UNIVERSITÀ DEGLI STUDI DI BERGAMO

UNIVERSITÀ DEGLI STUDI DI BERGAMO

DOCTORAL THESIS

Data-driven Model Predictive Control strategies for blood glucose regulation in Artificial Pancreas

---

Author:
Beatrice SONZOGNI

Supervisor:
Prof. Antonio FERRAMOSCA

Co-supervisor:
Prof. José María MANZANO CRESPO

UNIVERSITÀ DEGLI STUDI DI BERGAMO

Abstract

Department of Management, Information and Production Engineering

Doctor of Philosophy

Data-driven Model Predictive Control strategies for blood glucose regulation in Artificial Pancreas

by Beatrice SONZOGNI

The Type 1 Diabetes Mellitus (T1DM) is a chronic metabolic disorder characterized by the autoimmune destruction of the pancreatic beta cells that produce insulin. As a consequence, the Blood Glucose (BG) levels may be too high (i.e., in hyperglycemia, with BG higher than 180 mg/dL). Thus, T1DM patients require exogenous insulin daily injections, which decrease BG. Such amounts have to be carefully computed, in order to avoid hypoglycemia (i.e., BG lower than 70 mg/dL). To try to ease the injection process and BG management, the Artificial Pancreas (AP) can be exploited. This is a system composed of three devices: (i) a Continuous Glucose Monitoring (CGM), which is a sensor that measures the glucose at the interstitial level, (ii) a control algorithm that computes the amounts of insulin, which are then injected into the subcutaneous tissue with (iii) an insulin pump. The core part is the algorithm, which is usually based on a model of the system. However, the T1DM disease varies significantly inter- and intra-patients, so it is complex to identify an accurate and customized model. This is why, in this thesis, machine learning-based controllers are proposed to learn the BG-insulin-meals dynamics directly from the patients' data.

Specifically, Model Predictive Control (MPC) is used as a control algorithm, in which the control action is obtained by minimizing a cost function (representing the goal), fulfilling the constraints, in a receding horizon fashion. The Componentwise Hölder Kinky Inference (CHoKI) is used as a learning method, which is an approach based on Kinky Inference. The proposed customized controllers are then tested on the 10

in-silico adult T1DM patients of the UVA/Padova simulator, which is FDA-accepted for preclinical studies.

The control goal is to drive and maintain the BG values inside the euglycemic range (i.e., between 70 and 180 mg/dL). To achieve it, the CHoKI-based MPC aims to control the basal insulin injections, leaving the meal boluses control to the bolus calculator of the simulator.

In this thesis, the first step was to identify the CHoKI learning method for each in-silico patient. Then, several CHoKI-based MPC schemes were designed. For example, including slack variables in the constraints to avoid infeasibilities, weighting differently the hypo- and hyperglycemia conditions in the cost function, or adding a limitation based on the Insulin On Board (IOB) estimations to avoid injecting too much insulin. The tightening of the constraints due to the use of the CHoKI approach tends to be too restrictive, leading to short control horizons; thus, stochastic constraints were proposed to try to broaden them. Then, since patients' BG-insulin behavior may change over time, an online updating of the CHoKI dataset was proposed, in order to learn and to react to these variations. A hybrid modeling is also proposed, merging a linear model with the CHoKI to improve the model BG predictions. The proposals are tested with different simulation settings, varying the meals, the Physical Activity (PA) session, and with circadian variability of Insulin Sensitivity (S_I).

The final results are quite satisfactory, especially given that the proposed controllers are able to reduce hypoglycemic events, when compared to standard therapy, which injects constant basal insulin.

Keywords. Artificial Pancreas, Model Predictive Control, Learning-based control, CHoKI

Acknowledgements

I would like to express my gratitude to my supervisor, Professor *Antonio Ferramosca*, for his guidance and patience throughout this journey, allowing me to grow professionally and personally.
I would also like to thank Professor *Fabio Previdi* for giving me this opportunity, as well as the entire *Control Systems & Automation Lab* for making me feel welcome, and Professor *Antonio Russo* and Professor *Mirko Mazzoleni* for their helpful suggestions.
Thanks also go to all my past and present colleagues, *Marco*, *Davide C.*, *Leandro*, *Nicholas*, *Davide P.*, *Giulia*, *Pietro*, *Matteo*, *Francesco*, *Stefano* and *Lorenzo*, for all the time spent together, the help they offered, the collected memories, and the many conversations we had. And to *Nicola*, for his support and patience, and for sharing not only our travels, but also the joys that only the AP can offer.
I would like to express my heartfelt thanks to Professor *José María Manzano* for his many helpful insights and guidance, and to everyone at the *Optimization and Control of Distributed Systems* research group at the Universidad Loyola in Seville for their warm welcome and for all the moments we spent together.
I am particularly grateful to *Andrea* and *Javi* for making me feel home, and, among the many, to *Diego*, *Dani*, *Javi R.*, *Enrique*, *Gabi*, *Thalia*, *Fede*, *Erid*, *Lari*, *Agustin*, *Jesus* and *Andrea B.*, who made me leave a piece of my heart in Seville.
I would also like to thank all the PhD students I have met and spent time with over the years, for their support and advice, and for all the laughs we have shared.
I would like to thank all my friends for their constant support and encouragement. I would especially like to thank *Greta*, *Letizia* and *Samanta*, my lifelong partners in adventure, for their patience and for celebrating each of my milestones as if they were their own. I would also like to thank *Martina* and *Elisa* for their affectionate and cheerful support over the years.
I am deeply grateful to my *parents* for enabling me to follow this path. I would also like to thank my sister *Veronica*, who has always been my role model, and my nieces, *Ginevra* and *Giulia*, for all the joy they have brought to my life.

Contents

List of Acronyms

ADA American Diabetes Association. 16, 66, 87, 114

AHCL Advanced Hybrid Closed Loop. 10, 15, 23, 24

AP Artificial Pancreas. iii, iv, 1, 2, 9, 10, 12–14, 19, 23, 49, 65, 79, 90, 105, 113, 122, 147, 148

ARX Autoregressive eXogenous. 37

BG Blood Glucose. iii, iv, 1–9, 11–13, 15–19, 21, 23, 24, 50–60, 64, 65, 67, 69, 70, 75–84, 86–91, 94, 95, 97, 101–106, 111, 113–118, 120–123, 132–144, 146–152

BW Body Weight. 19

CF Correction Factor. 8, 19

CGM Continuous Glucose Monitoring. iii, 8–11, 13, 15, 20, 55, 91, 97

CHoKI Componentwise Hölder Kinky Inference. iii, iv, 2, 23–25, 27, 38, 41–47, 49, 52–55, 57–61, 66, 75, 84, 85, 90, 94–98, 100–107, 110, 113, 116, 122, 123, 125, 126, 131–136, 138, 139, 141, 142, 145, 147–151

CR Carbohydrate-to-Insulin Ratio. 8, 19, 21, 94, 152

CSII Continuous Subcutaneous Insulin Infusion. 8, 9

CVGA Control-Variability Grid Analysis. 17, 18, 65, 66, 68–71, 75, 77, 80–82, 85, 86, 88, 89, 113–116, 119, 120, 122, 135, 139, 140

DMMS Diabetes Mellitus Metabolic Simulator. 19

DMMS.R Diabetes Mellitus Metabolic Simulator for Research. 19, 24, 84

DPP-4 Dipeptidyl Peptidase 4. 5

FDA U.S. Food and Drug Administration. iv, 2, 8, 9, 19, 147

FL Fuzzy Logic. 14

GIP Gastric Inhibitory Peptide. 5

GLP-1 Glucagon-Like Peptide 1. 5

GoF Goodness of Fitting. 58, 101

GRI Glycemia Risk Index. 16, 17, 65, 66, 68–71, 75, 77, 80–82, 85, 86, 88, 89, 113–116, 119, 120, 122, 135, 139, 140

HbA1c Glycated Hemoglobin. 4

HCL Hybrid Closed Loop. 13

IAD Insulin Action Duration. 70, 72, 74, 79

IOB Insulin On Board. iv, 8, 70, 72–84, 86–89, 94, 105, 117, 120, 122, 139, 145, 148–151

ISpS Input-to-State practically Stable. 162

ISS Input-to-State Stability. 35, 44, 45, 49, 131, 162

ISTAT Istituto nazionale di statistica. 1

KI Kinky Inference. iii, 24, 37–40, 42, 44–46, 48, 125–127

LACKI Lazily Adapted Constant Kinky Inference. 39, 41, 42, 45, 57, 126, 127, 131

LQR Linear Quadratic Regulator. 61, 62, 64, 113

LS Least-Squares. 94

MDII Multiple Daily Insulin Injections. 8, 9

MIMO Multiple Inputs Multiple Outputs. 37

MPC Model Predictive Control. iii, iv, 2, 14, 15, 23, 24, 27–31, 33–38, 40, 44–46, 49–54, 59, 61–67, 70, 72, 73, 75, 76, 78, 81, 84, 90, 105, 107, 111, 113–123, 128, 136, 138, 139, 141, 142, 145–152

NARX Nonlinear Autoregressive eXogenous. 37, 42, 53, 55

NN Neural Network. 37, 52

PA Physical Activity. iv, 84, 85, 87–89, 102, 104, 148

PID Proportional-Integrative-Derivative. 14, 15

RLS Regularized Least-Squares. 93

RMSE Root Mean Squared Error. 57, 58, 98, 101

RPI Robust Positively Invariant. 159

S_I Insulin Sensitivity. iv, 52, 84, 85, 87, 89, 90, 96, 98, 120, 121, 134, 137, 138, 142, 148, 149

SAP Sensor Augmented Pump. 9

SD Standard Deviation. 15

SGLT2 Sodium-Glucose Cotransporter 2. 5

SMBG Self-Monitoring of Blood Glucose. 20

T1DM Type 1 Diabetes Mellitus. iii, iv, ix, 1–6, 8, 9, 13, 15, 19–21, 23, 24, 51, 53, 55, 64, 74, 85, 87, 90, 96, 105, 109, 113, 117, 122, 125, 132, 142, 147, 149

T2DM Type 2 Diabetes Mellitus. 4, 5, 19

TAR Time Above Range. 16, 117, 120, 123, 139, 142

TAR_1 Time Above Range, level 1. 16

TAR_2 Time Above Range, level 2. 16

TBR Time Below Range. 16

TBR_1 Time Below Range, level 1. 16

TBR_2 Time Below Range, level 2. 16

TEG The Epsilon Group. 19

TIR Time In Range. 15, 16, 65, 66, 68, 69, 71, 75, 77–80, 83, 85, 87, 88, 113, 114, 116, 117, 119–121, 123, 135, 142, 145

TITR Time In Tight Range. 15

TZDs Thiazolidinediones. 5

UVA University of Virginia. iv, 2, 19, 20, 24, 25, 53, 55, 63, 64, 72, 74, 79, 84, 85, 87, 90, 96, 105, 106, 113, 117, 122, 134, 137, 138, 145, 147–149, 153

WHO World Health Organization. 1

List of Notations

Scalars, vectors and matrices

- $v \in \mathbb{R}^n$ - n-dimensional column vector v.
- $|v|$ - Elementwise absolute value.
- $\|v\|$ - Euclidean norm (unless otherwise indicated).
- $\|v\|_\infty$ - Infinity norm.
- (v, w) - Concatenation of vectors. $(v, w) := [v^T, w^T]^T$.
- $v \leq w$ - Componentwise inequality.
- $\mathbb{1}_n$ - n-dimensional column vector of ones.
- 0_n - n-dimensional column vector of zeros.
- $\tilde{y}$ - Noisy measurement of y.
- $\hat{y}$ - Predicted value of y.
- $\hat{y}(j|k)$ - Predicted value of y at time step j, given the measurement at time step k.
- $A \in \mathbb{R}^{m \times n}$ - m-n-dimensional matrix.
- A_i - i-th row of the matrix A.
- I_n - n-dimensional identity matrix.
- $\mathbb{1}_{m \times n}$ - m-n-dimensional matrix of ones.
- $0_{m \times n}$ - m-n-dimensional matrix of zeros.

