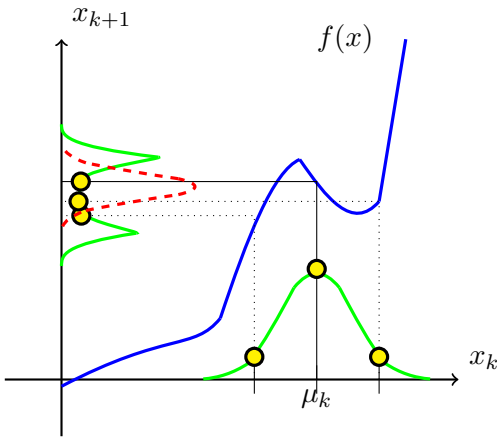

Figure 3.2. Illustration of sigma point collapse. The solid green lines represent the real distribution of x_k and x_{k+1}. The solid blue line represents the nonlinear mapping $f(x)$. The incorrectly estimated distribution of x_{k+1} is illustrated with red dashed line. The yellow dots represent sigma points before and after the mapping.

commonly used method for determining $\mathbf{\Sigma}^{\frac{1}{2}}$, since $\mathbf{\Sigma}$ is positive definite. μ is not limited to state variables only. It can contain disturbances and measurement noises as well. Using these sigma points, one can estimate the mean and covariance of the distribution of $f(X)$ as a weighted sum of $f(\mathcal{X})$:

$$\begin{aligned} \mathrm{E}\left(f(\mathbf{x})\right) &\approx \sum_{i=1}^{N} \omega_i^{(m)} f(\xi_i) = \bar{\mathbf{x}} \\ cov\left(f(\mathbf{x})\right) &\approx \sum_{i=1}^{N} \omega_i^{(c)} \left((f(\xi_i) - \bar{\mathbf{x}})(f(\xi_i) - \bar{\mathbf{x}})^T\right) \end{aligned} \tag{3.5}$$

There are various strategies to choose φ_i and the weights $\omega_i^{(m)}$ and $\omega_i^{(c)}$. A comparative summary of the sigma point strategies referenced in this work can be found in Table 3.1.4. The following properties were used for comparison:

- *Number of sigma points*: it indicates the computational power required to execute the filter algorithms. Hence, using a large amount of sigma points can be a limiting factor on embedded systems. Moreover, numerical instability may increase with the increasing number of points.
- *Number of parameters*: How configurable the method is.
- *Assumption of distribution*: Some methods rely on the assumption that the prior statistics of estimation error, disturbances, and noises are Gaussian, while others are less restrictive.

- *Nonnegative weights*: When calculating the estimated covariance matrices (3.5), negative weights can lead to singular or non-definite results.

Cubature Kalman filter

Nonlinear filtering in the Gaussian domain essentially reduces to a problem of how to compute integrals of nonlinear functions on Gaussian distribution (3.6):

$$\int_{\mathbb{R}^L} f(\mathbf{x}) \exp\left(-\mathbf{x}^T\mathbf{x}\right) d\mathbf{x}. \tag{3.6}$$

The Cubature Kalman filter aims to provide numerically accurate estimation that is easily extendable to high-dimensional problems [99]. The term *cubature* describes the numerical computation of a multiple integral [106]. It uses only $2L$ sigma points derived from a well-chosen third-degree spherical-radial cubature rule. The weights and basis functions $\omega^{(m)}$, $\omega^{(c)}$ and φ of the CKF require no adjustable parameter.

$$\begin{aligned} \varphi_i &= \begin{cases} \mathbf{e}_i\sqrt{L} & i = 1,\ldots,L \\ -\mathbf{e}_i\sqrt{L} & i = L+1,\ldots,2L \end{cases} \\ \omega_i^{(m)} &= \omega_i^{(c)} = \frac{1}{L}, \quad \forall i \end{aligned} \tag{3.7}$$

where $\mathbf{e}_i$ denotes the unit vector in $\mathbb{R}^L$ where the only non-zero element is the $(i-1)$th.

Because the third-degree rule is exact up to third-degree polynomials, the CKF computes the posterior mean accurately but only approximates the estimation error covariance. Furthermore, it assumes Gaussian distribution.

Unscented Kalman filter

The Unscented Kalman filter relies on the unscented transform, computing posterior statistics of an L-dimensional stochastic variable transformed by a nonlinear function [98]. Unlike CKF, the prior distribution does not need to be strictly Gaussian but merely symmetric. The UKF has three adjustable parameters: κ, α, and β. The weights and basis

functions ω and φ in the case of a UKF are:

$$\begin{aligned}
\varphi_i &= \begin{cases} \mathbf{0} & i = 1 \\ \mathbf{e}_i\sqrt{L+\lambda} & i = 2,\dots,L+1 \\ -\mathbf{e}_i\sqrt{L+\lambda} & i = L+2,\dots,2L+1 \end{cases} \\
\omega_i^{(m)} &= \begin{cases} \dfrac{\lambda}{L+\lambda} & i = 1 \\ \dfrac{1}{2(L+\lambda)} & i = 2,\dots,2L+1 \end{cases} \\
\omega_i^{(c)} &= \begin{cases} \dfrac{\lambda}{L+\lambda} + 1 - \alpha^2 + \beta & i = 1 \\ \dfrac{1}{2(L+\lambda)} & i = 2,\dots,2L+1 \end{cases}
\end{aligned} \tag{3.8}$$

where $\lambda = \alpha^2(L+\kappa) - L$ is a scaling parameter [97]. The constant α determines the spread of sigma points around μ, and is usually set to a small positive value (e.g., $10^{-4} \leq \alpha \leq 1$). The constant κ is a second scaling parameter usually set to $3 - L$ so that the kurtosis of the sigma points agrees with that of the Gaussian distribution [98]. β is used to incorporate prior knowledge of the distribution of μ and is usually set to 2 for the Gaussian distribution.

Reduced Sigma Point filter

Reduced Sigma Point filter uses a significantly smaller number of sigma points than CKF or UKF. A minimum set of $L+1$ asymmetric sigma points can fully capture the mean and covariance of a distribution. However, to incorporate minimal information regarding the third central moment (skewness), at least one more sigma point is necessary.

$$\begin{aligned}
\varphi_i &= \begin{cases} \mathbf{0} & i = 1 \\ \left(\dfrac{-2^{(L-1)/2}}{\sqrt{1-p_0}} \cdots \dfrac{-1}{\sqrt{1-p_0}}\right)^T & i = 2 \\ \left(\dfrac{2^{(L-1)/2}}{\sqrt{1-p_0}}, \dfrac{-2^{(L-2)/2}}{\sqrt{1-p_0}} \cdots \dfrac{-1}{\sqrt{1-p_0}}\right)^T & i = 3 \\ \left(0\dots0, \dfrac{2^{(L+2-i)/2}}{\sqrt{1-p_0}}, \dfrac{-2^{(L+1-i)/2}}{\sqrt{1-p_0}} \cdots \dfrac{-1}{\sqrt{1-p_0}}\right)^T & i = 4,\dots,L+1 \\ \left(0\dots0, \dfrac{1}{\sqrt{1-p_0}}\right)^T & i = L+2 \end{cases} \\
\omega_i^{(m)} &= \begin{cases} p_0 & i = 1 \\ \frac{1-p_0}{2^L} & i = 2 \\ 2^{i-3}\omega_2^{(m)} & i = 3,\dots,L+2 \end{cases} \qquad \omega_i^{(c)} = \omega_i^{(m)},
\end{aligned} \tag{3.9}$$

where $p_0 \in [0, 1]$ is the only adjustable parameter. Although it has reduced accuracy and precision, RSPF is useful when there is a need to minimize computation time.

Central Difference Kalman filter

Central Difference Kalman filter share several properties with UKF, including the number of sigma points ($2L + 1$), how their spread is configurable with a parameter (h), and that they both work with any symmetrical prior distribution. Furthermore, h is usually set to $\sqrt{3}$, which matches $\alpha = 1$ and $\kappa = 3 - L$ settings of UKF:

$$\varphi_i = \begin{cases} \mathbf{0} & i = 1 \\ \mathbf{e}_i h & i = 2, \ldots, L + 1 \\ -\mathbf{e}_i h & i = L + 2, \ldots, 2L + 1. \end{cases} \tag{3.10}$$

The mean value is calculated similarly as well, with the following weights:

$$\omega_i^{(m)} = \begin{cases} \dfrac{h^2 - L}{h^2} & i = 1 \\ \dfrac{1}{2h^2} & i = 2, \ldots, L + 2. \end{cases} \tag{3.11}$$

The novelty lies in how the covariance matrix is approximated from the sigma points. Instead of the formula used by other sigma point filters (3.5), CDKF uses the Stirling polynomial interpolation formula [100]:

$$\begin{aligned} cov\left(f(\mathbf{x})\right) \approx \quad & \textstyle\sum_{i=1}^{L} \quad \omega_i^{(c1)} \Big(f(\xi_{i+1}) - f(\xi_{i+L+1}) \Big) \\ & \Big(f(\xi_{i+1}) - f(\xi_{i+L+1}) \Big)^T + \\ & \textstyle + \sum_{i=1}^{L} \quad \omega_i^{(c2)} \Big(f(\xi_{i+1}) + f(\xi_{i+L+1}) - 2f(\xi_1) \Big) \\ & \Big(f(\xi_{i+1}) + f(\xi_{i+L+1}) - 2f(\xi_1) \Big)^T, \end{aligned} \tag{3.12}$$

where the two types of weights used are as follows:

$$\begin{aligned} \omega_i^{(c1)} &= \frac{1}{4h^2} \quad \forall i \\ \omega_i^{(c2)} &= \frac{h^2 - 1}{4h^4} \quad \forall i. \end{aligned} \tag{3.13}$$

The estimated mean value is not necessary for the covariance calculation.

Gauss-Hermite Quadrature filter

The Gauss-Hermite Quadrature filter is derived similarly to the the Cubature Kalman filter but relies on the Gauss-Hermite quadrature rule. Unlike the cubature rule, which deals with multiple integral, the *quadrature rule* deals with numerical computation of a univariate integral [106]. Arasaratnam et al. [101] propose a GHQF algorithm, which defines m number of sigma points for each scalar stochastic variable. These sigma points are then combined into m^L sigma points. m is an adjustable positive integer-valued parameter. For a single variable, the weights and basis functions ω and φ are defined using a symmetric $m \times m$ tridiagonal matrix $\mathbf{J}$:

$$\begin{aligned} J_{i,i} &= 0 \\ J_{i,i+1} &= \sqrt{i/2}, \quad 1 \geq i \geq m-1. \end{aligned} \tag{3.14}$$

Then, if ϵ_i represents the i-th eigenvalue of $\mathbf{J}$, while $v_{1,i}$ is the first element of the i-th normalized eigenvector of $\mathbf{J}$, the weights and basis functions are as follows:

$$\begin{aligned} \varphi_i &= \sqrt{2}\epsilon_i \\ \omega_i^{(m)} &= v_{1,i}^2 \\ \omega_i^{(c)} &= \omega_i^{(m)} \end{aligned} \tag{3.15}$$

Furthermore, GHQF shares the assumption with CKF that the prior statistics of estimation error, disturbances, and noises are Gaussian. Gaussian Sum Quadrature Kalman filter is an extension of GHQF to non-Gaussian distributions [101]. However, it requires even more sigma points and hence is not discussed further here.

Sparse-Grid Quadrature filter

Sparse-Grid Quadrature filter aims to provide the same capabilities as GHQF, but with the number of sigma points scaling polynomially instead of exponentially with dimension. It uses the Sparse-grid quadrature (SGQ) rule [102], a method that is meant to alleviate the curse of dimensionality problem in numerical integration. SGQ can be derived for different accuracy levels. The SGQ with accuracy level P is exact for all multivariate polynomials with the total order of $2P-1$. Both CKF and UKF can be considered special cases of level-2 SGQF. Level-3 SGQF can provide a similar estimation performance as GHQF with 7^L sigma points but requires maximum $2L^2+4L+1$ sigma points. The exact number depends on how the three free parameters – p_1, p_2 and p_3 – are chosen. Similarly to the GHQF, these parameters are selected from the perspective of a univariate estimation, where the points $\mu + \{-p_1, 0, p_1\}$ and $\mu + \{-p_3, -p_2, 0, p_2, p_3\}$ are used to estimate certain moments of an univariate Gaussian distribution transformed by a nonlinear function.

If all parameters are unique ($p_1 \neq p_2$, $p_2 \neq p_3$ and $p_3 \neq p_1$), the sigma points of SGQF are as follows:

$$\varphi_i = \begin{cases} \mathbf{0} & i = 1 \\ \mathbf{e}_i p_1 & i = 2, \ldots, L+1 \\ -\mathbf{e}_i p_1 & i = L+2, \ldots, 2L+1 \\ \mathbf{e}_i p_2 & i = 2L+2, \ldots, 3L+1 \\ -\mathbf{e}_i p_2 & i = 3L+2, \ldots, 4L+1 \\ \mathbf{e}_i p_3 & i = 4L+2, \ldots, 5L+1 \\ -\mathbf{e}_i p_3 & i = 5L+2, \ldots, 6L+1 \\ \mathbf{e}_i p_1 + \mathbf{e}_j p_1 & i = 6L+2, \ldots, 6L+1+C \\ -\mathbf{e}_i p_1 + \mathbf{e}_j p_1 & i = 6L+2+C, \ldots, 6L+1+2C \\ \mathbf{e}_i p_1 - \mathbf{e}_j p_1 & i = 6L+2+2C, \ldots, 6L+1+3C \\ -\mathbf{e}_i p_1 - \mathbf{e}_j p_1 & i = 6L+2+3C, \ldots, 6L+1+4C, \end{cases} \tag{3.16}$$

where $C = L(L-1)/2$ and $j \neq i$. Let us introduce the following variables for the weight calculation:

$$\begin{gathered} \hat{\omega}_1 = 1 - \frac{1}{p_1^2} \quad \hat{\omega}_2 = \frac{1}{2p_1^2} \quad \hat{\omega}_3 = 1 - 2\hat{\omega}_4 - 2\hat{\omega}_5 \\ \hat{\omega}_4 = \frac{3 - p_3^2}{2p_2^2(p_2^2 - p_3^2)} \quad \hat{\omega}_5 = \frac{3 - p_2^2}{2p_3^2(p_3^2 - p_2^2)}. \end{gathered} \tag{3.17}$$

Using these variables, the weights are as follows:

$$\begin{aligned} \omega_1^{(m)} &= \frac{(L-1)(L-2+L\hat{\omega}_1^2)}{2} - L(L-1)\hat{\omega}_1 + L\hat{\omega}_3 \\ \omega_i^{(m)} &= (L-1)\hat{\omega}_2(\hat{\omega}_1 - 1) \begin{cases} i = 2, \ldots, L+1 \\ i = L+2, \ldots, 2L+1 \end{cases} \\ \omega_i^{(m)} &= \hat{\omega}_4 \begin{cases} i = 2L+2, \ldots, 3L+1 \\ i = 3L+2, \ldots, 4L+1 \end{cases} \\ \omega_i^{(m)} &= \hat{\omega}_5 \begin{cases} i = 4L+2, \ldots, 5L+1 \\ i = 5L+2, \ldots, 6L+1 \end{cases} \\ \omega_i^{(m)} &= \hat{\omega}_2^2 \begin{cases} i = 6L+2, \ldots, 6L+1+C \\ i = 6L+2+C, \ldots, 6L+1+2C \\ i = 6L+2+2C, \ldots, 6L+1+3C \\ i = 6L+2+3C, \ldots, 6L+1+4C \end{cases} \end{aligned} \tag{3.18}$$

The same values are used for both $\omega_i^{(m)}$ and $\omega_i^{(c)}$.

3.1.5 Sigma point filter algorithm

An example algorithm for a sigma point filter is presented below, with the assumption, that the observed discrete-time system is of the form (3.3).

Table 3.1. Sigma point filters with different sigma point selection strategies

Name	Points	Parameters	Assumption on distribution	Non-negative weights
CKF	$2L$	0	Gaussian	yes
UKF	$2L+1$	3	Symmetric	yes*
RSPF	$L+2$	1	Symmetric	yes
CDKF	$2L+1$	1	Symmetric	yes*
GHQF	m^L**	1	Gaussian	yes
SGQF	$2L^2+4L+1$	3	Gaussian	no

* Depends on the choice of parameters.
** m is an adjustable parameter.

0. *Initialization*:

   (a) The initial estimation $\hat{\mathbf{x}}_0$ and $\mathbf{\Sigma}_0$ are chosen.

   (b) Initial sigma point set of N sigma points $\mathcal{X}_0$ is calculated from $\hat{\mathbf{x}}_0$ and the factors of $\mathbf{\Sigma}_0$, $\mathbf{Q}_0$ and $\mathbf{R}_1$ using a chosen sigma point selection strategy.

1. *Estimation*:
   $\mathcal{X}_k$ is known. It can be divided to $\begin{pmatrix} \mathcal{X}_k^{(x)T} & \mathcal{X}_k^{(w)T} & \mathcal{X}_k^{(v)T} \end{pmatrix}^T$ if the disturbance and the noise are embedded into the sigma points.

   (a) Propagate the sigma points through the nonlinear function of the system
   $$\mathcal{X}_{k+1}^{(\bar{x})} = \mathrm{f}(\mathcal{X}_k^{(x)}, \mathcal{X}_k^{(w)}) \tag{3.19}$$

   (b) The weighted mean of the propagated sigma points $\bar{\mathbf{x}}_{k+1}$ is the initial estimation of the state vector $\mathbf{x}_{k+1}$.
   If $\mathcal{X}_{k+1}^{(\bar{x})} = \{\xi_{k+1,1}^{(\bar{x})}, \ldots, \xi_{k+1,N}^{(\bar{x})}\}$, then the result is as follows:
   $$\bar{\mathbf{x}}_{k+1} = \sum_{i=1}^{N} \omega_i^{(m)} \xi_{k+1,i}^{(\bar{x})} \tag{3.20}$$

   (c) Calculate the covariance matrix of the state variables as follows:
   $$\begin{aligned} \mathbf{p}_{xx,i} &= (\xi_{k+1,i}^{(\bar{x})} - \bar{\mathbf{x}}_{k+1})(\xi_{k+1,i}^{(\bar{x})} - \bar{\mathbf{x}}_{k+1})^T \\ \mathbf{P}^{(xx)} &= \textstyle\sum_{i=1}^{N} \omega_i^{(c)} \mathbf{p}_{xx,i}. \end{aligned} \tag{3.21}$$

The formula is different in the case of CDKF:

$$\begin{aligned}
\mathbf{p}_{xx,i}^{(1)} &= (\xi_{k+1,i}^{(\bar{x})} - \xi_{k+1,j}^{(\bar{x})})(\xi_{k+1,i}^{(\bar{x})} - \xi_{k+1,j}^{(\bar{x})})^T \\
\mathbf{p}_{xx,i}^{(2)} &= (\xi_{k+1,i}^{(\bar{x})} + \xi_{k+1,j}^{(\bar{x})} - 2\xi_{k+1,1}^{(\bar{x})}) \\
&\quad (\xi_{k+1,i}^{(\bar{x})} + \xi_{k+1,j}^{(\bar{x})} - 2\xi_{k+1,1}^{(\bar{x})})^T \\
\mathbf{P}^{(xx)} &= \textstyle\sum_{i=2}^{(N-1)/2} (\omega_i^{(c1)} \mathbf{p}_{xx,i}^{(1)} + \omega_i^{(c2)} \mathbf{p}_{xx,i}^{(2)}),
\end{aligned} \tag{3.22}$$

where $j = i + (N-1)/2$.

(d) Propagate the transformed sigma points to the output:

$$\mathcal{X}_{k+1}^{(y)} = \mathrm{h}(\mathcal{X}_{k+1}^{(\bar{x})}, \mathcal{X}_k^{(v)}). \tag{3.23}$$

(e) The initial estimation of the measured output is the weighted mean of the propagated sigma points.
If $\mathcal{X}_{k+1}^{(y)} = \{\xi_{k+1,1}^{(y)}, \ldots, \xi_{k+1,N}^{(y)}\}$, then the result is as follows:

$$\bar{\mathbf{y}}_{k+1} = \sum_{i=1}^{N} \omega_i^{(m)} \xi_{k+1,i}^{(y)} \tag{3.24}$$

(f) Calculate the covariance matrix of the output and the cross-covariance between the output and the states as follows:

$$\begin{aligned}
\mathbf{p}_{yy,i} &= (\xi_{k+1,i}^{(y)} - \bar{y}_{k+1})(\xi_{k+1,i}^{(y)} - \bar{\mathbf{y}}_{k+1})^T \\
\mathbf{P}^{(yy)} &= \textstyle\sum_{i=1}^{N} \omega_i^{(c)} \mathbf{p}_{yy,i} \\
\mathbf{p}_{xy,i} &= (\xi_{k+1,i}^{(\bar{x})} - \bar{\mathbf{x}}_{k+1})(\xi_{k+1,i}^{(y)} - \bar{\mathbf{y}}_{k+1})^T \\
\mathbf{P}^{(xy)} &= \textstyle\sum_{i=1}^{N} \omega_i^{(c)} \mathbf{p}_{xy,i}.
\end{aligned} \tag{3.25}$$

Similarly to step 1c, the calculation is different for CDKF:

$$\begin{aligned}
\mathbf{p}_{yy,i}^{(1)} &= (\xi_{k+1,i}^{(y)} - \xi_{k+1,j}^{(y)})(\xi_{k+1,i}^{(y)} - \xi_{k+1,j}^{(y)})^T \\
\mathbf{p}_{yy,i}^{(2)} &= (\xi_{k+1,i}^{(y)} + \xi_{k+1,j}^{(y)} - 2\xi_{k+1,1}^{(y)}) \\
&\quad (\xi_{k+1,i}^{(y)} + \xi_{k+1,j}^{(y)} - 2\xi_{k+1,1}^{(y)})^T \\
\mathbf{P}^{(yy)} &= \textstyle\sum_{i=2}^{(N-1)/2} (\omega_i^{(c1)} \mathbf{p}_{yy,i}^{(1)} + \omega_i^{(c2)} \mathbf{p}_{yy,i}^{(2)}) \\
\mathbf{p}_{xy,i}^{(1)} &= (\xi_{k+1,i}^{(\bar{x})} - \xi_{k+1,j}^{(\bar{x})})(\xi_{k+1,i}^{(y)} - \xi_{k+1,j}^{(y)})^T \\
\mathbf{p}_{xy,i}^{(2)} &= (\xi_{k+1,i}^{(\bar{x})} + \xi_{k+1,j}^{(\bar{x})} - 2\xi_{k+1,1}^{(\bar{x})}) \\
&\quad (\xi_{k+1,i}^{(y)} + \xi_{k+1,j}^{(y)} - 2\xi_{k+1,1}^{(y)})^T \\
\mathbf{P}^{(xy)} &= \textstyle\sum_{i=2}^{(N-1)/2} (\omega_i^{(c1)} \mathbf{p}_{xy,i}^{(1)} + \omega_i^{(c2)} \mathbf{p}_{xy,i}^{(2)}).
\end{aligned} \tag{3.26}$$

Note, that the way $\mathbf{P}^{(xy)}$ is calculated is different from the algorithm presented by Nørgaard et al. [100].

2. *Update*:
The actual value of output $\mathbf{y}_{k+1}$ is available through measurement.

(a) Calculate the Kalman gain:

$$\mathbf{K}_{k+1} = \mathbf{P}^{(xy)} \left(\mathbf{P}^{(yy)} \right)^{-1}. \tag{3.27}$$

(b) Update the state vector and the estimation error covariance matrix as follows:

$$\begin{aligned} \hat{\mathbf{x}}_{k+1} &= \bar{\mathbf{x}}_{k+1} + \mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{k+1}) \\ \mathbf{\Sigma}_{k+1} &= \mathbf{P}^{(xx)} - \mathbf{K}_{k+1}\mathbf{P}^{(yy)}\mathbf{K}_{k+1}^{T}. \end{aligned} \tag{3.28}$$

(c) Calculate $\mathbf{\Sigma}_{k+1}^{\frac{1}{2}}$ using Cholesky decomposition.

(d) Define the new sigma point set $\mathcal{X}_{k+1}$ using the following mean and covariance matrix factor, with the chosen sigma point selection strategy:

$$\begin{pmatrix} \hat{\mathbf{x}}_{k+1} \\ \bar{\mathbf{w}}_{k+1} \\ \bar{\mathbf{v}}_{k+2} \end{pmatrix} \quad \begin{pmatrix} \mathbf{\Sigma}_{k+1}^{\frac{1}{2}} & 0 & 0 \\ 0 & \mathbf{Q}_{k+1}^{\frac{1}{2}} & 0 \\ 0 & 0 & \mathbf{R}_{k+2}^{\frac{1}{2}} \end{pmatrix} \tag{3.29}$$

3. $k = k + 1$ and continue from *Estimation*.

3.1.6 Extensive transform

Extensive transform [107] is an attempt to enhance the accuracy of sigma point filters by preserving the statistical independence of $\mathbf{\Sigma}_k$ from $\mathbf{Q}_k$ and $\mathbf{R}_k$ when approximating their joint predictive distribution. Estimation error can be separated from disturbances and measurement noise. Furthermore, it is possible to apply different sigma point selection strategies to each component as long as the sum of weights used to determine the covariance matrices equals one. The only exception is CDKF, where covariance matrices are calculated from sigma points differently. Let us have two sigma point sets χ_1 and χ_2 with n_1 and n_2 number of points, furthermore ω_1 and ω_2 weights associated with them. We can use the combined sigma point set χ with $n_1 \cdot n_2$ sigma points to acquire information on their joint distribution.

$$\begin{gathered} \chi \ni \xi_k = \begin{pmatrix} \xi_{1,i} \\ \xi_{2,j} \end{pmatrix} \quad \xi_{1,i} \in \chi_1 \quad \xi_{2,j} \in \chi_2 \\ w_k = w_{1,i} w_{2,j} \quad k = (n_2 - 1)i + j \\ i = 1, \ldots, n_1 \qquad j = 1, \ldots, n_2 \end{gathered} \tag{3.30}$$

3.1.7 Square-root filtering

In practical applications, the calculated estimation error covariance matrix $\mathbf{\Sigma}_{k+1}$ may not be strictly positive definite. Numerical inaccuracies can accumulate so that $\mathbf{\Sigma}_{k+1}$ is close to singularity or indefinite. The Cholesky decomposition can lead to poor numerical solutions, depending on the model and the number of state variables. Sigma point collapse [105] can also result in singular covariance matrices.

However, it is possible to provide a more robust and numerically better conditioned solution by avoiding the Cholesky decomposition altogether and calculating $\mathbf{\Sigma}_{k+1}^{\frac{1}{2}}$ directly. This approach is referred to as square-root filtering [108]. QR-decomposition is an essential tool for the square-root filtering algorithm. Let us introduce the notation $Triang$ for the transpose of the upper triangular result $\mathbf{\Gamma}$ of the QR decomposition (3.31):

$$\begin{aligned} \mathbf{\Gamma}^T &= Triang(\mathbf{A}) \\ \mathbf{\Theta\Gamma} &= QR(\mathbf{A}^T), \end{aligned} \tag{3.31}$$

where $\mathbf{A}$ is a real valued matrix. There are various versions of square-root filtering algorithms in the literature. The approach presented by Mark G. Rutten [108] requires disturbance and noise sources to be embedded into the sigma points. Furthermore, all weights $\omega^{(c)}$ need to be nonnegative.

The algorithm presented in Section 3.1.5 remains mostly unchanged except for 1c, 1f, 2a, 2b and 2c.

The square-root sigma point filter algorithm is as follows:

0. *Initialization*:

   (a) The initial estimation $\hat{\mathbf{x}}_0$ and $\mathbf{\Sigma}_0$ are chosen.

   (b) Initial sigma point set $\mathcal{X}_0$ is calculated from $\hat{\mathbf{x}}_0$ and the factors of $\mathbf{\Sigma}_0$, $\mathbf{Q}_0$ and $\mathbf{R}_0$ using a chosen sigma point selection strategy.

1. *Estimation*:
   $\mathcal{X}_k = \begin{pmatrix} \mathcal{X}_k^{(x)T} & \mathcal{X}_k^{(w)T} & \mathcal{X}_k^{(v)T} \end{pmatrix}^T$ is known.

   (a) Propagate the sigma points through the nonlinear function of the system:

   $$\mathcal{X}_{k+1}^{(\bar{x})} = \mathrm{f}(\mathcal{X}_k^{(x)}, \mathcal{X}_k^{(w)}) \qquad \mathcal{X}_{k+1}^{(\bar{x})} = \{\xi_{k+1,1}^{(\bar{x})}, \ldots, \xi_{k+1,N}^{(\bar{x})}\}. \tag{3.32}$$

   (b) Initial estimation of the state vector:

   $$\bar{\mathbf{x}}_{k+1} = \sum_{i=1}^{N} \omega_i^{(m)} \xi_{k+1,i}^{(\bar{x})}. \tag{3.33}$$

(c) Calculate the following:

$$\mathbf{P}^{(x)} = \begin{pmatrix} \sqrt{\omega_1^{(c1)}}(\xi_{k+1,2}^{(\bar{x})} - \xi_{k+1,L+2}^{(\bar{x})})^T \\ \vdots \\ \sqrt{\omega_L^{(c1)}}(\xi_{k+1,L+1}^{(\bar{x})} - \xi_{k+1,2L+1}^{(\bar{x})})^T \\ \sqrt{\omega_1^{(c2)}}(\xi_{k+1,2}^{(\bar{x})} + \xi_{k+1,L+2}^{(\bar{x})} - 2\xi_{k+1,1}^{(\bar{x})})^T \\ \vdots \\ \sqrt{\omega_L^{(c2)}}(\xi_{k+1,L+1}^{(\bar{x})} + \xi_{k+1,2L+1}^{(\bar{x})} - 2\xi_{k+1,1}^{(\bar{x})})^T \end{pmatrix}^T \tag{3.34}$$

for CDKF and

$$\mathbf{P}^{(x)} = \begin{pmatrix} \sqrt{\omega_1^{(c)}}(\xi_{k+1,1}^{(\bar{x})} - \bar{\mathbf{x}}_{k+1})^T \\ \vdots \\ \sqrt{\omega_N^{(c)}}(\xi_{k+1,N}^{(\bar{x})} - \bar{\mathbf{x}}_{k+1})^T \end{pmatrix}^T \tag{3.35}$$

for any of the other filters.

(d) Propagate the transformed sigma points to the output:

$$\mathcal{X}_{k+1}^{(y)} = \mathrm{h}(\mathcal{X}_{k+1}^{(\bar{x})}, \mathcal{X}_k^{(v)}) \qquad \mathcal{X}_{k+1}^{(y)} = \{\xi_{k+1,1}^{(y)}, \ldots, \xi_{k+1,N}^{(y)}\}. \tag{3.36}$$

.

(e) Calculate the initial estimation of the measured output:

$$\bar{\mathbf{y}}_{k+1} = \sum_{i=1}^{N} \omega_i^{(m)} \xi_{k+1,i}^{(y)}. \tag{3.37}$$

(f) Calculate the following for CDKF and for the other filters respectively:

$$\mathbf{P}^{(y)} = \begin{pmatrix} \sqrt{\omega_1^{(c1)}}(\xi_{k+1,2}^{(y)} - \xi_{k+1,L+2}^{(y)})^T \\ \vdots \\ \sqrt{\omega_L^{(c1)}}(\xi_{k+1,L+1}^{(y)} - \xi_{k+1,2L+1}^{(y)})^T \\ \sqrt{\omega_1^{(c2)}}(\xi_{k+1,2}^{(y)} + \xi_{k+1,L+2}^{(y)} - 2\xi_{k+1,1}^{(y)})^T \\ \vdots \\ \sqrt{\omega_L^{(c2)}}(\xi_{k+1,L+1}^{(y)} + \xi_{k+1,2L+1}^{(y)} - 2\xi_{k+1,1}^{(y)})^T \end{pmatrix}^T \tag{3.38}$$

$$\mathbf{P}^{(y)} = \begin{pmatrix} \sqrt{\omega_1^{(c)}}(\xi_{k+1,1}^{(y)} - \bar{\mathbf{y}}_{k+1})^T \\ \vdots \\ \sqrt{\omega_N^{(c)}}(\xi_{k+1,N}^{(y)} - \bar{\mathbf{y}}_{k+1})^T \end{pmatrix}^T \tag{3.39}$$

2. *Update*:
   The actual value of output $\mathbf{y}_{k+1}$ is available through measurement.

   (a) Calculate the factor $\mathbf{\Sigma}_{k+1}^{\frac{1}{2}}$ as follows:

$$\begin{pmatrix} \tilde{\mathbf{P}}^{(yy)} & 0 & 0 \\ \tilde{\mathbf{P}}_k^{(xy)} & \mathbf{\Sigma}_{k+1}^{\frac{1}{2}} & 0 \end{pmatrix} = Triang \begin{pmatrix} \mathbf{P}^{(y)} \\ \mathbf{P}^{(x)} \end{pmatrix}. \tag{3.40}$$

   (b) Calculate the Kalman gain:

$$\mathbf{K}_{k+1} = \tilde{\mathbf{P}}^{(xy)} \left(\tilde{\mathbf{P}}^{(yy)} \right)^{-1}. \tag{3.41}$$

   Note, that $\tilde{\mathbf{P}}^{(xy)}$ and $\tilde{\mathbf{P}}^{(yy)}$ are not equal to the covariance and cross-covariance matrices $\mathbf{P}^{(xy)}$ and $\mathbf{P}^{(yy)}$.

   (c) Update the state vector:

$$\hat{\mathbf{x}}_{k+1} = \bar{\mathbf{x}}_{k+1} + \mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{k+1}) \tag{3.42}$$

   (d) Define the new sigma point set $\mathcal{X}_{k+1}$ using $\mathbf{\Sigma}_{k+1}^{\frac{1}{2}}$ and the sigma point selection strategy.

3. $k = k + 1$ and continue from *Estimation*.

Some of the other notable updates of sigma point filters include Iterative filtering and $\mathcal{H}_\infty$ filtering. See Section A.2.1 and Section A.2.2 for further details.

3.2 State Estimation for T1DM Models

Sigma point filters are good candidates for state estimation of the T1DM models presented in Chapter 2 [109] [SBK16] [110]. These are nonlinear models with stochastic disturbances and measurement noise. However, there are certain aspects of T1DM models that must be addressed.