Sets

- $\mathbb{I}_a^b$ - Set of integers from a to b.
- $\mathbb{N}_0$ - Set of natural numbers and zero.
- $\mathbb{R}^{m\times n}$ - m-n-dimensional set of real numbers.
- $\mathcal{A} \subseteq \mathbb{R}^n$ - n-dimensional set $\mathcal{A}$.
- $\|\mathcal{A}\|_\infty$ - Infinity norm of the set $\mathcal{A}$. $\|\mathcal{A}\|_\infty := \max_{a\in\mathcal{A}} \|a\|_\infty$.
- $\mathcal{A} \oplus \mathcal{B}$ - Minkowski sum. $\mathcal{A} \oplus \mathcal{B} := \{a + b : a \in \mathcal{A}, b \in \mathcal{B}\}$.
- $\mathcal{A} \ominus \mathcal{B}$ - Pontryagin difference. $\mathcal{A} \ominus \mathcal{B} := \{c : c + b \in \mathcal{A}, \forall b \in \mathcal{B}\}$.
- $\mathcal{A} \cup \mathcal{B}$ - Union. $\mathcal{A} \cup \mathcal{B} = \{x : x \in \mathcal{A} \text{ or } x \in \mathcal{B}\}$.
- $\mathcal{A} \times \mathcal{B}$ - Cartesian product. $\mathcal{A} \times \mathcal{B} := \{(a, b) : a \in \mathcal{A}, b \in \mathcal{B}\}$.
- $\mathcal{B}(r)$ - Ball of radius $r \in \mathbb{R}^n := \{x : |x| \leq r\} \subseteq \mathbb{R}^n$.
- $\mathcal{B}(\mathcal{A})$ - Cartesian closed topological hull of $\mathcal{A}$.
- $\mathbb{B}(r)$ - Positive box of radius $r \in \mathbb{R}^n$. $\mathbb{B}(r) := \{x : 0 \leq x \leq r\} \subset \mathbb{R}^n$.
- $\mathfrak{d}(a, \mathcal{A})$ - Distance metric from point a to set $\mathcal{A}$.
- $\text{Proj}_{\mathcal{A}}(v)$ - Projection of v onto the set $\mathcal{A}$. $\text{Proj}_{\mathcal{A}}(v) = \arg\min_{a\in\mathcal{A}} \|a - v\|$.

Symbols and functions

- k - Time step.
- $\mathbb{P}(\cdot)$ - Probability.
- $\mathbb{P}(\cdot, \cdot)$ - Joint probability.
- $\mathbb{P}(\cdot|\cdot)$ - Conditional probability.
- x^* - Optimal value of x.
- x^+ - Subsequent value of $x(k)$, i.e., $x(k + 1)$.

CHAPTER 1

Introduction

1.1 Motivation

Diabetes is a chronic metabolic disease characterized by elevated levels of sugar in the blood. It affects more than 830 million people worldwide, according to the World Health Organization (WHO) estimates in 2022, a number that has been steadily increasing over the past decades. In Italy, in 2022 ISTAT stated that around 3.9 million people declared to have diabetes [57]. The proportion of total deaths related to diabetes among adults (20-79 years) in 2024 is 29% [9]. Diabetes not only has a social impact, but also an economic one, since the total economic burden of diabetic patients amounted to €20.3 billion/year [88].

Among the several types of diabetes, Type 1 Diabetes Mellitus (T1DM), also known as insulin-dependent or juvenile, is characterized by insufficient insulin amount, due to the autoimmune destruction of the pancreatic beta cells that produce it. To face this lack, T1DM patients require daily exogenous insulin administrations as therapy. This can be achieved through conventional therapy, i.e., basal-bolus therapy, which can be administered in various ways. The insulin can be injected through a syringe or a pen, whose amount is computed by the patient before meals, knowing the actual Blood Glucose (BG) level (with finger prick tests) and the grams of carbohydrates of the meal, by using a simple formulation with some parameters defined by the physicians, considering the patient's behavior [23]. Another option is the so-called Artificial Pancreas (AP), which exploits a sensor that continuously measures the BG values and then automatically injects the insulin through a pump

attached to the body, and a customized control algorithm performs the calculation of the dose [94].

Diabetes, especially T1DM, affects the quality of life of the patients and of their caregivers, requiring lifelong therapy and self-monitoring education. But it is difficult to maintain optimal BG levels through human intervention alone. This leads to the need for innovative therapies that are increasingly less invasive and more automated. In addition, this is a disorder that varies greatly inter- and intra-patient, which means looking for a therapy that is more precise, personalized and customized on the subject needs.

This thesis aims to address the problem just described. It focuses on the use of AP as treatment for T1DM subjects, to reduce the patients' intervention. In detail, it consists in the design of customized control algorithms, with the objective of computing the optimal insulin amount needed for the patient to reach and maintain the BG inside the safe range. Especially avoiding hypoglycemic conditions (i.e., when the BG is less than 70 mg/dL), which is particularly dangerous and can also lead the patient to diabetic coma or death in the short term. To do that, Model Predictive Control (MPC) is used as control algorithm. This requires a model of the system to make predictions of its future behavior, but due to the complexity of the disease, the identification of an accurate model is challenging. In particular, it is difficult to take into account all the factors that could influence the insulin-glucose relationship and also to be able to consider all the variations between different patients and even over time within the individual. That is why in this work a learning method called Componentwise Hölder Kinky Inference (CHoKI) is exploited to understand the behavior of the insulin-meal-glucose system, allowing to learn the patients' response directly from their data. Several controllers are proposed, and these are all tested on the T1DM in-silico patients of the University of Virginia (UVA)/Padova simulator, which is accepted by the U.S. Food and Drug Administration (FDA) for pre-clinical studies.

1.2 Diabetes

Diabetes is a family of metabolic disorders characterized by the body's inability to correctly balance the blood sugar level, due to the absence or insufficient insulin

production by the pancreas, or due to the body's malfunction in its use. As a consequence, it is characterized by high BG levels, which could lead to micro- and macrovascular complications [9, 30].

1.2.1 Introduction

The *pancreas* is an organ located in the upper left part of the abdomen, behind the stomach. It has both exocrine and endocrine functions. The exocrine function involves the production of digestive enzymes such as trypsin and chymotrypsin which digest proteins, ipase which breaks down fats, and amylase which digests carbohydrates. While the endocrine function involves the secretion of hormones produced by the islets of Langerhans into the bloodstream. Specifically, insulin, produced by beta cells, is needed to lower BG levels, and it must cooperate with glucagon, produced by alpha cells, which raises BG when necessary [118].

Since glucose is an essential nutrient for all cells in the organism, the human body has an autonomous regulatory system consisting of insulin and glucagon. This system allows the body to maintain the BG to a relatively constant level over time. In detail, the physiological relationship among insulin, glucagon, and glucose works as follows: after a meal, the carbohydrates are digested into glucose and then released into the bloodstream. This leads to an increase in BG levels, which is a signal for the pancreas to produce insulin. Insulin helps cells recognise that they need to absorb glucose from the bloodstream. Some cells use this glucose for energy, while others, such as those in the liver and muscles, store any excess glucose by converting it into a substance called glycogen. The glucagon hormone has the opposite action: its production is activated when the BG level decreases. This hormone prompts liver and muscle cells to convert glycogen into glucose, which is then released into the bloodstream, where it can be used by other cells as an energy source. The glucagon-insulin system is always active, ensuring that BG levels remain within a physiologically safe range and with constant energy supply for all cells [102].

When the aforementioned mechanism does not work correctly, the diabetic disease could occur. There exist different types of diabetes:

- Type 1 Diabetes Mellitus (T1DM): it is a chronic condition that is most often diagnosed in children and adolescents. It is characterized by the autoimmune destruction of pancreatic beta cells, leading to high BG level, due to the absence or not enough insulin production.

- Type 2 Diabetes Mellitus (T2DM): it appears mainly in adulthood. It affects how the body uses glucose for energy, since it is a combination of inadequate insulin secretion and insulin resistance of the cells, which causes its inability to act properly.
- Pre-diabetes: for the cases in which the BG level is higher than normal, but not as in T2DM.
- Gestational diabetes: which is high BG levels during pregnancy.

The exact causes are unknown. For T1DM, it could be due to genes or viruses that attack the immune system, and for T2DM, it could be due to a combination of genetics and lifestyle factors [30, Chapter 2].

In most cases, the diagnosis of diabetes is made only after the presence of severe symptoms, caused by extremely high or low BG levels, and in the more serious cases, it can be diagnosed even just after hospitalization. To help in the disease diagnosis, new innovations were introduced in the late 1970s and early 1980s, including the Glycated Hemoglobin (HbA1c) testing and the patient self-glucose testing, to enhance both the detection and control. The primary tests for diagnosis include blood tests to measure the levels of sugar in the bloodstream and urine tests to assess glucose and ketone levels (ketones in blood or urine indicate that the body is obtaining energy by burning fats instead of glucose). For the first, the concentration of the HbA1c is examined (i.e., also known as glycosylated hemoglobin, which is a form of hemoglobin chemically modified by the attachment of glucose molecules). It gives information about the average blood glucose levels of the previous 2-3 months (knowing that the average lifespan of a blood cell is approximately 3 months), allowing one to see the possible damage caused by an enduring high level of glucose. An optimal value of HbA1c is about 7%, but it could vary according to the patient. The guidelines for laboratory analysis for diabetes diagnosis are further detailed in [105].

In order to manage the diabetic disorder, the key point is the restoration of the BG concentration within the physiological range. In the majority of the population, this range, also known as the *euglycemic range*, is between 70 and 180 mg/dL, even if it can be changed according to patients' characteristics. Above this threshold, the person is in a state of *hyperglycemia*, and below it, in a state of *hypoglycemia*. For the first one, a series of long-term complications could appear, such as cardiovascular diseases, neuropathies (worsening of the vessels that are connected with the nerves),

kidney damages (reduction in the filtering action, with possible dialytic sessions required), diabetic retinopathy (possible problems such as glaucoma, cataracts and blindness) or urinary infections. For hypoglycemia, there is the risk of more severe problems in a short time, such as confusion, seizures, or collapse, leading to diabetic coma [9].

Thus, after the diabetes diagnosis, the first steps that must be taken are a healthy lifestyle and a balanced diet. The first treatment was the Allen diet, with low-carbohydrate and low-calorie foods. But these are not enough, and specifically, T1DM patients require daily insulin injections (thanks to the insulin discovery by Sir Frederick G Banting's team in 1922 [62]). Some patients with T2DM may also need some drugs to help manage their BG levels in addition to insulin injections [30, Chapter 9]. Some of the commonly used classes of medications include:

- Metformin: which is a biguanide medication, it limits the liver's ability to release sugar and improves cells' sensitivity to insulin.
- Dipeptidyl Peptidase 4 (DPP-4) inhibitors: by interfering in the process that breaks down Glucagon-Like Peptide 1 (GLP-1) and Gastric Inhibitory Peptide (GIP). GLP-1 and GIP are hormones that reduce BG levels in the body, but they are broken down very quickly. DPP-4 inhibitors allow these hormones to remain active in the body longer, lowering BG levels only when they are elevated.
- GLP-1 and dual GLP-1/GIP receptor agonists: similar to before, to help the use of these hormones to improve BG management.
- Sodium-Glucose Cotransporter 2 (SGLT2) inhibitors: SGLT2 works in the kidney to reabsorb glucose; SGLT2 inhibitors block this action, causing excess glucose to be eliminated in the urine.
- Sulfonylureas: stimulate pancreatic beta cells to release more insulin.
- Thiazolidinediones (TZDs): help insulin work better in muscle and fat, and reduce glucose production in the liver.

1.2.2 Type 1 Diabetes Mellitus

T1DM is characterized by high BG due to insulin deficiency, caused by the autoimmune destruction of the pancreatic beta cells. Its peak in diagnosis is between the

ages of 5 and 7 years. The clinical presentation varies individually, but typical symptoms include polydipsia, polyuria, lack of energy or fatigue, constant hunger, sudden weight loss, blurred vision, and diabetic ketoacidosis. Ketoacidosis is a serious metabolic condition that occurs when the body, due to a lack of insulin, begins to burn fat for energy. This process produces ketone bodies (which are produced by the liver and serve as an energy source when glucose is not available), which can, in excess, lead to acidification of the blood [8].

T1DM patients to achieve their glycemic target require lifelong exogenous insulin injections, which need to be tailored based on each patient's health status and any coexisting conditions. Besides insulin therapy, also glucose monitoring, disease-specific education (including self-BG monitoring and self-insulin administration, and ways to avoid and treat hypoglycemia), healthy diet, and lifestyle modifications are required to obtain an appropriate T1DM management.

1.2.3 Conventional therapies

T1DM patients require daily insulin injections to cope with the autoimmune destruction of the insulin-producing pancreatic beta cells. The first insulin injections were performed in 1922 at Toronto General Hospital, using a pancreatic extract from a dog, which resulted in a decrease of BG, but with a sterile abscess at one of the injection sites. After further purification of the pancreatic extract through alcohol treatment, in 1992, a reduction in BG and disappearance of ketonuria were observed, with little or no toxic reactions after the subcutaneous administration. In 1982, the first biosynthetic human insulin (i.e., Humulin) was approved for clinical use [19].

The main goal of the insulin therapy is to try to mimic the functioning of a healthy pancreas. Physiologically, the beta cells continuously produce and release insulin in the bloodstream, to manage the BG levels throughout the day. After a meal, insulin production is increased due to the increment in sugar levels. Based on that, to reproduce this action, it is possible to divide it into two parts: basal and boluses. The *basal* is the insulin amount continuously given to the patient to manage the BG in fasting periods, while the *boluses* are bigger amounts needed to face the effects of the carbohydrate ingestion or unexpected hyperglycemic events.

There exist several types of insulin, namely [35]:

- Rapid-acting insulin: it begins to work about 15 minutes after the injection, it has its peak in about one or two hours, and it lasts between two to four hours.
- Regular or short-acting insulin: it usually reaches the bloodstream within 30 minutes after injection, it peaks in two/three hours, and it is effective for approximately three to six hours.
- Intermediate-acting insulin: it generally reaches the bloodstream in about two to four hours after injection, it peaks four to 12 hours later, and it is effective for about 12 to 18 hours.
- Long-acting insulin: it reaches the bloodstream several hours after injection, and it tends to lower glucose levels up to 24 hours.
- Ultra long-acting insulin: it reaches the bloodstream in six hours, it does not peak, and it lasts about 36 hours or longer.

Usually, the boluses are made by faster insulin, like the rapid-acting or short-acting. The basal could be continuously injected (in the case of therapy with the pump), thus using fast insulin, or can be injected once (or twice) per day, with an intermediate, long-acting, or ultra long-acting insulin.

The insulin is injected into the fatty tissue under the skin, through a small needle, using a syringe or a pen. The first manufactured insulin pen (i.e., NovoPen, Novo Nordisk) was introduced in 1985, before syringes were used (the first specialized syringe for insulin injection was manufactured in 1924). Among the advantages of the pens over the syringes, there are greater accuracy, convenience, patient preference, and adherence to the therapy [62, 106].

The first method of administering conventional therapy is to inject the required amount of insulin with a pen after taking a BG measurement using a fingerstick. The procedure is the following: the patient uses a lancet device to prick the fingertip and then squeezes it to get a drop of blood. This has to be applied on a test strip, which is inserted in the glucose meter that will display the BG value. Once the measurement is done, the patient needs to compute and select the insulin dose on the pen (mainly based on the glucose level and the meal amount), decide the injection site (usually the abdomen, thigh, or upper arm), and inject by pushing the button on the pen. The needle of the insulin pen, the lancet, and the strip are disposable devices.

The insulin dose I_{bolus} can be computed with the following formulation:

$$I_{\text{bolus}} = \frac{g_{CHO}}{CR} + \frac{BG - BG_{\text{tar}}}{CF} - IOB,$$

where g_{CHO} is the grams of the carbohydrate intake, Carbohydrate-to-Insulin Ratio (CR) represents how many grams of carbohydrates are covered by one unit of insulin, BG is the actual blood glucose value, BG_{tar} is the BG target value, and Correction Factor (CF) (also known as insulin sensitivity factor) represents how much one unit of insulin lowers BG level, whose approximation can be obtained by dividing the total daily insulin dose by 1800. The estimations of the total daily dose, CR, BG_{tar}, and CF are usually made by the physician. The Insulin On Board (IOB) represents the current insulin still active in the body [23].

This is the Multiple Daily Insulin Injections (MDII) strategy. The patients need to repeat this procedure at least before each meal and also whenever they feel the need. Thus, this impacts the quality of the patients' lives, but also of their caregivers. Especially, considering, for example, T1DM children, who need their parents' supervision to do that. For this reason, other devices were developed, aiming to ease patients' and caregivers' lives, but also to have more precise BG control to improve diabetes management.

The MDII therapy can be replaced with the Continuous Subcutaneous Insulin Infusion (CSII) one, thanks to the development of the insulin pump. The first prototype of an insulin pump was designed by Dr. Kadish in 1963, but it has to be carried like a backpack due to its huge dimensions. In 1974, Dr. Pfeiffer developed the "Biostator", an insulin pump with intravenous continuous glucose monitoring and closed-loop intravenous insulin infusion features. Then, the first wearable insulin pump was designed by Kamen in 1976, allowing the introduction of insulin pump therapy. In 1983, MiniMed (then acquired by Medtronic in 2001) introduced their first insulin pump, MiniMed 502. The new generation, released in the 1990s, is comparatively small, compact, and effective. In 2003, Medtronic (Northridge, CA) introduced the first insulin pump in which the BG readings were wirelessly transmitted to the insulin pump, and the insulin doses were suggested by a bolus calculator. MiniMed also introduced the implantable insulin pump to deliver insulin intraperitoneally in 1986, but then this idea was dropped in 2007, due to issues with the pump and catheter, as well as concerns about under-delivery of insulin [28, 62].

In 1999, the FDA approved the first CGM, which collected data for three days. In

2004, Medtronic introduced the Guardian real-time CGM system, which could notify users of potentially dangerous hyperglycemia or hypoglycemia, and the same company released the first integrated pump and sensor (Sensor Augmented Pump (SAP)) by 2006. That same year, Dexcom (San Diego, CA) introduced its first real-time CGM. In 2008, the FreeStyle Navigator by Abbott (Alameda, CA) was released in the United States. At the beginning, all the CGM devices required to make a BG confirmation check before taking insulin decisions [55, 94].

The two aforementioned devices, i.e., the CGM and the insulin pump, started to be utilized together, leading to the development of the Artificial Pancreas (AP).

In addition to the conventional MDII, CSII, and AP therapies, there also exist other possibilities. One is the breathable insulin. In 2004, the FDA approved Afrezza, which is the only ultra rapid-acting inhaled insulin. It lowers the BG levels in adults patients in around 12-15 minutes. The device consists of an inhaler and cartridges, which contain Technosphere Insulin (TI) powder. TI is composed of recombinant human insulin and fumaryl diketopiperazine (FDKP), which is an inert excipient. Once inhaled, the particles carry insulin into the alveoli, where they dissolve and are absorbed through the alveolar walls. Then insulin is absorbed into the bloodstream, while FDPK is excreted, mainly by the kidneys [63]. It is a less invasive method, but the most common side effects are hypoglycemic events, cough, throat pain or irritation, and lung problems.

Another possibility for T1DM patients is the pancreatic transplantation. However, it is not widely practiced due to the risks associated with using immunosuppressive drugs and undergoing surgery [94].

1.3 Artificial Pancreas

To minimize the patient intervention that is needed in the conventional MDII therapy and to improve diabetic disease, the Artificial Pancreas (AP) can be employed. It is a system composed of three different devices that work together wirelessly: a sensor to measure the BG levels, a pump to deliver the insulin injections, whose amount has to be computed by a control algorithm. The system is represented in Figure 1.1, and the devices are detailed further in the following.

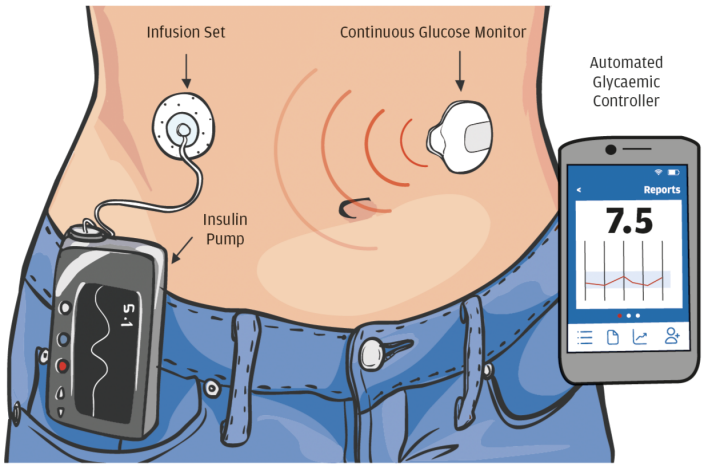


FIGURE 1.1: AP system, with a CGM that measures the glucose levels, a control algorithm that computes the insulin amount, which is then injected through a pump.

This is an Advanced Hybrid Closed Loop (AHCL) system, since it automatically manages the basal insulin amount, but still requires the intervention of the patient to insert the carbohydrate amount of the meal and to confirm the bolus injection [94].

1.3.1 Glucose sensor

The Continuous Glucose Monitoring (CGM) is a small non-invasive sensor that measures glucose values in the interstitial fluid at regular intervals (usually, every 5 min), thanks to a small needle inserted under the skin, commonly on the belly or arm.

A CGM is made of three parts: a wearable tiny sensor inserted under the skin, a transmitter that wirelessly sends readings to the receiver that stores information, which could be on a smartphone, on an insulin pump, or on a separate device.

The measures are made by converting glucose levels in the interstitial fluid into electronic signals. This electric current (which is proportional to the glucose concentration) is then sent to a reader from a transmitter attached to the sensor [64]. In detail, the sensor is a three-electrode electrochemical enzymatic glucose sensor, based on the transformation of glucose molecules ($C_6H_{12}O_6$) into hydrogen peroxide (H_2O_2) by the enzyme glucose oxidase (GO_x). The H_2O_2 concentration is then anodically detected at the surface of a platinum working electrode, biased with respect to an integrated iridium oxide (IrOx) electrode, when a bias of $0.5V$ is applied, thus providing a current proportional to the initial glucose concentration, according

(A) Dexcom G7 (B) Freestyle Libre 3 (Abbott) (C) Guardian Sensor 3 (Medtronic)

FIGURE 1.2: Continuous Glucose Monitoring examples. Images taken from: *www.dexcom.com*, *www.freestyle.abbott*, and *www.medtronic-diabetes.com*

to the reactions:

$$\text{Glucose} + O_2 \xrightarrow{GO_x} H_2O_2 + \text{Gluconic Acid}$$
$$H_2O_2 \xrightarrow{0.5V\ vs.IrOx} O_2 + 2H^+ + 2e^-$$

A counter platinum electrode closes the electrical circuit [99]. Then, the raw current sensor signal (i.e., $2e^-$) is translated into the BG concentration estimate through a calibration process, to deal also with the time delay of the glucose concentration in the plasma and in the interstitial fluid.

Sensors must be replaced at specific times, typically every 7-14 days, depending on the type of sensor. It is also important to rotate the application site of the sensor, to prevent skin irritation and ensure accurate measurements.

Among the ones on the market, the most used are shown in Figure 1.2. Namely, Dexcom G7, Freestyle Libre 3 (Abbott), and Guardian Sensor 3 (Medtronic).

1.3.2 Insulin pump

The insulin pump was introduced in the late 1970s. This is a small device worn on an external belt or attached with an adhesive patch, with a fine cannula implanted under the skin, to deliver insulin into the subcutaneous fatty tissue. From there, the insulin is absorbed into the bloodstream, with a rate that depends on its composition [51]. It requires monitoring and insulin refill, to ensure its functioning and patient safety. Additionally, the positioning of the infusion set needs to be changed to prevent skin irritation, infections, and poor insulin absorption (for example, due

(A) Thethered version (Tandem t:slim X2)

(B) Patch version (Insulet Omnipod 5)

FIGURE 1.3: Insulin pump examples. Images taken from: *www.tandemdiabetes.com* and *www.omnipod.com*

to lipohypertrophy, a condition where lumps or soft spots develop under the skin, blocking insulin delivery) [117].

There exist two types of pumps: tethered and patch. The former is attached to the body by a small tube connected to the cannula. The system is composed of an infusion set, a reservoir, a thin tube, and a cannula. The latter is tubeless, attached directly to the body. The system is made of a pod (which is the capsule with insulin applied directly to the body, with the needle) and a controller (that could be an external device or a smartphone) [62].