3.2.1 Preserving Nonnegativity

First, the state variables, inputs, and outputs of a T1DM model are all associated with strictly nonnegative physical quantities [111] [112]. One way to consider this in the state estimation is to use the natural logarithm of these variables, effectively representing them with the lognormal distribution. In this case, the covariance matrix of the estimation error is only valid for the transformed state vector. It is not strictly necessary to transform all variables, but it is enough to focus on

the nonlinear parts of the model and the states affected by disturbances instead. Let $\mathcal{T}_x$ denote the transformation of selected state variables to their natural logarithm: $\varkappa_{i,k} = \ln x_{i,k}$, where i is the index of a single state variable in the state vector $\mathbf{x}$ and k indicates discrete time. A similar transformation can be introduced for the disturbances denoted with $\mathcal{T}_w$. Meal intake and physical activity are likely candidates for this transformation, as both are strictly nonnegative. The transformed discrete time state space T1DM model (3.3) used during the estimation is as follows:

$$\begin{aligned} \varkappa_k &= \mathcal{T}_x \mathbf{x}_k \\ \varkappa_{k+1} &= \mathcal{T}_x \mathrm{f}(\mathcal{T}_x^{-1}\varkappa_k, \mathcal{T}_w^{-1}\mathbf{w}_k, k) \\ \mathbf{y}_k &= \mathrm{h}(\mathcal{T}_x^{-1}\varkappa_k, \mathbf{z}_k, k). \end{aligned} \tag{3.43}$$

The same method applies to the Extended Kalman and sigma point filters. In addition, potentially other nonlinear state estimation methods, such as particle filters and sliding mode observers, can benefit from this approach. Furthermore, this method is not limited to lognormal distribution. Other positive or nonnegative valued distributions may be applicable as well. Although state estimation can be improved this way, it is even more helpful in the long-term prediction of plasma glucose concentration.

Disturbance modeling is also peculiar in the state estimation of T1DM models, especially unannounced meal intake. The occurrence and the amount of carbohydrates ingested by the patient may be stochastic in nature, but Gaussian or lognormal distribution is a relatively poor model of the meal intake signal. However, suppose a dynamic model of the meal ingestion is available, and is part of the T1DM model used for state estimation. In that case, the estimated output of the meal ingestion model can be sufficiently accurate for control purposes.

If the chosen control method is realized via state feedback, certain state variables associated with meal ingestion may be inaccurate. This inaccuracy must be considered when choosing the model for controller design and synthesis. Section 3.3.2 provides more information regarding announced meal intake modeling.

3.2.2 Estimation of an uncertain model

As presented in Section 2.3, most T1DM models are inherently inaccurate. This work uses additive and multiplicative output uncertainty functions to capture this inaccuracy (Section 2.3). Let us use the following state-space representation for additive ($W_{out,a}$) and multiplicative ($W_{out,m}$) uncertainty functions respectively:

$$\begin{aligned} \mathbf{x}_{(out,a),k+1} &= \mathbf{A}_{out,a}\mathbf{x}_{(out,a),k} + \mathbf{B}_{out,a}d_{(out),k} \\ z_{(out,a),k} &= \mathbf{C}_{out,a}\mathbf{x}_{(out,a)_k} + D_{out,a}d_{(out),k} \end{aligned} \tag{3.44}$$

$$\begin{aligned} \mathbf{x}_{(out,m),k} &= \mathbf{A}_{out,m}\mathbf{x}_{(out,m),k} + \mathbf{B}_{out,m}y_k \\ z_{(out,m),k} &= \mathbf{C}_{out,m}\mathbf{x}_{(out,m),k} + D_{out,m}y_k, \end{aligned} \tag{3.45}$$

where $d_{(out)}$ is a white noise of standard normal distribution. The parameters of both functions can be derived from:

- the remainder error of a parameter identification task,
- a priori knowledge of intra-patient variability,
- known effects of model simplification,

among others. When performing state estimation, the uncertainty functions must be part of the model. $d_{(out)}$ is part of the disturbances. Furthermore, $\mathbf{x}_{(out,a)}$ and $\mathbf{x}_{(out,m)}$ are estimated like any other state variable of the T1DM model, which is beneficial for state feedback type of control algorithms.

While integrating additive uncertainty function to a state observer is relatively straightforward, a sigma point filter has no direct means to consider the Δ unstructured uncertainty (see Figure 2.2). To overcome this limitation, the additive and multiplicative uncertainty weighting function can be combined to calculate the measured output $y_{(meas)}$ as follows:

$$y_{(meas),k} = y_k + z_{(out,a),k}\left(1 + \frac{z_{(out,m),k}}{G_{add}}\right), \tag{3.46}$$

where G_{add} is the DC gain of the additive uncertainty weighting function. The combined additive and multiplicative output uncertainty is illustrated in Figure 3.3.

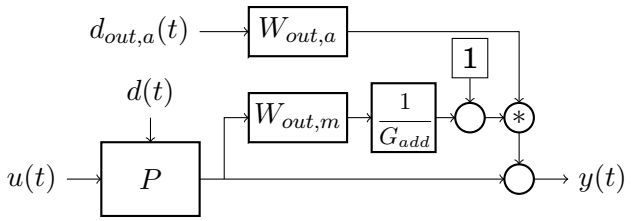

Figure 3.3. Model P extended with combined $W_{out,a}$ additive and $W_{out,m}$ multiplicative output uncertainty weighting functions for state estimation purposes.

The limitation of this approach is that it introduces a strong dependency between additive and multiplicative uncertainty. However, this may not be a disadvantage if the parameters of these weighting functions are derived from the same source, which is not uncommon in practice. Alternatively, introducing a new weighting function similar

to $W_{out,a}$, and combining it with $W_{out,m}$ instead can eliminate the dependency at the cost of increasing model and disturbance dimensions. Figure 3.4 illustrates this case.

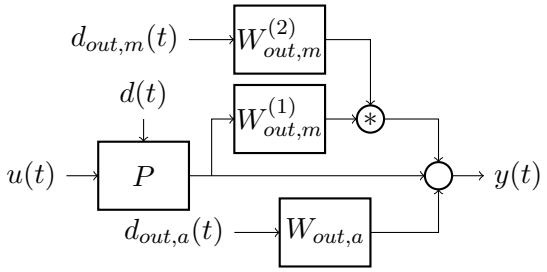

Figure 3.4. Model P extended with additive and multiplicative output uncertainty weighting functions for state estimation purposes. $W^{(1)}_{out,m}$ is the original multiplicative uncertainty weighting function, while $W^{(2)}_{out,m}$ is a low pass filter driven by white noise.

3.3 Prediction

Kalman filters and sigma point filters can also be used for model-based prediction. The *Estimation* step in the previously presented algorithms is already a model-based prediction, which is then corrected in the *Update* step based on the difference between the predicted and measured output. Thus, a Kalman filter-based predictor merely needs to repeatedly perform the *Estimation* step. As an additional benefit, this kind of predictor can also assess its accuracy of state variable and output prediction via $\mathbf{P}^{(xx)}$ and $\mathbf{P}^{(yy)}$ respectively. The progression of these covariance matrices can tell how far ahead the algorithm can predict so that the predicted values are still useful.

Using the same *Estimation* step for prediction is sufficient in open loop control or when the controller in a closed-loop configuration does not use the estimated values of the state observer. However, if the controller uses some of the estimated values of the state observer, the predictor should consider how future estimation errors can influence the system. In this case, the control algorithm is part of the model used by the predictor:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathrm{f}(\mathbf{x}_k, \mathbf{w}_k, \mathbf{u}_k, k) \\ \mathbf{x}_{(c),k+1} &= \mathrm{f_c}(\mathbf{x}_{(c),k}, (\mathbf{x}_k + \mathbf{w}_{(e),k}), k) \\ \mathbf{u}_k &= \mathrm{h}_c(\mathbf{x}_{(c),k}, (\mathbf{x}_k + \mathbf{w}_{(e),k}), k) \\ \mathbf{y}_k &= \mathrm{h}(\mathbf{x}_k, \mathbf{v}_k, \mathbf{u}_k, k). \end{aligned} \tag{3.47}$$

In this combined system of model and controller, $\mathbf{u}$ denotes the con-

trol signal, which is a stochastic signal due to the influence of the state estimation error $\mathbf{w}_{(e)}$. Functions f_c and h_c represent the control algorithm. If the controller has a dynamic behavior, it necessitates the introduction of new state variables, denoted with $\mathbf{x}_{(c)}$. $\mathbf{x}_{(c)}$ is unnecessary for static algorithms, such as state feedback control. Note that although the estimated state vector is the input argument of both f_c and h_c, the controller does not necessarily use all of them. As an example, if the controller uses the estimated output of the model, the operation $\mathrm{h}((\mathbf{x}_k + \mathbf{w}_{(e),k}), 0, \mathbf{u}_k, k)$ is incorporated into $\mathrm{f_c}$ and h_c.

$\mathbf{w}_{(e)}$ state estimation error is assumed to have zero mean value, but the predictor needs to estimate its covariance matrix $\mathbf{\Sigma}_{(e)}$ and the Cholesky factor $\mathbf{\Sigma}_{(e)}^{\frac{1}{2}}$. A modified state observer running parallel to the predictor can provide this information [SBK16].

In each prediction step, the modified observer performs *Estimation* and *Update* using only the nominal model (3.3). However, since there is no actual measurement available, both *Estimation* and *Update* can omit to calculate the following values:

- initial estimation of the measured output: $\bar{\mathbf{y}}$,
- Kalman gain: $\mathbf{K}$,
- updated state vector: $\hat{\mathbf{x}}$.

The only relevant output of the simplified observer is the updated factor of the estimation error covariance matrix: $\mathbf{\Sigma}_{(e)}^{\frac{1}{2}}$. Furthermore, to decouple the calculation from the prediction, the control signal $\mathbf{u}$ is substituted with its mean value, denoted with $\bar{\mathbf{u}}$.

Let us introduce the following notations for the predictor to differentiate it from the observer.

- The lower index j denotes j-th prediction step. Therefore, the j-th predicted value starting from the k-th estimation step has the lower index $k + j$, e.g., x_{k+j}.
- While the estimation of a variable x is denoted as $\hat{x}$, prediction of x is indicated with $\breve{x}$.

3.3.1 Predictor Algorithm

The sigma point filter-based predictor algorithm is as follows:

0. *Initialization*:

(a) At the end of the k-th estimation step the predictor starts from the estimated state variable vector and the state variables of the controller (if applicable):

$$\begin{aligned} \breve{\mathbf{x}}_{k+0} &= \hat{\mathbf{x}}_k \\ \breve{\mathbf{x}}_{(c),k+0} &= \mathbf{x}_{(c),k} \end{aligned} \tag{3.48}$$

(b) Set the initial mean and sigma points for the control signal:

$$\begin{aligned} \bar{\mathbf{u}}_{k+0} &= \mathrm{h}_c(\mathbf{x}_{(c),k}, \hat{\mathbf{x}}_k, k) \\ \mathcal{X}^{(u)}_{k+0} &= \{\bar{\mathbf{u}}, \ldots, \bar{\mathbf{u}}\} \end{aligned} \tag{3.49}$$

(c) Both estimation error and prediction error covariance matrix factor start from the same value:

$$\mathbf{\Sigma}^{\frac{1}{2}}_{(p),k+0} = \mathbf{\Sigma}^{\frac{1}{2}}_{(e),k+0} = \mathbf{\Sigma}^{\frac{1}{2}}_{k} \tag{3.50}$$

(d) Expand $\mathcal{X}_k = \begin{pmatrix} \mathcal{X}^{(x)T}_k & \mathcal{X}^{(w)T}_k & \mathcal{X}^{(z)T}_k \end{pmatrix}^T$ with the state variables of the control algorithm (if applicable). However, since these states are known at the beginning, the number of sigma points should not be increased and $\mathbf{\Sigma}^{\frac{1}{2}}_{(e),k+0}$ should not be used in the initialization.

$$\mathcal{X}_{(p),k+0} = \begin{pmatrix} \mathcal{X}^{(x)}_k \\ \left(\mathbf{x}_{(c),k}, \ldots, \mathbf{x}_{(c),k}\right) \\ \mathcal{X}^{(w)}_k \\ \mathcal{X}^{(z)}_k \end{pmatrix} \tag{3.51}$$

1. *Prediction*:
$\mathcal{X}_{(p),k+j} = \begin{pmatrix} \mathcal{X}^{(x)T}_{(p),k+j} & \mathcal{X}^{(w)T}_{(p),k+j} & \mathcal{X}^{(z)T}_{(p),k+j} \end{pmatrix}^T$ and $\mathcal{X}^{(u)}_{k+j}$ sigma point sets are available after the j-th prediction step. Note, that $\mathcal{X}^{(x)T}_{(p),k+j}$ incorporates both model and controller state variables, (x and $\mathbf{x}_{(c)}$) while $\mathcal{X}^{(w)T}_{(p),k+j}$ accounts for both disturbances and state estimation error (w and $\mathbf{w}_{(e)}$). The only exception is $j = 0$, where $\mathbf{w}_{(e)}$ is not present.

   (a) Propagate the sigma points through the model and controller:

$$\begin{aligned} \mathcal{X}^{(\bar{x})}_{(p),k+j+1} = & \begin{pmatrix} \mathrm{f}(\mathcal{X}^{(x)}_{(p),k+j}, \mathcal{X}^{(w)}_{(p),k+j}, \mathcal{X}^{(u)}_{k+j}) \\ \mathrm{f}_c(\mathcal{X}^{(x)}_{(p),k+j}, \mathcal{X}^{(w)}_{(p),k+j}) \end{pmatrix} = \\ & = \{\xi^{(\bar{x})}_{k+j+1,1}, \ldots, \xi^{(\bar{x})}_{k+j+1,N}\}. \end{aligned} \tag{3.52}$$

(b) Calculate the $j+1$-th predicted value of the combined vector of $\breve{\mathbf{x}}_{k+j+1}$ model and $\breve{\mathbf{x}}_{(c),k+j+1}$ controller state variables:

$$\begin{pmatrix} \breve{\mathbf{x}}_{k+j+1} \\ \breve{\mathbf{x}}_{(c),k+j+1} \end{pmatrix} = \sum_{i=1}^{N} \omega_i^{(m)} \xi_{k+j+1,i}^{(\bar{x})}. \tag{3.53}$$

(c) For a regular predictor, calculate $\mathbf{P}^{(xx)}$ the same way as in the *Estimation* step of a sigma point filter. The new prediction error covariance matrix factor $\mathbf{\Sigma}_{(p),k+j+1}^{\frac{1}{2}}$ is the Cholesky factor of $\mathbf{P}^{(xx)}$.

Alternatively, calculate $\mathbf{P}^{(x)}$ the same way as for a square root filter. Then, use QR decomposition to calculate $\mathbf{\Sigma}_{(p),k+j+1}^{\frac{1}{2}}$:

$$\mathbf{\Sigma}_{(p),k+j+1}^{\frac{1}{2}} = Triang(\mathbf{P}^{(x)}) \tag{3.54}$$

(d) For output prediction, propagate the transformed sigma points:

$$\begin{aligned} \mathcal{X}_{(p),k+j+1}^{(y)} = \quad & \mathrm{h}(\mathcal{X}_{(p),k+j+1}^{(\bar{x})}, \mathcal{X}_{(p),k+j+1}^{(z)}, \mathcal{X}_{k+j}^{(u)}) = \\ & = \{\xi_{k+j+1,1}^{(y)}, \ldots, \xi_{k+j+1,N}^{(y)}\}. \end{aligned} \tag{3.55}$$

(e) Calculate the predicted value of the measured output:

$$\breve{\mathbf{y}}_{k+j+1} = \sum_{i=1}^{N} \omega_i^{(m)} \xi_{k+j+1,i}^{(y)}. \tag{3.56}$$

(f) Calculate the covariance matrix of the predicted output the same way as $\mathbf{P}^{(yy)}$ in a sigma point filter.

2. *Estimation error update*:

(a) Using the sigma point set $\mathcal{X}_{k+j}^{(u)} = \{\xi_{k+j,1}^{(u)}, \ldots, \xi_{k+j,N}^{(u)}\}$, calculate the mean value of the control signal:

$$\bar{\mathbf{u}}_{k+j} = \sum_{i=1}^{N} \omega_i^{(m)} \xi_{k+j,i}^{(u)}. \tag{3.57}$$

(b) Perform all steps of *Estimation* the same way as in a sigma point filter using $\mathcal{X}_{(e),k+j}$ and $\bar{\mathbf{u}}_{k+j}$.

(c) Use $\mathbf{P}^{(xx)}$, $\mathbf{P}^{(xy)}$ and $\mathbf{P}^{(yy)}$ calculated in the previous step to acquire the new estimation error covariance matrix, and its Cholesky factor:

$$\begin{aligned}\boldsymbol{\Sigma}_{(e),k+j+1} &= \mathbf{P}^{(xx)} - \mathbf{P}^{(xy)}\Big(\mathbf{P}^{(yy)}\Big)^{-1}\mathbf{P}^{(xy)T} \\ \boldsymbol{\Sigma}^{\frac{1}{2}}_{(e),k+j+1} &= Chol\Big(\boldsymbol{\Sigma}_{(e),k+j+1}\Big).\end{aligned} \tag{3.58}$$

Alternatively, use $\mathbf{P}^{(x)}$ and $\mathbf{P}^{(y)}$ to calculate $\boldsymbol{\Sigma}^{\frac{1}{2}}_{(e),k+j+1}$ directly:

$$\begin{pmatrix} * & 0 & 0 \\ * & \boldsymbol{\Sigma}^{\frac{1}{2}}_{(e),k+j+1} & 0 \end{pmatrix} = Triang \begin{pmatrix} \mathbf{P}^{(y)} \\ \mathbf{P}^{(x)} \end{pmatrix}. \tag{3.59}$$

(d) Calculate the new $\mathcal{X}_{(p),k+j+1}$ sigma point set for the *Prediction* with the chosen sigma point selections strategy from the vector $\begin{pmatrix} \breve{\mathbf{x}}^T_{k+j+1} & \bar{\mathbf{w}}^T_{k+j+1} & \bar{\mathbf{v}}^T_{k+j+2} & 0^T \end{pmatrix}^T$ and covariance matrix factor: $\operatorname{diag}\Big(\boldsymbol{\Sigma}^{\frac{1}{2}}_{(p),k+j+1}, \mathbf{Q}^{\frac{1}{2}}_{k+j+1}, \mathbf{R}^{\frac{1}{2}}_{k+j+2}, \boldsymbol{\Sigma}^{\frac{1}{2}}_{(e),k+j+1}\Big)$.

(e) Use the new sigma point set $\mathcal{X}_{(p),k+j+1}$ to define the next sigma point set for the control signal using the control algorithm:

$$\mathcal{X}^{(u)}_{(p),k+j+1} = \mathrm{h}_c(\mathcal{X}_{(p),k+j+1}) \tag{3.60}$$

(f) Calculate the new sigma point $\boldsymbol{\Sigma}^{\frac{1}{2}}_{(e),k+j+1}$ using the vector $\begin{pmatrix} \breve{\mathbf{x}}^T_{k+j+1} & \bar{\mathbf{w}}^T_{k+j+1}c & \bar{\mathbf{v}}^T_{k+j+2} \end{pmatrix}^T$ and covariance matrix factor: $\operatorname{diag}\Big(\boldsymbol{\Sigma}^{\frac{1}{2}}_{(e),k+j+1}, \mathbf{Q}^{\frac{1}{2}}_{k+j+1}, \mathbf{R}^{\frac{1}{2}}_{k+j+2}\Big)$, for the *Estimation error update* with the chosen sigma point selection strategy.

3. $j = j + 1$ and continue from *Prediction*.

3.3.2 Meal Intake Announcement

Using an unannounced meal intake model for the predictor and observer provides the benefit of assessing the glucose control's expected quality, e.g., the likelihood of hypoglycemia. Because there is no a priori information regarding meal intakes, one must assume that the glucose flux resulting from ingestion has high variability. Thus, the predictor can only provide a relatively large standard deviation for the predicted glucose concentration [SBK16]. This is usually unsuited for event detection. However, if the meal intakes are announced, the predictor can be more versatile.

First, the predictor needs a different meal intake model. Let us denote the disturbance signal associated with meal intake as w_{meal}. The parameters used for w_{meal} are usually constant when meals are unannounced, with values that cover a wide range of inputs. For announced meals, the parameters should be a function of time, capturing the estimated amount of CHO in the meal and uncertainties both in amount and timing. Two potential meal intake functions are illustrated in Figure 3.5, showing the $\bar{w}_{meal}(t)$ mean value and $\sigma_{w_{meal}}(t)$ standard deviation of stochastic signal $w_{meal}(t)$ over time. More advanced models can consider CHO content and other selected nutritional factors as well [113].

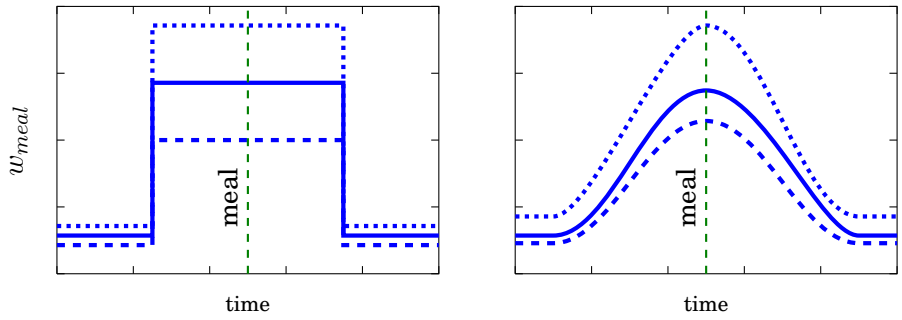

Figure 3.5. Illustration of possible meal intake models. The blue lines display $\bar{w}_{meal}(t)$ mean value and $\sigma_{w_{meal}}(t)$ standard deviation of the stochastic variable $w_{meal}(t)$. Solid line represent the former, while dashed and dotted line the latter. The vertical green dashed lines mark a single meal intake announcement in each plot. The plot on the left is an example where the meal intake is modeled by a time-varying normal distribution, while it has lognormal distribution in the plot on the right.

When validating a meal announcement, at least two predictions are necessary: one with parameters from the announcement and one assuming no meal intake occurred. Then, as more and more glucose measurements become available, a separate logic can assess which predictions have a higher likelihood. The decision can be made based on e.g., fixed thresholds, statistical techniques [114, 115] or soft computing methods [116].

3.3.3 Fault and Event Detection

Unannounced meals and unexpected events can be detected in a rather similar manner. However, in this case, there is no direct trigger for executing the prediction. The AP may perform a new prediction and test for anomalies using an earlier one for every new measurement data. However, this can be rather resource intensive and impractical for embedded

hardware. Alternatively, it is possible to combine the predictor with a supervisor logic and perform prediction only when necessary. Predictions validate or reject hypotheses regarding the event or fault in these cases.

For example, suppose the measured glucose levels have consistently low likelihood based on $\mathbf{P}^{(yy)}$ calculated by the state observer. In that case, the supervisor logic can trigger multiple predictions from an earlier point in time. Some of the use cases the predictor may be able to cover are as follows:

- there was an unannounced meal intake,
- there is ongoing physical activity,
- there is a sensor fault with some characteristic signature in data,
- the pump provides no insulin.

The specificity of the changes introduced by these events defines the applicability of the predictor. The supervisor logic executes predictions associated with the relevant events above and compares them with the hypothesis that "no event occurred".

3.4 Simulation

Chapter 5 presents in silico validation of various observer algorithms with details of the corresponding simulation environment and setup. To illustrate the behavior of the observer, Figure 3.6 displays an example of a single 12 hours simulation. The simulation setup is identical to the one presented in Section 5.1, which includes the parameters of the Cambridge model that represents the ground truth, model reduction, parameters of the uncertainty functions, parameters of sigma point selection strategy,initial estimation error, and disturbance and noise parameters. Both the reference model and observer used the parameter set of patient no. 1 as presented in Table 5.1. The meal intake is based on *Scenario 3*, presented in Section 5.3.

The observer is based on the Unscented Kalman filter. In addition, it applies additive and multiplicative uncertainty models, lognormal transform, and square-root filtering. During the simulation, a state feedback controller uses the estimated state variables to define the insulin input. The controller is the hybrid $\mathcal{H}_2/\mathcal{H}_\infty$ controller referenced in Section 5.3.4.

Figure 3.6 shows the actual and estimated values of the Cambridge model, except for the meal intake and ingestion subsystem, where only

the output G_2 is relevant. The subplot displaying the estimated subcutaneous glucose measurement also illustrates the output uncertainty considered by the observer with cyan patches. The last two subplots display the outputs of the multiplicative and additive uncertainty weighting functions, respectively.

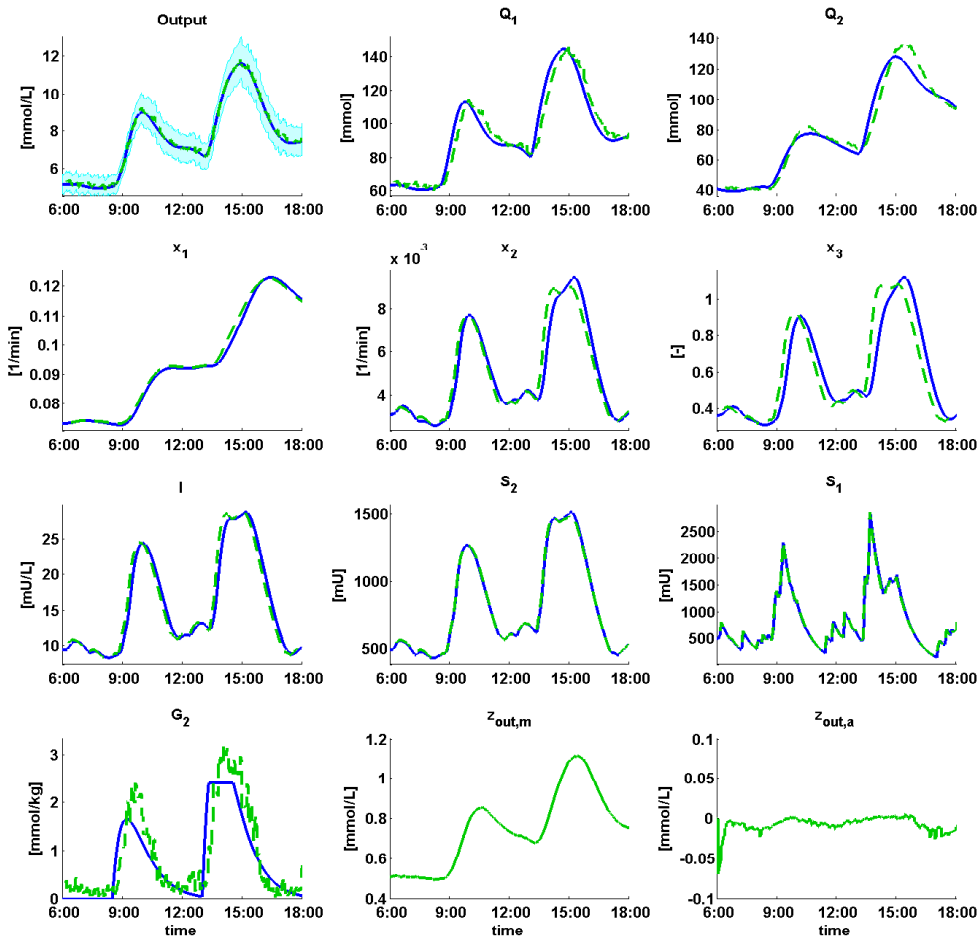

Figure 3.6. Illustration of an UKF based T1DM state observer. Blue solid and green dashed lines represent the actual and estimated value of the observed quantities indicated on the top of each subplot, respectively. The first subplot visualizes the output uncertainty that the observer considers with cyan patches. The last two subplots display the output of the multiplicative and additive output uncertainty functions, respectively, with solid green lines.

The effect of model simplification is visible in the deviation of the actual and estimated values of state variables x_2, x_3, and I. The estimated glucose flux (G_2) shows some delay, which is then compensated with larger estimated values shortly after each peak of the actual G_2 signal.

Figure 3.7 displays the simulation results of the same observer and controller as before but in predictor mode. The figure visualizes a single 180-minute-long prediction following a meal intake announcement. The predictor considers ± 10 g of uncertainty in the amount of CHO intake and ± 10 minutes uncertainty in timing. The meal model is as illustrated in the left subplot of Figure 3.5.

The predicted mean value of the subcutaneous glucose concentration, marked with a dashed green line, roughly follows the actual glucose concentration, marked with a solid blue line. The predicted standard deviation increases along the prediction horizon, which is indicated by cyan patches.

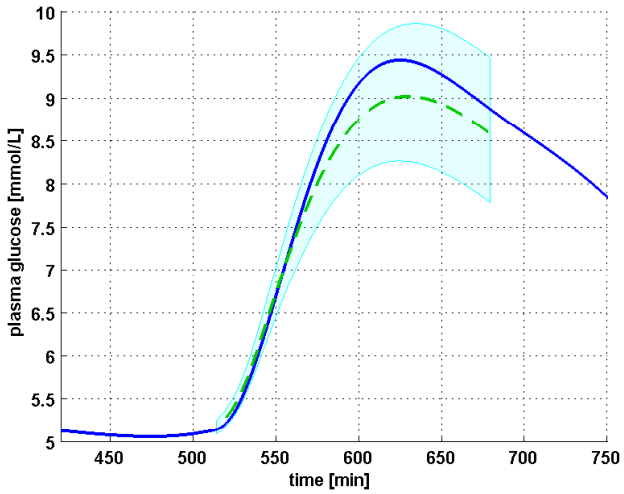

Figure 3.7. Simulation of a 180 minutes prediction horizon following a meal intake announcement. The solid blue line represents the actual subcutaneous glucose concentration. The green dashed line and cyan patches illustrate the predicted mean and standard deviation of the glucose concentration.

4

LINEAR PARAMETER-VARYING MODELING AND CONTROL

This chapter addresses the second thesis group: a robust nonlinear control algorithm for Artificial Pancreas. The controller addresses the nonlinearity, the nonnegativity, and the intra-patient variability of the glucose-insulin interaction in a T1DM patient. The contributions that belong to this thesis group are the following:

1. A linear parameter-varying approximation of the well-known Cambridge T1DM model enabled the synthesis of quasi linear parameter varying controllers via the solution of linear matrix inequalities.

2. A $\mathcal{H}_2/\mathcal{H}_\infty$ controller for Artificial Pancreas, which considers model uncertainties and intra-patient variability, as well as constraints on the control signal.

3. A method to automatically scale performance and multiplicative uncertainty outputs of a T1DM model so that robust stability constraints are ensured, and nominal performance is optimized, as long as they are feasible.

The related publications are: [KSZ11, KSF$^+$11, KTSS12, KS12, KSF$^+$12, KKSE13, SEK$^+$13, KSAB13, KSS$^+$13, KKS$^+$14, SEK14, KS16, SDKew].

4.1 Requirements of Blood Glucose Control

As discussed in Section 1.2.1, the design of the control algorithm of the Artificial Pancreas must consider various expectations and constraints. The controller presented in this Chapter addresses the following requirements:

- The controller shall be model-based, using the Cambridge model as its core.
- The controller shall consider the nonlinearity of the model.
- The controller shall consider the uncertainty resulting from modeling inaccuracies, time-varying parameters, and model simplifications.
- The controller shall keep the output in the range of normoglycemia.
- The controller shall reduce the time spent above the normal range (hyperglycemia), and avoid going below the range (hypoglycemia).
- The controller shall not request a negative control signal.
- The controller shall not rely on external information regarding disturbances (e.g., meal intake).

4.1.1 Literature Overview

The proposed controller relies only on the hardware most frequently used in clinical practice: a continuous subcutaneous glucose sensor and continuous subcutaneous insulin infusion. Although the use of intraperitoneal insulin infusion proposed by Barnes and Jones [24] or an additional glucagon pump as presented by El-Khatib et al. [81] and partially by Zhu et al. [30] can be highly efficient, they require hardware that currently has practical limitations. Furthermore, there is no reliance on patient intervention, as in the solution proposed by Beneyto et al. [27].

The proposed control signal addresses the nonnegativity of the insulin input but not as a hard constraint like the positive sliding mode control presented by Leyva et al. [28]. Moreover, it is a nonlinear technique, similarly to the nonlinear state feedback of Cai et al. [29] and the robust nonlinear feedback of Rigatos et al. [48].

There are two main approaches to handling the changes in the system dynamics over time: adaptivity and robustness. Shi et al. [34] and Kovács et al. [35] present notable recent examples, while Turksoy et al. [33] provide an overview of adaptive control for T1DM. Moreover, Iman

et al. [42] present an adaptive MPC controller, while the work of Abuin et al. introduces a circadian insulin sensitivity scheme [41].

However, the controller presented in this thesis group applies robustness to address intra-patient variability. One of the first solutions employing $\mathcal{H}_\infty$-norm based robust control for T1DM is presented by Doyle et al. [47]. A more recent example extends the $\mathcal{H}_\infty$-based robustness to nonlinear control [48]. The approach presented by Mirzaee et al. [50] uses a robust LPV controller, but partially relies on meal intake announcements. The controller proposed in this work requires no meal intake information.

The most widely accepted approach is model predictive control [36, 37, 38]. However, based on the results obtained from clinical trials, there is a need to address the robustness of these methods. Paoletti et al. [45] increase the robustness of an MPC by deriving uncertainty sets capturing the distribution of meal intake history and patient habits, hence eliminating the need for meal announcements. Siket et al. [46] addresses parameter uncertainties with particle swarm optimization.

4.2 Methods

4.2.1 Linear Parameter-Varying Model

Models of physiological systems are often expressed in state space form, in which the state variables represent a physical quantity or concentration, as presented in Chapter 2. For a linear system expressed in state space form, there is a wide range of controller synthesis methods that use the state space matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ as input. These methods include robust control, which could address the severe parameter and modeling uncertainties that burden T1DM models. However, the vast majority of T1DM models have a varying degree of nonlinearity. Hence, one cannot apply linear control synthesis methods directly. However, linear parameter-varying (LPV) models can bridge the gap between the nonlinearity of the T1DM models and the powerful techniques for linear state space systems [117].

A dynamic, continuous time state space LPV model with n states, m inputs and r outputs is as follows:

$$\begin{aligned} \dot{\mathrm{x}}(t) &= \mathbf{A}(\rho(t))\mathrm{x}(t) + \mathbf{B}(\rho(t))\mathrm{u}(t) \\ \mathrm{y}(t) &= \mathbf{C}(\rho(t))\mathrm{x}(t) + \mathbf{D}(\rho(t))\mathrm{u}(t) \end{aligned} \tag{4.1}$$

$$\begin{aligned} \mathbf{A}(\rho(t)) &= \mathbf{A}_0 + \textstyle\sum_{i=1}^{p} \rho_i(t)\mathbf{A}_i \quad \mathbf{B}(\rho(t)) = \mathbf{B}_0 + \textstyle\sum_{i=1}^{p} \rho_i(t)\mathbf{B}_i \\ \mathbf{C}(\rho(t)) &= \mathbf{C}_0 + \textstyle\sum_{i=1}^{p} \rho_i(t)\mathbf{C}_i \quad \mathbf{D}(\rho(t)) = \mathbf{D}_0 + \textstyle\sum_{i=1}^{p} \rho_i(t)\mathbf{D}_i \end{aligned}$$

where $x(t) \in \mathbb{R}^n$ is the vector of state variables, $u(t) \in \mathbb{R}^m$ is the vector

of inputs and $y(t) \in \mathbb{R}^r$ is the vector of outputs. $\mathbf{A}_i$, $\mathbf{B}_i$, $\mathbf{C}_i$, and $\mathbf{D}_i$ are $n \times n$, $n \times m$, $r \times n$, and $r \times m$ real matrices respectively for all $i \in \{1, ..., p\}$.