In Figure 1.3, two examples are reported: on the left Tandem t:slim X2 (a tethered one), while on the right Insulet Omnipod 5 (a patch one). Both with the controller on an external device.

1.3.3 Control algorithm

The core part of AP is the control algorithm, since it defines the quality of the therapy performances, as it is the part that computes the insulin amount that will be then injected to maintain the BG within the euglycemic range.

Control problem

In general, a control problem concerns the design of a strategy that allows a dynamic system to behave as desired, even in the presence of external disturbances or uncertainties (which could be measurable or not, but cannot be controlled). It is a

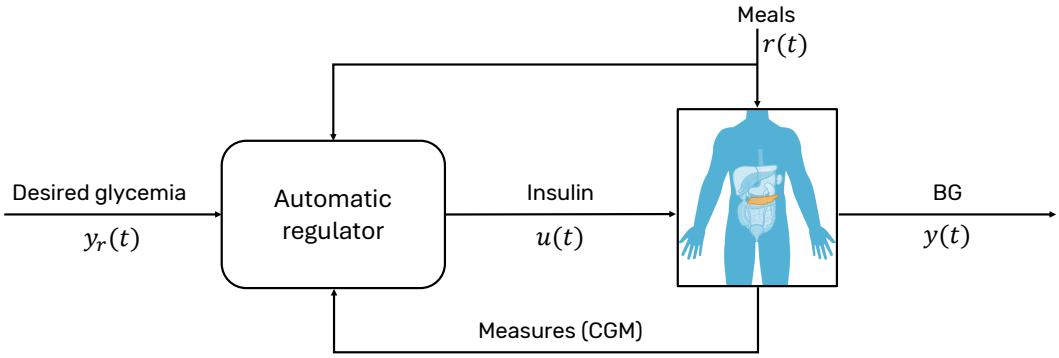


FIGURE 1.4: HCL schema of the AP: composed of the T1DM patient (i.e., the system), the controller that receives the values of the desired BG condition, the meal amount, and the measured BG from the sensor, and gives the insulin amount as output.

question of determining how to set the inputs (i.e., control actions) of a system in order to obtain a certain behavior of the outputs (i.e., the reference signals), respecting any physical or operational constraints.

To analyse and solve the control problem, control theory is used. It provides theoretical tools to model dynamic systems, to establish control objectives, and to design control algorithms to drive the system behavior. This is appliable in various fields, including industrial automation, robotics, economics and biological systems [52].

AP control problem In this case, the analysed control problem is related to the T1DM, and in particular, Figure 1.4 shows the control schema for the insulin-glucose management with the AP. In detail, the system is the T1DM patient, the control action is the insulin, the feedback is given by the CGM measurements, the output (i.e., the controlled variable) is the BG that needs to be maintained within the euglycemic zone (which is the desired reference behavior). The meals are considered as a disturbance of the system, which could be measured (since the patients have to enter the grams of carbohydrates they are going to eat) or not (if not announced), depending on the control type.

In this case, there is only one control action (i.e., insulin), which means that this has to be carefully computed, since this cannot be removed nor there is a counterpart that can be added to face it. This is particularly important in the case of an excessive amount of injected insulin, which drives the patient to hypoglycemia, which is very dangerous since it can lead to diabetic coma or even to death in the short term.

Control algorithms for AP

Among the most widespread control algorithms, there are the Fuzzy Logic (FL), the Proportional-Integrative-Derivative (PID), and Model Predictive Control (MPC) controllers that were used in the AP case [60, 89, 114, 116].

Fuzzy Logic (FL) The FL controllers take a set of inputs and apply conditional logic to obtain an output control, allowing the implementation of human expertise in controller design [122].

Proportional-Integrative-Derivative (PID) The PID controllers are widely used thanks to their simplicity and flexibility. It requires the tuning of the parameters, which can also be done using just empirical rules. Its functioning can be summarized as follows: the controller receives the input signals, it computes the proportional, integral, and derivative responses, and then the output (i.e., $u(t)$, the control action) is obtained by summing those three components, as in the following equation:

$$u(t) = K_p e(t) + K_i \int_{t_0}^{t} e(t)dt + K_d \frac{de(t)}{dt}, \tag{1.1}$$

where K_p is the proportional action coefficient, K_i is the integral action coefficient, and K_d the derivative action coefficient, and $e(t) = y_m(t) - y^0(t)$ is the error (that is, the input to the controller, where y_m is the current measure of the controlled variable and y^0 is the aimed set-point). The interpretation of the three action is that the proportional part is related to the amount of the error (i.e., higher the error, higher the control needed), the integral part is related to the mean value error (i.e., for polarization), and the derivative part is related to the error tendency (i.e., if the error is growing, more control is needed) [59].

Model Predictive Control (MPC) The MPC is an optimization-based control technique. It is made of three ingredients: (i) a dynamic model to make predictions about the future behavior of the system, evaluating possible inputs; (ii) the constraints that have to be satisfied; and (iii) the function that has to be minimized, representing the goal of the controller. At each sampling instant, the optimization problem is solved online, and the outcome is the sequence of the best control actions that have to be applied to the system to reach the goal at the end of the prediction horizon, according to the actual inputs and handling the constraints, which could be both on inputs and states. The first component of the sequence is applied to the

system, and the entire procedure is repeated at the next sampling instant, with the updated input (in a receding horizon fashion) [98]. Further details are provided in Chapter 2.

AP on the market

Among the AHCL systems currently on the market in Italy, the three used at Hospital Papa Giovanni XXIII (Bergamo, Italy) were compared to analyse the night-time effectiveness in reaching the glycemic targets, considering different control algorithms. In detail, MiniMed 780G is associated with CGM Guardian and SmartGuard algorithm (which is a PID), Tandem t:slimX2 is associated with CGM Dexcom G6 with Control IQ algorithm (i.e., MPC), and Accu-Check Insight is associated with CGM Dexcom G6 with Diabeloop Generation 1 (DBLG1) algorithm (i.e., MPC) [39]. The results of the study were published in [20]. This is a retrospective analysis, considering 55 patients (22 with the SmartGuard algorithm, 18 with the Control IQ one, and 15 with the DBLG1 one), during the night-time period (i.e., from 00:00 to 07:00) of 14 days, to compare the different systems under similar conditions (thus with fewer influence of external factors, such as physical activity or meals). As outcome, the glucose trends were similar in the three cases, all allowing for a good management of nocturnal BG levels.

1.3.4 Performance indexes

To represent the patient's behavior and to assess the quality of the insulin therapy, the first graph that can be done contains the BG values measured by the CGM, the amount of the insulin injections, and the meals. This allows one to see the overall situation, and it is also useful to check for glucose fluctuations. However, to better evaluate the performance of the T1DM management, some useful indexes can be used. Among the possibilities, there are [30, Chapter 6]:

- Mean glucose: average BG value (mg/dL).
- Standard Deviation (SD): standard deviation of BG values (mg/dL).
- Time In Range (TIR): percentage of time spent with BG in the euglycemic range (i.e., between 70 and 180 mg/dL).
- Time In Tight Range (TITR): percentage of time spent with BG between 70 and 140 mg/dL.

- Time Below Range (TBR): percentage of time in hypoglycemia, that could be divided into:
  - Time Below Range, level 1 (TBR_1): percentage of time spent between with BG 55 and 70 mg/dL;
  - Time Below Range, level 2 (TBR_2): percentage of time spent with BG below 55 mg/dL.
- Time Above Range (TAR): percentage of time in hyperglycemia, that could be divided into:
  - Time Above Range, level 1 (TAR_1): percentage of time spent with BG between 180 and 250 mg/dL;
  - Time Above Range, level 2 (TAR_2): percentage of time spent with BG above 250 mg/dL.
- Number of severe hypoglycemic events: number of times a patient registers BG values lower than 55 mg/dL.

According to the American Diabetes Association (ADA), the glycemic targets are: $< 5\%$ of time with BG in TAR_2, $< 25\%$ for TAR_1, $> 70\%$ for TIR, $< 4\%$ for TAR_1 and $< 1\%$ for TAR_2 [11].

There exist some other indexes that could help to graphically visualize the quality of the diabetes management.

Glycemia Risk Index (GRI) An additional useful tool can be the GRI, which is a quantitative measure designed to provide a comprehensive evaluation of an individual's susceptibility to hypoglycemia or hyperglycemia. The GRI was designed to estimate the clinicians' percentile rankings (with 0 being the best and 100 the worst) [65]. It is obtained from

$$\text{GRI} = \Big(3.0 \cdot \text{TBR}_2 + 2.4 \cdot \text{TBR}_1\Big) + \Big(1.6 \cdot \text{TAR}_2 + 0.8 \cdot \text{TAR}_1\Big), \tag{1.2}$$

where the first part is the hypoglycemic component and the second is the hyperglycemic one.

The GRI can be displayed graphically on a grid with the hypoglycemia component on the horizontal axis and the hyperglycemia component on the vertical axis. Diagonal lines divide the graph into five zones (quintiles) based on overall glycemia

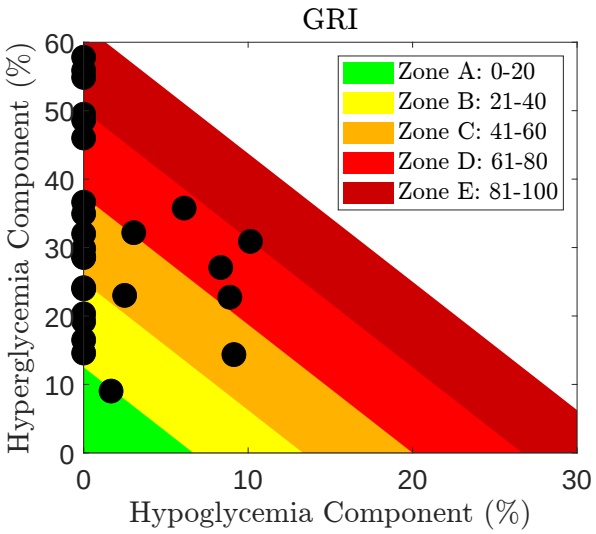
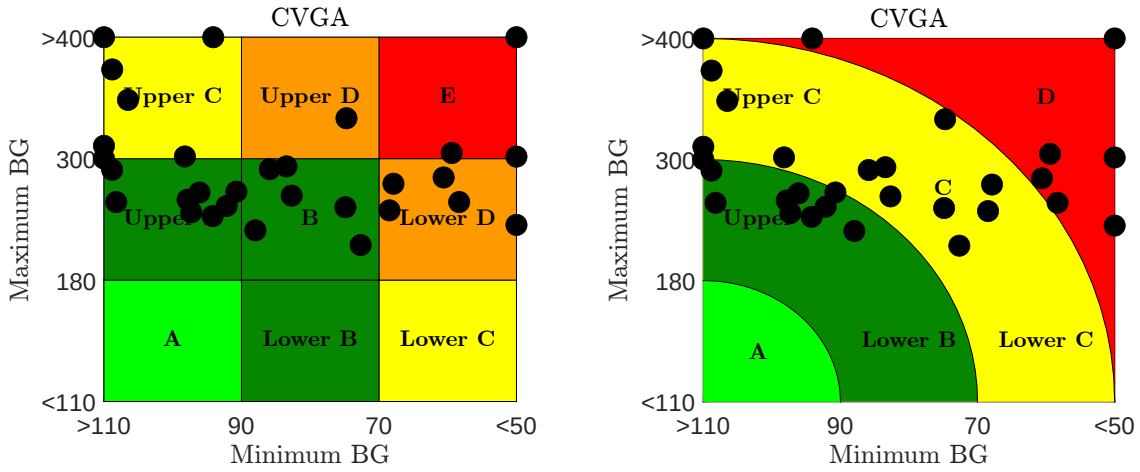


FIGURE 1.5: Glycemia Risk Index example. Each black dot represents a patient during a given period, on the y-axis there is the hyperglycemic risk component, and on the x-axis the hypoglycemic risk component. The goal is to have all the dots in the green A zone.

quality, from best ($1^{st} - 20^{th}$ percentile) to worst ($81^{st} - 100^{th}$ percentile). In Figure 1.5 an example is visible, where each point represents a specific subject. This grid also allows the tracking of sequential changes within an individual.

Control-Variability Grid Analysis (CVGA) Up to now, the average results are evaluated, but to have a more complete analysis, also the CVGA can be assessed. The CVGA is a graphical representation that provides both visual and numerical information about the quality of glycemic control, since it visualizes the "extreme" patient condition. The plane is partitioned into nine regions, detailed in Table 1.1, corresponding to different levels of BG control quality, from A (best) to E (worst). Then each subject is represented as a dot on the graph, where the x-coordinate is the minimum BG value and the y-coordinate is the maximum BG value. As recommended by [81], the lower bound of CVGA is set at 2.5% of the distribution of data and the upper bound at 97.5%, to avoid this index being sensitive to noise and to statistical outliers. In Figure 1.6 are reported two versions of the CVGA, according to the shape of the regions, but with the same meaning. Specifically, Figure 1.6(A) shows an example of the version just described, while Figure 1.6(B) shows four circular regions (detailed in Table 1.2), with the advantage that all patients suffering from severe hypo- or hyper-glycemic episodes fall in zone D [113].



(A) Rectangular version

(B) Rounded version

FIGURE 1.6: Control-Variability Grid Analysis examples. Each black dot represents a specific patient during a given period, showing the worst condition, since on the x-axis there is the minimum BG value, and on the y-axis the maximum BG value. There are two possible ways of dividing the plane, in rectangles in (A) and in circles in (B); but with the same goal: trying to stay inside the green zone A.

Zone	Description
A	Accurate control: X range 110–90 mg/dL and Y range 110–180 mg/dL
Lower B	Benign deviations into hypoglycemia: X = 90–70 mg/dL Y = 110–180 mg/dL
B	Benign control deviations: X = 90–70 mg/dL and Y = 180–300 mg/dL
Upper B	Benign deviations into hyperglycemia: X = 110–90 mg/dL, Y = 180–300 mg/dL
Lower C	Over-Correction of hyperglycemia: X < 70 mg/dL, Y = 110–180 mg/dL
Upper C	Over-Correction of hypoglycemia: X = 110–90 mg/dL, Y > 300 mg/dL
Lower D	Failure to Deal with hypoglycemia: X < 70 mg/dL, Y = 180–300 mg/dL
Upper D	Failure to Deal with hyperglycemia: X = 90–70 mg/dL, Y > 300 mg/dL
E	Erroneous control: X < 70 mg/dL and Y > 300 mg/dL

TABLE 1.1: Description of the nine zones of the CVGA (Figure 1.6(A))

Zone	Description
A	Accurate control
Lower B	Benign deviations into hypoglycemia
Upper B	Benign deviations into hyperglycemia
Lower C	Over-Correction of hyperglycemia
Upper C	Over-Correction of hypoglycemia
C	Over-Correction of both hypo- and hyper-glycemia
D	Failure to Deal with glycemia control

TABLE 1.2: Description of the four circle zones of the CVGA (Figure 1.6(B))

1.3.5 UVA/Padova simulator

Dealing with the collection of data and with the making of experiments on T1DM patients is very complicated, especially due to the danger of hypoglycemic events. This is why virtual patients and in-silico trials were developed: to facilitate testing and accelerate the regulatory processes of AP research. This allows glucose sensors, insulin delivery technologies, and control algorithms to be tested in more extreme conditions. Specifically, the UVA/Padova simulator was approved in 2008 by FDA as a substitute for animal trials and to perform preclinical studies. Then, in 2013, the new version was approved, i.e., Diabetes Mellitus Metabolic Simulator (DMMS) [115] (with the commercial version Diabetes Mellitus Metabolic Simulator for Research (DMMS.R)) [33]. The distributed software is available from The Epsilon Group (TEG).

The simulator consists of three parts: a database that contains metabolic characteristics of in-silico subjects (with a set of parameters used to describe their metabolic response to carbohydrates, insulin, drugs, and physical exercises), a mathematical simulation model that applies a set of input data to a selected subject (whose output is the estimation over time of the subjects' responses, including their BG and insulin levels), and a user interface to interact with the database and the simulation model. The design of this model is based on studies of glucose/insulin metabolism in humans [34].

The available DMMS.R contains 10 virtual patients (and an extra one representing their average values) for each of the categories, i.e., T1DM children, T1DM adolescents, T1DM adults, T2DM adults, and prediabetic adults. The characteristics of the virtual adult patients with T1DM are shown in Table 1.3, since they are the ones analysed in this thesis. Specifically, it reports their name, the basal fasting BG (Gb), Body Weight (BW), age, optimal daily basal insulin amount, Carbohydrate-to-Insulin Ratio (CR), and Correction Factor (CF).

Simulation settings

In order to obtain an adequate simulation, it is possible to define its settings, including the selection of the subject, the duration of the simulation, the meals (i.e., their time, carbohydrate amount, and duration), the physical activity (i.e., time, duration, and intensity of the exercise session), the presence of some medications (only

Name	Gb (mg/dL)	BW (kg)	Age (years)	Daily Basal Insulin (U)	CR (g/U)	CF (mg/dL/U)
#1	122.02	79.80	32	29.37	19.16	43.85
#2	123.44	80.34	22	32.37	22.48	42.10
#3	119.41	70.87	42	35.99	14.55	35.15
#4	119.39	67.48	24	22.82	19.70	53.20
#5	119.65	67.11	47	22.05	13.47	48.02
#6	115.57	73.01	23	45.65	8.99	25.89
#7	115.06	46.06	47	29.98	18.14	42.60
#8	130.02	98.66	56	25.40	8.79	37.06
#9	113.57	67.78	24	22.70	19.76	53.44
#10	113.31	80.79	31	29.97	13.77	39.50

TABLE 1.3: Characteristics of the virtual T1DM adult patients of the UVA/Padova simulator

Metformin is now available), and the activation of the insulin sensitivity circadian variability. Then, there is the possibility to set the pump, sensor, and controller to define the therapy.

In detail, for the virtual glucose sensor it is possible to choose from the available options and configure the desired characteristics. Among the possibilities, the ideal CGM sensor is defined to always provide the subcutaneous glucose concentration, without noise, lag, or any other error. The general configurable sensor allows for the selection of a pre-defined sensor from which to load the configuration values. The following characteristics can then be configured: range of output possible values, bias to add or multiply to the output value, time lag, sensor lifespan, the sample time, and the noise (Gaussian white noise or Autoregressive noise). The SMBG sensor provides the plasma glucose concentration, with Gaussian white noise applied to every sample. The magnitude of the noise is defined as the maximum permitted by one of the ISO specifications for SMBG (that is, an international standard that specifies the performance requirements for in vitro glucose monitoring systems used by individuals with diabetes for self-testing, for example, the ISO15197 2013). It also allows the integration of a customized sensor, implemented as a MATLAB or JavaScript plugin element.

As for the sensor, there is an ideal pump (operating without errors or delays), a general configurable one (defining the characteristic of the basal and bolus channels, including the frequency of administration, the smallest possible amount of insulin that can be injected, and the noise level) and the possibility of testing a pump with a plugin delivery element, developed in MATLAB or JavaScript.

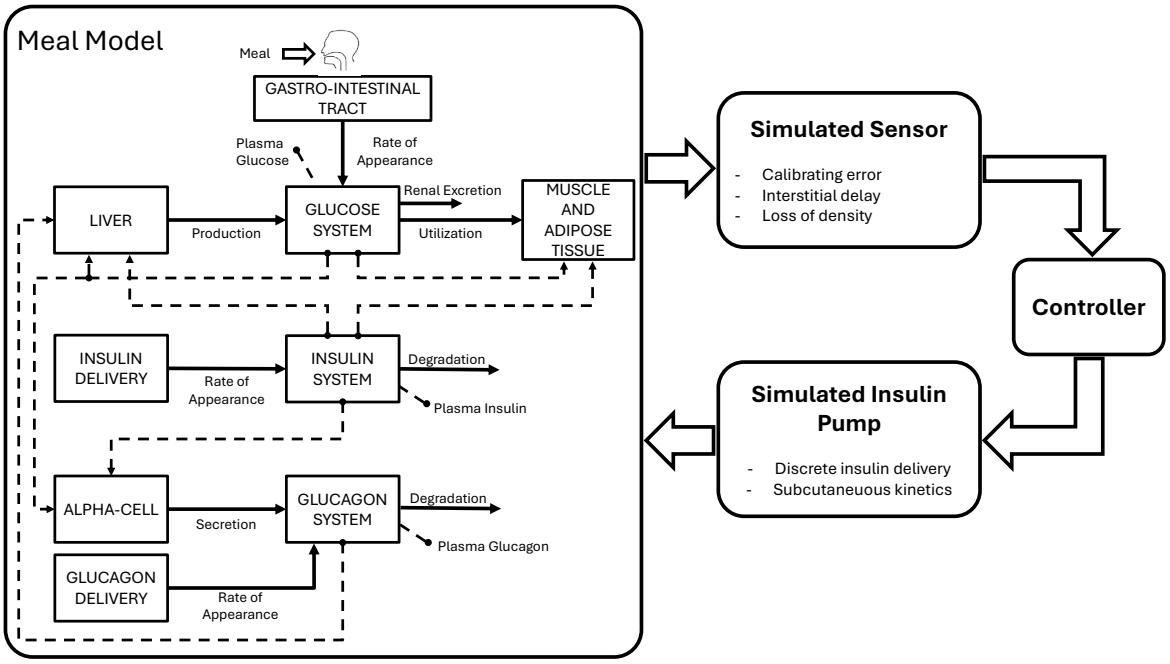


FIGURE 1.7: Scheme of the model included in the T1DM simulator

For the controller part, the basal insulin therapy could be delivered as a constant and continuous amount of fasting-active insulin or as a single (or more) boluses with long-acting insulin. Both amounts are based on the total daily insulin amount defined for each single subject, or it can be modified. The boluses can be injected at meal time (or with a delay of [-30,30] minutes) and the amount is defined by making g_{CHO}/CR. It is also possible to define a CR or use the standard one in the subject database, to modify a bolus multiplier (and also a modified one before an exercise session), to deliver directly to the plasma, to define the insulin type, and to decide if apply a random error to the meal carbohydrate estimate. There is also the possibility of defining correction boluses, as meal-time correction, post-prandial correction or in case of high BG alarm-based corrections, selecting the BG thresholds. As for the previous cases, also for the controller it is possible to evaluate algorithms made with a MATLAB or JavaScript plugin element.

Model

The glucose-insulin model is described in detail in [67], and its schematic representation is shown in Figure 1.7. The detailed model equations can be found in Appendix A.

Briefly, the model puts in relation plasma concentrations, that is, glucose G and insulin I, with glucose fluxes, that is, endogenous glucose production (EGP), glucose rate of appearance (Ra), glucose utilization (U), renal extraction (E), and insulin fluxes, that is, rate of insulin appearance from the subcutaneous tissue (SC) and insulin degradation (D). There are two subsystems:

- Glucose subsystem: it is made of a 2-compartmental model, the first one represents plasma and rapidly equilibrating tissues (thus, the insulin-independent utilization), and the second one represents slowly equilibrating tissues (thus, the insulin-dependent utilization). The glucose kinetics are described by the following equations:

$$\begin{cases} \dot{G}_p(t) = EGP(t) + Ra(t) - U_{ii}(t) - E(t) - k_1 \cdot G_p(t) + k_2 \cdot G_t(t) \\ G_p(0) = G_{pb} \\ \dot{G}_t(t) = -U_{id}(t) + k_1 \cdot G_p(t) - k_2 \cdot G_t(t) \\ G_t(0) = G_{tb} \\ G(t) = \frac{G_p}{V_G} \\ G(0) = G_b \end{cases} \tag{1.3}$$

  Where $G_p(t)$ is the glucose mass in plasma, $G_t(t)$ is the glucose mass in the tissues, the suffix b denotes basal state, U_{ii} and U_{id} are the insulin-independent and dependent glucose utilizations respectively, V_G is the distribution volume of glucose, and k_1, k_2 are the rate parameters.

- Insulin subsystem: it is made of 2 compartments as well, to represent liver and plasma. The insulin kinetics are described by the following equations:

$$\begin{cases} \dot{I}_p(t) = -(m_2 + m_4) \cdot I_p(t) + m_1 \cdot I_l(t) + R_{ai}(t) \\ I_p(0) = I_{pb} \\ \dot{I}_l(t) = -(m_1 + m_3) \cdot I_l(t) + m_2 \cdot I_p(t) \\ I_{pl}(0) = I_{lb} \\ I(t) = \frac{I_p(t)}{V_I} \end{cases} \tag{1.4}$$

  Where I_p is the insulin mass in the plasma, I_l is the insulin mass in the liver, S is the insulin secretion, V_I is the distribution volume of insulin, m_1, m_2, m_4 are rate parameters and $m_3 = \frac{HE(t) \cdot m_1}{1 - HE(t)}$, where HE is the hepatic extraction of insulin.