The vector of scheduling variables $\rho(t) \in \mathbb{R}^p$ represents the time-varying changes in the system. If (4.1) is an LPV model, the scheduling variables must satisfy the following requirements:

1. The scheduling variables must be bounded, and the bound is known: $\rho_{i,min} \leq \rho_i(t) \leq \rho_{i,max}, \forall i \in \{1, ..., p\}$

2. Each scheduling variable is continuously differentiable: $\rho_i(t) \in \mathcal{C}^1(\mathbb{R}, \mathbb{R}), \forall i \in \{1, ..., p\}$

3. The time derivate of the scheduling variables is also bounded: $d\rho_{i,min} \leq \dot{\rho}_i(t) \leq d\rho_{i,max}, \forall i \in \{1, ..., p\}$

The majority of T1DM models, that are defined in continuous time and use integer order nonlinear differential equations, can be transformed into a linear parameter-varying model [118] (4.1). This includes the following models referenced in Chapter 2:

- Both the minimal [71] and the extended [72] Bergman model [Kovács et al., 2011b].
- Most models used in intensive care [Kovács et al., 2011b].
- The Sorensen model [KSAB13].
- The Magni model [118].
- The Cambridge model [Kovács et al., 2011] [SEK+13] [SEK14].

It is important to note that if the state variables of T1DM models represent physical quantities and concentrations in a physiological system, they all meet the requirements for scheduling variables. Hence, any state variable of a T1DM system can be used as a scheduling variable in an LPV representation.

The rest of this Section will propose LPV representation of the Cambridge models presented in Section 2.1.1 with the intent to use them in robust linear and LPV controller synthesis. The scheduling variables in these LPV models aim to capture the nonlinear aspects only without addressing parametric uncertainties. Furthermore, a simplified linear worst-case meal intake replaces the meal absorption (2.4). Section A.3.1 presents a similar LPV representation of the Magni model.

4.2.2 Robust Control of Uncertain Model

The nonlinearity of a T1DM model is not the only aspect to consider for realizing AP. As discussed in Section 2.3 there are uncertainties and disturbances to any T1DM model.

Let P denote a linear MIMO model, $\mathrm{d}_P(t)$ a vector of disturbances, while $\mathrm{u}(t)$ control signals coming from K linear controller. Let $\mathrm{z}_P(t)$ a vector of outputs that indicate control performance, such as tracking error, while $\mathrm{y}(t)$ is the vector of measured outputs used for control. As referenced in Section 2.3.1, Δ is unstructured uncertainty, with $\mathrm{z}_\Delta(t)$ input and $\mathrm{d}_\Delta(t)$ output. The closed-loop control of the uncertain linear model can be arranged into a ΔPK structure [119] as shown in Figure 4.1.

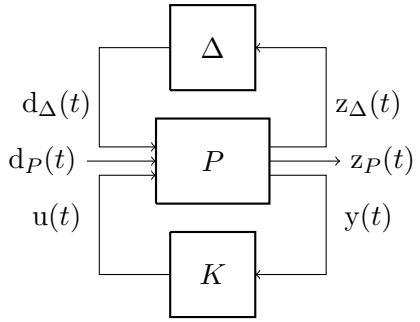

Figure 4.1. ΔPK structure of a linear model.

$\mathrm{z}_P(t)$ and $\mathrm{z}_\Delta(t)$, as well as $\mathrm{d}_P(t)$ and $\mathrm{d}_\Delta(t)$ can be grouped into a single vector as follows:

$$\mathrm{d}(t) = \begin{pmatrix} \mathrm{d}_\Delta(t) \\ \mathrm{d}_P(t) \end{pmatrix} \quad \mathrm{z}(t) = \begin{pmatrix} \mathrm{z}_\Delta(t) \\ \mathrm{z}_P(t) \end{pmatrix} \tag{4.2}$$

The controller K ensures Robust Stability (RS) if $\mathbf{M}(j\omega)$, the combined transfer matrix of P and K with $\mathrm{d}(t)$ input and $\mathrm{z}_\Delta(t)$ has only poles that have negative real value, and its $\mathcal{H}_\infty$ norm is smaller than one [119]:

$$\|\mathbf{M}(j\omega)\|_\infty = \sup_{\omega \in \mathbb{R}} \bar{\sigma}(\mathbf{M}(j\omega)) < 1, \tag{4.3}$$

where $\bar{\sigma}$ is the largest singular value of a matrix.

Similarly, K ensures nominal performance (NP) for a predefined positive γ value if the transfer from $\mathrm{d}(t)$ input to $\mathrm{z}_P(t)$ output meets a chosen norm constraint $\|.\|_p$ so that

$$\|\mathbf{M}(j\omega)\|_p < \gamma. \tag{4.4}$$

Some of the most commonly used norms based on transfer matrix for performance are $\mathcal{H}_\infty$ and $\mathcal{H}_2$ norms [117], from which the latter is calculated as follows:

$$\|\mathbf{M}(j\omega)\|_2 = \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty} Trace\big(\mathbf{M}(j\omega)\mathbf{M}^*(j\omega)\big)d\omega}, \tag{4.5}$$

Note, that $\|\mathbf{M}(j\omega)\|_2$ is only finite if $\mathrm{M}(j\omega)$ has no direct feedthrough.

Furthermore, there are two other nominal performance norms presented in [117]: generalized $\mathcal{H}_2$ ($\mathcal{H}_{2g}$) norm and $\mathcal{L}_1$ or peak-to-peak norm. They are defined as follows, for a given positive γ value:

$$\|\mathbf{M}(j\omega)\|_{2,g} = \frac{\|\mathrm{z}(t)\|_\infty}{\|\mathrm{d}(t)\|_2} < \gamma, \tag{4.6}$$

$$\|\mathbf{M}(j\omega)\|_1 = \frac{\|\mathrm{z}(t)\|_\infty}{\|\mathrm{d}(t)\|_\infty} < \gamma. \tag{4.7}$$

Both $\mathrm{d} : \mathbb{R} \to \mathbb{R}^p$ and $\mathrm{z} : \mathbb{R} \to \mathbb{R}^q$ are Lebesgue-measurable functions, and their norms are calculated as follows:

$$\|\mathrm{d}\|_\infty = \max_{i=1,\ldots,p} \sup_{t\geq 0} |d_i(t)|, \tag{4.8}$$

$$\|\mathrm{z}\|_\infty = \max_{j=1,\ldots,q} \sup_{t\geq 0} |z_j(t)|, \tag{4.9}$$

$$\|\mathrm{d}\|_2 = \int_0^\infty \mathrm{d}(t)^T \mathrm{d}(t) dt. \tag{4.10}$$

If a controller can ensure both Robust Stability and Nominal Performance, then Robust Performance is achieved.

4.2.3 Controller Synthesis with Linear Matrix Inequalities

It is possible to synthesize a controller for a linear or LPV model given in state space form by solving a set of Linear Matrix Inequalities (LMI), given that the controller is feasible [119, 117]. An LMI is an expression of the following form [117]:

$$\mathbf{M}(\mathrm{x}) := \mathbf{M}_0 + x_1\mathbf{M}_1 + \ldots + x_m\mathbf{M}_m \geq 0, \tag{4.11}$$

where $\mathrm{x} = (x_1, \ldots, x_m)$ is a vector of real numbers, and $\mathbf{M}_i$ are real symmetric $n \times n$ matrices ($i = 1, \ldots, m$, and $n \in \mathbb{Z}_+$). The inequality $\geq$ means that $\mathrm{M}(\mathrm{x})$ is positive semidefinite. LMI-s are particularly interesting in control for the following reasons:

- An LMI defines a convex constraint on x.

- Convex optimization problems either have a single global optimum or are infeasible.

- A wide range of controller synthesis problems can be formulated as a convex optimization problem with convex constraints.

Let us consider the following LPV system in state space form:

$$\begin{aligned} \dot{\mathrm{x}}(t) &= \mathbf{A}(\rho(t))\mathrm{x}(t) + \mathbf{B}_1(\rho(t))\mathrm{d}(t) + \mathbf{B}_2\mathrm{u}(t) \\ \mathrm{z}(t) &= \mathbf{C}_1(\rho(t))\mathrm{x}(t) + \mathbf{D}_{11}(\rho(t))\mathrm{d}(t) + \mathbf{D}_{12}u(t) \\ \mathrm{y}(t) &= \mathbf{C}_2(\rho(t))\mathrm{x}(t) + \mathbf{D}_{21}(\rho(t))\mathrm{d}(t), \end{aligned} \tag{4.12}$$

where $\mathrm{x}(t) \in \mathbb{R}^{n_x}$ is the vector of state variables, $\mathrm{d}(t) \in \mathbb{R}^{n_d}$ is the vector of disturbances while $\mathrm{u}(t) \in \mathbb{R}^{n_u}$ is the vector of input signals. $\mathrm{z}(t) \in \mathbb{R}^{n_z}$ and $\mathrm{y}(t) \in \mathbb{R}^{n_y}$ are the vectors of performance signals and measured outputs respectively. $\rho \in \mathbb{R}^{n_\rho}$ is the vector of scheduling variables. $\mathbf{A}(\rho(t)) : \mathbb{R}^{n_\rho} \to \mathbb{R}^{n_x \times n_x}$, $\mathbf{B}_1(\rho(t)) : \mathbb{R}^{n_\rho} \to \mathbb{R}^{n_x \times n_d}$, $\mathbf{C}_1(\rho(t)) : \mathbb{R}^{n_\rho} \to \mathbb{R}^{n_z \times n_x}$, and $\mathbf{D}_{11}(\rho(t)) : \mathbb{R}^{n_\rho} \to \mathbb{R}^{n_z \times n_d}$ are matrix valued affine functions of $\rho(t)$, while $\mathbf{B}_2$, $\mathbf{C}_2$ $\mathbf{D}_{12}$ and $\mathbf{D}_{21}$ are real matrices of appropriate sizes. One of the most commonly used control schemes for this model is the following [119, 117]:

$$\begin{aligned} \dot{\mathrm{x}}_K(t) &= \mathbf{A}_K(\rho(t))\mathrm{x}_K(t) + \mathbf{B}_K(\rho(t))\mathrm{y}(t) \\ \mathrm{u}(t) &= \mathbf{C}_K(\rho(t))\mathrm{x}_K(t) + \mathbf{D}_K(\rho(t))\mathrm{y}(t). \end{aligned} \tag{4.13}$$

In this case, the controller is a dynamic LPV system as well. $\mathrm{x}_K(t)$ and $\mathbf{A}_K(\rho(t))$ have the same dimensions as $\mathrm{x}(t)$ and $\mathbf{A}(\rho(t))$ respectively, while $\mathbf{B}_K(\rho(t)) : \mathbb{R}^{n_\rho} \to \mathbb{R}^{n_x \times n_y}$, $\mathbf{C}_K(\rho(t)) : \mathbb{R}^{n_\rho} \to \mathbb{R}^{n_u \times n_x}$ and $\mathbf{D}_K(\rho(t)) : \mathbb{R}^{n_\rho} \to \mathbb{R}^{n_u \times n_y}$ are matrix-valued affine functions of ρ. If the state variables are available for measurement, it is also possible to realize the controller via state feedback:

$$\mathrm{u}(t) = \mathbf{G}(\rho(t))\mathrm{x}(t) \tag{4.14}$$

Both model and controller are LPV, affine functions of scheduling variable ρ. Let ρ_c be a convex polytope spanned by a set of M points $\mathrm{conv}\{\rho_1, \ldots, \rho_M\}$, so that $\rho(t)$ is always contained in ρ_c.

$$\rho(t) \in \rho_c, \quad \forall t \in [0, +\infty]. \tag{4.15}$$

By solving LMIs for points $\rho_1, \ldots, \rho_M$ makes the LMIs valid for the entire convex polytope [117]. Note that the more scheduling variables are in vector ρ, the more points are needed for the polytope, and the growth is exponential.

Linear Matrix Inequalities for State Feedback Control

Successfully solving the following LMIs:

$$\mathcal{X} > 0 \tag{4.16}$$

$$\begin{pmatrix} \mathcal{A}(\rho_j)^T + \mathcal{A}(\rho_j) & \mathcal{B}(\rho_j) & \mathcal{C}(\rho_j)^T \\ \mathcal{B}(\rho_j)^T & -\mathbf{I} & \mathcal{D}(\rho_j)^T \\ \mathcal{C}(\rho_j) & \mathcal{D}(\rho_j) & -\mathbf{I} \end{pmatrix} < 0 \tag{4.17}$$

for

- $n_x \times n_x$ symmetric real matrix $\mathbf{X}$,
- $\tilde{\mathbf{G}}(\rho_j) = \mathbf{G}(\rho_j)\mathbf{X}$,

where $\mathcal{A}(\rho_j)$, $\mathcal{B}(\rho_j)$, $\mathcal{C}(\rho_j)$, $\mathcal{D}(\rho_j)$, and $\mathcal{X}$ denotes:

$$\begin{aligned} \mathcal{X} &= \mathbf{X} \\ \mathcal{A}(\rho_j) &= \mathbf{A}(\rho_j)\mathbf{X} - \mathbf{B}_2\tilde{\mathbf{G}}(\rho_j) \\ \mathcal{B}(\rho_j) &= \mathbf{B}_1(\rho_j) \\ \mathcal{C}(\rho_j) &= \mathbf{C}_1(\rho_j)\mathbf{X} - \mathbf{D}_{12}\tilde{\mathbf{G}}(\rho_j) \\ \mathcal{D}(\rho_j) &= \mathbf{D}_{11}(\rho_j). \end{aligned} \tag{4.18}$$

provides the state feedback controller (4.14) that can make the LPV model (4.12) asymptotically stable. Furthermore, it ensures that the transfer from $\mathrm{d}(t)$ disturbances to $\mathrm{z}(t)$ performance output has $\mathcal{H}_\infty$ norm smaller than one ($\gamma = 1$) for all $j = 1, ...M$ [117].

Note that it is necessary to introduce $\tilde{\mathbf{G}}(\rho_j)$, and acquire $\mathbf{G}(\rho_j)$ indirectly to avoid the multiplication on unknown variables, and keep the matrix inequalities linear.

$\mathbf{D}_{11}(\rho_j)$ must be zero in order to be able to set an $\mathcal{H}_2$ constraint instead of an $\mathcal{H}_\infty$ one. Furthermore, there is a new variable introduced by the LMI for $\mathcal{H}_2$ norm: affine function $\mathbf{Z}(\rho) : \mathbb{R}^{n_\rho} \to \mathbb{R}^{n_y \times n_y}$. $\mathbf{Z}(\rho(t))$ is a symmetric real matrix for all $\rho(t)$. The LMI constraints for $\mathcal{H}_2$ norm are as follows:

$$\begin{gathered} \begin{pmatrix} \mathcal{A}(\rho_j)^T + \mathcal{A}(\rho_j) & \mathcal{B}(\rho_j) \\ \mathcal{B}(\rho_j)^T & -\mathbf{I} \end{pmatrix} < 0 \\ \begin{pmatrix} \mathcal{X} & \mathcal{C}(\rho_j)^T \\ \mathcal{C}(\rho_j) & \mathbf{Z}(\rho_j) \end{pmatrix} > 0 \\ Trace(\mathbf{Z}(\rho_j)) < 1. \end{gathered} \tag{4.19}$$

The LMI constraints for $\mathcal{H}_{2,g}$ norm is rather similar to $\mathcal{H}_2$, with the exception that $\mathbf{Z}(\rho)$ is not present:

$$\begin{gathered} \begin{pmatrix} \mathcal{A}(\rho_j)^T + \mathcal{A}(\rho_j) & \mathcal{B}(\rho_j) \\ \mathcal{B}(\rho_j)^T & -\mathbf{I} \end{pmatrix} < 0 \\ \begin{pmatrix} \mathcal{X} & \mathcal{C}(\rho_j)^T \\ \mathcal{C}(\rho_j) & \mathbf{I} \end{pmatrix} > 0. \end{gathered} \tag{4.20}$$

Unlike the previous norms, solving LMIs for $\mathcal{L}_1$ norm is an iterative process. Each iteration uses a different positive real λ value. Furthermore, solution requires a new positive real scalar-valued function $\mu(\rho) : \mathbb{R}^{n_\rho} \to \mathbb{R}^+$. The LMIs of controller synthesis for $\mathcal{L}_1$ norm are as follows:

$$\begin{gathered}\begin{pmatrix} \mathcal{A}(\rho_j)^T + \mathcal{A}(\rho_j) + \lambda\mathcal{X} & \mathcal{B}(\rho_j) \\ \mathcal{B}(\rho_j)^T & -\mu(\rho_j)\mathbf{I} \end{pmatrix} < 0 \\ \begin{pmatrix} \lambda\mathcal{X} & 0 & \mathcal{C}(\rho_j)^T \\ 0 & (1-\mu(\rho_j))\mathbf{I} & \mathcal{D}(\rho_j)^T \\ \mathcal{C}(\rho_j) & \mathcal{D}(\rho_j) & \mathbf{I} \end{pmatrix} > 0 \\ \mu(\rho_j) > 0. \end{gathered} \tag{4.21}$$

Linear Matrix Inequalities for Dynamic Controller

The dynamics controller usually has an equal or less amount of state variables than the controlled process. This work focuses on the former case. However, it is impossible to derive an LMI using the dynamic controller parameters directly, and a transformation is necessary to make the matrix inequalities linear [117]. The transformed LMIs for all the four previously referenced norms are solved for the following variables:

- $\mathbf{X}$, $\mathbf{Y}$ $n_x \times n_x$ real symmetric matrices and
- $\mathbf{K}(\rho_j)$, $\mathbf{L}(\rho_j)$, $\mathbf{M}(\rho_j)$, $\mathbf{N}(\rho_j)$ real matrix valued affine functions.

The inequalities (4.17), (4.19), (4.20) and (4.21) are still valid for the dynamic controller case, but the meaning of $\mathcal{A}(\rho_j)$, $\mathcal{B}(\rho_j)$, $\mathcal{C}(\rho_j)$ and $\mathcal{D}(\rho_j)$ are different from (4.18):

$$\begin{aligned} \mathcal{A}(\rho_j) &= \begin{pmatrix} \mathbf{A}(\rho_j)\mathbf{Y} + \mathbf{B}_2\mathbf{M}(\rho_j) & \mathbf{A}(\rho_j) + \mathbf{B}_2\mathbf{N}(\rho_j)\mathbf{C}_2 \\ \mathbf{K}(\rho_j) & \mathbf{X}\mathbf{A}(\rho_j) + \mathbf{L}(\rho_j)\mathbf{C}_2 \end{pmatrix} \\ \mathcal{B}(\rho_j) &= \begin{pmatrix} \mathbf{B}_1(\rho_j) + \mathbf{B}_2\mathbf{N}(\rho_j)\mathbf{D}_{21} \\ \mathbf{X}\mathbf{B}_1(\rho_j) + \mathbf{L}(\rho_j)\mathbf{D}_{21} \end{pmatrix} \\ \mathcal{C}(\rho_j) &= \begin{pmatrix} \mathbf{Y}\mathbf{C}_1(\rho_j) + \mathbf{M}(\rho_j)^T\mathbf{D}_{12}^T \\ \mathbf{C}_1(\rho_j)^T + \mathbf{C}_2^T\mathbf{N}(\rho_j)^T\mathbf{D}_{12}^T \end{pmatrix}^T \\ \mathcal{D}(\rho_j) &= \mathbf{D}_{11}(\rho_j) + \mathbf{D}_{12}\mathbf{N}(\rho_j)\mathbf{D}_{21} \\ \mathcal{X} &= \begin{pmatrix} \mathbf{Y} & \mathbf{I} \\ \mathbf{I} & \mathbf{X} \end{pmatrix}. \end{aligned} \tag{4.22}$$

Regardless of the used norm, $\mathcal{X}$ must be positive definite, similarly to the state feedback case:

$$\mathcal{X} > 0 \tag{4.23}$$

If the LMIs are feasible and successfully solved, transforming $\mathbf{K}(\rho_j)$, $\mathbf{L}(\rho_j)$, $\mathbf{M}(\rho_j)$ and $\mathbf{N}(\rho_j)$ provides the (4.13) controller parameters, for

$j = 0, \ldots, M$.

$$\begin{aligned} \mathcal{T}_1 &= \begin{pmatrix} \mathbf{U} & \mathbf{X}\mathbf{B}_2 \\ 0 & \mathbf{I} \end{pmatrix} \\ \mathcal{T}_2 &= \begin{pmatrix} \mathbf{V}^T & 0 \\ \mathbf{C}_2\mathbf{Y} & \mathbf{I} \end{pmatrix} \\ \begin{pmatrix} \mathbf{A}_{K,j} & \mathbf{B}_{K,j} \\ \mathbf{C}_{K,j} & \mathbf{D}_{K,j} \end{pmatrix} &= \mathcal{T}_1^{-1} \begin{pmatrix} \mathbf{K}_j - \mathbf{X}\mathbf{A}_j\mathbf{Y} & \mathbf{L}_j \\ \mathbf{M}_j & \mathbf{N}_j \end{pmatrix} \mathcal{T}_2^{-1} \end{aligned} \tag{4.24}$$

where

$$\mathbf{U}\mathbf{V}^T = \mathbf{I} - \mathbf{X}\mathbf{Y}. \tag{4.25}$$

Note that even if there is a numerically well-conditioned solution for the LMIs of a dynamic controller, (4.25) may produce matrices $\mathbf{U}$ and $\mathbf{V}$ that are close to singularity.

Constraints on Closed-loop Poles

It is possible to set constraints on the controlled system beyond stability and norms. Two of these that have high relevance in AP [SEK14]:

1. Limiting fastest poles in the closed-loop system,
2. Limiting angle of complex conjugate pole pairs.

Several of the most widely used T1DM models are defined in continuous time. However, it is very likely, that the AP executes its control algorithm in fixed discrete time intervals. If that is the case, the dynamics of the model that are faster than the sampling time T_s lose their relevance. Moreover, the controller shall not accelerate the system beyond what the sampling time allows. The LMI below ensures that real value of the poles of the closed-loop system are larger than p_{max} for both controller types [117]. Setting p_{max} to $-\frac{1}{2T_s}$ enables executing the control algorithm in T_s time intervals.

$$\mathcal{A} + \mathcal{A}^T - 2p_{max}\mathcal{X} > 0 \tag{4.26}$$

As mentioned earlier in Section 4.4.1, weighting functions alone cannot guarantee nonnegative control signal. However, the need for negative $u(t)$ is greatly decreased if there are no oscillatory transients. Hence, it is beneficial to limit the angle of complex conjugate pole pairs of the closed-loop system, using the following LMI [117]:

$$\begin{pmatrix} \mathcal{A}(\rho_j) + \mathcal{A}(\rho_j) & 2\cos(\alpha)\mathcal{A}(\rho_j) \\ 2\cos(\alpha)\mathcal{A}(\rho_j)^T & \mathcal{A}(\rho_j) + \mathcal{A}(\rho_j) \end{pmatrix} < 0 \tag{4.27}$$

Figure 4.2 illustrates the combined effect of (4.26) and (4.27).

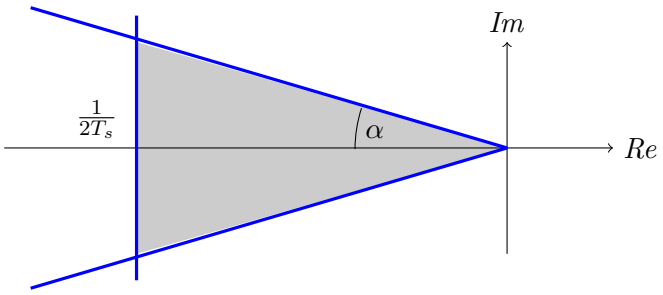

Figure 4.2. Constraints for the poles of the controlled system: T_s is the sampling time of the CGM sensor, α is the maximum angle of the complex conjugate pole pairs

4.3 LPV representation of the Cambridge model

The LPV-transformed Cambridge model needs scheduling variables that represent the following nonlinearities:

- Remote effect of insulin on glucose distribution: $x_1(t)Q_1(t)$
- Remote effect of insulin on glucose disposal: $x_2(t)Q_2(t)$
- The Michelis-Menten function of non-insulin-dependent glucose flux: $\frac{F_{01}Q_1(t)}{Q_1(t)+V_G}$
- Endogenous glucose production: $EGP_0 \max\{0, 1 - x_3(t)\}$
- Renal extraction: $R_{cl} \max\{0, Q_1(t) - R_{thr}V_G\}$

For the remote effect of insulin, we have the product of an insulin-related and a glucose-related state variable. For both of the products, we can choose either of the states as scheduling variables. The potential configurations are as follows:

1. Both scheduling variables are insulin related:

$$\rho_1(t) = x_1(t) \quad \rho_2(t) = x_2(t) \tag{4.28}$$

2. Mixed configurations:

$$\rho_1(t) = Q_1(t) \quad \rho_2(t) = x_2(t) \tag{4.29}$$

or

$$\rho_1(t) = x_1(t) \quad \rho_2(t) = Q_2(t) \tag{4.30}$$

3. Both scheduling variables are glucose related:

$$\rho_1(t) = Q_1(t) \quad \rho_2(t) = Q_2(t) \tag{4.31}$$

In the case of option 1 the system is hardly controllable. Endogenous glucose production is the only connection between the glucose and insulin subsystem. Furthermore, insulin is the control signal of AP. Thus, all states linked to insulin can change in a rather wide range during closed-loop control. Consequently, the LPV model will have drastically different dynamics at the extreme values of the scheduling variables, making the controller synthesis rather difficult. On the other hand, AP aims to keep the glucose concentration in a narrow range. Therefore, the controller synthesis needs to consider smaller bounds for scheduling variables for options 2 and 3. Option 3 is the most straightforward approach. Not only does it offer the smallest polytope for the scheduling variables, but there is a connection between ρ_1 and the nonlinearity of non-insulin-dependent glucose flux, which can be exploited. However, both Q_1 and Q_2 are affected by disturbances, while x_1 and x_2 are not. If the scheduling variables are not available for measurement, estimation of Q_1 and Q_2 will be less accurate than x_1 and x_2. A reasonable compromise between option 1 and 3 is (4.29) of option 2. The model is still controllable, and the estimation of $x_2(t)$ is potentially more accurate and precise than the estimation of Q_2. Despite this, option 3 is still the recommended LPV modeling strategy [SEK+13].

There are two alternatives for the Michelis-Menten function of non-insulin-dependent glucose flux. The trivial approach is to define a scheduling variable as:

$$\rho_f(t) = \frac{F_{01}}{Q_1(t) + V_G} \tag{4.32}$$

$\rho_f(t)$ is a function of $Q_1(t)$. This dependency can be exploited to define a minimal convex polytope for $\rho(t)$. If there is a need to reduce the dimension of the scheduling variables, linear approximation (4.33) can be accurate enough to use in controller design [SEK14]. The various configurations are illustrated on Figure 4.3.

$$\frac{F_{01}}{Q_1(t) + V_G} \approx F_a \rho_1(t) + F_b \tag{4.33}$$

Reducing the number of scheduling variables can be beneficial because - as discussed in Section 4.2.3 - the computational power needed to execute controller synthesis for an LPV system increases exponentially with the number of scheduling variables.

Endogenous glucose production and renal extraction introduce a "switching" effect to the model. The former is active unless plasma glucose

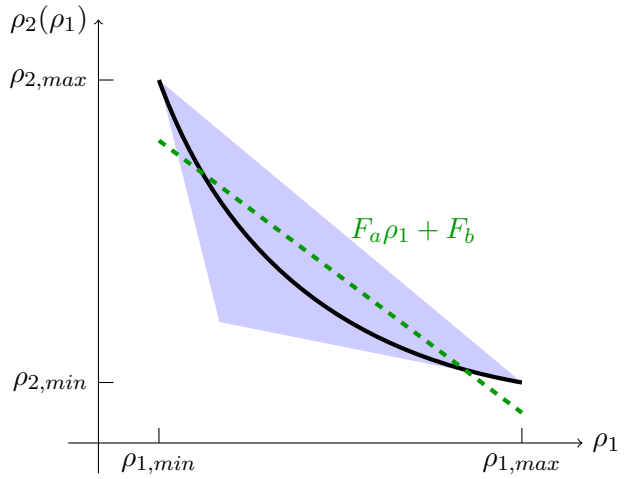

Figure 4.3. Different configurations for scheduling variable ρ_2. Solid black line: ρ_2 as a function of ρ_1; blue area: convex polytope for ρ_1 and ρ_2; dashed line: linear approximation of ρ_2.

is elevated. The latter activates when the plasma glucose concentration exceeds the threshold R_{th}. There are four different approaches to handling such nonlinearity in an LPV model:

- Introduce a scheduling variable approximating the switching with a sigmoid function.
- Replace with a linear approximation and ensure that switching never occurs on a controller synthesis level.
- Address the switching nature of the nonlinearity with switching control.
- Treat it as a disturbance.

The endogenous glucose production only depends on $x_3(t)$, which is an insulin-specific state variable. Therefore, it is possible to set a constraint for the control signal that $x_3(t)$ is always smaller than one. The corresponding nonlinearity can be ignored if this constraint is satisfied. On the other hand, renal extraction can be considered a disturbance.

Using (4.31), (4.32), and the worst-case meal intake model (2.4) the LVP representation chosen for the Cambridge model is as follows:

$$\begin{aligned}
\dot{C}(t) = & -k_{a,int}C(t) + \frac{k_{a,int}}{V_G}Q_1(t) \\
\dot{Q}_1(t) = & -\rho_f(t)Q_1(t) - \rho_1(t)x_1(t) + k_{12}Q_2(t) - dist_{Rcl}(t) - \\
& -dist_{Phy}(t) + EGP_0\left(1 - x_3(t)\right) + \frac{U_{G,ceil}}{t_{max}}\tilde{G}(t) \\
\dot{Q}_2(t) = & \rho_1(t)x_1(t) - k_{12}Q_2(t) - \rho_2(t)x_2(t) \\
\dot{x}_1(t) = & -k_{b1}x_1(t) + S_{IT}k_{b1}I(t) \\
\dot{x}_2(t) = & -k_{b2}x_2(t) + S_{ID}k_{b2}I(t) \\
\dot{x}_3(t) = & -k_{b3}x_3(t) + S_{IE}k_{b3}I(t) \\
\dot{I}(t) = & \frac{k_a}{V_I}S_2(t) - k_eI(t) \\
\dot{S}_2(t) = & -k_aS_2(t) + k_aS_1(t) \\
\dot{S}_1(t) = & -k_aS_1(t) + u(t) \\
\dot{\tilde{G}}(t) = & -\frac{1}{t_{max}}\tilde{G}(t) + dist_{meal}(t)
\end{aligned} \tag{4.34}$$

where $dist_{Rcl}(t)$ is a disturbance representing renal extraction.

If model reduction is necessary (see Figure 2.1), the resulting LPV model will be:

$$\begin{aligned}
C(t) \approx & \frac{1}{V_G}Q_1(t) \\
\dot{Q}_1(t) = & -\rho_f(t)Q_1(t) - \rho_1(t)x_1(t) + k_{12}Q_2(t) - dist_{Rcl}(t) - \\
& -dist_{Phy}(t) + EGP_0\left(1 - \tfrac{k_aS_{IE}}{V_Ik_e}S_2(t)\right) + \frac{U_{G,ceil}}{t_{max}}\tilde{G}(t) \\
\dot{Q}_2(t) = & \rho_1(t)x_1(t) - k_{12}Q_2(t) - \rho_2(t)\tfrac{k_aS_{ID}}{V_Ik_e}S_2(t) \\
\dot{x}_1(t) = & k_{b1}\left(\tfrac{k_aS_{IT}}{V_Ik_e}S_2(t) - x_1(t)\right) \\
\dot{S}_2(t) = & -k_aS_2(t) + k_aS_1(t) \\
\dot{S}_1(t) = & -k_aS_1(t) + u(t) \\
\dot{\tilde{G}}(t) = & -\frac{1}{t_{max}}\tilde{G}(t) + dist_{meal}(t)
\end{aligned} \tag{4.35}$$

If there is a need to reduce the number of scheduling variables, the equation for $\dot{Q}_1(t)$ using (4.33) is as follows:

$$\begin{aligned}
\dot{Q}_1(t) = & -\left(F_a\rho_1(t) + F_b\right)(t)Q_1(t) - \rho_1(t)x_1(t) + k_{12}Q_2(t) - \\
& -dist_{Rcl}(t) - dist_{Phy}(t) + EGP_0\left(1 - x_3(t)\right) + \\
& + \frac{U_{G,ceil}}{t_{max}}\tilde{G}(t).
\end{aligned} \tag{4.36}$$

As previously mentioned in Section 2.2 if $1/k_{12}$ is comparable to the CGMS sample time, then further model reduction is possible by replacing $Q_1(t)$ and $Q_2(t)$ with $\tilde{Q}(t) \approx Q_t(t) - x_1(t)/k_{12}$ as follows:

$$\begin{aligned}\dot{C}(t) &= -k_{a,int}C(t) + \frac{k_{a,int}}{V_G}\tilde{Q}(t) - \frac{k_{a,int}}{k_{12}V_G}x_1(t) \\ \dot{Q}_1(t) &= -\rho_f(t)\tilde{Q}(t) + (F_c\rho_1(t) + F_d)\,x_1(t) - dist_{Rcl}(t) - \\ & dist_{Phy}(t) - \rho_2(t)x_2(t) + EGP_0\,(1 - x_3(t)) + \frac{U_{G,ceil}}{t_{max}}\tilde{G}(t)\end{aligned} \tag{4.37}$$

where $F_c\rho_1(t) + F_d \approx \dfrac{Q_1(t)F_{01}}{k_{12}(Q_1(t) + V_G)}$. Furthermore, the output can be approximated with $C(t) \approx \dfrac{1}{V_G}\left(\tilde{Q}(t) - \dfrac{1}{k_{12}}x_1(t)\right)$, if necessary. Note, that even though $Q_1(t)$ and $Q_2(t)$ are not part of the model (4.37), the scheduling variables $\rho_1(t)$ and $\rho_2(t)$ are still present.

The scheduling variables must be available for measurement to use an LPV controller based on these LPV models. However, this is not the case in clinical practice. Hence, a state observer must provide accurate estimations for the control algorithm. Using estimation instead of measurement [Kovács et al., 2011] makes the controller a quasi LPV controller.

4.4 Robust Controller Design

4.4.1 Extended Model

The model used for robust controller design is not limited to the model itself. Extending the model with additional static or dynamic weighting functions is advantageous in general, and especially in AP [119, 120] [Kovács et al., 2011, KS12, KSAB13, SEK+13, SEK14]. For example, some of the control relevant signals may have different units, ranges of values, or different significance. Weighting functions are essential to make them comparable. Furthermore, they can capture additional constraints and performance specifications. Finally, dynamic weighting functions assign varying influences to different frequency ranges.

Sensor Noise Model

For the sake of simplicity, the measurement noise of the CGM sensor is modeled with additive white noise. The noise has standard normal distribution, multiplied with W_n = 0.1 [mmol/L] weighting function.

Quantifying Disturbance Rejection Performance

The regulation of the glucose concentration in a T1DM patient is essentially a disturbance rejection problem. One approach to solving it

with a closed-loop control is to minimize the gain from meal intake (and other disturbances) to an error signal $e(t)$. This error signal is the deviation of glucose concentration from a reference signal $r(t)$, representing the desired output. The weighting function W_{per} specifies our expectation of the error signal in the frequency domain. It takes a considerable amount of time for a healthy metabolism to normalize the glucose levels after a meal intake with high carbohydrate content. Hence, it is advantageous to choose a W_{per} akin to a low pass filter. The W_{per} chosen in this work is as follows:

$$W_{per}(j\omega) = \frac{1}{180j\omega + 1}. \tag{4.38}$$

It specifies that the controller should mainly minimize tracking error below $\frac{2\pi}{180}$ [rad/min]. The value was chosen based on the 3-hour duration of OGTT. Moreover, the direct feedthrough is zero to enable $\mathcal{H}_2$ norm-based performance optimization.

Reference Dynamics

A healthy individual's fasting plasma glucose (FPG) is below 4.9 mmol/L or 90 mg/dL, which is a good candidate for a constant reference signal r_0. Alternatively, if $\hat{G}_{flux}(t)$ estimation of glucose flux from meal intake is available, one can define a weighting function W_{ref}. The input of W_{ref} is the estimated glucose flux, and the sum of its output with a constant value of $r_0 = 4.9$ mmol/L serves as reference signal $r(t)$. W_{ref} should mimic the glucose trends of a healthy patient after meal intake. In this work, the parameters of W_{ref} are patient-specific:

$$W_{ref}(j\omega) = \frac{11}{U_{G,ceil}} \frac{1}{60j\omega + 1} \tag{4.39}$$

Control Signal Constraints

The ΔPK structure allows placing constraints on the control signal. In practical applications, the control signal cannot be infinitely large or small and may have limited resolution. Furthermore, actuator dynamics may place additional constraints, either because it is physically unable to provide specific signals or certain behavior can lead to damage or degradation of the equipment. When it comes to model-based control based on the Cambridge model (2.3), there are four constraints to consider:

1. Maximum insulin injection rate u_{max}.
2. Resolution of injected insulin u_q.

3. Ensuring that endogenous glucose production does not reach below zero, as mentioned in Section 4.3.

4. In a single hormone AP, the control signal cannot be negative.

Figure 4.4 illustrates how to extend the nominal model P to accommodate for these limitations.

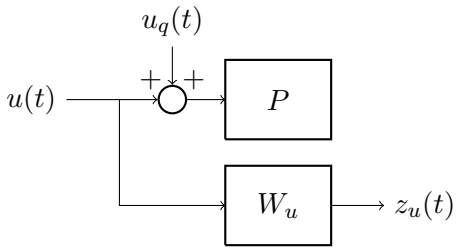

Figure 4.4. Limitations of the control signal.

$u_q(t)$ is quantization noise resulting from the resolution of the control signal. W_u is a weighting function with patient-specific parameters intended to ensure both less than u_{max} control signal and nonnegative endogenous glucose production. Let W_{EGP} denote the transfer from $u(t)$ to $x_3(t)$ in the original Cambridge model (2.3):

$$W_{EGP}(j\omega) = \frac{\frac{S_{IE}}{V_I k_e}}{(\frac{1}{k_a}s+1)^2(\frac{1}{k_e}s+1)(\frac{1}{k_{b3}}s+1)} \tag{4.40}$$

W_u should incorporate W_{EGP} while considering model reduction and u_{max}. Figure 4.5 illustrates the case where the state variable I has negligible time constant k_e, and hence the transfer function of W_u is as follows:

$$\begin{aligned} W_u(j\omega) &= \frac{\frac{S_{IE}}{V_I k_e}(\tau s+1)^3}{(\frac{1}{k_a}s+1)^2(\frac{1}{k_{b3}}s+1)} \\ \tau &= \sqrt[3]{\frac{u_{max}V_I k_e}{S_{IE}k_a^2 k_{b3}}}. \end{aligned} \tag{4.41}$$

However, with a linear or LPV system, it is not possible to limit a signal asymmetrically. Therefore, W_u alone cannot guarantee a nonnegative control signal.

Extended LPV Cambridge Model

Figure 4.6 displays the Cambridge model with its proposed LPV form (4.34) extended with weighting functions W_n, W_{per}, W_{ref} and W_u pre-

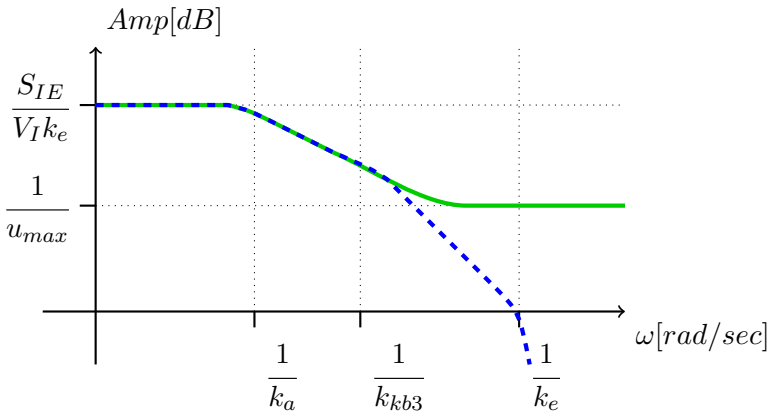

Figure 4.5. Configuring W_u based on model parameters. The dashed blue and solid green lines display the amplitude diagrams of W_{EGP} and W_u, respectively. In this example, parameter k_e is only relevant when setting the DC gain of W_u.

sented in this Section. It also includes uncertainty functions $W_{o,m}$ and $W_{o,a}$ presented in Section 2.3.1, shortened from $W_{out,m}$ and $W_{out,a}$.

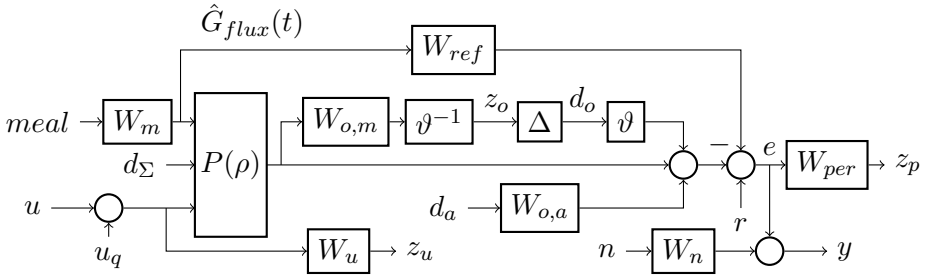

Figure 4.6. Extended LPV model for controller synthesis.

$P(\rho)$ represents the LPV model (4.34) excluding the state components regarding meal intake and ingestion, which is separated into a weighting function W_m. As mentioned earlier, the estimated glucose flux $\hat{G}_{flux}(t)$ is the input of W_{ref}, which is shown to be the output of W_m. To account for the estimation error $G_{flux}(t) - \hat{G}_{flux}(t)$, it is incorporated into d_Σ, which the sum of all disturbances affecting $P(\rho)$ directly:

- $dist_{Phy}(t)$ physical activity,
- $dist_{Rcl}(t)$ renal extraction,
- u_q quantization noise,

- and the aforementioned glucose flux estimation error.

The constant scaling factor ϑ is necessary to tune the model during controller synthesis to satisfy Robust Stability constraints at the expense of Nominal Performance.

4.4.2 Controller Synthesis for Artificial Pancreas

Although the LMIs presented in Section 4.2.3 address many of the control problems posed by AP, there are additional ways to customize them for blood glucose control:

- Scale nominal performance for better disturbance rejection.

- Scale output multiplicative uncertainty.

γ should be smaller than one to ensure both RS and NP. However, while this is a hard constraint for robust stability, it may be sufficient to find an "as low as possible" γ value for nominal performance. Alternatively, a fixed γ_{max} constraint may be specified to satisfy minimum performance requirements, but the controller should still be optimized to patient parameters. Let $J \subset \{1, \ldots, n_z\}$ be a set of indices that select some of the $\mathrm{z}(t)$ signals of the LPV model (4.12), that are associated with nominal performance. Furthermore, let Ψ be an $n_z \times n_z$ real diagonal matrix, for which the following properties hold:

$$\Psi_{i,i} = \begin{cases} 1 & i \notin J \\ \psi & i \in J \end{cases}, \tag{4.42}$$

where ψ is a positive real number. Let us take the LMI for $\mathcal{H}_\infty$ norm (4.17), although the same approach can be applied to all other norms, and scale the nominal performance outputs with Ψ:

$$\begin{pmatrix} \mathcal{A}(\rho_j)^T + \mathcal{A}(\rho_j) & \mathcal{B}(\rho_j) & \mathcal{C}(\rho_j)^T\Psi \\ \mathcal{B}(\rho_j)^T & -\mathbf{I} & \mathcal{D}(\rho_j)^T\Psi \\ \Psi\mathcal{C}(\rho_j) & \Psi\mathcal{D}(\rho_j) & -\mathbf{I} \end{pmatrix} < 0. \tag{4.43}$$

As long as the feasibility of the LMI is maintained, the larger ψ is the better the disturbance rejection performance, while still satisfying robust stability and other hard constraints (e.g., for control signal). However, (4.43) in its current form is not linear. Therefore, the following

transformation is necessary:

$$\mathcal{T} = \begin{pmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \Psi^{-1} \end{pmatrix}$$
$$\mathcal{T} \begin{pmatrix} \mathcal{A}(\rho_j)^T + \mathcal{A}(\rho_j) & \mathcal{B}(\rho_j) & \mathcal{C}(\rho_j)^T\Psi \\ \mathcal{B}(\rho_j)^T & -\mathbf{I} & \mathcal{D}(\rho_j)^T\Psi \\ \Psi\mathcal{C}(\rho_j) & \Psi\mathcal{D}(\rho_j) & -\mathbf{I} \end{pmatrix} \mathcal{T}^T = \tag{4.44}$$
$$= \begin{pmatrix} \mathcal{A}(\rho_j)^T + \mathcal{A}(\rho_j) & \mathcal{B}(\rho_j) & \mathcal{C}(\rho_j)^T \\ \mathcal{B}(\rho_j)^T & -\mathbf{I} & \mathcal{D}(\rho_j)^T \\ \mathcal{C}(\rho_j) & \mathcal{D}(\rho_j) & -\Psi^{-2} \end{pmatrix}.$$

Introducing a new variable $\tilde{\gamma} = \frac{1}{\psi^2}$ makes (4.44) an LMI. Solving it for minimal $\tilde{\gamma}$ provides a controller that satisfies robust performance, with the nominal performance tuned for individual patients.

Since the solution is for all of $\{\rho_1, \ldots, \rho_M\}$, $\tilde{\gamma}$ is a valid upper limit for the entire convex polytope. However, this is usually a suboptimal solution. It may be possible to achieve a smaller $\tilde{\gamma}$ value for some points of the polytope by changing $\tilde{\gamma}$ into an affine real positive valued function of ρ.

$$\tilde{\gamma}(\rho) : \mathbb{R}^{n_\rho} \to \mathbb{R}^+. \tag{4.45}$$

Let us introduce the average of the points spanning ρ_c:

$$\bar{\rho} = \frac{1}{M} \sum_{j=1}^{M} \rho_j. \tag{4.46}$$

Instead of minimizing $\tilde{\gamma}$ when solving the LMIs, minimize the value of $\tilde{\gamma}(\bar{\rho})$, so that $\tilde{\gamma}(\rho_j)$ is positive for all $j = 1, \ldots, M$.

The performance output is not the only one that should be scaled. The glucose concentration in the subcutaneous tissue, which is the output of the Cambridge model, can reach up to 17 mmol/L even during adequate control. It might be infeasible to achieve robust stability depending on how the weighting function of multiplicative uncertainty $W_{out,m}$ is set. Thus, the scaling mentioned earlier in Section 4.4.1 and displayed in Figure 4.6 denoted with ϑ can prove useful. Let us introduce other sets of indices, J_{in} and J_{out}, similarly to J. Performance outputs indexed with members of J_{out} are associated with multiplicative uncertainty, while disturbance inputs indexed with members of J_{in} come from the unstructured uncertainty block. The latter is denoted with Δ in Figure 4.6. Let Θ_{in} and Θ_{out} be $n_d \times n_d$ and $n_z \times n_z$ real

diagonal matrices respectively, similar to Φ:

$$\Theta_{in,(i,i)} = \begin{cases} 1 & i \notin J_{in} \\ \vartheta & i \in J_{in} \end{cases}$$
$$\Theta_{out,(i,i)} = \begin{cases} 1 & i \notin J_{pit} \\ \dfrac{1}{\vartheta} & i \in J_{out} \end{cases} \tag{4.47}$$

The multiplicative uncertainty scaling applied for $\mathcal{H}_\infty$ norm is as follows:

$$\begin{pmatrix} \mathcal{A}(\rho_j)^T + \mathcal{A}(\rho_j) & \mathcal{B}(\rho_j)\Theta_{in} & \mathcal{C}(\rho_j)^T\Theta_{out} \\ \Theta_{in}\mathcal{B}(\rho_j)^T & -\mathbf{I} & \Theta_{in}\mathcal{D}(\rho_j)^T\Theta_{out} \\ \Theta_{out}\mathcal{C}(\rho_j) & \Theta_{out}\mathcal{D}(\rho_j)\Theta_{in} & -\mathbf{I} \end{pmatrix} < 0. \tag{4.48}$$

While the nominal performance scaling was identical to state feedback and dynamic control, the same does not hold for uncertainty scaling. For state feedback, the following transformation is applicable:

$$\mathcal{T} = \begin{pmatrix} \mathbf{I} & 0 & 0 \\ 0 & \Theta_{in} & 0 \\ 0 & 0 & \Theta_{out}^{-1} \end{pmatrix}$$
$$\mathcal{T} \begin{pmatrix} \mathcal{A}(\rho_j)^T + \mathcal{A}(\rho_j) & \mathcal{B}(\rho_j)\Theta_{in} & \mathcal{C}(\rho_j)^T\Theta_{out} \\ \Theta_{in}\mathcal{B}(\rho_j)^T & -\mathbf{I} & \Theta_{in}\mathcal{D}(\rho_j)^T\Theta_{out} \\ \Theta_{out}\mathcal{C}(\rho_j) & \Theta_{out}\mathcal{D}(\rho_j)\Theta_{in} & -\mathbf{I} \end{pmatrix} \mathcal{T}^T = \tag{4.49}$$
$$= \begin{pmatrix} \mathcal{A}(\rho_j)^T + \mathcal{A}(\rho_j) & \mathcal{B}(\rho_j)\Theta_{in}^2 & \mathcal{C}(\rho_j)^T \\ \Theta_{in}^2\mathcal{B}(\rho_j)^T & -\Theta_{in}^2 & \Theta_{in}^2\mathcal{D}(\rho_j)^T \\ \mathcal{C}(\rho_j) & \mathcal{D}(\rho_j)\Theta_{in}^2 & -\Theta_{out}^{-2} \end{pmatrix}.$$

Once ϑ^2 is substituted with $\tilde{\vartheta}$, then (4.49) reverts back to an LMI, since $\mathcal{B}$ and $\mathcal{D}$, as defined in (4.18), does not depend on any of the controller parameters or the other unknowns. Furthermore, since J and J_{out} are disjointed sets, the two scaling methods can be combined.

The validity of this method is not limited to $\mathcal{H}_\infty$ norm, as the same $\mathcal{T}$ transformation is applicable for the LMIs of $\mathcal{H}_2$ and $\mathcal{H}_{2g}$ norms. However, the inequalities for $\mathcal{L}_1$ norm are not linear due to the term $\Theta_{in}^2(1 - \mu(\rho))$. Hence, an iterative solution is necessary.

Unfortunately, the same method does not make the matrix inequality linear for the dynamic controller since in (4.22) both $\mathcal{B}$ and $\mathcal{D}$ depend on controller parameters. Thus, for a dynamic controller, finding the optimal ϑ scaling value, where the problem is feasible, can only be determined via an iterative process. If performance output is scaled, the goal is to find the ϑ value that allows the best performance.

Figure 4.7 shows an example of glucose control of multiple virtual T1DM patients with a hybrid $\mathcal{H}_2/\mathcal{H}_\infty$ robust qLPV state feedback controller. Chapter 5 provides more simulation results.

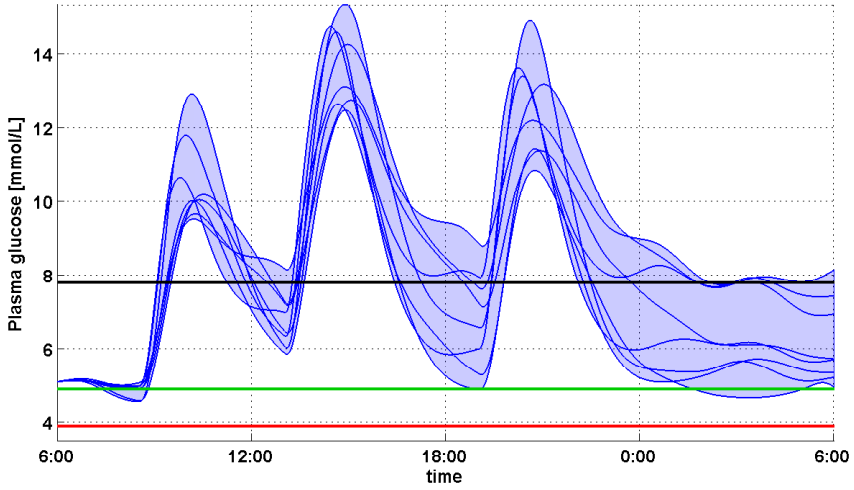

Figure 4.7. Example of glucose control. Solid blue line represents subcutaneous glucose concentration. The threshold for hypo- and hyperglycemia is indicated with red and black lines, respectively. Green line marks the optimal glucose level.

4.4.3 Known Limitations

The robust controller synthesis method presented in Section 4.2.3 have certain limitations. First, solving the LMIs introduced in this Section results in an LPV controller. It is more capable of controlling an LPV system than a purely linear algorithm. However, there are most probably more efficient nonlinear controllers. Merely not limiting the matrix-valued functions $\mathbf{A}_K(\rho)$, $\mathbf{B}_K(\rho)$, $\mathbf{C}_K(\rho)$, $\mathbf{D}_K(\rho)$, and $\mathbf{G}(\rho)$ to affine functions of ρ could lead to a more powerful controller. In this case, $\mathcal{X}$ should not be constant but a function of ρ. However, the matrix inequalities would no longer be linear.

Using a constant $\mathcal{X}$ has another consequence as well. Although the LMIs hold at each point of the convex polytope spanned by the scheduling variables, they are only valid *locally*. The transfer from one point of the polytope to another, precisely the transient behavior, can be problematic. For example, insulin affects the glucose levels in a T1DM model faster if the glucose concentration is already high than when it is low. Hence, the insulin injected by an LPV controller for a specific scheduling variable value may not be optimal as the scheduling variables change. This can lead to oscillatory transients even if the complex conjugate pole pairs of the closed-loop system are constrained.

5

IN SILICO VALIDATION

This chapter addresses the third thesis group: in silico validation of the observer and controller algorithms proposed in thesis groups 1 and 2. The contributions that belong to this thesis group are the following:

1. Evaluation of the effect of the number of sigma points on state estimation accuracy for a commonly used T1DM model.

2. In silico validation of a robust qLPV $\mathcal{H}_2/\mathcal{H}_\infty$ state feedback controller.

The related publications are: [KSAB13, KSF+13, KSS+13, KKSE13, SEK14, KKS+14, KSE+14, KFS+15, SDKew].

The effectiveness of an AP control algorithm can be assessed in a simulated environment. The FDA has approved the UVA-Padova Type 1 Diabetes Simulator as an alternative to animal testing, as mentioned earlier in Section2.1 [84]. The Cambridge model has its own freely available simulator [85]. The Jacobs T1DM simulator [121] employs the Cambridge model as well but contains an embedded physical activity sub-model. Finally, the AIDA simulator is a freeware computer program for demonstration and teaching purposes that has been available online since 1996 [122].

5.1 Simulation Environment

The state estimation and controller algorithm presented in earlier chapters were tested via a simulator based on the Cambridge model [17]. The Cambridge model serves as both a virtual patient (ground truth) and the basis of controller and observer design. [Szalay et al., 2012]

Table 5.1. Virtual patient parameters for the Cambridge simulator

	Patient ID					
	1	2	4	6	8	10
$k_{a,int}$	0.0736	0.151	0.1103	0.0974	0.0689	0.0898
V_G	12.26	13.05	15.5	15.58	13.70	13.3
F_{01}^s	0.893	0.879	0.704	1.0999	0.986	0.3999
k_{12}	0.1095	0.0509	0.0635	0.0307	0.0293	0.0537
R_{cl}	0.0119	0.0130	0.0105	0.00965	0.0115	0.0111
R_{thr}	11.70	9.22	10.07	7.75	7.75	7.55
EGP_0	1.354	0.819	0.718	1.1224	1.069	0.4393
t_{max}	43	55	29	26	42	52
$U_{G,ceil}$	2.415	2.415	2.415	2.415	2.415	2.415
k_{b1}	0.0021	0.0006	0.0014	0.0007	0.0032	0.0048
k_{b2}	0.3956	0.0136	0.1377	0.0369	0.2195	0.0442
k_{b3}	0.0803	0.0202	0.0210	0.0339	0.0323	0.0166
S_{IT}	0.00771	0.0011	0.00119	0.00124	0.003	0.00054
S_{ID}	0.000314	0.000158	0.000664	0.000153	0.000170	0.0012
S_{IE}	0.0377	0.0073	0.0116	0.0114	0.0219	0.0053
k_a	0.0198	0.016	0.0253	0.0257	0.0289	0.0244
V_I	7.79	11.79	11.3	12.3	9.72	13.02
k_e	0.132	0.101	0.14	0.177	0.120	0.167

evaluates the observability of the model for the latter case. Each simulation uses one of the six patient parameters published for the Cambridge simulator [85]. The parameters of the patients are presented in in Table 5.1.

The simulation environment is part of a larger framework built with MATLAB® version R2009b and the corresponding Simulink®. This framework allows the creation and testing of a wide range of observers, predictors, and controllers for glucose control. The framework is implemented as a collection of scripts, graphical user interfaces (GUI), and Simulink® diagrams. Storing the framework in Bitbucket with Git version control system allows for joint research work in AP development. Furthermore, the framework generates standalone simulation environments, aiding the reproducibility of published results.

Completing the first six of the following seven steps in sequence results in a generated simulation environment:

1. *Patient parameters*: The six parameter sets presented in Table 5.1 are hard-coded. More parameter sets can be generated based on the parameter ranges presented in [85]. Furthermore, this step assesses the worst-case values for certain state variables.

2. *Observer generation*: This step uses templates to generate the observer and, optionally, the predictor algorithm code.

3. *Model definition*: This step defines the model used for controller synthesis. It may perform any of the following:

- model simplification (also applies to the observer),
- exact linearization (not in the scope of the theses, but was used in other T1DM research [Kovács et al., 2011]), and
- Scheduling variable definition for LPV representation.

The step concludes with extending the model with weighting functions presented in Section 4.4.1.

4. *Controller synthesis*: The synthesis consists of two main parts: LMI generation and solving the LMIs. The CVX toolbox [123, 124] version 2.2 is used for the latter. The synthesis may be performed once or iteratively. See Section 4.2.3 for further details.

5. *Controller evaluation*: Since the controller synthesis relies on numerical methods, the resulting controller may only satisfy some of the required constraints due to numerical inaccuracies. This step verifies whether the synthesized controller is acceptable.

6. *Simulation*: This step combines the artifacts from the previous five to create a standalone simulator. It consists of a Simulink® diagram generated from a template, observer, predictor, and controller algorithms scripts, a script to initialize and run the simulation, scenario definition, and data files storing patient-specific parameters. The generated simulator is tested by running a single simulation for each virtual patient using a set of simple meal intake scenarios.

7. *Evaluation*: The final step generates data from the results of the previous step. This data supports visualizing the capabilities of the control algorithm.

Logs are generated and stored in the case of an error, and all consecutive steps are canceled. Each of the steps above result in reusable artifacts. Different configurations may share these artifacts depending on their similarities and differences.

Figures 5.1, 5.2, and 5.3 show the main GUI. The check marks on the bottom indicate the completion state of the seven main steps listed above. The display on the left illustrates the extended model. The model changes based on configuration. Clicking on elements of the extended model results in the parameter editor on the right displaying the properties of the selected element. For example, Figure 5.1 shows the parameters of the weighting function labeled W_{p1}. Figure 5.2 displays the combined settings of model reduction and observer parameters, including the sigma point strategy and distribution of state variables.

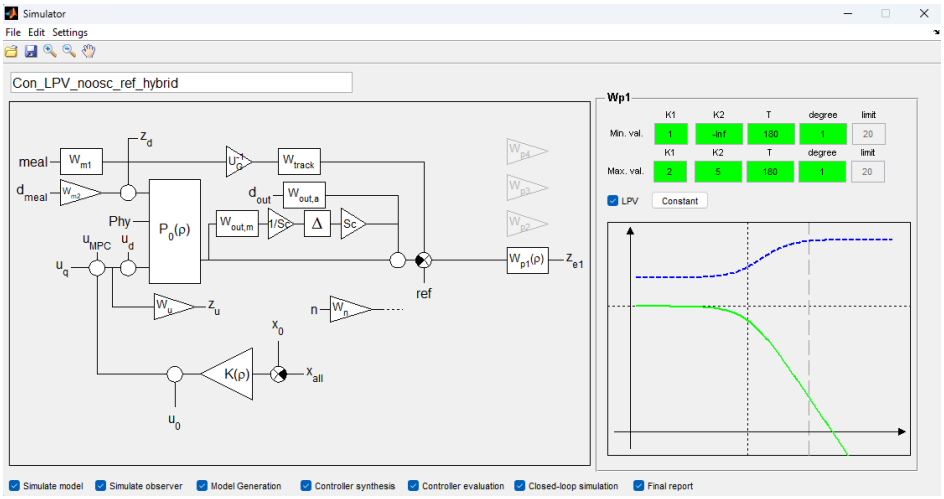

Figure 5.1. Primary GUI, with W_{p1} performance weighting function selected. The parameters indicate LPV configuration with transfer functions displayed in the bottom of the panel labeled *Wp1*. Based on the progress indicator on the bottom, all of the seven steps have been completed successfully.

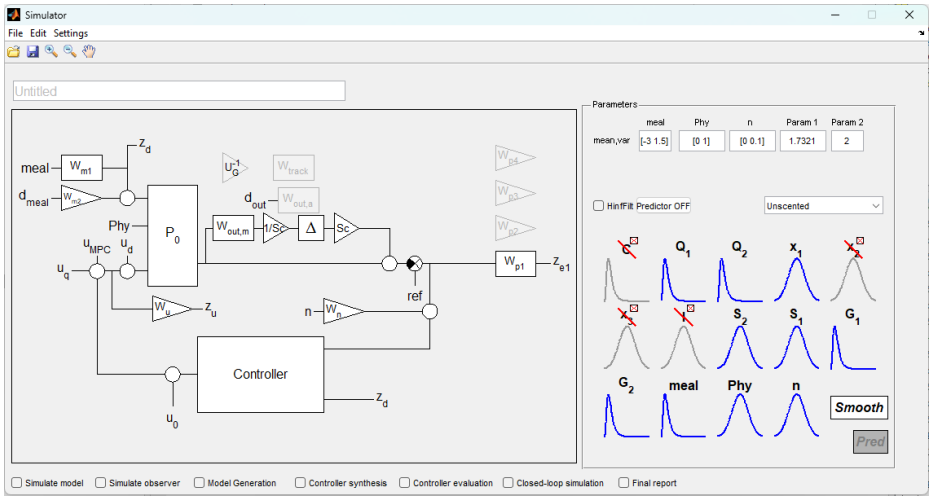

Figure 5.2. Primary GUI with the model node P_0 selected. Observer parameters and model reduction configuration is shown in the *Parameters* panel on the right. None of the seven steps have been executed.

Finally, Figure 5.3 reveals controller-specific parameters. The user can switch between state feedback and dynamic control scheme, define

norm constraints by assigning performance weighting functions, and configure scheduling variable selection for LPV control. Clicking on the illustration in the bottom right corner opens an additional GUI, revealing more settings, as presented in Figure 5.4.

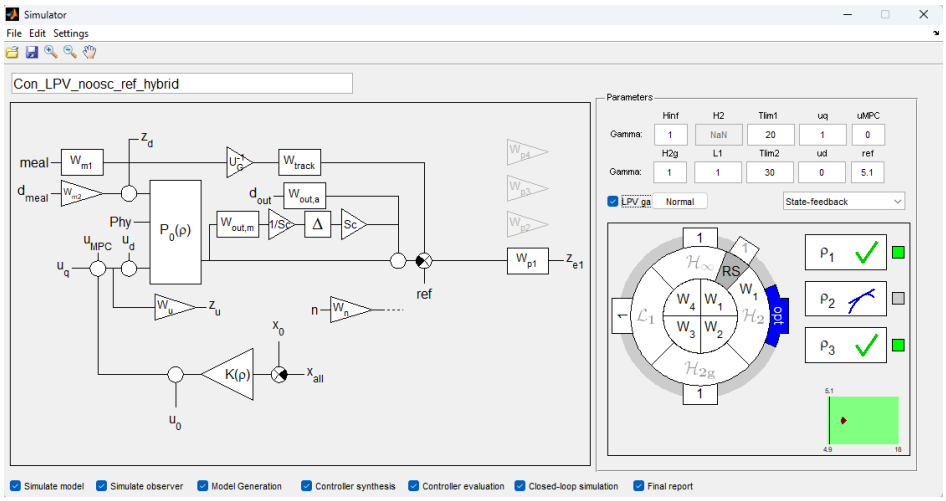

Figure 5.3. Primary GUI with the controller node $K(\rho)$ selected. The *Parameters* panel on the right display settings for a qLPV state feedback controller.

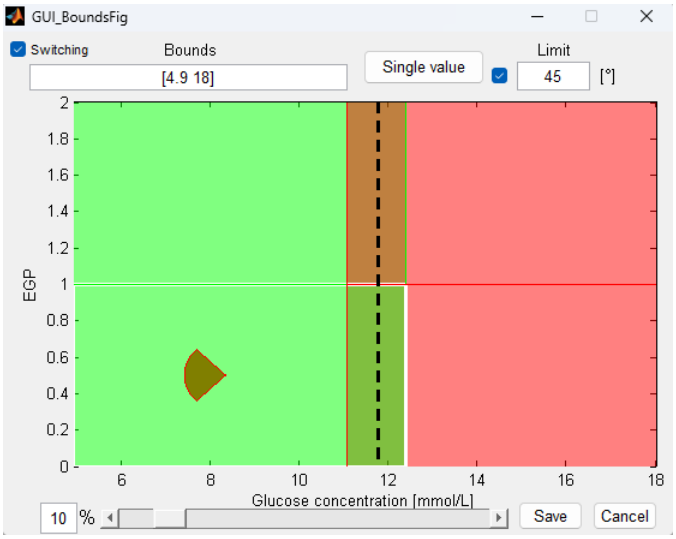

Figure 5.4. Illustration of secondary GUI with switching control and 45° constraint set on the angle of closed-loop pole pairs.

The framework enables comparing different controller configurations using the artifacts generated by the last execution step, as illustrated in Figure 5.5.

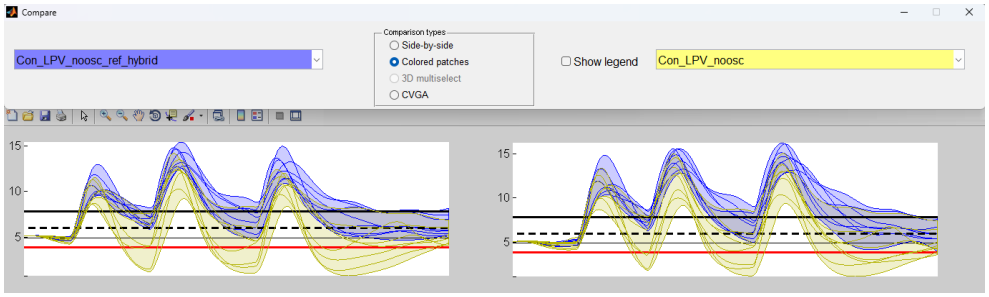

Figure 5.5. Example of comparison of different controller configurations.

Figure 5.6 shows an example of a generated Simulink® diagram. The nominal model, labeled *Extended system*, is simulated with variable step size *ode45* solver. In contrast, the observer, predictor, and controller algorithms are executed in fixed-time instances based on the sampling time of the CMG sensor by the *AP* node. Figure 5.7 displays further details on the Extended system node.

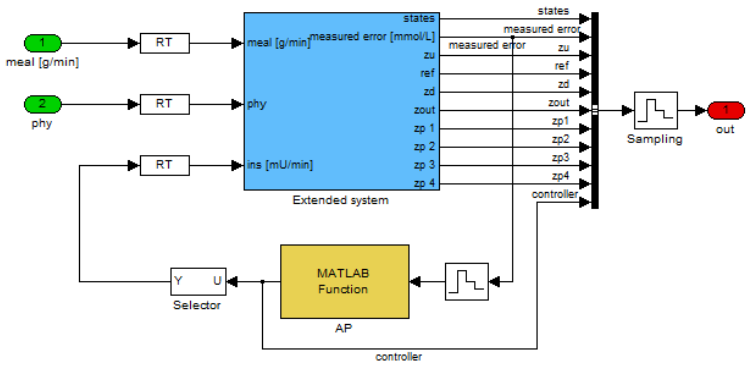

Figure 5.6. Simulink® diagram for closed-loop control simulation.