Subcutaneous insulin kinetics is represented by a 3-compartment model. Endogenous glucose production is assumed to be linearly dependent on plasma glucose concentration and a delayed insulin signal. Glucose rate of appearance is described with a model of glucose transit through the stomach and intestine, with the stomach represented by 2 compartments, while a single compartment is used to describe the gut; the rate constant of gastric emptying is a nonlinear function of the amount of carbohydrates in the stomach. Glucose utilization during a meal has 2 components: the insulin-independent utilization by the brain and the erythrocytes, which takes place in the first compartment and it is constant, while the insulin-dependent utilization takes place in the second compartment and depends nonlinearly on glucose in the tissues [33].

Given the mentioned devices, control algorithms and models, there are still open challenges to address, like taking into account the highly varying conditions inter- and intra-patients in the glucose-insulin dynamics or guaranteeing efficient a safe operation of the AP. Hence, this thesis contributes to this field, as is outlined in what follows.

1.4 Contributions and thesis outline

1.4.1 Contributions

As detailed above, the quality of the therapy and thus the performance of the T1DM management strongly depends on the model of the system and on the control algorithm used in the AP. The models used are usually general, and their quality is quite low considering the complexity of the disease. Moreover, the control strategy and the parameters of the devices are commonly set manually by the physicians. This is the main reason why, in this thesis, the data-driven learning methods are implemented. The idea is to develop models derived directly from the patients' data, which means obtaining highly personalized models. The goal is to try to mimic the AHCL on the market, automatically finding the optimal basal insulin amount to maintain the BG inside the euglycemic range.

To achieve this, the CHoKI learning method is exploited, which is based on an inference method, that basically learns the effect of the present and past input (i.e., insulin, meals, and BG) on the actual BG value. This technique is implemented in the MPC to make glucose forecasts, to be able to obtain the optimal insulin amount

to be injected into the patient, in order to reach the safe range and to remain inside it. The idea is to develop personalized control algorithms, trying to mimic and improve the AHCL systems on the market. Several control algorithms were tested on the virtual patients of the DMMS.R, with changes in the constraints' kind and shape, including the value of the insulin still active from previous boluses to avoid the risk of hypoglycemia, and testing them on more challenging scenarios, with the insulin sensitivity variability, and during physical activity sessions. Then, to take into consideration also the variations within the patient over time, an online updating of the CHoKI method is proposed, refreshing its dataset with meaningful new data. Another study was performed by using the CHoKI to learn the non-linearities that the linear model cannot catch.

1.4.2 Outline

The remainder of this thesis is organized as follows:

- **Chapter 2** details the theory of the MPC algorithms, focusing also on its data-driven version and on its application in the glucose management field. The Kinky Inference learning method is described, highlighting the CHoKI approach.
- **Chapter 3** presents the MPC CHoKI-based, and it also describes the results of applying the proposed controller to the in-silico patients of the UVA/Padova simulator. In this chapter, several versions of the CHoKI-based MPC are proposed and compared, making variations in constraints, simulation conditions, and variability. At the end of the chapter, a hybrid model is also provided to make BG predictions. It consists of the merge of a linear model with corrections made with the CHoKI.
- **Chapter 4** presents a stochastic version of the CHoKI-based MPC, with chance constraints on the BG, trying to obtain a less conservative controller. This proposal is also tested on the UVA/Padova simulator.
- **Chapter 5** presents an online version of the CHoKI, in which the dataset is updated online, adding only the data that is informative. The aim of this chapter is to try to deal with the T1DM patient variability (such as due to circadian rhythm of insulin sensitivity, stressful situations, physical activity, growth of

the subject, ...), trying to adapt the CHoKI method over time. This proposal is also tested on the UVA/Padova simulator.

- **Chapter 6** draws some conclusions on this thesis and describes some possible future developments.
- **Appendix A** contains the equations of the model in the UVA/Padova simulator, and **Appendix B** contains all the useful definitions that are not included in Chapter 3.

1.4.3 List of publications

Journal Papers:

1. B. Sonzogni, J. M. Manzano, M. Polver, F. Previdi, and A. Ferramosca. CHoKI-based MPC for blood glucose regulation in artificial Pancreas. *IFAC Journal of Systems and Control*, page 100294, 2025.
2. N. D. Borella, A. Ferramosca, G. Castagna, S. Ippolito, S. Ceresoli, A. Taverna, B. Sonzogni, R. Trevisan, and G. Lepore. Comparison of the night-time effectiveness in achieving glycemic targets in adults with type 1 diabetes of three advanced hybryd closed-loop systems. *Acta Diabetologica*, pages 1–9, 2024.

International Conference Papers:

1. B. Sonzogni, J. M. Manzano, M. Polver, F. Previdi, and A. Ferramosca. CHoKI-based MPC for blood glucose regulation in artificial Pancreas. *IFAC-PapersOnLine*, 56(2), 9672–9677, 2023.
2. B. Sonzogni, J. M. Manzano, M. Polver, F. Previdi, and A. Ferramosca. CHoKI-Based MPC for Blood Glucose Regulation in Artificial Pancreas with Probabilistic Constraints. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 1619–1624. IEEE, 2023.
3. B. Sonzogni, J. M. Manzano, F. Previdi, and A. Ferramosca. Insulin on Board safety constraint effect in a CHoKI-based MPC for Artificial Pancreas. *IFAC-PapersOnLine*, 58(24):257–262, 2024.

4. B. Sonzogni, J. M. Manzano, F. Previdi, and A. Ferramosca. Hybrid Modeling of the Insulin-Glucose System: Combining Linear and Data-Driven Models for Artificial Pancreas. *IFAC-PapersOnLine*, 59(2):109–114, 2025.
5. M. Polver, B. Sonzogni, M. Mazzoleni, F. Previdi, and A. Ferramosca. Artificial pancreas under a zone model predictive control based on gaussian process models: toward the personalization of the closed loop. *IFAC-PapersOnLine*, 56(2):9642–9647, 2023.
6. N. Licini, B. Sonzogni, P. Abuin, F. Previdi, A. H. González, and A. Ferramosca. Artificial Pancreas under stable pulsatile Model Predictive Control: including the Physical Activity effect. In *63rd IEEE conference on decision and control (CDC)*, 2024.
7. N. Licini, B. Sonzogni, P. Abuin, F. Previdi, A. H. González, and A. Ferramosca. A novel dynamic modeling of Insulin Sensitivity in the Blood Glucose Minimal Model. In *2025 American Control Conference (ACC)*, 2025.
8. N. Licini, B. Sonzogni, P. Abuin, F. Previdi, A. H. González, and A. Ferramosca. Optimized carbohydrate suggestions and insulin dosing in Artificial Pancreas: dual-action Pulsatile Zone MPC with non-standard IOB constraints for physical activity management. In *European Control Conference (ECC)*, 2025.

Other Papers:

This is the list of other papers that are not strictly connected with the main topic of this thesis:

1. F. Cadamuro, M. Piazzoni, E. Gamba, B. Sonzongni, F. Previdi, F. Nicotra, A. Ferramosca, and L. Russo. Artificial Intelligence tool for prediction of ECM mimics hydrogel formulations via click chemistry. *Biomaterials Advances*, page 214323, 2025.
2. P. A. Bonaffini, F. Stanco, L. Dulcetta, G. Poli, P. Brambilla, P. Marra, C. Valle, F. L. Lorini, M. Mazzoleni, B. Sonzogni, et al. Chest X-ray at Emergency Admission and Potential Association with Barotrauma in Mechanically Ventilated Patients: Experience from the Italian Core of the First Pandemic Peak. *Tomography*, 9(6):2211–2221, 2023.

CHAPTER 2

Model Predictive Control and CHoKI learning method

In this chapter, the theory of the techniques utilized to achieve the goal mentioned in the previous chapter is detailed. In particular, starting from the Model Predictive Control (MPC) algorithm, also showing its data-driven version. Then, among the possible data-driven learning methods, the Componentwise Hölder Kinky Inference (CHoKI) is presented.

2.1 Model Predictive Control

MPC is a control method that uses a dynamic model to forecast the behavior of a system and optimize the forecast to produce the best control move at the current time [98]. There are three main ingredients: a model of the system, a cost function that has to be minimised (used to represent, for example, energy consumption, response time, or deviation from a setpoint), and constraints on the inputs (such as limits on system power), on the states (such as safety, physical or operating limitations), and on the outputs (such as desired range). These are visible in the generic MPC formulation in (2.3). Specifically, MPC resolves an optimization problem at each time step: it uses the model of the system to forecast the future evolution along a finite horizon, it computes an optimal sequence of control actions, then applies to the system only the first value of this sequence, and, at the next sampling instant, the current state is measured and the procedure is repeated (i.e., with a receding horizon strategy, defined in Definition 2.1). Among the advantages, there are the

good performances also in the presence of noise or uncertainties, flexibility in the objective formulation, the possibility of dealing with constraints and of handling multivariable control problems. On the other hand, the disadvantages might be the need for an accurate model and the computation complexity (especially for cases with nonlinear systems or long horizons) [98].

Definition 2.1: Receding Horizon Principle

At any time k, solve the optimal control problem over the prediction horizon $[k, k+N_p]$ and apply only the first input $u^*(k)$ of the optimal sequence $\mathbf{u}^*(k)$. At time $k+1$, move the prediction window one step ahead, measure the new state and repeat the optimization over the prediction horizon $[k+1, k+N_p+1]$.

To show the generic formulation of a nominal, discrete MPC, first a nonlinear time-invariant discrete-time system is considered, described by the following equation

$$x(k+1) = f\big(x(k), u(k)\big), \tag{2.1}$$

where $x \in \mathbb{R}^{n_x}$ represents the state (which is assumed to be known and measurable), and $u \in \mathbb{R}^{n_u}$ represents the input. The state and input of the system can be subject to some constraints, such that $\forall k \geq 0$, the following conditions have to be satisfied

$$x(k) \in \mathcal{X} \subset \mathbb{R}^{n_x}, \quad u(k) \in \mathcal{U} \subset \mathbb{R}^{n_u}. \tag{2.2}$$

The aim in regulation problems is to drive the system state to an equilibrium, $x_s = f(x_s, u_s)$.

Then, (2.3) shows the generic formulation of the optimization problem $P_{N_p}(x(k))$ that has to be solved at each time instant $k \geq 0$.

$$\begin{aligned}
\min_{\mathbf{u}(k)} \quad & V_{N_p}(x(k), \mathbf{u}(k)) := \sum_{j=0}^{N_p-1} \ell\left(\hat{x}(j|k), u(j|k)\right) + V_f\left(\hat{x}(N_p|k)\right) && \text{(2.3a)}\\
\text{subject to} \quad & \hat{x}(0|k) = x(k), && \text{(2.3b)}\\
& \hat{x}(j|k) = f\big(\hat{x}(j-1|k), u(j-1|k)\big), \ \ \forall j \in \mathbb{I}_1^{N_p}, && \text{(2.3c)}\\
& \hat{x}(j|k) \in \mathcal{X}, \ \ \forall j \in \mathbb{I}_0^{N_p}, && \text{(2.3d)}\\
& u(j|k) \in \mathcal{U}, \ \ \forall j \in \mathbb{I}_0^{N_p-1}, && \text{(2.3e)}\\
& \hat{x}(N_p|k) \in \mathcal{X}_f. && \text{(2.3f)}
\end{aligned}$$

Where, in detail:

- $\mathbf{u}(k) := \{u(0|k), ..., u(N_p - 1|k)\}$ is the control sequence, which contains the variables that have to be optimized;
- $N_p \in \mathbb{N} \setminus 0$ is the prediction horizon (i.e., represents how many steps ahead the model predicts the system's future behavior);
- the cost functional $V_{N_p}(\cdot)$ in (2.3a) represents the control goal and it is made of two components: the stage cost $\ell(\cdot)$ and the terminal cost $V_f(\cdot)$;
- (2.3b) imposes the initial condition $\hat{x}(0|k)$ to be equal to the current state $x(k)$;
- the state predictions need to be consistent with the dynamics of the system, so the model (2.3c) is considered;
- the constraints on the states (i.e., (2.3d)) and the ones on the control action (i.e., (2.3e)) must be met;
- (2.3f) is the stability-related terminal constraint, used to reach the desired condition (that is, reaching the set $\mathcal{X}_f$) at the end of the prediction horizon.