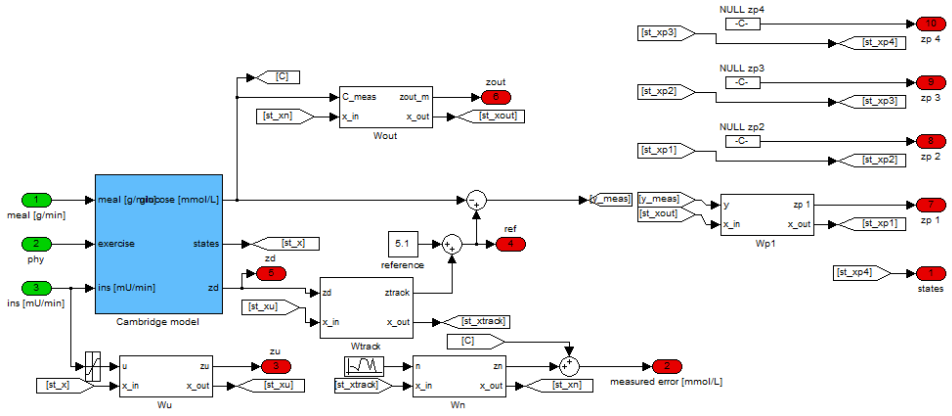

Figure 5.7. Simulink® diagram of extended model.

The subcutaneous glucose measurements are generated via a simple CGMS model that consists of an additive Gaussian white sensor noise with zero mean value and 0.1 [mmol/L] standard deviation. The sampling time of the CGMS is five minutes. For the sake of simplicity, no sensor dynamics, drift [SSBK14], or other types of sensor faults were considered.

The measurement noise and the starting phase of the time-varying parameters were randomized for each execution.

Extended Model Parameters

The state estimator and the controller use the same extended model displayed in Figure 4.6. The model reduction was patient-specific: state variables that would introduce time constants smaller than 20 minutes were removed. Additive and multiplicative uncertainty weighting functions account for the difference between the actual and simplified model. The parameters of these functions are not patient-specific and hence derived in a way to cover the effects of model simplification and time-varying parameters for all six virtual patients. Table 5.1 presents the parameter values for the weighting functions.

5.2 State Estimation Evaluation

Table 5.2 contains the parameters used for all sigma point selection strategies referenced in this Chapter. The sensitivity of these parameters was not the focus of this work, and hence the values were chosen as recommended in their corresponding publications.

Table 5.2. Parameters of the weighting functions of the extended model

Name	Transfer function	Description
$W_{out,m}$	$\frac{0.1}{30j\omega+1}$	Represents $\pm$ 10 % multiplicative output uncertainty.*
$W_{out,a}$	$\frac{0.5}{30j\omega+1}$	Represents $\pm$ 0.5 mmol/L additive output uncertainty.*
W_{ref}	$\frac{11}{U_{G,ceil}(60j\omega+1)}$	Parameter are derived from [125] and adjusted empirically.
W_{per}	$\frac{1}{180j\omega+1}$	Weighting function was tuned empirically.
W_n	0.1	Represents CGM sensor noise with 0.1 [mmol/L] standard deviation.
W_u	$\frac{S_{IE}\left(\frac{1}{k_a 4500}j\omega+1\right)}{V_I k_e\left(\frac{1}{k_a}j\omega+1\right)^2}$	Uses patient-specific parameters to avoid endogenous glucose production saturation and keep insulin infusion rate below 4500 mU/min.

* The uncertainties are only relevant on frequencies below $2\pi/30$ [rad/min].

Table 5.3. Sigma point filter parameters.

Name	Parameter	Value	Description
CDKF	h	$\sqrt{3}$	Results in the same sigma point spread as UKF with $\alpha=1$ and $\kappa=3-L$*.
GHQF	m	2	Number of sigma points for each scalar stochastic variable.
RSPF	p_0	0	This value results in the smallest sigma point spread.
SGQF	p_1	1.2556	Spread of sigma points.
	p_2	$\sqrt{3}$	
	p_3	2.857	
UKF	κ	3-L	Scaling parameter.*
	α	1	Spread of sigma points.
	β	2	Used to incorporate prior knowledge of the distribution.

*L indicates the dimension of the observed system.

The following metrics are used for the evaluation and comparison of state estimation algorithms [SMM+14]:

1. Root-mean-square error (RMSE) averaged over simulations. For the j-th state variable and N simulations consisting of M samples each:
$$R_1^{(x_j)} = \frac{1}{N}\sum_{i=1}^{N}\sqrt{\frac{1}{M}\sum_{k=0}^{M}\left(x_j^{(i)}[k] - \hat{x}_j^{(i)}[k]\right)^2} \tag{5.1}$$

2. Maximum root-mean-square error based on N simulations:
$$R_2^{(x_j)} = \max_{i=1,\ldots,N}\sqrt{\frac{1}{M}\sum_{k=0}^{M}\left(x_j^{(i)}[k] - \hat{x}_j^{(i)}[k]\right)^2} \tag{5.2}$$

3. Worst case absolute error:
$$R_3^{(x_j)} = \max_{i=1,\ldots,N}\max_{k=1,\ldots,M}\left|x_j^{(i)}[k] - \hat{x}_j^{(i)}[k]\right| \tag{5.3}$$

4. RMSE as a function of time averaged over the N simulations:
$$R_4^{(x_j)}[k] = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x_j^{(i)}[k] - \hat{x}_j^{(i)}[k]\right)^2} \tag{5.4}$$

5. A function capturing the trends in RMSE:
$$R_5^{(x_j)}[k] = \frac{1}{N}\sum_{i=1}^{N}\sqrt{\frac{1}{k}\sum_{l=0}^{k}\left(x_j^{(i)}[l] - \hat{x}_j^{(i)}[l]\right)^2} \tag{5.5}$$

The simulations used two different scenarios. One is the Oral Glucose Tolerance Test (OGTT) [126, 127], commonly used in clinical practice to diagnose type 1 diabetes mellitus. The simulation takes three hours, with a one-time 75 mg carbohydrate intake at the beginning. *Scenario 1* refers to OGTT simulation from now on.

However, OGTT does not include any insulin injections. Hence, another scenario (*Scenario 2*) is necessary. To cover the AP use case, closed-loop control provides insulin input. The meal intake protocol of this scenario has been frequently used to evaluate control algorithms [128]. It spans 24 hours and consists of three carbohydrate intakes:

- Breakfast containing 45g carbohydrate intake at 9:30.
- Lunch containing 75g carbohydrate at 13:30

Table 5.4. Comparison of sigma point filters with EKF using metrics $R_1 - R_3$ in their default configuration.

			CDKF	CKF	EKF	GHQF	RSPF	SGQF	UKF
Scenario 1	R_1	Q_1	18.86	18.83	39.68	18.82	19.41	18.81	18.81
		Q_2	10.84	10.95	47.78	10.85	10.50	10.87	11.04
		G_{flux}	0.98	0.98	1.50	0.98	0.97	0.98	0.98
	R_2	Q_1	25.75	25.42	53.40	25.84	28.00	25.51	25.77
		Q_2	33.53	32.40	106.09	32.67	31.45	32.51	32.75
		G_{flux}	1.20	1.20	1.85	1.20	1.24	1.21	1.21
	R_3	Q_1	40.84	40.77	76.80	40.86	48.66	40.73	40.96
		Q_2	45.07	43.94	149.07	44.40	40.50	44.18	44.52
		G_{flux}	3.45	3.45	3.44	3.45	3.45	3.45	3.45
	Time*		90%	120%	100%	1367%	92%	119%	91%
Scenario 2	R_1	Q_1	9.41	9.41	14.62	9.41	9.47	9.41	9.41
		Q_2	4.42	4.42	9.72	4.42	4.57	4.42	4.41
		G_{flux}	0.54	0.54	0.66	0.54	0.55	0.54	0.54
	R_2	Q_1	13.51	13.47	20.09	13.53	13.62	13.49	13.52
		Q_2	7.81	7.84	25.88	7.86	7.98	7.76	7.82
		G_{flux}	0.74	0.74	0.87	0.74	0.75	0.74	0.74
	R_3	Q_1	34.36	34.70	47.11	34.69	34.60	34.45	34.39
		Q_2	25.71	25.61	65.10	25.81	25.58	25.59	25.74
		G_{flux}	3.40	3.40	3.41	3.40	3.40	3.40	3.40

*Average execution time of a single simulation relative to EKF.

- Dinner containing 85g carbohydrate at 19:30.

Fifty simulations are executed for each virtual patient. In each case, the initial value of the estimated state vector is the equilibrium state at 5 [mmol/L] subcutaneous glucose concentration. The actual system starts with a maximum $\pm 5\%$ deviation from the initial estimation. The values are randomly generated for each simulation with uniform distribution. The metrics R_1 - R_5 combine all simulation results for all patients.

5.2.1 Sigma Point Filters with Default Parameters

The first set of simulations compares all filters with their default settings. None of the potential improvement options described in Section 3.1.7-3.1.6 are present.

Table 5.4 summarizes metrics $R_1 - R_3$ for both scenarios, while Figures 5.8 and 5.9 visualize R_4 and R_5 for *Scenario 1* and *Scenario 2* respectively. Based on Figures 5.8 and 5.9 the sigma point filters provide better estimation than Extended Kalman filter. It is not easy to distinguish between the other filters based on any of the metrics, except for the Reduced Sigma Point filter. It has slightly different error trends, providing worse estimation in some cases and better in others.

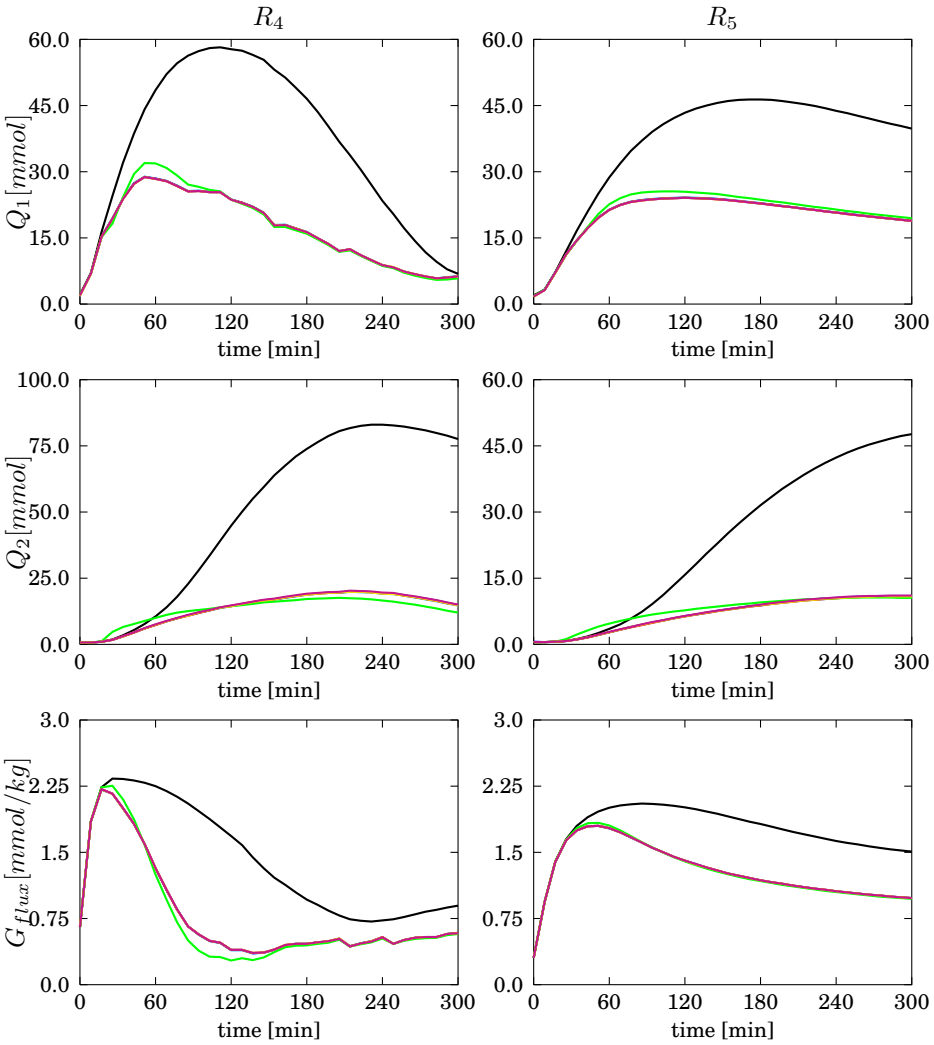

Figure 5.8. Comparison of sigma point filters with EKF on *Scenario 1* in their default configuration. Subplots on the left and right column display the R_4 and R_5 metrics. Each row of subplots displays metrics for a different state variable on the left. Each subplot displays results for all of the following filters: EKF (black), RSPF (green), UKF (magenta), CKF (magenta), CDKF (magenta), GHQF (magenta), and SGQF (magenta). Due to the large relative difference between EKF and the sigma point filters, it is difficult to distinguish between the latter five filters based on this figure.

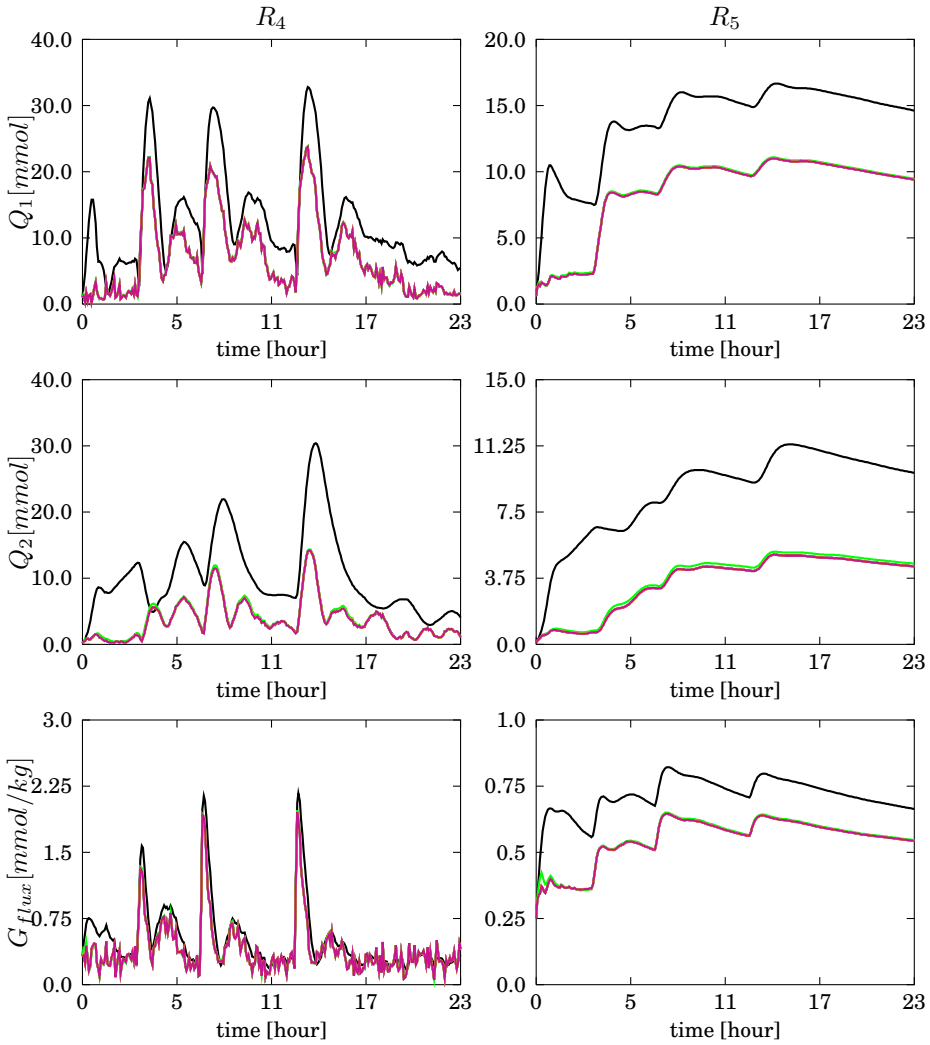

Figure 5.9. Comparison of sigma point filters with EKF on *Scenario 2* in their default configuration. Subplots on the left and right column display the R_4 and R_5 metrics. Each row of subplots displays metrics for a different state variable on the left. Each subplot displays results for all of the following filters: EKF (black), RSPF (green), UKF (magenta), CKF (magenta), CDKF (magenta), GHQF (magenta), and SGQF (magenta). Due to the large relative difference between EKF and the sigma point filters, it is difficult to distinguish between the latter five filters based on this figure.

5.2.2 Square Root Filtering

Using the filters in square root form does not intend to improve the accuracy of the filters directly. Instead, it improves their numerical stability and uses a less resource-intensive operation. Therefore, the focus was on those filters that use only nonnegative weights for calculating the covariance matrices, as presented in Table 3.1.4.

Based on the values presented in Table 5.5 there is no clear advantage or disadvantage in using a square-root filter in terms of accuracy.

5.2.3 Lognormal Distribution

The following simulations show the distribution of all glucose-specific state variables of the Cambridge model (C, Q_1, Q_2, G_1, G_1), and the meal intake input is modeled with lognormal distribution instead of normal distribution. The results are summarized in Table 5.6 as well as Figures 5.10 and 5.11.

Based on Table 5.6 in *Scenario 1* all lognormal filters provide better results than their regular counterpart, by a significant margin. EKF with lognormal transform can compete with sigma point filters with the default configuration. The only exception is the R_3 metric for G_{flux}, where the lognormal filters have a slightly larger worst-case error. In *Scenario 2* the differences are smaller but still in favor of lognormal versions of the filters, except for CKF. Lognormal CKF has worse performance than its original counterpart. Finally, RSPF did not create any valid result for either scenario. This and the poor performance of CKF is connected since these two filters have the largest spread of their sigma points among all the sigma point filters presented in this work. This larger spread of sigma points, combined with a lognormal transformation, can lead to numerical instability.

Figures 5.10 and 5.11 display the metrics R_4 and R_5 for the lognormal case. The original CKF is used as a reference, marked with a green dashed line. For *Scenario 1*, the lognormal filters show the largest difference for state variable Q_2, still a clear advantage for Q_1, and a minor difference for Q_{flux}. For *Scenario 2*, the differences are more subtle. Based on R_5, lognormal EKF and CKF perform worse than the original CKF, which also slightly outperforms GHQF and SGQF in the estimation of G_{flux} signal.

Using lognormal transformation can improve the estimation accuracy of the Cambridge model's sigma point filters and EKF alike, as long as the sigma point spread is moderate.

Table 5.5. Comparison of regular and square root form of sigma point filters using metrics $R_1 - R_3$. For each filter, left and right column show values for regular and square root version respectively. Improvements are indicated with green, while red highlights setbacks.

			CDKF		CKF		GHQF		RSPF		UKF	
Scenario 1		Q_1	18.86	18.82	18.83	18.84	18.82	18.86	19.41	19.58	18.81	18.82
	R_1	Q_2	10.84	10.92	10.95	10.85	10.85	10.84	10.50	11.09	11.04	10.95
		G_{flux}	0.98	0.98	0.98	0.98	0.98	0.98	0.97	0.97	0.98	0.98
		Q_1	25.75	25.60	25.42	25.87	25.84	25.75	28.00	26.87	25.77	25.42
	R_2	Q_2	33.53	33.31	32.40	32.16	32.67	33.55	31.45	30.71	32.75	32.39
		G_{flux}	1.20	1.20	1.20	1.20	1.20	1.20	1.24	1.24	1.21	1.20
		Q_1	40.84	40.72	40.77	41.29	40.86	40.84	48.66	45.96	40.96	40.76
	R_3	Q_2	45.07	44.96	43.94	43.58	44.40	45.09	40.50	39.99	44.52	43.93
		G_{flux}	3.45	3.45	3.45	3.45	3.45	3.45	3.45	3.44	3.45	3.45
Scenario 2		Q_1	9.41	9.41	9.41	9.41	9.41	9.40	9.47	9.35	9.41	9.41
	R_1	Q_2	4.42	4.42	4.42	4.42	4.42	4.41	4.57	4.48	4.41	4.42
		G_{flux}	0.54	0.54	0.54	0.54	0.54	0.54	0.55	0.54	0.54	0.54
		Q_1	13.51	13.54	13.47	13.54	13.53	13.51	13.62	13.49	13.52	13.49
	R_2	Q_2	7.81	7.79	7.84	7.83	7.86	7.78	7.98	7.82	7.82	7.76
		G_{flux}	0.74	0.74	0.74	0.74	0.74	0.73	0.75	0.74	0.74	0.74
		Q_1	34.36	34.49	34.70	34.39	34.69	34.43	34.60	34.74	34.39	34.46
	R_3	Q_2	25.71	25.76	25.61	25.72	25.81	25.56	25.58	25.24	25.74	25.58
		G_{flux}	3.40	3.39	3.40	3.40	3.40	3.40	3.40	3.44	3.40	3.40

*Average execution time of square root execution relative to the regular version.

Table 5.6. Comparison of sigma point filters with and without lognormal transformation using metrics $R_1 - R_3$. For each filter, left and right column show values for regular and lognormal version respectively. Improvements are indicated with green, while red highlights setbacks

			CDKF		CKF		EKF		GHQF		SGQF		UKF	
Scenario 1	R_1	Q_1	18.86	13.74	18.83	14.11	39.68	15.83	18.82	13.99	18.81	13.90	18.81	14.12
		Q_2	10.84	5.35	10.95	4.88	47.78	11.80	10.85	5.84	10.87	5.17	11.04	5.07
		G_{flux}	0.98	0.72	0.98	0.69	1.50	1.50	0.98	0.84	0.98	0.72	0.98	0.66
	R_2	Q_1	25.75	19.65	25.42	20.13	53.40	27.87	25.84	20.18	25.51	19.81	25.77	20.08
		Q_2	33.53	16.48	32.40	15.81	106.09	42.78	32.67	19.44	32.51	17.59	32.75	16.60
		G_{flux}	1.20	1.00	1.20	0.98	1.85	1.85	1.20	1.09	1.21	0.99	1.21	0.96
	R_3	Q_1	40.84	32.23	40.77	32.75	76.80	71.92	40.86	34.36	40.73	32.83	40.96	31.26
		Q_2	45.07	23.95	43.94	23.07	149.07	62.81	44.40	28.10	44.18	25.44	44.52	24.00
		G_{flux}	3.45	3.46	3.45	3.51	3.44	3.54	3.45	3.45	3.45	3.46	3.45	3.47
Scenario 2	R_1	Q_1	9.41	8.10	9.41	9.64	14.62	12.21	9.41	9.18	9.41	8.29	9.41	7.94
		Q_2	4.42	4.33	4.42	4.76	9.72	6.09	4.42	4.54	4.42	4.42	4.41	4.32
		G_{flux}	0.54	0.50	0.54	0.64	0.66	0.71	0.54	0.53	0.54	0.54	0.54	0.50
	R_2	Q_1	13.51	11.81	13.47	13.03	20.09	16.71	13.53	13.09	13.49	11.88	13.52	11.72
		Q_2	7.81	7.18	7.84	8.52	25.88	11.74	7.86	7.68	7.76	7.51	7.82	7.03
		G_{flux}	0.74	0.64	0.74	0.95	0.87	0.88	0.74	0.70	0.74	0.76	0.74	0.64
	R_3	Q_1	34.36	30.74	34.70	37.31	47.11	48.88	34.69	32.81	34.45	31.58	34.39	31.02
		Q_2	25.71	22.24	25.61	26.12	65.10	38.06	25.81	22.05	25.59	23.25	25.74	22.45
		G_{flux}	3.40	3.52	3.40	3.39	3.41	8.07	3.40	3.42	3.40	3.26	3.40	3.53

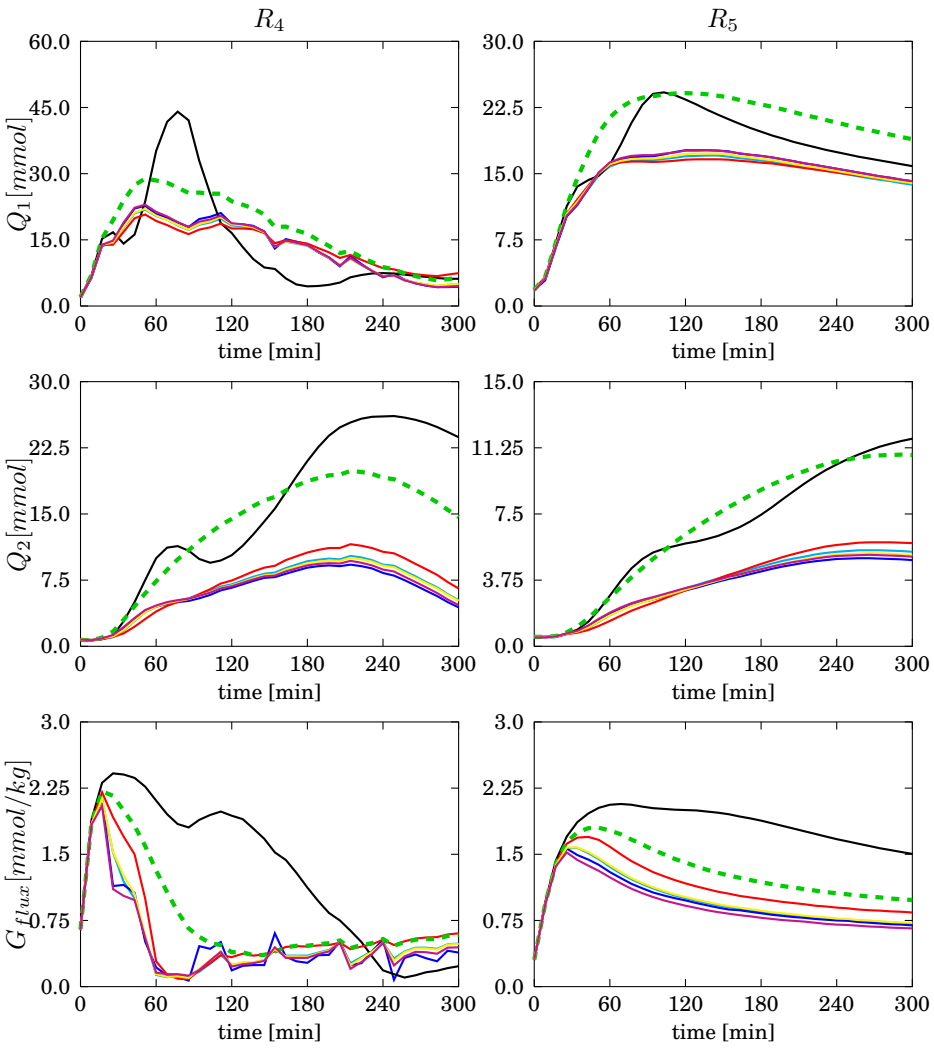

Figure 5.10. Comparison of sigma point filters using lognormal transformation on *Scenario 1*. Subplots on the left and right column display the R_4 and R_5 metrics. Each row of subplots displays metrics for a different state variable indicated on the left. Each subplot displays results for all of the following filters: EKF (black), CKF (blue), UKF (magenta), CDKF (cyan), GHQF (red), and SGQF (yellow). CKF in default configuration is used as a reference displayed with a dashed green line.

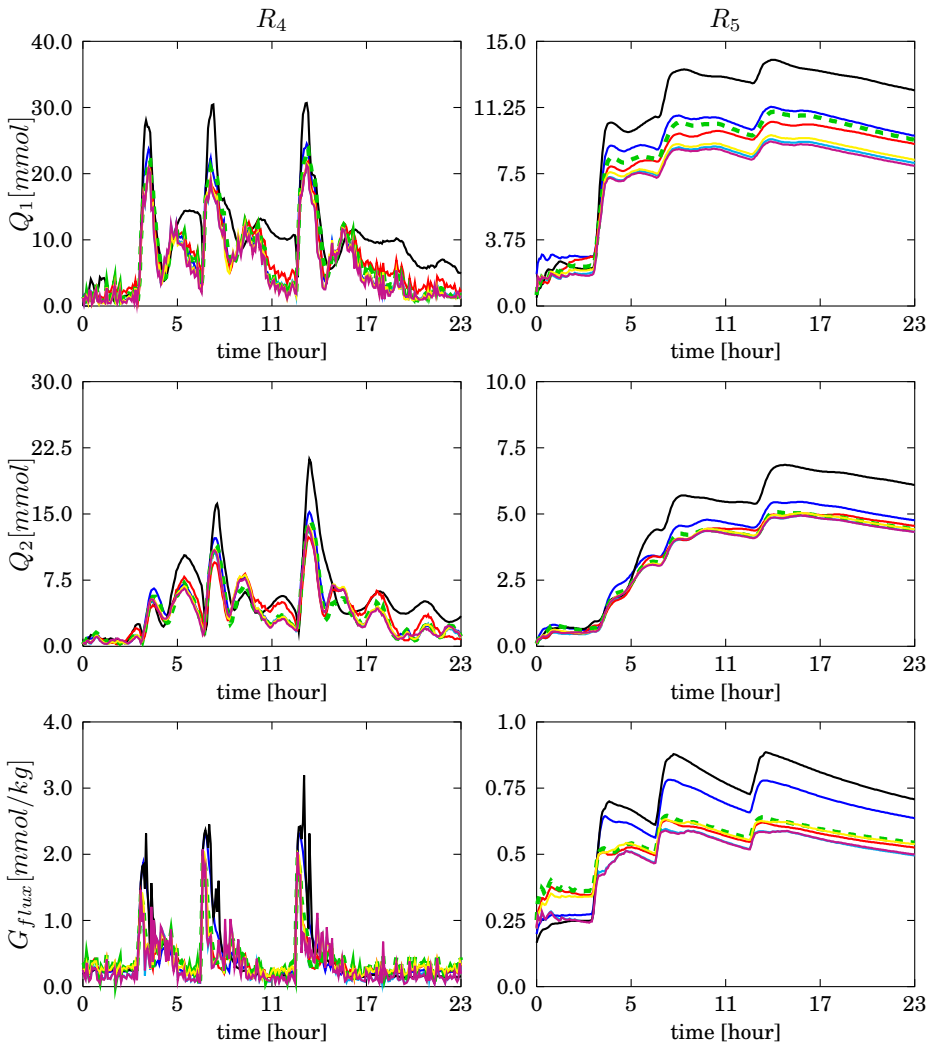

Figure 5.11. Comparison of sigma point filters using lognormal transformation on *Scenario 2*. Subplots on the left and right column display the R_4 and R_5 metrics. Each row of subplots displays metrics for a different state variable on the left. Each subplot displays results for all of the following filters: EKF (black), CKF (blue), UKF (magenta), CDKF (cyan), GHQF (red), and SGQF (yellow). CKF in default configuration is used as a reference, displayed with a green dashed line.

Table 5.7. Comparison of sigma point filters with and without extensive transform using metrics $R_1 - R_3$. For each filter, left and right column show values for regular and extensive version respectively. Improvements are indicated with green, while red highlights setbacks

			CKF		RSPF		SGQF		UKF	
Scenario 1	R_1	Q_1	14.11	13.40	19.41	18.53	13.90	13.09	14.12	13.36
		Q_2	4.88	5.19	10.50	11.59	5.17	5.24	5.07	4.67
		G_{flux}	0.69	0.70	0.97	0.98	0.72	0.71	0.66	0.70
	R_2	Q_1	20.13	18.95	28.00	24.91	19.81	18.70	20.08	19.02
		Q_2	15.81	17.24	31.45	34.63	17.59	18.46	16.60	15.41
		G_{flux}	0.98	0.97	1.24	1.18	0.99	0.95	0.96	1.01
	R_3	Q_1	32.75	28.84	48.66	39.64	32.83	27.97	31.26	31.40
		Q_2	23.07	25.24	40.50	47.63	25.44	27.38	24.00	23.43
		G_{flux}	3.51	3.34	3.45	3.44	3.46	3.38	3.47	3.42
Scenario 2	R_1	Q_1	9.64	7.88	9.47	9.23	8.29	8.15	7.94	11.51
		Q_2	4.76	4.26	4.57	4.33	4.42	4.35	4.32	5.19
		G_{flux}	0.64	0.51	0.55	0.54	0.54	0.54	0.50	0.72
	R_2	Q_1	13.03	11.53	13.62	13.29	11.88	11.91	11.72	15.42
		Q_2	8.52	7.35	7.98	7.66	7.51	7.55	7.03	9.70
		G_{flux}	0.95	0.66	0.75	0.73	0.76	0.76	0.64	1.06
	R_3	Q_1	37.31	29.92	34.60	34.24	31.58	29.87	31.02	42.54
		Q_2	26.12	24.53	25.58	26.49	23.25	24.62	22.45	31.48
		G_{flux}	3.39	3.50	3.40	3.38	3.26	3.25	3.53	3.36

5.2.4 Extensive Transform

Table 5.7 summarizes the difference between some of the sigma point filters with and without extensive transform, which is presented in Section 3.1.6. CDKF and GHKF have no extensive version. CDKF uses a different method for calculating covariance matrix than other sigma point filters. On the other hand, GHQF is already as if each state would have an individual extensive transform. All sigma point filters in Table 5.7 are in their lognormal form, except for RSPF.

Based on *Scenario 1* extensive transform improves the accuracy for state variable Q_1 at the cost of estimation accuracy of Q_2. UKF is an exception, where both Q_1 and Q_2 improve at the cost of G_{flux}.

However, the results are different for *Scenario 2*. Extensive transform is mostly advantageous for CKF and RSPF. In contrast, there is little to no improvement in SGQF, and UKF performs significantly worse when applying extensive transform.