The optimal control sequence resulting from the solution of $P_{N_p}(x(k))$ is denoted as $\mathbf{u}^*(x(k))$ (or $\mathbf{u}^*(k)$), where $\mathbf{u}^*(k) = \{u^*(0|k), \ldots, u^*(N_p-1|k)\}$. The optimal value of the cost function given by $\mathbf{u}^*(k)$ is $V^*_{N_p}(x(k)) := V_{N_p}(x(k), \mathbf{u}^*(k))$. The control law implied by this MPC scheme is defined as $\kappa_{N_p}(x) := u^*(0|k)$ [52].

2.1.1 Stability

The closed-loop stability of the system under a generic MPC strategy must be verified in the MPC design phase. First, the recursive feasibility property has to be guaranteed by design. The *recursive feasibility* property implies that if the optimization problem is feasible (i.e., when there is at least one solution that satisfies all the constraints imposed on the system over the prediction horizon) at one time step, it will remain feasible at all subsequent time steps, provided that the system's state evolves according to the system dynamics and constraints [98]. Which means that, starting from an initial condition $x(0) \in \mathcal{X}$ that makes $P_{N_p}(x(0))$ feasible, $P_{N_p}(x(k))$ remains feasible $\forall k > 0$. This ensures that the controller can operate continuously without any infeasibility issues, while satisfying the constraints [91].

The Lyapunov stability theory is the usual tool used to derive asymptotic stability results in the MPC context. Specifically, the optimal value of the objective function

must be a Lyapunov function. To reach this goal, terminal ingredients (terminal cost and/or terminal constraints) can be added to the optimization problem [98].

Lyapunov stability The optimal cost $V_{N_p}^*(\cdot)$ needs to be a Lyapunov function, which means that it has to satisfy the conditions in Definition 2.2.

Definition 2.2: Lyapunov function

A function $V : \mathbb{R}^{n_x} \to \mathbb{R}_{\geq 0}$ is said to be a Lyapunov function for the system $x^+ = f(x)$ and set Ω if there exist $\mathcal{K}_\infty$-functions α_1 and α_2, and a $\mathcal{PD}$ function α_3, such that for any $x \in \mathbb{R}^{n_x}$,

$$\begin{aligned} V(x) &\geq \alpha_1(|x|_\Omega), && \text{(2.4a)} \\ V(x) &\leq \alpha_2(|x|_\Omega), && \text{(2.4b)} \\ V(f(x)) - V(x) &\leq -\alpha_3(|x|_\Omega), && \text{(2.4c)} \end{aligned}$$

The support $\mathcal{K}$, $\mathcal{K}_\infty$ and $\mathcal{PD}$ (positive definite) functions are defined in Appendix B. For the MPC case, this means that there exist two $\mathcal{K}_\infty$ (or $\mathcal{K}$) functions $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ and a continuous $\mathcal{PD}$ function $\alpha_3(\cdot)$ such that, for any state $x \in \mathcal{X}$ that makes $P_{N_p}(x)$ feasible, verify the following conditions:

$$\begin{aligned} V_{N_p}^*(x) &\geq \alpha_1(\|x\|) && \text{(2.5a)} \\ V_{N_p}^*(x) &\leq \alpha_2(\|x\|) && \text{(2.5b)} \\ V_{N_p}^*\left(f(x, \kappa_{N_p}(x))\right) - V_{N_p}^*(x) &\leq -\alpha_3(\|x\|). && \text{(2.5c)} \end{aligned}$$

To ensure that $V_{N_p}^*(\cdot)$ is a Lyapunov function, the next assumptions are needed.

Assumption 2.1

The terminal constraint set $\mathcal{X}_f$ is compact and contains the origin. Moreover, there exists a locally stabilizing control law $\kappa_f(\cdot)$ such that $\kappa_f(\cdot) \in \mathcal{U} \ \ \forall x \in \mathcal{X}_f$ and

$$f\left(x, \kappa_f(x)\right) \in \mathcal{X}_f,$$

$$V_f\left(f(x, \kappa_f(x))\right) - V_f(x) \leq -\ell\left(x, \kappa_f(x)\right),$$

Assumption 2.2

There exist $\mathcal{K}_\infty$ functions $\alpha_\ell(\cdot)$ and $\alpha_f(\cdot)$ that verify the following conditions:

$$\ell(x,u) \geq \alpha_\ell(\|x\|), \forall (x,u) \in \mathcal{X} \times \mathcal{U},$$

$$V_f(x) \leq \alpha_f(\|x\|), \forall x \in \mathcal{X}_f.$$

Then, to ensure stability, different formulations of the terminal ingredients in the MPC can be implemented, according to the application [52]:

- MPC with terminal equality constraint: $x(N_p) = x_s^*$, where $x(N_p)$ is the state at the end of the prediction horizon and x_s^* is the desired steady state. This means to have $\mathcal{X}_f = x_s^*$, $V_f(x) = 0$ [52, Theorem 5.5].
- MPC with terminal cost: adding a new cost component (i.e., the terminal cost V_f) to penalize the state at the end of the prediction horizon [52, Theorem 5.13].
- MPC with inequality terminal constraint: replacing the terminal equality constraint with a set $\mathcal{X}_f$, such that $x(N_p) \in \mathcal{X}_f$, obtaining a larger domain of attraction and fewer numerical problems.
- MPC with terminal cost and constraint: adding a terminal cost to the cost function V_f and using an inequality terminal constraint with $\mathcal{X}_f$.

2.1.2 MPC for tracking

In many applications, certain conditions may change (such as variations in the cost of energy, the availability of resources, or the production goal) and thus a new target setpoint y_t needs to be provided to the MPC, which may differ from the previous optimal one. As a consequence, the MPC schemes provided before are no longer able to guarantee stability and recursive feasibility [42, 68].

There are many possible solutions to deal with this problem of changing setpoints, such as dual-model controllers [27], MPC algorithms that guarantee local stability and asymptotic tracking of constant references [82], or the use of reference governors [12, 44]. Then, a new MPC for tracking formulation is proposed in [73], which allows to track any admissible target steady state in an admissible evolution. Its two main new ingredients are: (i) the artificial reference $y_s(k)$ that is considered as an optimization variable, and (ii) the cost function used to penalise the difference

between the artificial reference and the desired target y_t, i.e., the offset cost function $V_O(\cdot) = y_s(k) - y_t$. Specifically, the additional artificial reference y_s, which is computed at each sampling time k, is used to ensure that the admissible evolution of the system converges to the desired reference. This controller drives the system to any admissible target steady state, and if this is not admissible, the system is steered to the closest admissible steady state.

Assumption 2.3

The output function $h(\cdot,\cdot)$ is such that, for any given y_s, there exists a unique equilibrium point (x_s, u_s) such that $x_s = f(x_s, u_s)$ and $y_s = h(x_s, u_s)$. Furthermore, there exist a locally Lipschitz function $g_x : \mathcal{Y}_s \to \mathbb{R}^{n_x}$ and a continuous function $g_u : \mathcal{Y}_s \to \mathbb{R}^{n_u}$ such that

$$x_s = g_s(y_s), \quad u_s = g_u(y_s). \tag{2.6}$$

A generic formulation of the optimal control problem $P_{N_p}(x(k), y_t)$ is the following:

$$\begin{aligned}
\min_{(\mathbf{u}(k), y_s(k))} \quad & V_{N_p}(x(k), y_t, \mathbf{u}(k), y_s(k)) && \text{(2.7a)}\\
\text{subject to} \quad & \hat{x}(0|k) = x(k), && \text{(2.7b)}\\
& \hat{x}(j|k) = f(\hat{x}(j-1|k), u(j-1|k)), \ \forall j \in [1, N_p], && \text{(2.7c)}\\
& (\hat{x}(j|k), u(j|k)) \in \mathcal{Z}, \ \forall j \in [0, N_p - 1], && \text{(2.7d)}\\
& x_s(k) = g_x(y_s(k)), \ u_s(k) = g_u(y_s(k)), && \text{(2.7e)}\\
& (\hat{x}(N_p|k), y_s(k)) \in \mathcal{T}. && \text{(2.7f)}
\end{aligned}$$

Where

- $\mathbf{u}(k) \coloneqq \{u(0|k), \ldots, u(N_p - 1|k)\}$ is the control sequence to be optimized, together with the artificial reference $y_s(k)$;
- $N_p \in \mathbb{N} \setminus \{0\}$ is the prediction horizon;
- the set $\mathcal{Z}$ is defined as $\mathcal{Z} \coloneqq \mathcal{X} \times \mathcal{U} \subset \mathbb{R}^{n_x} \times \mathbb{R}^{n_u}$;
- the cost function in (2.7a) is defined as

$$V_{N_p}(x(k), y_t, \mathbf{u}(k), y_s(k)) \coloneqq \sum_{j=0}^{N_p-1} \ell(\hat{x}(j|k) - x_s(k), u(j|k) - u_s(k)) +$$

$$V_f\left(\hat{x}(N_p|k) - x_s(k), y_s(k)\right) + V_O\left(y_s(k) - y_t\right), \quad (2.8)$$

that represents the control goal, and it is composed of the stage cost function $\ell(\cdot,\cdot)$, the terminal cost function $V_f(\cdot)$, and the offset cost function $V_O(\cdot)$;

- (2.7b) imposes the initial condition $\hat{x}(0|k)$ to be equal to the current state $x(k)$;
- (2.7c) imposes the predictions $\hat{x}(j|k)$ to be consistent with the system dynamics, defined by the function $f(\cdot,\cdot)$;
- (2.7d) imposes the state predictions $\hat{x}(j|k)$ and the chosen control actions $u(j|k)$ to verify the state and input constraints;
- (2.7e) explicates the relationship between the value taken by the artificial reference $y_s(k)$ and the corresponding equilibrium point $(x_s(k), u_s(k))$;
- (2.7f) is the terminal constraint for tracking to be imposed on the terminal state $\hat{x}(N_p|k)$ and the artificial reference $y_s(k)$ for stability reasons.

The optimal control sequence resulting from the solution of $P_{N_p}(x(k), y_t)$ is hereafter denoted as $(\mathbf{u}^0(x(k), y_t), y_s^0(x(k), y_t))$ (or $(\mathbf{u}^0(k), y_s^0(k))$), where

$$\mathbf{u}^0(k) = \left\{u^0(0|k), \ldots, u^0(N_c - 1|k)\right\},$$

while the optimal value of the cost function given by $(\mathbf{u}^0(k), y_s^0(k))$ is $V_{N_p}^0(x(k), y_t)$ $:= V_{N_p}(x(k), y_t, \mathbf{u}^0(k), y_s^0(k))$. The control law implied by this MPC scheme is defined as $\kappa_{N_p}(x(k), y_t) := u^0(0|k)$. Note that here the superscript 0 is used to represent the optimal values obtained by solving the optimization problem, while in the previous section it was denoted with $*$. This is because in tracking MPC schemes with artificial references, there is a difference between the best admissible steady output y_s^* and the artificial steady output y_s^0 that is selected at each instant k by the MPC.

As in the MPC for regulation detailed in the previous section, also in the case of the MPC for tracking changing setpoints, there exist different schemes to ensure recursive feasibility and stability of the best admissible equilibrium, whose properties are proved in [75]:

- MPC with terminal inequality constraint (with the terminal constraint set $\mathcal{T}$, i.e., $(\hat{x}(N_p|k), y_s(k)) \in \mathcal{T}$) and terminal cost function V_f. Here $\mathcal{T}$ needs to be an invariant set for tracking, as considered in Definition 2.3.

- MPC with terminal equality constraint, $\mathcal{T} = \{g_x(y_s)\} \times \mathcal{Y}_t$ and $V_f(x - x_s, y_s) = 0$.
- MPC with terminal cost function V_f and without terminal constraint, $\mathcal{T} = \mathcal{X} \times \mathcal{Y}_t$.

The purpose of this scheme is to make the artificial reference to converge to the desired setpoint y_s^*, such that the closed-loop system can also converge to it.