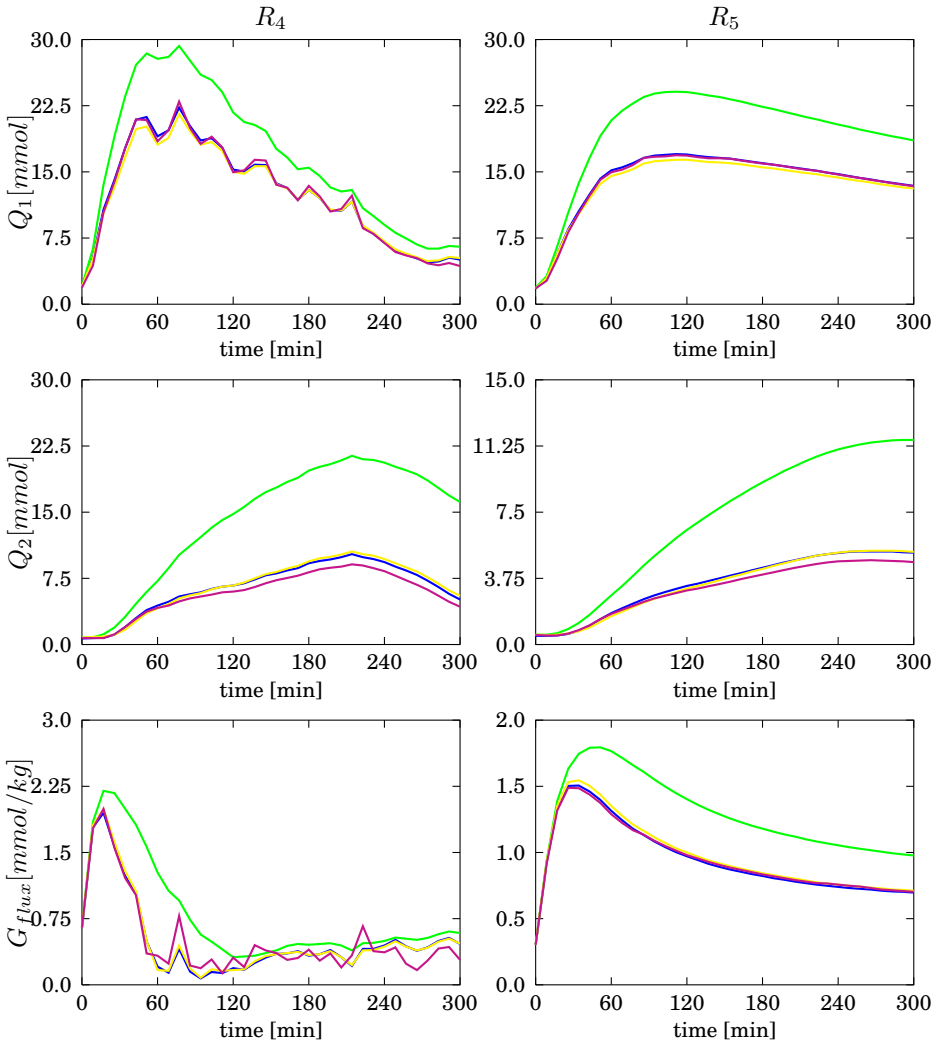

Figure 5.12. Comparison of sigma point filters using extensive transform on *Scenario 1*. Subplots on the left and right column display the R_4 and R_5 metrics. Each row of subplots displays metrics for a different state variable on the left. Each subplot displays results for all of the following filters: CKF (blue), UKF (magenta), RSPF (green), and SGQF (yellow). UKF without extensive transform is used as a reference, displayed with a solid green line.

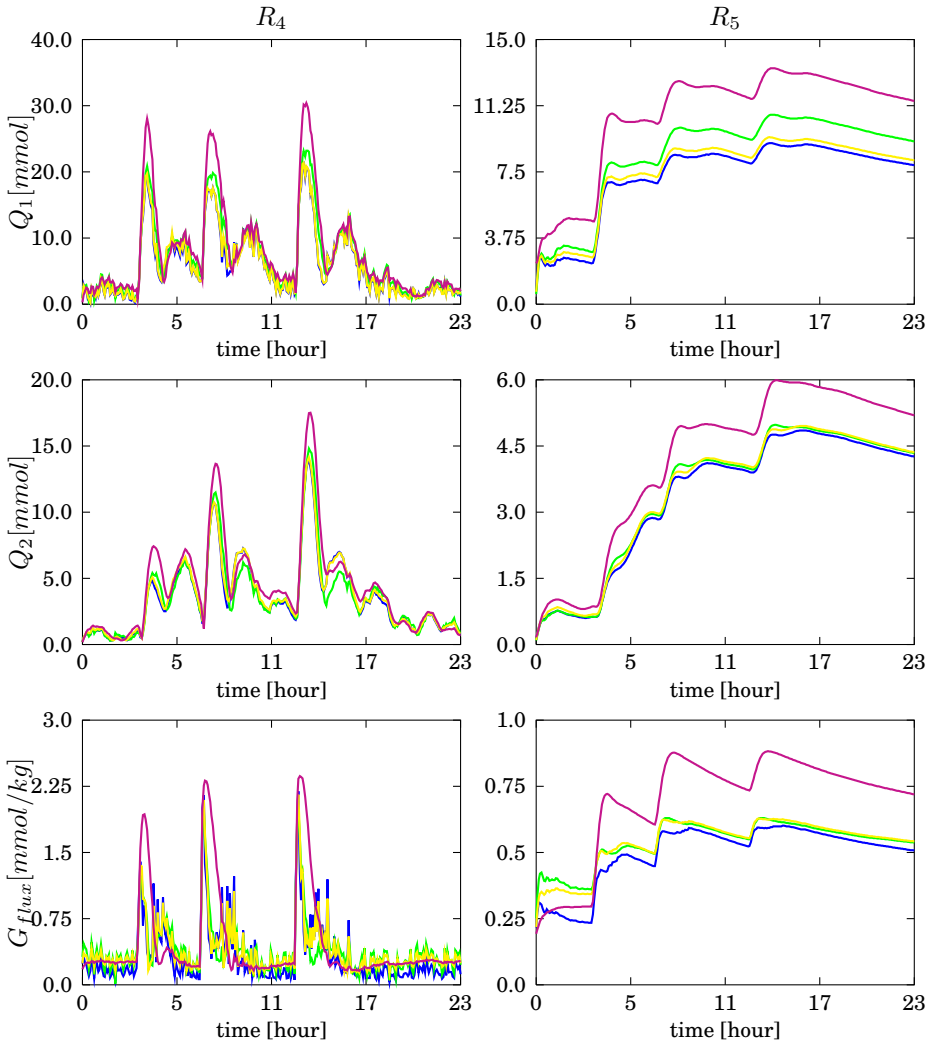

Figure 5.13. Comparison of sigma point filters using extensive transform on *Scenario 2*. Subplots on the left and right column display the R_4 and R_5 metrics. Each row of subplots displays metrics for a different state variable on the left. Each subplot displays results for all of the following filters: CKF (blue), UKF (magenta), RSPF (green), and SGQF (yellow). UKF without extensive transform is used as a reference, displayed with a solid green line.

5.2.5 Summary

Based on the simulation results, using lognormal transformation has a clear advantage for EKF and sigma point filters as long as the latter has a moderate spread of sigma points. However, perhaps using a different distribution can help overcome this limitation while preserving the nonnegativity of the state variables and disturbances.

Increasing the number of sigma points beyond a specific limit does not provide a significant advantage in terms of accuracy. However, the more sigma points a filter uses, the more computational power it requires, which is a disadvantage when running on embedded hardware. Therefore, using at least $2L$ sigma points in a symmetric configuration is sufficient, where L is the dimension of the model, including states, disturbances, and noises.

A high number of sigma points are even more demanding for square-root filtering since it requires the QR decomposition of a matrix that scales linearly with the number of sigma points. Square-root filtering itself did not provide significantly better accuracy in these simulations. However, based on [108], it has practical advantages in implementing an artificial pancreas in terms of numerical stability.

5.3 Controller Evaluation

For easy comparison with other control methods, the simulations were done using two commonly used meal intake scenarios. This allows easier comparison with other published methods. Both scenarios span 24 hours.

1. *Scenario 3*: 150 g of carbohydrate (CHO) intake per day. It consists of a 35 g CHO breakfast at 8:30, a 65 g CHO lunch at 13:00, and 50 g CHO dinner at 19:00.

2. *Scenario 4*: The meal intake protocol presented in [128], which was also used in Section 5.2 It consists of a 45 g CHO breakfast at 9:30, a 75 g CHO lunch at 13:30, and 85 g CHO dinner at 19:30.

The control algorithm does not announce the timing and amount of CHO intake. The main focus was meal intake; hence, none of the scenarios contain deterministic nor stochastic physical activity signals. The control algorithm provides the injected insulin, no manual intervention or open loop protocol is applied.

Control variability grid analysis (CVGA) is used to visualize and compare different controller configurations. It is a commonly used tool in AP development [128]. The x and y axis are the minimum and maximum glucose levels throughout the simulations, respectively, in

[mg/dL]. A single symbol (dot, square, etc.) represents a single simulation. The plot is divided into different areas with a letter assigned from *A* to *E*. Controllers that can keep their patients in regions *A* and *B* are considered sufficient for AP use. Minimum and maximum glucose is limited to 50 and 400 mg/dL respectively.

The model used for controller synthesis, unless stated otherwise, is as described in Section 4.4.1. The constant reference signal r_0 is increased from 4.9 to 5.1 mmol/L to lessen the chance of hypoglycemia further.

5.3.1 Linear $\mathcal{H}_\infty$ Controller

This is the simplest configuration presented in this section. Instead of the LPV representation, the Cambridge model is simply approximated with a linear system at the desired r_0 subcutaneous glucose concentration. The output multiplicative weighting function is increased from 10% to 25% to compensate for the model simplification. Furthermore, the extended model does not contain W_{ref} reference dynamics.

Figure 5.14 displays the simulation results for both dynamic controller and state feedback. Due to the linearization of the model at a lower glucose level, both controllers assume slow dynamics, which does not hold for postprandial glucose levels. However, the behavior of the two controller configurations is different. The state feedback controller administers more insulin than necessary, leading to lower maximum values and severe hypoglycemia (below 50 mg/dL). In contrast, the dynamic controller avoids hypoglycemia at the cost of higher maximum glucose levels.

Although both controllers could be adjusted for better overall behavior, the goal here is not to provide an optimal linear configuration. Instead, the aim is to highlight the effect of selected structural changes. Hence, the parameters of weighting functions are mostly the same for all versions.

5.3.2 qLPV $\mathcal{H}_\infty$ Controller

Figure 5.15 and Figure 5.16 present the CVGA comparison of linear and qLPV controllers for state feedback and dynamic controller respectively. Two out of six patients reached B zone for the state feedback controller or came very close for both meal scenarios.

For the dynamic controller, the maximum glucose levels are decreased, pushing the virtual patients from upper regions to B zone. However, two patients in both scenarios have severe hypoglycemia. The controller synthesis of those two patients was numerically poorly conditioned.

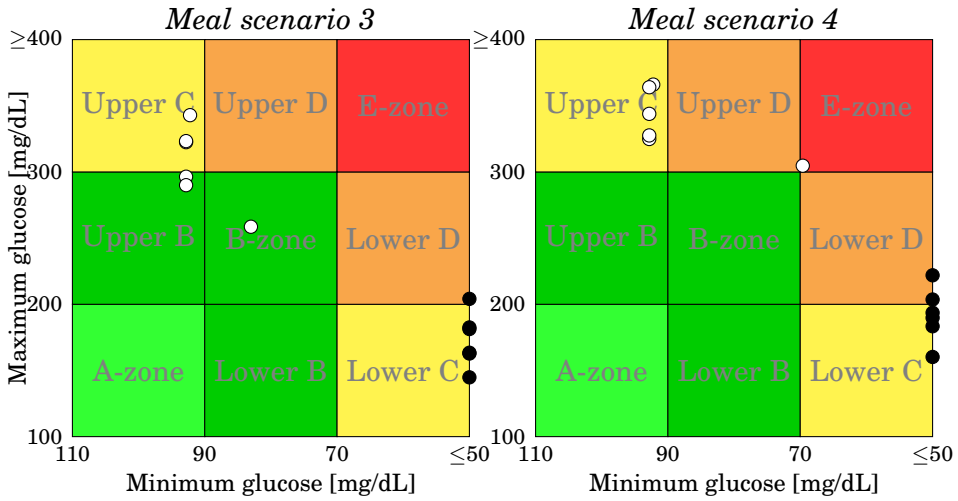

Figure 5.14. CVGA for linear $\mathcal{H}_\infty$ controllers for *meal scenario 3* (left) and *meal scenario 4* (right). Black and white circles represent simulations using state-feedback and dynamic controller respectively. The minimal glucose levels are limited to 50 mg/dL, but actual values may be lower.

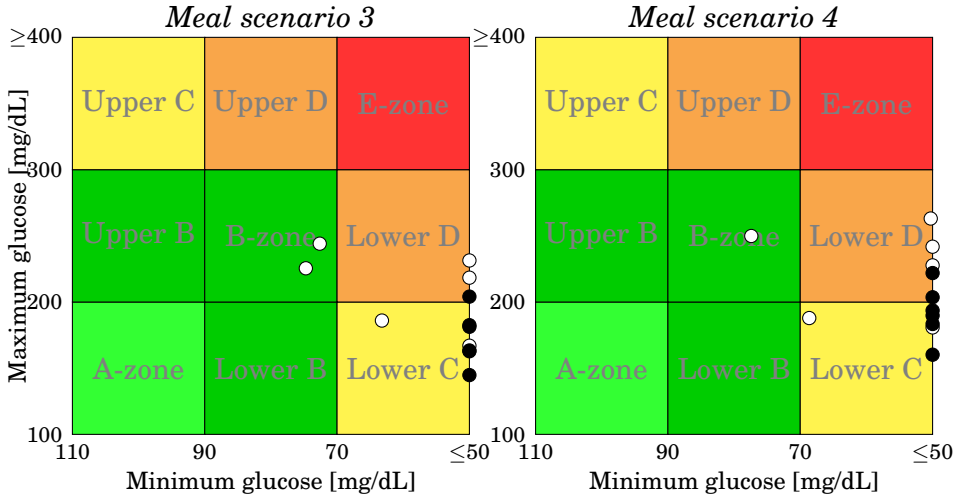

Figure 5.15. CVGA comparison of linear (black circles) and qLPV $\mathcal{H}_\infty$ (white circles) state feedback controllers. Left and right subplots show results for *meal scenario 3* and *meal scenario 4* respectively. Minimum glucose levels are limited to 50 mg/dL. Actual values may be lower.

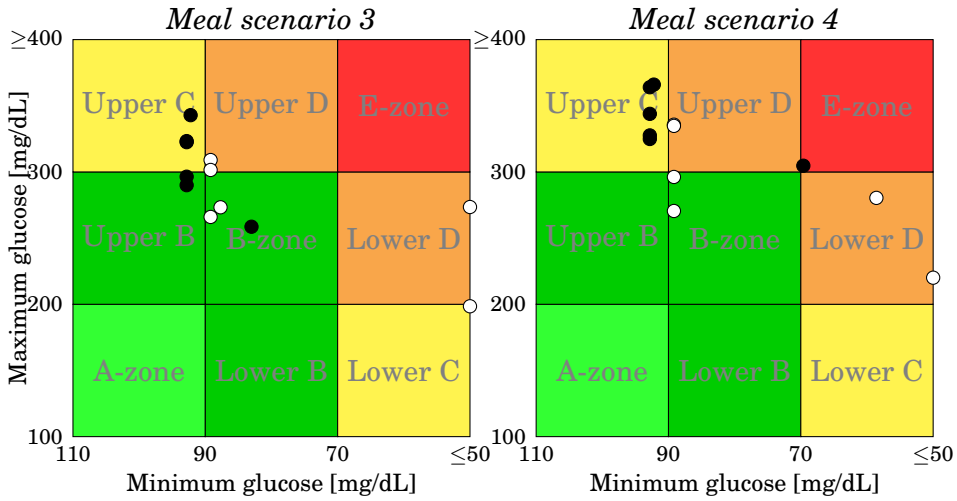

Figure 5.16. CVGA comparison of linear (black circles) and qLPV $\mathcal{H}_\infty$ (white circles) dynamic controllers. Left and right subplots show results for *meal scenario 3* and *meal scenario 4* respectively.

5.3.3 Reference Dynamics

Reference Dynamics can help to avoid hypoglycemic episodes. Instead of using a constant reference signal, the estimated glucose flux from meal intake is propagated through a weighting function to mimic a healthy individual's postprandial glucose concentration trend. An example of this reference signal is displayed in Figure 5.17. Figure 5.18 illustrates the difference between static and dynamic reference signal.

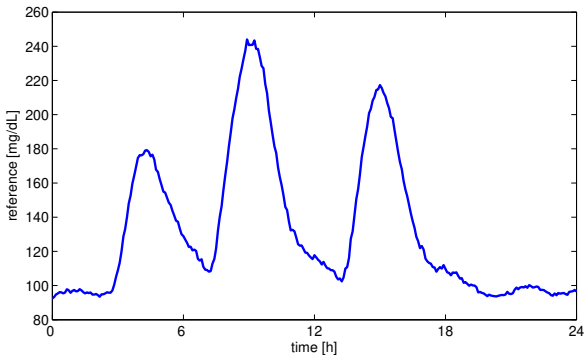

Figure 5.17. Example of reference signal generated with reference dynamics weighting function

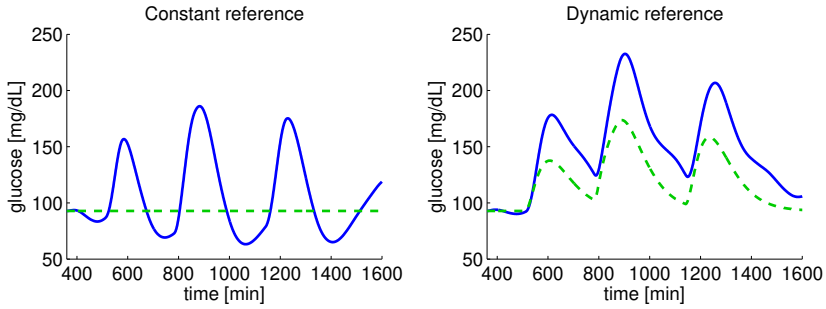

Figure 5.18. Difference between static and dynamic reference signal.

The simulation results for state feedback and dynamic controller are displayed in Figures 5.19 and 5.20 respectively. The hypoglycemic episodes have decreased for the state feedback controller but have not been eliminated. The reference dynamics weighting functions prevent hypoglycemia for the dynamic controller but increase the maximum glucose levels, pushing patients towards *Upper C* zone. A dynamic controller generally provides more consistent results than state feedback, but the latter provides better results overall.

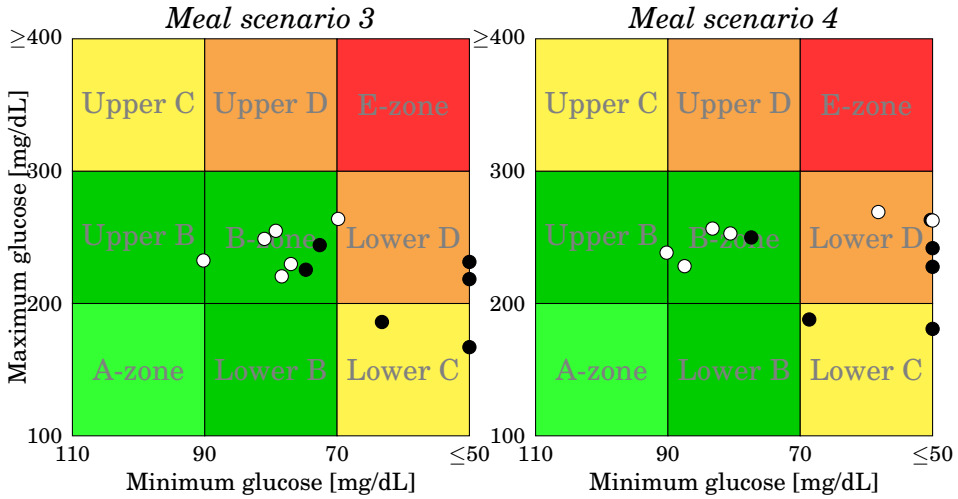

Figure 5.19. CVGA comparison of state feedback qLPV $\mathcal{H}_\infty$ controllers with and without W_{ref} reference dynamics. Left and right subplots show results for *meal scenario 3* and *meal scenario 4*, respectively. Black circles represent simulations without W_{ref}. White circles indicate simulations of with W_{ref}.

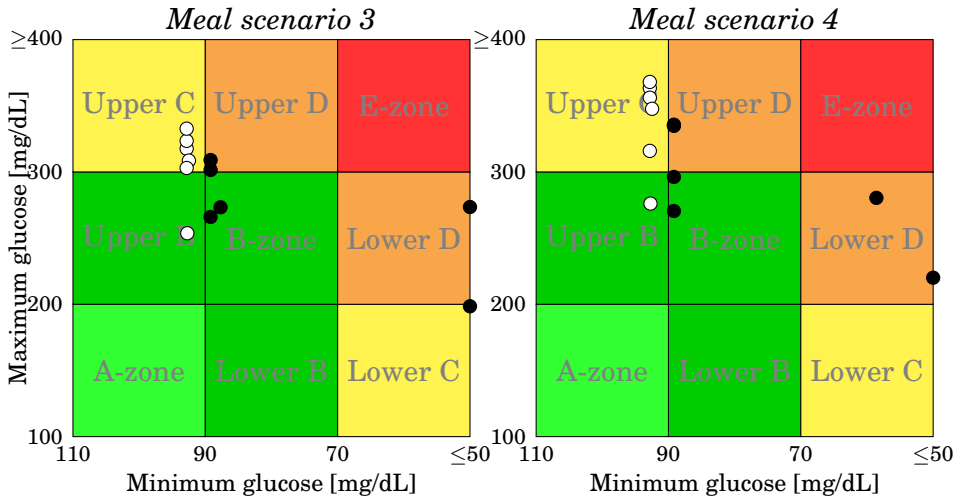

Figure 5.20. CVGA comparison of dynamic qLPV $\mathcal{H}_\infty$ controllers with and without W_{ref} reference dynamics. Left and right subplots show results for *meal scenario 3* and *meal scenario 4*, respectively. Black circles represent simulations without W_{ref}. White circles indicate simulations of with W_{ref}.

5.3.4 Hybrid Norm Approach

Up to this point, all controllers used exclusively $\mathcal{H}_\infty$ norm for both robust stability and nominal performance. Hence, the final round of simulations compares different hybrid norm controllers.

$\mathcal{H}_\infty$ norm constraints still ensure robust stability, but each symbol in Figure 5.21 and Figure 5.22 represent a controller where nominal performance use one of the following norms: $\mathcal{H}_2$, $\mathcal{H}_{2g}$, or $\mathcal{L}_1$. The former figure displays results for state feedback control while the latter shows dynamic control.

For state feedback control, the use of $\mathcal{H}_2$, $\mathcal{H}_{2g}$ or $\mathcal{L}_1$ norm help to avoid hypoglycemia. However, only $\mathcal{H}_2$ and $\mathcal{H}_{2g}$ keep all virtual patients within B zone. Using $\mathcal{L}_1$ norm leads to higher minimum glucose levels at the cost of higher maximum. At least half of the patients are in *Upper C* zone for both meal scenarios.

In a dynamic controller's case, using different norms does not bring significant benefits. There are only slight differences between norms, except for $\mathcal{L}_1$ norm, which leads to higher maximum glucose levels, similar to the state feedback control.

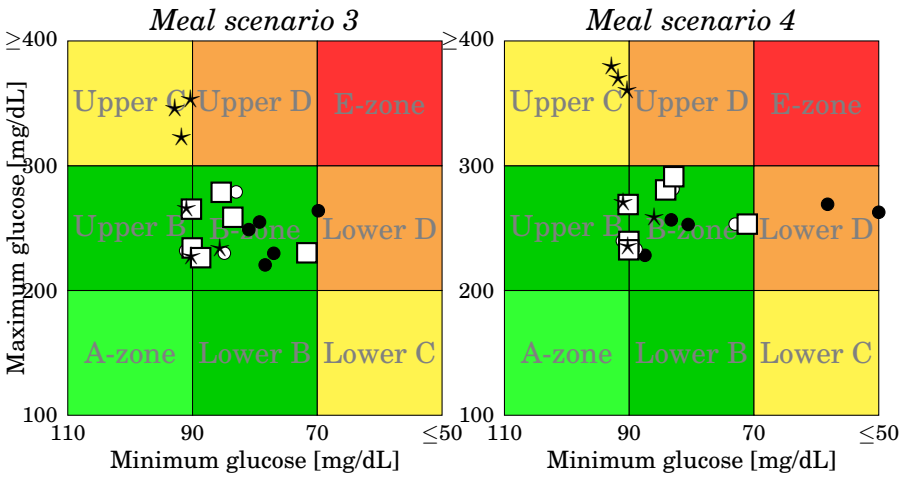

Figure 5.21. CVGA comparison of state feedback controllers using different norms for nominal performance. Left and right subplots show results for *meal scenario 3* and *meal scenario 4*, respectively. The used symbols are as follows: black circles denote $\mathcal{H}_\infty$ norm, white circles denote $\mathcal{H}_2$ norm, white squares denote $\mathcal{H}_{2g}$ norm, and black stars denote $\mathcal{L}_1$ norm.

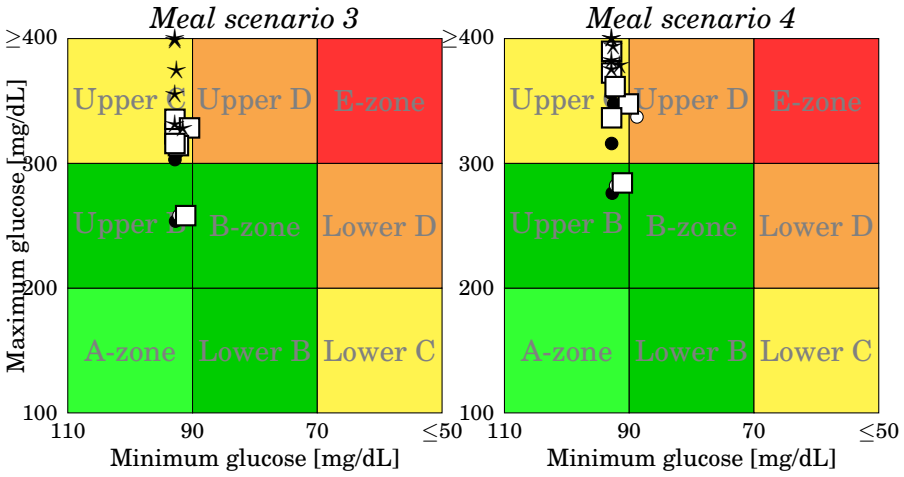

Figure 5.22. CVGA comparison of dynamic controllers using different norms for nominal performance. Left and right subplots show results for *meal scenario 3* and *meal scenario 4*, respectively. The used symbols are as follows: black circles denote $\mathcal{H}_\infty$ norm, white circles denote $\mathcal{H}_2$ norm, white squares denote $\mathcal{H}_{2g}$ norm, and black stars denote $\mathcal{L}_1$ norm.

To put these results into perspective, some of the recently published control methods reported the following CVGA outcomes:

- The sliding mode controller of Beneyto et al. [27] achieved 42.8 mg/dL average minimum glucose levels without and 59.2 mg/dL with meal intake announcements.
- The nonlinear state feedback control of Cai et al. [29] kept the patients in the *A* and *B* zones when meal intakes were announced. However, without meal intake information, some patients achieved maximum glucose levels above 300 mg/dL or below 70 mg/dL.
- Some patients were in the lower *D* zone with both the single and the dual hormone configuration of the Deep Reinforcement Learning based method presented by Zhu et al. [30].
- The multivariate Bayesian optimization-assisted parameter adaptation framework proposed by Shi et al. [34] kept the vast majority of the patients in the *B* zone. However, some patients still entered the *D* and *E* zones.
- The robust MPC presented by Siket et al. [46] provided similarly efficient results to [34], but the outliers improved to the lower *C* zone instead.
- The robust LPV approach of Mirzaee et al. [50] kept all patients in the *A* and *B* zones but partially relied on announced meal intakes.
- The MPC presented by Shi et al. [38] kept most of the patients in the most optimal *A* zone, with the use of CHO intake profiles of the patients.

It is important to note that an accurate comparison of these methods is difficult due to the differences in simulation environments and scenarios. Furthermore, some of these scenarios include physical activity and insulin sensitivity changes due to circadian rhythm, which was outside the scope of the in silico validation presented in this Chapter.

5.3.5 Summary

Based on control variability grid analysis, the combination of qALPV state feedback with reference dynamics and robust hybrid $\mathcal{H}_2/\mathcal{H}_\infty$ or $\mathcal{H}_{2g}/\mathcal{H}_\infty$ controller could achieve satisfactory blood glucose control for two commonly used meal intake scenarios.

With the dynamic controller, the same results could not be achieved. All presented controller configurations lead to high maximum glucose levels, exposing the patients to potentially severe hyperglycemia. The

controller synthesis for the dynamic controller is also more difficult to solve from a numerical point of view. Hence, it is recommended to use state feedback control over a dynamic controller.

6

THESES

Thesis Group 1 (Chapter 3)

I provided a new state observer framework for estimating the state variables of nonlinear T1DM models and the glucose flux resulting from meal intake. The state observer considers the measurement noise of CGM sensors, the nonlinearity, uncertainty, and nonnegativity of the model, and the glucose utilization resulting from physical activity.

Square root sigma point filters can provide a satisfactory estimation of the state variables of T1DM models, combined with meal intake and uncertainty dynamics. The meal intake, physical activity, and the estimation error of state variables directly or indirectly affected by these disturbances should be modeled with lognormal distribution.

Publications related to the theses are: [SEK14, SMM$^+$14, SSBK14, KS16, SBK16, SDKew].

Thesis 1.1

I provided a method for a generic stochastic state estimation algorithm to consider the nonnegativity of the model and the most significant disturbances: meal intake and physical activity.

Let $\mathcal{T}_x$ denote the transformation of selected state variables to their natural logarithm: $\varkappa_{i,k} = \ln x_{i,k}$, where i is the index of a single state variable in the state vector x and k indicates time. Let $\mathcal{T}_w$ denote the same transformation for disturbances. The transformed discrete time state space T1DM model used during the estimation is as follows:

$$\begin{aligned} \varkappa_k &= \mathcal{T}_x \mathrm{x}_k \\ \varkappa_{k+1} &= \mathcal{T}_x \mathrm{f}(\mathcal{T}_x^{-1}\varkappa_k, \mathcal{T}_w^{-1}\mathrm{w}_k, k) \\ \mathrm{y}_k &= \mathrm{h}(\mathcal{T}_x^{-1}\varkappa_k, \mathrm{z}_k, k). \end{aligned} \tag{6.1}$$

Publication related to this thesis: [SMM+14, SBK16, SDKew].

Thesis 1.2

I provided a new state observer framework that considers and estimates the additive and multiplicative output uncertainty of a T1DM model.

Extending the nominal T1DM model with uncertainty weighting functions driven by white noise can account for modeling uncertainties and intra-patient variability. Furthermore, this approach enables robust state feedback control that relies on the same weighting functions.

Publications related to this thesis: [SEK14, KS16, SDKew].

Thesis 1.3

I provided a state observer framework that can estimate the glucose flux resulting from meal intake and ingestion.

Extending the nominal T1DM model with a dynamic meal ingestion subsystem, driven by white noise with lognormal distribution, can provide a reliable estimation of the glucose flux resulting from meal intake, especially if the state observer algorithm is a sigma point filter.

Publications related to this thesis: [SEK14, SMM+14, SSBK14].

Thesis 1.4

I proposed a predictor algorithm that can provide a long-term prediction for a nonlinear and uncertain T1DM model using planned meal intake announcements.

The predicted distribution of the model output can be used to validate meal intake announcements, detect unannounced events, or early detection of potential hypoglycemic episodes.

The publication related to this thesis: [SBK16].

Thesis Group 2 (Chapter 4)

I provided a new robust nonlinear control algorithm for Artificial Pancreas. The controller addresses the nonlinearity, nonnegativity, and intra-patient variability of the glucose-insulin interaction in a T1DM patient.

The controller is realized via a quasi linear parameter-varying state feedback. The state observer proposed in Thesis 1 supplies the estimated state and scheduling variables. By extending the nominal model with appropriate weighting functions, the controller can be configured to meet robustness criteria, avoid severe hypoglycemia, and rely on a nonnegative control signal.

Publications related to the theses are: [KSZ11, KSF+11, KTSS12, KS12, KSF+12, KKSE13, SEK+13, KSAB13, KSS+13, KKS+14, SEK14, KS16, SDKew].

Thesis 2.1

I provided a linear parameter varying approximation of the well-known Cambridge T1DM model, which enabled the synthesis of quasi linear parameter-varying controllers via the solution of linear matrix inequalities.

Out of the four potential candidates, two glucose-related state variables are chosen as scheduling variables. The third scheduling variable is a Hill function of one of the first two scheduling variables. Saturation of endogenous glucose production is avoided by setting an additional constraint on the control signal. At the same time, the nonlinearity of the renal extraction is resolved by replacing it with a less conservative disturbance. The resulting model contains only matrix-valued affine functions of the scheduling variables, making it a linear parameter-varying representation.

Publications related to this thesis: [SEK+13, KSAB13, SEK14, KS16, SDKew].

Thesis 2.2

I provided a $\mathcal{H}_2/\mathcal{H}_\infty$ controller for Artificial Pancreas, which considers model uncertainties and intra-patient variability, as well as constraints on the control signal.

The controller synthesis is performed by solving appropriate linear matrix inequalities. Furthermore, the nominal T1DM model used for the controller synthesis is extended with weighting functions.

Publications related to this thesis: [KSZ11, KSF+11, KS12, KSF+12, KKSE13, KSS+13, KKS+14, SEK14, KS16, SDKew].

Thesis 2.3

I provided a method to automatically scale performance and multiplicative uncertainty outputs of a T1DM model so that robust stability constraints are ensured and nominal performance is optimized, as long as they are feasible.

I applied changes to particular well-known linear matrix inequalities commonly used for the synthesis of linear parameter-varying controllers. The changes enable automatic scaling for performance and uncertainty outputs for a state feedback controller. However, only performance scaling is available for dynamic controller synthesis.

Publications related to this thesis: [SEK14, SDKew].

Thesis Group 3 (Chapter 5)

I performed the in silico validation of the observer and controller algorithms proposed in Thesis groups 1 and 2.

The simulation environment is based on the Cambridge model and simulator.

Publications related to the theses are: [KSAB13, KSF$^+$13, KSS$^+$13, KKSE13, SEK14, SMM$^+$14, KKS$^+$14, KSE$^+$14, KFS$^+$15, SDKew].

Thesis 3.1

I examined the effect of the number of sigma points on state estimation accuracy for a commonly used T1DM model. The results indicate that using more than 2L+1 sigma points provide no significant benefit.

Based on the evaluation, the sigma points shall have a symmetric configuration containing at least $2L$ points, where L represents the dimension of the model. Increasing the number of sigma points beyond that value does not lead to significant benefits. Both Cubature and Unscented Kalman filters with $2L$ and $2L+1$ sigma points provide sufficient estimation capabilities.

Publication related to this thesis: [SMM$^+$14].

Thesis 3.2

I performed the in silico validation of a robust qLPV $\mathcal{H}_2/\mathcal{H}_\infty$ state feedback controller. An Artificial Pancreas undergoing clinical trial can incorporate the proposed control algorithm, provided that Cambridge model based representation of the participants is available.

Based on control variability grid analysis, the controller could provide satisfactory blood glucose control for two 24-hour meal intake scenarios.

Publications related to this thesis: [KSAB13, KSF$^+$13, KSS$^+$13, KKSE13, SEK14, KKS$^+$14, KSE$^+$14, KFS$^+$15, SDKew].

7

CONCLUSION

This dissertation is about the automated control of the blood glucose concentration of a type 1 diabetes patient, which is referred to as Artificial Pancreas in related literature. The main focus is on three aspects of the Artificial Pancreas: estimation, prediction, and control.

Since all three aspects are model-based, Chapter 2 provided an introduction to T1DM modeling, with a detailed description of a well-known and widely used example, the Cambridge model. This is a nonlinear state-space system defined in continuous time with ordinary differential equations. It consists of a linear insulin absorption component, a nonlinear subsystem capturing glucose utilization and transfer, and a nonlinear meal ingestion model. There is a freely accessible simulator for the Cambridge model with six sets of virtual patient parameters. The Chapter also addressed model reduction and uncertainty modeling. The latter is especially relevant for robust controller design.

Chapter 3 discussed the state and disturbance estimation, which also covers the first thesis group. The mathematical background is Kalman filters and sigma point filters. First, the Chapter presents some of the most commonly used sigma point selection strategies. The strategies are followed by updates addressing sigma point filter limitations, such as square root filtering for numerical stability and extensive transform for preserving the statistical independence between states and disturbances. Finally, the Chapter presents ways to customize sigma point filters for artificial pancreas use. This includes lognormal transform to preserve the nonnegativity of state variables and disturbances, the use of uncertainty models, and aspects of disturbance estimation. The Chapter concludes with possibilities of prediction. The state observer discussed earlier can be extended into a predictor. The predictor is then

combined with another state observer to address the long-term effects of state estimation error in a closed-loop setting. The proposed predictor algorithm can validate meal announcements and event hypotheses.

The second thesis group is detailed in Chapter 4, separated into two parts. The first half presents the multiple linear parameter-varying representations of the Cambridge model. It highlights the benefits and drawbacks of the different alternatives and proposes one which is suitable for controller design. The second half starts with a discussion on how to define constraints on the controller by extending the nominal LPV model with weighting functions. This includes control signal limitations, uncertainties, and performance requirements. Finally, the mathematical background of robust controller synthesis follows, including the definition of relevant norms and the presentation of the necessary linear matrix inequalities. The Chapter concludes with two AP-specific changes to the LMIs:

- Performance scaling, which enables hard constraints on robust stability and patient-specific optimization of nominal performance simultaneously, with a single LMI.

- Output multiplicative uncertainty scaling. For the state feedback controller, it provides automatic patient-specific tuning of uncertainty output in order to make the corresponding LMIs feasible with optimal nominal performance. For the dynamic controller, an iterative process defines the scaling factor.

The in silico validation of the previously introduced state observer and controller algorithms was carried out in Chapter 5. The first half focuses on state estimation, while the second half deals with the control algorithm. Although, there are overlaps since the observers are also examined in a closed-loop setting, and the controller relies heavily on estimated signals. Based on the results, lognormal transformation increases the accuracy of sigma point filters if the spread of the sigma points is moderate. Furthermore, using more than $2L + 1$ sigma points yield only little benefit, which may not justify the increase in computation time. The controller could remain in the A and B zones of control variability grid analysis, which indicates satisfactory blood glucose control.

The presented state observer can support a wide range of control algorithms that require accurate estimation of signals not available for measurement. Using square root filtering and a moderate number of sigma points makes it a lightweight algorithm that is easy to implement and use on embedded hardware. However, the ability to calculate the estimated covariance of the prediction error may be useful in some

applications. Furthermore, the predictor is easily implemented if the state observer is already available.

When designing the controller, the focus was more on robustness than disturbance rejection. Hence, it may be combined with other control algorithms that use different approach: e.g., adaptive, predictive, or soft-computing methods. In this kind of setup, the proposed controller can cover use cases where the deviation of the real system from the nominal model is large, or meal and other external information is unreliable. Furthermore, it has low computational power requirements, which makes it an attractive choice for low-performance embedded hardware or "power saver" use cases.

The Artificial Pancreas promises effective management of a chronic condition that is more and more prevalent worldwide, affecting the lives of millions of individuals. I sincerely hope that this dissertation can have a positive contribution to keeping that promise.

Appendix A

APPENDIX

A.1 Model

A.1.1 Magni model

The following differential equations describe the Magni-model:

$$\begin{aligned}
\dot{G_M}(t) = & -k_{sc}G_M(t) + \frac{k_{sc}}{V_G}G_p(t) \\
\dot{G}_p(t) = & -k_1 G_p(t) + k_2 G_t(t) - k_{e1}\max\{0, G_p(t) - k_{e2}\} + \\
& + f\frac{k_{abs}Q_{gut}(t)}{BW} + \max\{0, k_{p1} - k_{p2}G_p(t) - k_{p3}I_d(t)\} - \\
& - U_{ii}(t) \\
\dot{G}_t(t) = & \ k_1 G_p(t) - k_2 G_t(t) - \frac{(V_{mx}X(t) + V_{m0})G_t(t)}{K_m + G_t(t)} \\
\dot{X}(t) = & -p_{2U}(X(t) + I_b) + \frac{p_{2U}}{V_I}I_p(t) \\
\dot{I}_d(t) = & -k_i I_d(t) + k_i I_1(t) \\
\dot{I}_1(t) = & -k_i I_1(t) + \frac{k_i}{V_I}I_p(t) \\
\dot{I}_p(t) = & -(m_2 + m_4)I_p(t) + m_1 I_L(t) + k_{a2}S_2(t) + k_{a1}S_1(t) \\
\dot{I}_L(t) = & -(m_1 + m_3)I_L(t) + m_2 I_p(t) \\
\dot{S}_2(t) = & -k_{a2}S_2(t) + k_d S_1(t) \\
\dot{S}_1(t) = & -(k_{a1} + k_d)S_1(t) + \frac{1}{BW}u(t) \\
\dot{Q}_{sto1}(t) = & -k_{gri}Q_{sto1}(t) + d(t) \\
\dot{Q}_{sto2}(t) = & -k_{empt}(t)Q_{sto2}(t) + k_{gri}Q_{sto1}(t) \\
\dot{Q}_{gut}(t) = & -k_{abs}Q_{gut}(t) + k_{empt}(t)Q_{sto2}(t)
\end{aligned} \tag{A.1}$$

The state variables are as follows:

- $G_M(t)$ [mg/dL] subcutaneous glucose concentration;

- $G_p(t)$ [mg/kg] glucose in plasma and rapidly equilibrating tissues;
- $G_t(t)$ [mg/kg] glucose in slowly equilibrating tissues;
- $X(t)$ [pmol/L] insulin in interstitial fluid;
- $I_d(t)$ and $I_1(t)$ [pmol/L] state variables for delayed insulin signal;
- $I_p(t)$ [pmol/kg] insulin mass in plasma;
- $I_L(t)$ [pmol/kg] insulin mass in liver;
- $S_2(t)$ [pmol/kg] monomeric insulin in the subcutaneous tissue;
- $S_1(t)$ [pmol/kg] polymeric insulin in the subcutaneous tissue;
- Q_{sto1} [mg] amount of glucose in the stomach in solid phase;
- Q_{sto2} [mg] amount of glucose in the stomach in liquid phase;
- Q_{gut} [mg] glucose mass in the intestine.

The inputs of the system are $u(t)$ injected insulin flow [pmol/min], $d(t)$ amount of ingested glucose [mg/min], and $U_{ii}(t)$ insulin-independent glucose utilization [mg/kg/min]. $d(t)$ and $U_{ii}(t)$ are considered disturbance from AP perspective.

Table A.1 provides details on model parameters.

k_{empt} is a time-varying parameters [83], which is calculated as follows:

$$\begin{aligned} k_{empt}(t) = \;& k_{min} + \frac{k_{max} + k_{min}}{2} \Big(\tanh\Big(\alpha(Q_{sto}(t) - bD)\Big) - \\ & - \tanh\Big(\beta(Q_{sto}(t) - cD)\Big) + 2 \Big) \\ \alpha = \;& \frac{5}{2D\,(1-b)} \\ \beta = \;& \frac{5}{2Dc} \\ Q_{sto}(t) = \;& Q_{sto1}(t) + Q_{sto2}(t), \end{aligned} \tag{A.2}$$

where $k_{empt}(t)$ is on its maximum value (k_{max}) when the stomach contains D amount of ingested glucose. Then $k_{empt}(t)$ decreases with the rate of α to a minimal value (k_{min}), but shortly afterward it rises back to the maximum with the rate of β. c is the percentage of the dose for which $k_{empt}(t)$ decreases to the value $\frac{k_{max} + k_{min}}{2}$, and similarly b represents the percentage of the dose for which k_{empt} rises back from its minimal value to $\frac{k_{max}+k_{min}}{2}$. The change of $k_{empt}(t)$ is shown on Fig. A.1, where the usual amount of 60 g carbohydrate (CHO) intake used

Table A.1. Magni model parameters

Name	Unit	Description
V_G	dL/kg	distribution volume of glucose;
k_1, k_2	1/min	rate parameters of the glucose subsystem;
k_{e1}	1/min	renal glomerular filtration rate;
k_{e2}	mg/kg	renal threshold;
V_I	L/kg	insulin distribution volume;
$m_1, ..., m_4$	1/min	rate parameters of the insulin subsystem;
BW	kg	body weight;
k_{p1}	mg/kg/min	extrapolated endogenous glucose production at zero glucose and insulin;
k_{p2}	1/min	liver glucose effectiveness;
k_{p3}	mg $\cdot$ L/kg/min/pmol	indicator of effect of a delayed insulin signal;
k_i	1/min	parameter of delayed insulin signal;
I_b	pmol/L	basal level of plasma insulin concentration;
p_{2U}	1/min	rate constant of insulin action;
K_m	mg/kg	model parameter for insulin-dependent glucose utilization;
V_{m0}	1/min	model parameter for insulin-dependent glucose utilization;
V_{mx}	L/pmol/min	model parameter for insulin-dependent glucose utilization;
k_d	1/min	degradation constant;
k_{a1}, k_{a2}	1/min	absorption constants;
k_{sc}	1/min	rate constant for the subcutaneous glucose compartment;
k_{gri}	1/min	rate of grinding;
k_{abs}	1/min	rate constant of intestinal absorption;
f	-	fraction of intestinal absorption which actually appears in plasma;
k_{empt}	1/min	rate constant of gastric emptying.

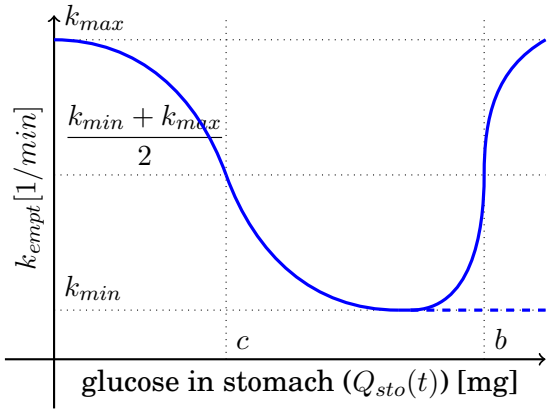

Figure A.1. Time-varying rate of gastric emptying

in the literature was considered. Furthermore, the equations (A.2) can be simplified into (A.3) if needed.

$$k_{empt}(t) = k_{min} + \frac{k_{max} + k_{min}}{2}\left(\tanh\left(\alpha(Q_{sto}(t) - bD)\right) + 1\right) \tag{A.3}$$

A.1.2 Non-invasive or Minimally Invasive Glucose Sensors

The following is a non exhaustive list of non-invasive or minimally invasive CMG sensors [92].

- Infrared absorption spectroscopy: the intensity of a beam of light of a specific wavelength is measured before and after interaction with matter.
- Kromoscopy: uses relative intensities of overlapping spectroscopic responses from multiple detectors recording spectra over wavelengths of near-infrared light.
- Thermal infrared measurement is based on the principle that the cutaneous microcirculation depends on the local glucose concentration.
- Raman spectroscopy: assesses scattering of single wavelength light.
- Polarimetry: measures the optical rotary dispersal of polarized light.
- Optical coherence tomography is a system that uses a low-power laser source, an in-depth scanning system, a sampling device, and

a light detector. The sensor irradiates the skin with coherent light and records the backscattered radiation.

- Occlusion spectroscopy uses scattering technique and exploits pulsatile arterial flow.
- Photoacoustic spectroscopy uses the principle that the absorption of light by tissue causes ultrasonic waves.
- Reverse iontophoresis is a transdermal technique where the interstitial fluid is accessed through the skin by applying a low electric current across the skin between two electrodes, causing charged and uncharged species to pass across the dermis at rates significantly greater than passive permeability.
- Sonophoresis is similar to reverse iontophoresis but uses low-frequency ultrasound to increase skin permeability.

A.2 Observer

A.2.1 Iterative filtering

If the function $h()$ in (3.3) is nonlinear, it is possible to further enhance the estimation of the sigma point filter by performing the *Update* step in an iterative manner [129]. Once the *Estimation* step is completed with either regular or square root sigma point filter algorithm, perform the following steps for *Update*:

0. Set up the iteration loop with the following initialization:

   - Set the iteration variable $j = 1$.
   - Set the initial estimation of the state vector: $\hat{\mathbf{x}}_{k+1,0} = \bar{\mathbf{x}}_{k+1}$
   - Set the initial estimation of the state vector and the initial estimation error covariance matrix, using either $\mathbf{\Sigma}_{k+1,0} = \mathbf{P}_{xx}$ or $\mathbf{\Sigma}_{k+1,0} = \mathbf{P}_x \mathbf{P}_x^T$.
   - Use the already propagated sigma points:

$$\mathcal{X}_{k+1,1} = \left[\begin{array}{ccc} \mathcal{X}_{k+1}^{(\bar{x})T} & \mathcal{X}_{k}^{(w)T} & \mathcal{X}_{k}^{(z)T} \end{array} \right]^T \tag{A.4}$$

   - Set the maximum number of iterations I_{max}

1. Calculate estimation of the measured output $\bar{\mathbf{y}}_{k+1,j}$, and either $\mathbf{P}_j^{(yy)}$ or $\mathbf{P}_j^{(y)}$ with steps 1d, 1e and 1f of *Estimation* based on either the regular of the square root sigma point filter algorithm. Use $\mathcal{X}_{k+1,j}$ as input.

2. Calculate the Kalman gain $\mathbf{K}_{k+1,j}$, and either $\mathbf{\Sigma}_{k+1,j}$ or $\mathbf{\Sigma}_{k+1,j}^{\frac{1}{2}}$ using step 2a and 2b from *Update*.

3. Get the jth candidate for state vector: $\hat{\mathbf{x}}_{k+1,j} = \hat{\mathbf{x}}_{k+1,j-1} + \mathbf{K}_{k+1,j}(\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{k+1,j})$.

4. Define the new sigma point set $\mathcal{X}_{k+1,j+1}$ using the chosen sigma point selection strategy.

5. If $j < I_{max}$ and the following two inequality holds:

$$\begin{aligned}(\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{k+1,j-1})^T \mathbf{R}_k^{-1} (\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{k+1,j-1}) \quad &> \\ (\hat{\mathbf{x}}_{k+1,j} - \hat{\mathbf{x}}_{k+1,j-1})^T \mathbf{\Sigma}_{k+1,j}^{-1} (\hat{\mathbf{x}}_{k+1,j} - \hat{\mathbf{x}}_{k+1,j-1}) \quad &+ \\ + (\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{k+1,j})^T \mathbf{R}_{k+1}^{-1} (\mathbf{y}_{k+1} - \bar{\mathbf{y}}_{k+1,j}) &\end{aligned} \tag{A.5}$$

   then increment j, define the new sigma point set $\mathcal{X}_{k+1,j+1}$ using a sigma point selection strategy, and continue from step 1. Otherwise, use $\hat{\mathbf{x}}_{k+1,j}$, $\mathbf{\Sigma}_{k+1,j}^{\frac{1}{2}}$ and $\mathcal{X}_{k+1,j+1}$ as $\hat{\mathbf{x}}_{k+1}$, $\mathbf{\Sigma}_{k+1}^{\frac{1}{2}}$ and $\mathcal{X}_{k+1}$ respectively, and continue with the $k+1$th *Estimation* step. Note, that the algorithm is identical to the regular or square root sigma point filter if $I_{max} = 1$.

Since $h()$ is nonlinear in the Cambridge model extended with output multiplicative uncertainties, the effect of iterative filtering was tested via simulations.

Based on simulation results presented in Table A.2 iterative filtering may provide some benefits. It improves the estimation of state variable Q_1 for all sigma point filters. However, the results are mixed for the other two state variables.

Table A.2. Comparison of sigma point filters with and without iterative estimation using metrics $R_1 - R_3$. For each filter, left and right column show values for regular and iterative version respectively. Improvements are indicated with green, while red highlights setbacks

			CDKF		CKF		GHQF		RSPF		SGQF		UKF	
Scenario 1	R_1	Q_1	13.74	12.54	14.11	13.27	13.99	12.59	19.41	17.41	13.90	12.71	14.12	13.06
		Q_2	5.35	5.29	4.88	4.74	5.84	5.81	10.50	9.58	5.17	5.30	5.07	5.07
		G_{flux}	0.72	0.67	0.69	0.70	0.84	0.79	0.97	0.89	0.72	0.68	0.66	0.64
	R_2	Q_1	19.65	18.25	20.13	19.09	20.18	18.40	28.00	24.33	19.81	18.31	20.08	18.95
		Q_2	16.48	18.36	15.81	16.42	19.44	19.04	31.45	29.17	17.59	18.22	16.60	17.50
		G_{flux}	1.00	0.90	0.98	0.99	1.09	1.00	1.24	1.10	0.99	0.91	0.96	0.90
	R_3	Q_1	32.23	28.66	32.75	32.78	34.36	27.30	48.66	38.56	32.83	29.04	31.26	30.05
		Q_2	23.95	27.36	23.07	24.63	28.10	28.09	40.50	38.64	25.44	26.93	24.00	26.20
		G_{flux}	3.46	3.27	3.51	3.38	3.45	3.39	3.45	3.44	3.46	3.28	3.47	3.12
Scenario 2	R_1	Q_1	8.10	7.84	9.64	9.20	9.18	8.59	9.47	8.81	8.29	7.94	7.94	7.73
		Q_2	4.33	4.21	4.76	4.53	4.54	4.35	4.57	4.35	4.42	4.29	4.32	4.21
		G_{flux}	0.50	0.50	0.64	0.64	0.53	0.49	0.55	0.51	0.54	0.53	0.50	0.52
	R_2	Q_1	11.81	11.59	13.03	12.45	13.09	12.35	13.62	12.68	11.88	11.63	11.72	11.24
		Q_2	7.18	7.08	8.52	8.44	7.68	7.39	7.98	7.47	7.51	7.39	7.03	7.22
		G_{flux}	0.64	0.65	0.95	0.96	0.70	0.66	0.75	0.70	0.76	0.73	0.64	0.66
	R_3	Q_1	30.74	29.93	37.31	34.26	32.81	31.45	34.60	63.72	31.58	30.02	31.02	30.32
		Q_2	22.24	23.50	26.12	27.54	22.05	22.69	25.58	25.10	23.25	24.39	22.45	23.84
		G_{flux}	3.52	3.51	3.39	3.36	3.42	3.49	3.40	3.41	3.26	3.30	3.53	3.50

A.2.2 $\mathcal{H}_\infty$ filtering

$\mathcal{H}_\infty$ filtering extends the state approximation problem with additional inequality constraints for a chosen γ value:

$$J_k = \frac{\sum_{j=0}^{k} \|\hat{\mathbf{z}}_j - \mathbf{L}\mathbf{x}_j\|^2}{\|\mathbf{x}_0 - \hat{\mathbf{x}}_0\|^2_{\mathbf{\Sigma}_0^{-1}} + \sum_{j=0}^{k-1} \|\mathbf{w}_j\|^2_{\mathbf{Q}_j^{-1}} + \sum_{j=0}^{k} \|\mathbf{v}_j\|^2_{\mathbf{R}_j^{-1}}} < \gamma^2 \tag{A.6}$$

where $\mathbf{z}_k = \mathbf{L}\mathbf{x}_k$ are virtual measurements of $\mathbf{x}_k$, $\hat{\mathbf{z}}_k$ represents the estimation of $\mathbf{z}_k$, while $\|\mathbf{x}\|^2_{\mathbf{M}}$ denotes the square of the weighted l_2 norm of $\mathbf{x}$: $\|\mathbf{x}\|^2_{\mathbf{M}} = \mathbf{x}^T\mathbf{M}\mathbf{x}$ [130]. It is possible to prioritize the estimation of certain state variables over others by configuring the matrix $\mathbf{M}$. This is beneficial when the estimated values are used in a control algorithm. For example, the Linear Parameter Varying control presented in Section 4.2.1 requires reliable estimation of Q_1 and Q_2 state variables of the Cambridge model (4.34).

Using the virtual measurements as follows extends the regular sigma point filter algorithm with $\mathcal{H}_\infty$ constraints (A.6) for a chosen γ value [130]:

$$\begin{aligned}
\mathbf{P}^{(zz)} &= \begin{pmatrix} \mathbf{P}^{(yy)} & \left(\mathbf{P}^{(xy)}\right)^T\mathbf{L}^T \\ \mathbf{L}\mathbf{P}^{(xy)} & \mathbf{L}\mathbf{P}^{(xx)}\mathbf{L}^T - \gamma^2\mathbf{I} \end{pmatrix} \\
\mathbf{P}^{(xz)} &= \begin{pmatrix} \mathbf{P}^{(xy)} & \mathbf{P}^{(xx)}\mathbf{L}^T \end{pmatrix} \\
\mathbf{\Sigma}_{k+1} &= \mathbf{P}^{(xx)} - \mathbf{P}^{(xy)}\left(\mathbf{P}^{(zz)}\right)^{-1}\left(\mathbf{P}^{(xz)}\right)^T.
\end{aligned} \tag{A.7}$$

However, the Kalman gain is calculated the same way. The resulting $\mathbf{\Sigma}_{k+1}$ may be indefinite depending on how the value of γ. Hence, γ is set in an iterative manner in each update step.

When testing $\mathcal{H}_\infty$ filtering with the Cambridge model, the γ iteration sporadically resulted in instability in the filters. Furthermore, it is difficult to combine it with square root filtering.

A.3 Control

A.3.1 LPV representation of the Magni model

The scheduling variables of the LPV-transformed Magni model must address the following nonlinearities:

- Endogenous glucose production: $\max\{0, k_{p1} - k_{p2}G_p(t) - k_{p3}I_d(t)\}$
- Insulin-dependent glucose utilization: $\dfrac{(V_{mx}X(t) + V_{m0})G_t(t)}{K_m + G_t(t)}$
- Renal extraction: $k_{e1}\max\{0, G_p(t) - k_{e2}\}$

Similarly to the Cambridge model, endogenous glucose production and renal extraction introduce a "switching" effect to the model. The former is active unless either plasma glucose, plasma insulin, or both are elevated. The latter activates only when the plasma glucose concentration exceeds the threshold k_{e2}.

One can use the following scheduling variables ρ_{EGP} and ρ_e, which approximate the switching with a sigmoid function:

$$\begin{aligned} \max\{0, k_{p1} - k_{p2}G_p(t) - k_{p3}I_d(t)\} \approx \quad & -\rho_{EGP}(t)\left(k_{p2}G_p(t) + \right. \\ & \left. + k_{p3}I_d(t)\right) + k_{p1}EGP_0(t) \\ k_{e1}\max\{0, G_p(t) - k_{e2}\} \approx \rho_e(t)G_p(t) - k_{e1}k_{e2}RE_0(t) & \\ \rho_{EGP}(t) = \frac{1}{1 + e^{\alpha(k_{p1} - k_{p2}G_p(t) - k_{p3}I_d(t))}} & \\ \rho_e(t) = \frac{1}{1 + e^{\alpha(k_{e2} - G_p(t))}} & \end{aligned} \tag{A.8}$$

The newly introduced parameter α defines the steepness of the sigmoid functions, and hence the bounds of the time derivate of $\rho_{EGP}(t)$ and $\rho_e(t)$. Some controller strategies relying on the LPV model can impose constraints on α. $EGP_0(t) \in [0, 1]$ and $RE_0(t) \in [0, 1]$ are considered as new disturbances from controller design perspective.

Alternatively it is possible to substitute the nonlinear endogenous glucose production with $k_{p1} - k_{p2}G_p(t) - k_{p3}I_d(t)$, and ensure that $k_{p1} - k_{p2}G_p(t) - k_{p3}I_d(t) \geq 0$. However, this can difficult to achieve depending on the parameter values, since the controller can directly set $I_d(t)$, but $G_p(t)$ is affected by various external disturbances. For same holds for renal extraction, however $G_p(t) \leq k_{e2}$ is a more reasonable constraint in comparison. This requires to controller to protect the T1DM patient from extreme hyperglycemia, which is the sole purpose of AP. Considering the small impact renal extraction can have on the plasma glucose concentration, it is feasible to simply treat is as disturbance, unlike endogenous glucose production.

There are two different LPV representations for insulin-dependent glucose utilization. The first approach is to use a single scheduling variable:

$$\rho_{X,G}(t) = \frac{V_{mx}X(t) + V_{m0}}{K_m + G_t(t)} \tag{A.9}$$

This LPV representation has drawbacks. $\rho_{X,G}(t)$ can have a wide range of values, since it is a function of two state variables. Hence, we need to consider all combination of their bounds when calculating $\rho_{X,G,min}$ and $\rho_{X,G,max}$. Furthermore, one of these variables - $X(t)$ - is linked to insulin concentration, which is regarded as the control signal

in AP. This means it can change in a rather wide range during closed-loop control, increasing the bounds of $\rho_{X,G}(t)$ even further. Finally, $G_t(t)$ will have no connection with $X(t)$ with this LPV representation, which in turn alters the controllability of the derived LPV model.

Consequently, the preferred LPV representation of insulin-dependent glucose utilization is $\rho_{G1}(t)G_t(t)+\rho_{G2}(t)X(t)$, with the following scheduling variables:

$$\rho_{G1}(t) = \frac{V_{m0}}{K_m + G_t(t)} \quad \rho_{G2}(t) = \frac{V_{mx}G_t(t)}{K_m + G_t(t)} \tag{A.10}$$

Using (A.10) the controllability of the model is retained, and the scheduling variables are only depend from a single state variable. In addition, that state variable - $G_t(t)$ - is linked to glucose concentration, which is kept in a narrow range if the controller works well.

$\rho_{G2}(t)$ can be expressed as a linear function of $\rho_{G1}(t)$ as shown in (A.11), and hence we can reduce the number of scheduling variables in the LPV model.

$$\rho_{G2}(t) = V_{mx} - \frac{K_m V_{mx}}{V_{m0}}\rho_{G1}(t) \tag{A.11}$$

Using scheduling variables ρ_{EGP} from (A.8) and ρ_{G2} from (A.10) and (A.11), as well as introducing disturbance $RE(t)$ to represent renal extraction, the LPV representation of the Magni model (A.1) with worst case meal intake is as follows:

$$\begin{aligned}
\dot{G_M}(t) = &\ -k_{sc}G_M(t) + \frac{k_{sc}}{V_G}G_p(t) \\
\dot{G}_p(t) = &\ -(k_1 + k_{p2}\rho_{EGP}(t))G_p(t) + k_2 G_t(t) + \frac{k_{abs}f}{t_Q BW}\tilde{Q}_{gut}(t) - \\
&\ -\rho_{EGP}(t)k_{p3}I_d(t) - k_{e1}RE(t) + k_{p1}EGP_0(t) - \\
&\ -U_{ii}(t) \\
\dot{G}_t(t) = &\ k_1 G_p(t) - (k_2 + \rho_{G1}(t))G_t(t) - \Big(V_{mx} - \\
&\ - \frac{K_m V_{mx}}{V_{m0}}\rho_{G1}(t)\Big) X(t) \\
\dot{X}(t) = &\ -p_{2U}(X(t) + I_b) + \frac{p_{2U}}{V_I}I_p(t) \\
\dot{I}_d(t) = &\ -k_i I_d(t) + k_i I_1(t) \\
\dot{I}_1(t) = &\ -k_i I_1(t) + \frac{k_i}{V_I}I_p(t) \\
\dot{I}_p(t) = &\ -(m_2 + m_4)I_p(t) + m_1 I_L(t) + k_{a2}S_2(t) + k_{a1}S_1(t) \\
\dot{I}_L(t) = &\ -(m_1 + m_3)I_L(t) + m_2 I_p(t) \\
\dot{S}_2(t) = &\ -k_{a2}S_2(t) + k_d S_1(t) \\
\dot{S}_1(t) = &\ -(k_{a1} + k_d)S_1(t) + \frac{1}{BW}u(t) \\
\dot{\tilde{Q}}_{gut}(t) = &\ -\frac{1}{t_Q}\tilde{Q}_{gut}(t) + d(t)
\end{aligned} \tag{A.12}$$

REFERENCES

[1] S. Wild, G. Roglic, A. Green, R. Sicree, and H. King, "Global prevalence of diabetes. Estimates for the year 2000 and projections for 2030," *Diabetes Care*, vol. 27, no. 5, pp. 1047–1053, 2004.

[2] W. H. Organization. Diabetes fact sheet, 2020 [Online].

[3] A. Fonyo and E. Ligeti, *Physiology*. Budapest: Medicina, third ed., 2008.

[4] R. Buzzetti, S. Zampetti, and P. Pozzilli, "Impact of obesity on the increasing incidence of type 1 diabetes," *Diabetes, Obesity and Metabolism*, vol. 22, no. 7, pp. 1009–1013, 2020.

[5] C. Cobelli, E. Renard, and B. Kovatchev, "Artificial pancreas: Past, present and future," *Diabetes*, vol. 60, no. 11, pp. 2672–2682, 2011.

[6] H. Thabit and R. Hovorka, "Coming of age: the artificial pancreas for type 1 diabetes," *Diabetologia*, vol. 59, no. 9, pp. 1795–1805, 2016.

[7] J. Tašić, M. Takács, and L. Kovács, "Control engineering methods for blood glucose levels regulation," *Acta Polytechnica Hungarica*, vol. 19, pp. 127–152, 08 2022.

[8] S. LesleAnn Hayward and W. Leah M., "New developments in glucagon treatment for hypoglycemia," *Drugs*, vol. 82, pp. 1179–1191, 07 2022.

[9] U. S. Food and Drug Administration, "Recently-approved devices - Medtronic's Minimed 670G System," Sep 2016.

Available online at: https://wayback.archive-it.org/7993/20170111141252/http://www.fda.gov/MedicalDevices/ProductsandMedicalProcedures/DeviceApprovalsandClearances/Recently-ApprovedDevices/ucm522764.htm.

[10] J. Kropff and J. H. DeVries, "Continuous glucose monitoring, future products, and update on worldwide artificial pancreas projects," *Diabetes Technology & Therapeutics*, vol. 18, no. S2, 2016.

[11] G. Forlenza, S. Deshpande, T. Ly, D. Howsmon, F. Cameron, N. Baysal, E. Mauritzen, T. Marcal, L. Towers, B. Bequette, L. Huyett, J. Pinsker, R. Gondhalekar, F. Doyle, D. Maahs, B. Buckingham, and E. Dassau, "Application of zone model predictive control artificial pancreas during extended use of infusion set and sensor: A randomized crossover-controlled home-use trial," *Diabetes Care*, vol. 40, p. dc170500, 06 2017.

[12] S. Brown, B. Kovatchev, D. Raghinaru, J. Lum, B. Buckingham, Y. Kudva, L. Laffel, C. Levy, J. Pinsker, R. Wadwa, E. Dassau, F. Doyle, S. Anderson, M. Church, V. Dadlani, L. Ekhlaspour, G. Forlenza, E. Isganaitis, D. Lam, and R. Beck, "Six-month randomized, multicenter trial of closed-loop control in type 1 diabetes," *New England Journal of Medicine*, vol. 381, pp. 1707–1717, 10 2019.

[13] M. Breton and B. Kovatchev, "One year real-world use of control-iq advanced hybrid closed-loop technology," *Diabetes Technology & Therapeutics*, vol. 23, 03 2021.

[14] P. Colmegna and R. Sánchez-Peña, "Analysis of three T1DM simulation models for evaluating robust closed-loop controllers," *Computer methods and programs in biomedicine*, vol. 113, 10 2013.

[15] M. Somogyi and M. Kirstein, "Insulin as a cause of extreme hyperglycemia and instability," *Weekly Bulletin of the St Louis Medical Society*, vol. 32, pp. 498–510, 1938.

[16] R. Defronzo, E. Ferrannini, P. Zimmet, and K. Alberti, eds., *International Textbook of Diabetes Mellitus*. Australia: Wiley-Blackwell, 4th ed., 2015.

[17] R. Hovorka, V. Canonico, L. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. Pieber, H. Schaller, L. Schaupp, T. Vering, and M. Wilinska, "Nonlinear model predictive control

of glucose concentration in subjects with type 1 diabetes," *Physiological measurement*, vol. 25, pp. 905–920, 2004.

[18] L. Magni, D. M. Raimondo, C. D. Man, G. D. Nicolao, B. Kovatchev, and C. Cobelli, "Model predictive control of glucose concentration in type 1 diabetic patients: An in silico trial," *Biomedical Signal Processing and Control*, pp. 338–346, 2009.

[19] A. Facchinetti, S. Del Favero, G. Sparacino, J. R. Castle, W. K. Ward, and C. Cobelli, "Modeling the glucose sensor error," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 3, pp. 620–629, 2014.

[20] M. Vettoretti, G. Cappon, G. Acciaroli, A. Facchinetti, and G. Sparacino, "Continuous glucose monitoring: Current use in diabetes management and possible future applications," *Journal of Diabetes Science and Technology*, vol. 12, no. 5, pp. 1064–1071, 2018.

[21] M. Gokul, M. Karthick, S. Sriram, S. R. Luckneshwaran, T. Arun Prasath, and N. A N, "Review on effectiveness of various continuous glucose monitoring systems for diabetic patients," in *Proceedings of the 8th International Conference on Smart Structures and Systems (ICSSS)*, pp. 01–04, 2022.

[22] S. Mehmood, I. Ahmad, H. Arif, U. Ammara, and A. Majeed, "Artificial pancreas control strategies used for type 1 diabetes control and treatment: A comprehensive analysis," *Applied System Innovation*, vol. 3, p. 31, 07 2020.

[23] J. Tašić, G. Eigner, and L. Kovács, "Review of algorithms for improving control of blood glucose levels," in *2020 IEEE 18th International Symposium on Intelligent Systems and Informatics (SISY)*, pp. 179–184, 10 2020.

[24] A. J. Barnes and R. W. Jones, "PID-based glucose control using intra-peritoneal insulin infusion: An in silico study," in *Proceedings of the 20019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp. 1057–1062, 2019.

[25] A.-L. Alshalalfah, G. B. Hamad, and O. A. Mohamed, "Towards safe and robust closed-loop artificial pancreas using improved PID-based control strategies," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 8, pp. 3147–3157, 2021.

[26] D. Calupiña, A. García, O. Camacho, A. Rosales, and P. Rivadeneira, "Non-linear PID and dynamic SMC for the artificial

pancreas control in the treatment of type 1 diabetes," in *IEEE Third Ecuador Technical Chapters Meeting (ETCM)*, pp. 1–6, 2018.

[27] A. Beneyto, A. Bertachi, J. Bondia, and J. Vehi, "A new blood glucose control scheme for unannounced exercise in type 1 diabetic subjects," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 2, pp. 593–600, 2020.

[28] H. Leyva, G. Quiroz, F. A. Carrillo, and R. Femat, "Insulin stabilisation in artificial pancreas: a positive control approach," *IET Control Theory & Applications*, vol. 13, no. 7, pp. 970–978, 2019.

[29] D. Cai, J. Song, J. Wang, and D. Shi, "Glucose regulation for subjects with type 1 diabetes using active disturbance rejection control," in *Proceedings of the 2019 Chinese Control Conference (CCC)*, pp. 6970–6975, 2019.

[30] T. Zhu, K. Li, P. Herrero, and P. Georgiou, "Basal glucose control in type 1 diabetes using deep reinforcement learning: An in silico validation," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, pp. 1223–1232, 04 2021.

[31] J. C. Peiró, "Comparing artificial pancreas controlled by hybrid "closed-loop" machine learning (ML) trained algorithm to multi-daily injection (MDI), insulin pump without CGM and "sensor assisted" insulin pump therapies for diabetes type 1 (DT1) treatment," in *Proceedings of the 2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)*, pp. 1–6, 2020.

[32] A. Mohammadzadeh and T. Kumbasar, "A new fractional-order general type-2 fuzzy predictive control system and its application for glucose level regulation," *Applied Soft Computing*, vol. 91, p. 106241, 2020.

[33] K. Turksoy and A. Cinar, "Adaptive control of artificial pancreas systems - a review," *Journal of Healthcare Engineering*, vol. 5, 2014.

[34] D. Shi, E. Dassau, and F. J. Doyle, "A multivariate Bayesian optimization framework for long-term controller adaptation in artificial pancreas," in *Proceedings of the 2018 IEEE Conference on Decision and Control (CDC)*, pp. 276–283, 2018.

[35] B. Czakó, D. Drexler, and L. Kovács, "Control of a T1DM model using robust fixed-point transformations based control with disturbance rejection," in *Proceedings of the 2022 IEEE International*

Conference on Automation, Quality and Testing, Robotics (AQTR), pp. 1–6, 05 2022.

[36] B. Kovatchev, C. Cobelli, and E. Renard, "Multi-national study of subcutaneous model-predictive closed-loop control in type 1 diabetes: summary of the results," *Journal of Diabetes Science and Technology*, vol. 4, pp. 1374–1381, 2010.

[37] S. Del Favero, D. Bruttomesso, F. Di Palma, G. Lanzola, R. Visentin, A. Filippi, R. Scotton, C. Toffanin, M. Messori, S. Scarpellini, P. Keith-Hynes, B. P. Kovatchev, J. H. DeVries, E. Renard, L. Magni, A. Avogaro, C. Cobelli, and on behalf of the AP@home Consortium, "First Use of Model Predictive Control in Outpatient Wearable Artificial Pancreas," *Diabetes Care*, vol. 37, pp. 1212–1215, 04 2014.

[38] D. Shi, E. Dassau, and F. J. Doyle III, "Multivariate learning framework for long-term adaptation in the artificial pancreas," *Bioengineering & Translational Medicine*, vol. 4, no. 1, pp. 61–74, 2019.

[39] P. Colmegna, F. Garelli, H. De Battista, F. Bianchi, and R. S. Sánchez-Peña, "The arg algorithm: clinical trials in Argentina," in *The Artificial Pancreas* (R. S. Sánchez-Peña and D. R. Cherñavvsky, eds.), pp. 79–104, Academic Press, 2019.

[40] A. A. Embaby, Z. Nossair, and H. Badr, "Adaptive nonlinear model predictive control algorithm for blood glucose regulation in type 1 diabetic patients," in *Proceedings of the 2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, pp. 109–115, 2020.

[41] P. Abuin, J. E. Sereno, A. Ferramosca, and A. H. Gonzalez, "Closed-loop MPC-based artificial pancreas: Handling circadian variability of insulin sensitivity," in *Proceedings of the 2020 Argentine Conference on Automatic Control (AADECA)*, pp. 1–6, 2020.

[42] I. Hajizadeh, N. Hobbs, M. Sevil, M. Rashid, M. R. Askari, R. Brandt, and A. Cinar, "Performance monitoring, assessment and modification of an adaptive MPC: Automated insulin delivery in diabetes," in *Proceedings of the 2020 European Control Conference (ECC)*, pp. 283–288, 2020.

[43] M. Siket, K. Novak, L. Kovács, and G. Eigner, "Automatically estimated meals in model predictive control-moving horizon esti-

mation control strategy," in *Proceedings of the 2022 13th Asian Control Conference (ASCC)*, pp. 1367–1372, 05 2022.

[44] D. Boiroux, M. Hagdrup, Z. Mahmoudi, K. Poulsen, H. Madsen, and J. B. Jørgensen, "An ensemble nonlinear model predictive control algorithm in an artificial pancreas for people with type 1 diabetes," in *Proceedings of the 2016 European Control Conference (ECC)*, pp. 2115–2120, 2016.

[45] N. Paoletti, K. S. Liu, H. Chen, S. Smolka, and S. Lin, "Data-driven robust control for a closed-loop artificial pancreas," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 17, pp. 1981–1993, 11 2020.

[46] M. Siket, K. Novák, H. Redjimi, J. Tar, L. Kovács, and G. Eigner, "Control of type 1 diabetes mellitus using particle swarm optimization driven receding horizon controller," *IFAC-PapersOnLine*, vol. 54, pp. 293–298, 01 2021.

[47] R. Parker, F. Doyle, J. Ward, and N. Peppas, "Robust $\mathcal{H}_\infty$ glucose control in diabetes using a physiological model," *American Institute of Chemical Engineers*, vol. 46, no. 12, pp. 2537–2549, 2000.

[48] G. Rigatos, P. Siano, and A. Melkikh, "A nonlinear optimal control approach of insulin infusion for blood glucose levels regulation," *Intellectual Industrial Systems*, vol. 3, pp. 91–102, 11 2017.

[49] G. Eigner, I. Böjthe, A. Mészáros, and L. Kovács, "Robust $\mathcal{H}_\infty$ controller design for T1DM based on relaxed LMI conditions," in *Proceedings of the 2019 IEEE 23rd International Conference on Intelligent Engineering Systems (INES)*, pp. 000363–000368, 2019.

[50] A. Mirzaee, M. Dehghani, and M. Mohammadi, "Robust LPV control design for blood glucose regulation considering daily life factors," *Biomedical Signal Processing and Control*, vol. 57, 2020.

[51] J. Bondia, S. Romero-Vivo, B. Ricarte, and J. L. Diez, "Insulin estimation and prediction: A review of the estimation and prediction of subcutaneous insulin pharmacokinetics in closed-loop glucose control," *IEEE Control Systems Magazine*, vol. 38, no. 1, pp. 47–66, 2018.

[52] E. Fushimi, P. Colmegna, H. De Battista, F. Garelli, and R. Sánchez-Peña, "Artificial pancreas: Evaluating the ARG algorithm without meal announcement," *Journal of Diabetes Science and Technology*, vol. 13, pp. 1035–1043, 07 2019.

[53] Z. Mahmoudi, D. Boiroux, M. Hagdrup, K. Nørgaard, N. K. Poulsen, H. Madsen, and J. B. Jørgensen, "Application of the continuous-discrete extended Kalman filter for fault detection in continuous glucose monitors for type 1 diabetes," in *Proceedings of the 2016 European Control Conference (ECC)*, pp. 714–719, 2016.

[54] D. Boiroux, T. K. S. Ritschel, N. Kjølstad Poulsen, H. Madsen, and J. B. Jørgensen, "Efficient computation of the continuous-discrete extended Kalman filter sensitivities applied to maximum likelihood estimation," in *Proceedings of the 2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 6983–6988, 2019.

[55] Z. Mahmoudi, K. Nørgaard, N. K. Poulsen, H. Madsen, and J. B. Jørgensen, "Fault and meal detection by redundant continuous glucose monitors and the unscented Kalman filter," *Biomedical Signal Processing and Control*, vol. 38, pp. 86–99, 2017.

[56] L. Ortmann, D. Shi, E. Dassau, F. J. Doyle, B. J. Misgeld, and S. Leonhardt, "Automated insulin delivery for type 1 diabetes mellitus patients using Gaussian process-based model predictive control," in *Proceedings of the 2019 American Control Conference (ACC)*, pp. 4118–4123, 2019.

[57] C. M. Ramkissoon, A. Bertachi, A. Beneyto, J. Bondia, and J. Vehi, "Detection and control of unannounced exercise in the artificial pancreas without additional physiological signals," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 1, pp. 259–267, 2020.

[58] I. Sala-Mira, M. Siket, L. Kovács, G. Eigner, and J. Bondia, "Effect of model, observer and their interaction on state and disturbance estimation in artificial pancreas: An in-silico study," *IEEE Access*, vol. 9, pp. 143549–143563, 2021.

[59] W. Alam, Q. Khan, R. A. Riaz, R. Akmeliawati, I. Khan, and K. S. Nisar, "Gain-scheduled observer-based finite-time control algorithm for an automated closed-loop insulin delivery system," *IEEE Access*, vol. 8, pp. 103088–103099, 2020.

[60] H. Chen, N. Paoletti, S. A. Smolka, and S. Lin, "Committed moving horizon estimation for meal detection and estimation in type 1 diabetes," in *Proceedings of the 2019 American Control Conference (ACC)*, pp. 4765–4772, 2019.

[61] R. Sanz, P. García, J.-L. Díez, and J. Bondia, "Artificial pancreas system with unannounced meals based on a disturbance observer

and feedforward compensation," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 1, pp. 454–460, 2021.

[62] I. Sala Mira, J. L. Diez, B. Ricarte, and J. Bondia, "Sliding-mode disturbance observers for an artificial pancreas without meal announcement," *Journal of Process Control*, vol. 78, pp. 68–77, 2019.

[63] X. Yu, M. Rashid, J. Feng, N. Hobbs, I. Hajizadeh, S. Samadi, M. Sevil, C. Lazaro, Z. Maloney, E. Littlejohn, L. Quinn, and A. Cinar, "Online glucose prediction using computationally efficient sparse kernel filtering algorithms in type-1 diabetes," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 1, pp. 3–15, 2020.

[64] T. Kushner, D. Bortz, D. M. Maahs, and S. Sankaranarayanan, "A data-driven approach to artificial pancreas verification and synthesis," in *Proceedings of the 2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, pp. 242–252, 2018.

[65] L. Meneghetti, M. Terzi, S. Del Favero, G. A. Susto, and C. Cobelli, "Data-driven anomaly recognition for unsupervised model-free fault detection in artificial pancreas," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 1, pp. 33–47, 2020.

[66] A. Güemes, G. Cappon, B. Hernandez, M. Reddy, N. Oliver, P. Georgiou, and P. Herrero, "Predicting quality of overnight glycaemic control in type 1 diabetes using binary classifiers," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 5, pp. 1439–1446, 2020.

[67] G. Eigner, M. Nagy, and L. Kovács, "Machine learning application development to predict blood glucose level based on real time patient data," pp. 1–6, 10 2020.

[68] A. Aliberti, I. Pupillo, S. Terna, E. Macii, S. Di Cataldo, E. Patti, and A. Acquaviva, "A multi-patient data-driven approach to blood glucose prediction," *IEEE Access*, vol. 7, pp. 69311–69325, 2019.

[69] K. Li, J. Daniels, C. Liu, P. Herrero, and P. Georgiou, "Convolutional recurrent neural networks for glucose prediction," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, pp. 603–613, 02 2020.

[70] T. Zhu, L. Kuang, K. Li, J. Zeng, P. Herrero, and P. Georgiou, "Blood glucose prediction in type 1 diabetes using deep learning on the edge," in *Proceedings of the 2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2021.

[71] R. N. Bergman, Y. Z. Ider, C. R. Bowden, and C. Cobelli, "Quantitative estimation of insulin sensitivity.," *American Journal of Physiology-Endocrinology and Metabolism*, vol. 236, no. 6, p. E667, 1979.

[72] R. Bergman, L. Phillips, and C. Cobelli, "Physiological evaluation of factors controlling glucose tolerance in man," *Journal of Clinical Investigation*, vol. 68, pp. 1456–1467, 1981.

[73] X. Wong, J. Chase, G. Shaw, C. Hann, T. Lotz, J. Lin, I. Singh-Levett, L. Hollingsworth, O. Wong, and S. Andreassen, "Model predictive glycaemic regulation in critical illness using insulin and nutrition input: A pilot study," *Medical Engineering & Physics*, vol. 28, no. 7, pp. 665–681, 2006.

[74] T. F. Lotz, J. G. Chase, K. A. McAuley, D. S. Lee, J. Lin, C. E. Hann, and J. I. Mann, "Transient and steady-state euglycemic clamp validation of a model for glycemic control and insulin sensitivity testing," *Diabetes Technology & Therapeutics*, vol. 8, no. 3, pp. 338–346, 2006.

[75] A. Le Compte, J. G. Chase, A. Lynn, C. Hann, G. Shaw, X.-W. Wong, and J. Lin, "Blood glucose controller for neonatal intensive care: virtual trials development and first clinical trials," *Journal of diabetes science and technology*, vol. 3, p. 1066—1081, September 2009.

[76] J. Sorensen, *A Physiologic Model of Glucose Metabolism in Man and its Use to Design and Assess Improved Insulin Therapies for Diabetes*. PhD thesis, Massachusetts Institute of Technology, 1985.

[77] R. S. Dubey, D. Baleanu, M. Mishra, and P. Goswami, "Solution of modified bergman minimal blood glucose-insulin model using Caputo-Fabrizio fractional derivative," *Computer Modeling in Engineering & Sciences*, vol. 128, pp. 1247–1263, 01 2021.

[78] P. Palumbo, S. Panunzi, and A. D. Gaetano, "Qualitative behavior of a family of delay-differential models of the glucose-insulin system," *Discrete & Continuous Dynamical Systems*, vol. 7, no. 2, pp. 399–424, 2007.

[79] S. Contreras, D. Medina-Ortiz, C. Conca, and A. Olivera-Nappa, "A novel synthetic model of the glucose-insulin system for patient-wise inference of physiological parameters from small-size OGTT data," *Frontiers in Bioengineering and Biotechnology*, vol. 8, p. 195, 2020.

[80] P. L. Brubaker, E. L. Ohayon, L. M. D'Alessandro, and K. H. Norwich, "A mathematical model of the oral glucose tolerance test illustrating the effects of the incretins," *Annals of biomedical engineering*, vol. 35, pp. 1286——1300, July 2007.

[81] F. H. El-Khatib, S. J. Russell, K. L. Magyar, M. Sinha, K. McKeon, D. M. Nathan, and E. R. Damiano, "Autonomous and Continuous Adaptation of a Bihormonal Bionic Pancreas in Adults and Adolescents With Type 1 Diabetes," *The Journal of Clinical Endocrinology & Metabolism*, vol. 99, pp. 1701–1711, 05 2014.

[82] S. Salinari, A. Bertuzzi, and G. Mingrone, "Intestinal transit of a glucose bolus and incretin kinetics: a mathematical model with application to the oral glucose tolerance test," *American Journal of Physiology-Endocrinology and Metabolism*, vol. 300, no. 6, pp. 955–965, 2011.

[83] C. Dalla Man, R. Rizza, and C. Cobelli, "Meal simulation model of the glucose-insulin system," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 10, pp. 1740–1749, 2007.

[84] C. D. Man, F. Micheletto, D. Lv, M. Breton, B. Kovatchev, and C. Cobelli, "The UVA/PADOVA type 1 diabetes simulator: New features," *Journal of Diabetes Science and Technology*, vol. 8, no. 1, pp. 26–34, 2014.

[85] M. Wilinska, L. Chassin, C. Acerini, J. Allen, D. Dunger, and R. Hovorka, "Simulation environment to evaluate closed-loop insulin delivery systems in type 1 diabetes," *J.Diab.Sci.Techn.*, vol. 4, no. 1, pp. 132–144, 2010.

[86] D. Enns, "Model reduction with balanced realizations: an error bound and a frequency weighted generalization," *In Proceedings of the 23rd IEEE Conference on Decision and Control*, pp. 127–132, 1984.

[87] D. Kumar and S. Nagar, "Square-root optimal hankel norm approximation technique for order reduction of non-minimal systems," in *Proceedings of the 2014 International Electrical Engineering Congress (iEECON)*, pp. 1–4, 2014.

[88] K. Turksoy, S. Samadi, J. Feng, E. Littlejohn, L. Quinn, and A. Cinar, "Meal detection in patients with type 1 diabetes: A new module for the multivariable adaptive artificial pancreas control system," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 1, pp. 47–54, 2016.

[89] L. Meneghetti, A. Facchinetti, and S. D. Favero, "Model-based detection and classification of insulin pump faults and missed meal announcements in artificial pancreas systems for type 1 diabetes therapy," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 1, pp. 170–180, 2021.

[90] B. Lantos, *Theory and design of control systems I., in Hungarian*. Budapest: Akademia Press, 2003.

[91] O. Bhat, D. Khan, and R. Yousuf, "Automated glucose control: A review," *International Journal of E-Health and Medical Communications*, vol. 12, 09 2021.

[92] N. S. Oliver, C. Toumazou, A. E. G. Cass, and D. G. Johnston, "Glucose sensors: a review of current and emerging technology," *Diabetic Medicine*, vol. 26, no. 3, pp. 197–210, 2009.

[93] O. Amft, H. Junker, and G. Troster, "Detection of eating and drinking arm gestures using inertial body-worn sensors," in *Proceedings of the 9th IEEE International Symposium on Wearable Computers (ISWC'05)*, pp. 160–163, 2005.

[94] R. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME, Ser. D, Journal of Basic Engineering*, vol. 82, pp. 34–45, 1960.

[95] M. Grewal and A. Andrews, *Kalman Filtering: Theory and Practice*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[96] B. Lantos, *Theory and design of control systems II., in Hungarian*. Budapest: Akademia Press, 2003.

[97] S.Haykin, *Kalman Filtering and Neural Networks*. New York, USA.: John Wiley & Sons, Inc., 2002.

[98] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[99] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1254–1269, 2009.

[100] M. Nørgaard, N. Poulsen, and O. Ravn, "New developments in state estimation of nonlinear systems," *Automatica*, vol. 36, pp. 1627–1638, 11 2000.

[101] I. Arasaratnam, S. Haykin, and R. Elliott, "Discrete-time nonlinear filtering algorithms using Gauss–Hermite quadrature," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 953–977, 2007.

[102] B. Jia, M. Xin, and Y. Cheng, "Sparse-grid quadrature nonlinear filtering," *Automatica*, vol. 48, no. 2, pp. 327–341, 2012.

[103] N. Gordon, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings of Communications, Radar and Signal Processing*, vol. 140, pp. 107–113(6), April 1993.

[104] S. Julier and J. Uhlmann, "Reduced sigma point filters for the propagation of means and covariances through nonlinear transformations," in *Proceedings of the 2002 American Control Conference (IEEE Cat. No.CH37301)*, vol. 2, pp. 887–892, 2002.

[105] R. Turner and C. Rasmussen, "Model based learning of sigma points in unscented Kalman filtering," in *Proceedings of the 2010 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 178–183, 2010.

[106] J. R. Rice, *Numerical Methods in Software and Analysis (Second Edition)*. San Diego: Academic Press, second edition ed., 1993.

[107] F. Faubel and D. Klakow, "A transformation-based derivation of the kalman filter and an extensive unscented transform," in *Proc. 15th Workshop on Statistical Signal Processing of IEEE Signal Processing Society*, (Cardiff, United Kingdom), pp. 161–164, 2009.

[108] M. G. Rutten, "Square-root unscented filtering and smoothing," in *Proceedings of the 2013 IEEE 8th International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 294–299, 2013.

[109] Z. Mahmoudi, K. Nørgaard, N. K. Poulsen, H. Madsen, and J. B. Jørgensen, "Fault and meal detection by redundant continuous glucose monitors and the unscented Kalman filter," *Biomedical Signal Processing and Control*, vol. 38, pp. 86–99, 2017.

[110] E. Aguirre-Zapata, J. Cardenas-Cartagena, and J. Garcia-Tirado, "Glycemic monitoring in critical care using nonlinear state estimators," *Proceedings of the 20th IFAC World Congress*, vol. 50, no. 1, pp. 4430–4435, 2017.

[111] A. Duun-Henriksen, S. Schmidt, R. Røge, J. Møller, K. Nørgaard, J. Jørgensen, and H. Madsen, "Model identification using stochastic differential equation grey-box models in diabetes," *Journal of diabetes science and technology*, vol. 7, pp. 431–440, 03 2013.

[112] D. A. Drexler, J. Sápi, and L. Kovács, "Positive nonlinear control of tumor growth using angiogenic inhibition," *Proceedings of the 20th IFAC World Congress*, vol. 50, no. 1, pp. 15068 – 15073, 2017.

[113] L. Angrisani, G. Annuzzi, P. Arpaia, L. Bozzetto, A. Cataldo, A. Corrado, E. D. Benedetto, V. Di Capua, R. Prevete, and E. Vallefuoco, "Neural network-based prediction and monitoring of blood glucose response to nutritional factors in type-1 diabetes," in *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, pp. 1–6, 2022.

[114] A. E. Fathi, E. Palisaitis, B. Boulet, L. Legault, and A. Haidar, "An unannounced meal detection module for artificial pancreas control systems," in *Proceedings of the 2019 American Control Conference (ACC)*, pp. 4130–4135, 2019.

[115] K. Kölle, A. L. Fougner, K. A. Frelsøy Unstad, and O. Stavdahl, "Fault detection in glucose control: Is it time to move beyond CGM data?," *Proceedings of the 10th IFAC Symposium on Biological and Medical Systems BMS 2018*, vol. 51, no. 27, pp. 180–185, 2018.

[116] L. Dénes-Fazakas, L. Szilágyi, J. Tasic, L. Kovács, and G. Eigner, "Detection of physical activity using machine learning methods," in *Proceedings of the 2020 IEEE 20th International Symposium on Computational Intelligence and Informatics (CINTI)*, pp. 167–172, 2020.

[117] C. Scherer and S. Weiland, *Linear Matrix Inequalities in Control*. Lecture Notes, Delft Center for Systems and Control, Delft, The Netherlands, 2004.

[118] F. Bianchi, M. Moscoso-Vásquez, P. Colmegna, and R. Sánchez-Peña, "Invalidation and low-order model set for artificial pancreas robust control design," *Journal of Process Control*, vol. 76, pp. 133–140, 02 2019.

[119] K. Zhou, *Robust and Optimal Control*. New Jersey: Prentice Hall, 1996.

[120] L. Kovács, A. György, B. Kulcsár, P. Szalay, B. Benyó, and Z. Benyó, "Robust control of type 1 diabetes using μ-synthesis," in *Proceedings of the UKACC International Conference on Control 2010*, pp. 1–6, 2010.

[121] N. Resalat, J. El Youssef, N. Tyler, J. Castle, and P. G. Jacobs, "A statistical virtual patient population for the glucoregulatory

system in type 1 diabetes with integrated exercise model," *PLOS ONE*, vol. 14, pp. 1–17, 07 2019.

[122] R. LaPorte, S. Akazawa, A. Drash, C. Gamboa, H. Lee, I. Libman, E. Marler, S. Orschiedt, T. Songer, and N. Tajima, "Diabetes and the internet," *Diabetes Care*, vol. 18, pp. 890–895, 6 1995.

[123] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1." `http://cvxr.com/cvx`, Mar. 2014.

[124] M. Grant and S. Boyd, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control* (V. Blondel, S. Boyd, and H. Kimura, eds.), Lecture Notes in Control and Information Sciences, pp. 95–110, Springer-Verlag Limited, 2008. `http://stanford.edu/~boyd/graph_dcp.html`.

[125] C. D. Man, R. A. Rizza, and C. Cobelli, "Mixed meal simulation model of glucose-insulin system," in *Proceedings of the 2006 International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 307–310, 2006.

[126] K. Pippitt, M. Li, and H. E. Gurgle, "Diabetes mellitus: Screening and diagnosis.," *American family physician*, vol. 93, pp. 103–109, January 2016.

[127] G. Adanaş Aydın, P. Özdemir Akdur, and G. Özgen, "The effect of glucose tolerance test on fetoplacental circulation," *Taiwanese Journal of Obstetrics and Gynecology*, vol. 60, no. 4, pp. 723–727, 2021.

[128] L. Magni, D. Raimondo, C. D. Man, M. Breton, S. Patek, G. D. Nicolao, C. Cobelli, and B. Kovatchev, "Evaluating the efficacy of closed-loop glucose regulation via control-variability grid analysis," *J Diab Sci Techn*, vol. 2, no. 4, pp. 630–635, 2008.

[129] G. Sibley, G. Sukhatme, and L. Matthies, "The iterated sigma point Kalman filter with applications to long range stereo," vol. 8, 08 2006.

[130] B. Jia and M. Xin, "Sparse-grid quadrature $\mathcal{H}_\infty$ filter for discrete-time systems with uncertain noise statistics," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 3, pp. 1626–1636, 2013.

PUBLICATIONS OF THE AUTHOR RELATED TO THE THESES

[KFS+15] L. Kovács, T. Ferenci, J. Sápi, G. Eigner, J. Klespitz, P. Szalay, M. Kozlovszky, and I. Rudas. Physiological modeling and control at Óbuda university. In *Proceedings of the 2015 IEEE 10th Jubilee International Symposium on Applied Computational Intelligence and Informatics*, pages 21–25, 2015.

[KKS+14] Levente Kovács, Miklós Kozlovszky, Péter Szalay, György Eigner, Péter István Sas, Tamás Ferenci, Zsuzsanna Almássy, Enikő Felszeghy, Győző Kocsis, József Fövényi, Krisztina Wudi, Anna Körner, László Kautzky, Hajnalka Soós, Andrea Orbán, Tamás Niederland, Andrea Juhászné Tuifel, Tímea Tóthné Sebestyén, Mária Hocsi, Andrea Soós, András Török, and László Barkai. Magyar mesterséges hasnyálmirigy projekt. Eredmények és távlatok. *Diabetologia Hungarica*, 22:73–76, 2014.

[KKSE13] Levente Kovács, Miklós Kozlovszky, Péter Szalay, and Péter István Eigner, György és Sas. A magyar mesterséges hasnyálmirigy projekt legújabb eredményei. In *A Magyar Gyermekorvosok Társasága és a Magyar Diabétesz Társaság XXX. Gyermekdiabétesz tudományos ülése*, 10 2013.

[KS12] Levente Kovács and Peter Szalay. $\mathcal{H}_\infty$ robust control of a T1DM model. *IFAC Proceedings Volumes*, 45(18):61–66,

2012. 8th IFAC Symposium on Biological and Medical Systems.

[KS16] Levente Kovács and Péter Szalay. *Uncertainties and Modeling Errors of Type 1 Diabetes Models*, pages 211–225. Springer International Publishing, 2016.

[KSAB13] Levente Kovács, Péter Szalay, Zsuzsanna Almássy, and László Barkai. Applicability results of a nonlinear model-based robust blood glucose control algorithm. *Journal of diabetes science and technology*, 7(3):708–716, 2013.

[KSE+14] Levente Kovács, Johanna Sápi, György Eigner, Tamás Ferenci, Péter Szalay, József Klespitz, Balázs Kurtán, Miklós Kozlovszky, Dániel A. Drexler, Péter Pausits, István Harmati, Zoltán Sápi, and Imre J. Rudas. Model-based healthcare applications at Óbuda university. In *Proceedings of the 2014 IEEE 9th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI)*, pages 183–187, 2014.

[KSF+11] L Kovács, P. Szalay, T. Ferenci, D. A. Drexler, J. Sápi, I. Harmati, and Z. Benyó. Modeling and optimal control strategies of diseases with high public health impact. In *Proceedings of the 2011 15th IEEE International Conference on Intelligent Engineering Systems*, pages 23–28, 2011.

[KSF+12] L Kovács, P. Szalay, T. Ferenci, J. Sápi, P. Sas, D.A. Drexler, I. Harmati, B. Benyó, and A. Kovács. Model-based control algorithms for optimal therapy of high-impact public health diseases. In *Proceedings of the 2012 IEEE 16th International Conference on Intelligent Engineering Systems (INES)*, pages 531–536, 2012.

[KSF+13] L. Kovács, J. Sápi, T. Ferenci, P. Szalay, D.A. Drexler, Gy. Eigner, P.I. Sas, B. Kiss, I. Harmati, M. Kozlovszky, and Z. Sápi. Model-based optimal therapy for high-impact diseases. In *Proceedings of the 2013 IEEE 17th International Conference on Intelligent Engineering Systems (INES)*, pages 209–214, 2013.

[KSS+13] Levente Kovács, Péter Szalay, Péter István Sas, György Eigner, Zsuzsanna Almássy, Enikő Felszeghy, Győző Kocsis, József Fövényi, Anna Körner, László Kautzky, Hajnalka Soós, Andrea Orbán, Tamás Niederland, Andrea Juhászné Tuifel, Tímea Tóthné Sebestyén, Andrea Soós,

András Török, and László Barkai. Preliminary model-free results of a Hungarian robust artificial pancreas algorithm. *Diabetes Technology and Therapeutics*, 15(Suppl1):A–96, 2013.

[KSZ11] Levente Kovács, Péter Szalay, and Almássy Zsuzsanna. Validation results of a modern robust control algorithm for type 1 diabetes. In *Proceedings of the MACRo 2011 – 3d International Conference on Recent Achievements in Mechatronics, Automation, Computer Sciences and Robotics*. Sapientia Kiadó, 2011.

[KTSS12] Levente Kovács, Ferenci Tamás, Johanna Sápi, and Peter Szalay. Népegészségügyi problémák számítógépes modellezése. *Informatika és Menedzsment az Egészségügyben: Az Egészségügyi Vezetők Szaklapja*, 11:49–55, 01 2012.

[SBK16] Péter Szalay, Zoltán Benyó, and Levente Kovács. Long-term prediction for T1DM model during state-feedback control. In *Proceedings of the 2016 12th IEEE International Conference on Control and Automation (ICCA)*, pages 311–316, 2016.

[SDKew] Péter Szalay, Dániel András Drexler, and Levente Kovács. Exploring robustness in blood glucose control with unannounced meal intake for type-1 diabetes patient. *Acta Polytechnica Hungarica*, under review.

[SEK+13] P Szalay, Gy Eigner, M Kozlovszky, I Rudas, and L Kovacs. The significance of LPV modeling of a widely used T1DM model. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, 2013:3531—–3534, 2013.

[SEK14] Péter Szalay, György Eigner, and Levente A. Kovács. Linear matrix inequality-based robust controller design for type-1 diabetes model. *Proceedings of the 19th IFAC World Congress*, 47(3):9247–9252, 2014.

[SMM+14] Peter Szalay, Adrienn Molnár, Márk Müller, György Eigner, Imre Rudas, Zoltán Benyó, and Levente Kovács. Comparison of sigma-point filters for state estimation of diabetes models. In *Proceedings of the 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2476–2481, 2014.

[SSBK14] Péter Szalay, László Szilágyi, Zoltán Benyó, and Levente Kovács. Sensor drift compensation using fuzzy interference system and sparse-grid quadrature filter in blood glucose control. In *Neural Information Processing*, pages 445–453, Cham, 2014. Springer International Publishing.

OTHER PUBLICATIONS OF THE AUTHOR

[Énzsöly et al., 2014] Énzsöly, A., Szabó, A., Kántor, O., Dávid, C., Szalay, P., Szabó, K., Szél, Á., Németh, J., and Lukáts, Á. (2014). Pathologic alterations of the outer retina in streptozotocin-induced diabetes. *Investigative ophthalmology & visual science*, 55(6):3686—3699.

[György et al., 2011] György, A., Szalay, P., Drexler, D. A., Benyó, B., Benyó, Z., and Kovács, L. (2011). Quasi model based optimal control of type 1 diabetes mellitus*. *Proceedings of the 18th IFAC World Congress*, 44(1):5012–5017.

[Kovács et al., 2012] Kovács, L., Szalay, P., Benyó, B., and Chase, J. G. (2012). Optimal tight glycaemic control supported by differential geometric methods. In *Proceedings of the 5th European Conference of the International Federation for Medical and Biological Engineering*, pages 351–354, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Kovács et al., 2011] Kovács, L., Szalay, P., Almássy, Z., Benyo, Z., and Barkai, L. (2011). Quasi in-silico validations of a nonlinear LPV model-based robust glucose control algorithm for type 1 diabetes. *Proceedings of the 18th IFAC World Congress*, 44(1):7114–7119.

[Kovács et al., 2011] Kovács, L., Szalay, P., Benyó, B., and Chase, G. J. (2011). Asymptotic output tracking in blood glucose control. A case study. In *Proceedings of the 2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 59–64.

[Kovács et al., 2011a] Kovács, L., Szalay, P., Benyó, B., and Geoffrey Chase, J. (2011a). Nonlinear control analysis of an ICU model for tight glycaemic control. *Proceedings of the 18th IFAC World Congress*, 44(1):1739–1744.

[Kovács et al., 2011b] Kovács, L., Szalay, P., Benyó, B., and Geoffrey Chase, J. (2011b). Robust tight glycaemic control of ICU patients. *Proceedings of the 18th IFAC World Congress*, 44(1):4995–5000.

[Kovács et al., 2012] Kovács, L., Szalay, P., Sas, P. I., Benyó, B. I., Benyó, Z., Almássy, Z., Felszeghy, E., Kocsis, G., Fövényi, J., Krisztina, W., Madarász, E., Zs, K., Körner, A., Kautzky, L., Grósz, A., Soós, A., Orbán, A., Török, A., and Barkai, L. (2012). Closing the Loop – Mesterséges hasnyálmirigy szabályozási algoritmusának in silico validációja 1-es típusú magyar diabetesesek adatain. *Diabetologia Hungarica*, 20(1).

[Szalay and Kovács, 2012] Szalay, P. and Kovács, L. (2012). *Applicability of Asymptotic Tracking in Case of Type 1 Diabetes*, pages 249–260. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Szalay et al., 2012] Szalay, P., Sas, P. I., Barkai, L., and Kovács, L. (2012). Nonlinear analysis of type 1 diabetes models by differential geometric approach. *Proceedings of the 8th IFAC Symposium on Biological and Medical Systems*, 45(18):55–60.