Definition 2.3: Invariant set for tracking

Given a constraint set $\mathcal{Z}$, a set of feasible setpoints $\mathcal{Y}_t \subseteq \mathcal{Y}_s$ and a local control law $\kappa_f(x, y_s)$, a set $\mathcal{T} \subset \mathbb{R}^n \times \mathbb{R}^p$ is an admissible invariant set for tracking for system

$$x(k+1) = f(x(k), u(k)), \;\; y(k) = h(x(k), u(k)),$$

if, for all $(x, y_s) \in \mathcal{T}$, the local control law $\kappa_f(x, y_s)$ is feasible and $\left(f\left(x, \kappa_f(x, y_s)\right), y_s\right) \in \mathcal{T}$.

2.1.3 Robust MPC

Up to now, the MPC for regulation and the one for tracking changing setpoints are proposed, which were both designed by considering a nominal nonlinear time-invariant discrete-time system $x(k+1) = f(x(k), u(k))$. However, in practice, there are model mismatches, and external disturbances (i.e., perturbations) or measuring noises, resulting in a discrete-time system that can be described by the equation

$$x(k+1) = f\left(x(k), u(k)\right) + w_d(k), \tag{2.9}$$

where $x \in \mathbb{R}^{n_x}$ is the system state, $u \in \mathbb{R}^{n_u}$ is the system input, and $w_d \in \mathbb{R}^{n_x}$ is the perturbation action on the system. The disturbance w_d in constrained optimal control problems is usually assumed to be bounded since it is impossible to ensure that a system with unbounded disturbances satisfies the usual state and control constraints. Thus, it is assumed that w_d satisfies the bound $w_d \in \mathcal{W}_d$, where $\mathcal{W}_d$ is a compact subset of $\mathbb{R}^{n_x}$ containing the origin [98].

The system state and input are subject to hard constraints

$$x(k) \in \mathcal{X} \subset \mathbb{R}^{n_x}, \quad u(k) \in \mathcal{U} \subset \mathbb{R}^{n_u}, \quad \forall k \geq 0. \tag{2.10}$$

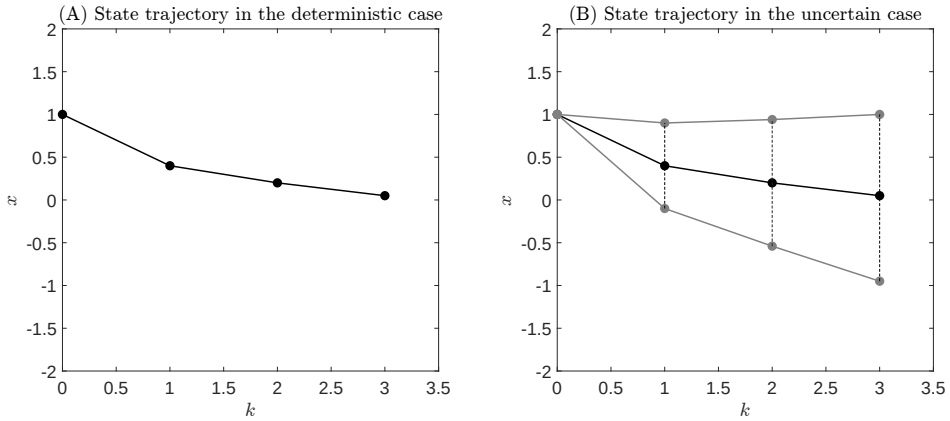


FIGURE 2.1: (A) shows the deterministic state trajectory, while (B) represents the state tube trajectory in the case with uncertainty.

Robust controllers compute the predicted behavior based on the nominal system $z(k) = g(z(k), u(k))$, which is not identical to real behavior, considering the error $e(k) = x(k) - z(k)$. To control these noisy signals the use of robust control strategies is required. A control system is robust when stability is maintained and the performance specifications are met for a specified range of model variations and a class of noise signals [14]. Therefore, it is fundamental to consider uncertainty in the stability analysis. A possible way to do it is to consider the Input-to-State Stability (ISS) theory, which offers a method for making a robust analysis of noisy systems. In fact, an ISS system implies that bounded uncertainties have a bounded effect on the evolution of the system [96, 98, 123].

Within the class of Robust MPC methods, two of the most studied and applied approaches are those based on min-max optimization, as in [3, 13, 61, 69, 76], and on the tube-based control, as in [90, 92, 121].

Tube-based MPC

The tube-based MPC is a variation designed to deal with the uncertainties of the models (such as modeling errors or disturbances), maintaining the stability and system performance [74].

Figure 2.1 shows the difference between the nominal deterministic state trajectory and the tube trajectory of the system behavior in the presence of disturbances. With suitable tube choice, satisfaction of state and control constraints may be guaranteed for each realization of the disturbance sequence [98].

The basic concept of tube-based robust MPC is to combine a standard MPC problem, to compute the desired state and control trajectory, with a feedback control, that is used to keep the real state within an invariant tube around the nominal trajectory, which represents the center of the tubes, and it is computed by solving the MPC problem. This means having two separate phases: an offline evaluation of the constraints, to ensure that the uncertain future paths lie in these tubes, and an online standard MPC applied to the nominal trajectories. The disturbances acting on the system are unknown and thus cannot be included in the prediction model. However, through this method, it is possible to approximate the maximum effect that they have on the predicted output trajectory. Based on this maximum effect, the control problem can be modified by ensuring that the constraints are satisfied for all the possible disturbance sequences [31, 43, 77, 85, 90, 92, 121].

2.1.4 Data-driven MPC

As said in Chapter 1, the controllers are usually based on a model of the system, being one of the main ingredients in (2.3). In recent years, many studies have focused on considering learning methods to substitute the model, especially in cases in which the model is difficult to identify or it could be too complex to be implemented on a controller [85]. Therefore, rather than relying on physical or phenomenological models, the controller learns the system's behavior directly from historical data or controlled experiments. This leads to the concept of the data-driven MPC, whose structure of the optimization problem remains that of the MPC described in the previous sections.

In the literature, there has been growing interest in the development of data-driven MPC, as reviewed in [54], a set of approaches that use data collected from the system to learn dynamic models [26, 86, 100]. Another possibility is to avoid the explicit identification of a physical model, as in [16, 17], in which the input-output data from the system are exploited to directly design the controller. The learning-based MPC [7] also allows for online updates to the model to improve performance, as in [6, 21].

Data-based models

There are several possible data-driven methods that could be exploited to make predictions inside an MPC structure. Among the many, system identification techniques can be exploited to obtain a model from data, such as Autoregressive eXogenous (ARX). More recent approaches exploit nonlinear models learned from data with machine learning techniques, such as NARX, Kinky Inference (KI), Gaussian Processes or Neural Network (NN) [124].

NARX models The NARX models are shortly detailed here since they are needed to better understand the learning method presented in Section 2.2.

Consider a general nonlinear, discrete time, output-feedback, Multiple Inputs Multiple Outputs (MIMO) system, where $y(k) \in \mathbb{R}^{n_y}$ is the vector of outputs, $x(k) \in \mathbb{R}^{n_x}$ is the state, $u(k) \in \mathbb{R}^{n_u}$ is the manipulable inputs, and time step $k \in \mathbb{N}_0$. Assuming that only input-output samples are available, a regressive form of the state must be constructed to describe the system. Nonlinear Autoregressive eXogenous (NARX) model can be used. First, the model is said to be autoregressive since $y(k+1)$ depends on previous outputs up to memory horizon n_a, and since it depends on the external previous inputs $u(k-1)$ up to $u(k-n_b)$ is said to be exogenous.

$$y(k+1) = f\big(y(k), ..., y(k-n_a), u(k), ..., u(k-n_b)\big). \tag{2.11}$$

where $n_a \in \mathbb{N}_0$ represents the memory horizon for the outputs, and $n_b \in \mathbb{N}_0$ the memory horizon for the inputs. Under certain observability conditions, previous outputs and inputs can reconstruct the state

$$x(k) = \big(y(k), ..., y(k-n_a), u(k-1), ..., u(k-n_b)\big), \tag{2.12}$$

such that it is possible to write

$$y(k+1) = f\big(x(k), u(k)\big). \tag{2.13}$$

It is possible to rewrite the NARX model in (2.11) as

$$y(k+1) = f(w(k)), \tag{2.14}$$

by defining the regressor $w \in \mathbb{R}^{n_w}$ as

$$w(k) = \big(x(k), u(k)\big) = \big(y(k), ..., y(k-n_a), u(k-1), ..., u(k-n_b), u(k)\big). \tag{2.15}$$

2.2 Componentwise Hölder Kinky Inference

As explained in the previous section, MPC requires the knowledge of a model that describes the system to forecast its behavior, and this model could be replaced with a data-driven learning method. A recent technique that can be used for this purpose is the CHoKI, which arises from the class of Lipschitz interpolation techniques.

2.2.1 Introduction

The term Kinky Inference (KI) was introduced by Jan-Peter Calliess in [24], referring to the sudden changes (i.e., the *kinks*) in the first derivative of the predictions. KI is a class of learning approaches evolving from Lipschitz interpolation, which is a technique based on the Lipschitz continuity of the ground truth function, defined as follows.

Definition 2.4: Lipschitz continuity

A function $f : \mathcal{W} \to \mathcal{Y}$ is Lipschitz continuous if there exist a real constant $L_f \geq 0$ such that for all $w_1, w_2 \in \mathcal{W}$,

$$\|f(w_1) - f(w_2)\| \leq L_f \|w_1 - w_2\|. \tag{2.16}$$

There exists an extension of the Lipschitz continuity, named Hölder continuity, in which the function has to satisfy the following more generalized condition:

Definition 2.5: Hölder continuity

A function $f : \mathcal{W} \to \mathcal{Y}$ is Hölder continuous if there exist two real constants $L \geq 0$ and $0 < p \leq 1$ such that for all $w_1, w_2 \in \mathcal{W}$,

$$\|f(w_1) - f(w_2)\| \leq L \|w_1 - w_2\|^p, \tag{2.17}$$

where L represents the smallest Lipschitz constant and p is called the Hölder exponent, $\mathcal{W} \subseteq \mathbb{R}^{n_w}$ is the input space and $\mathcal{Y} \subseteq \mathbb{R}^{n_y}$ is the output space.

In the case of $p = 1$, it means to have Lipschitz continuity [86].

The standard KI method estimates the predictor as follows:

Definition 2.6: Standard KI predictor

Given a dataset of $N_{\mathcal{D}}$ input-output observations, $\mathcal{D} = \left\{(\tilde{y}_i, w_i), i \in \mathbb{I}_1^{N_{\mathcal{D}}}\right\}$, and certain Hölder parameters grouped in $\theta_{KI} = (L, p)$, the prediction of an unseen query point q is computed as

$$\hat{\mathfrak{f}}\,(q; \theta_{KI}, \mathcal{D}) = \frac{1}{2} \min_{i=1,\dots,N_{\mathcal{D}}} \left(\tilde{y}_i + L\|w_i - q\|^p\right) + \frac{1}{2} \max_{i=1,\dots,N_{\mathcal{D}}} \left(\tilde{y}_i - L\|w_i - q\|^p\right). \tag{2.18}$$

This consists of the computation of the Euclidean norm of the difference between two known regressors (i.e., q and w_i, build as in (2.15), with $\mathcal{D} = (\tilde{y}_i, w_i)$) at the power of the Hölder exponent p and multiplied by the Lipschitz constant L, which are then once summed and once subtracted from the output $\tilde{y}_i$, which corresponds to the regressor w_i. Then, the minimum of the sum and the maximum of the difference are considered to obtain the upper and lower bounds of the function, respectively. The average of the two values is computed to get the output prediction $\hat{\mathfrak{f}}$ corresponding to the query regressor q. A visual representation of the KI predictor is visible in Figure 2.2. Where the real function $f(\cdot)$ that has to be estimated is the yellow line, and the red dots represent the collected dataset $\mathcal{D}$. The purple and green dotted lines represent the upper and lower bounds, respectively, and their average allows to obtain the KI predictions $\hat{f}(\cdot)$, which is the blue dotted line.

Lazily Adapted Constant Kinky Inference (LACKI) Looking at KI predictions in (2.18), the value of the Lipschitz constant L can be known a priori or obtained by exploiting the available data in $\mathcal{D} = \left\{(\tilde{y}_i, w_i),\ i \in \mathbb{I}_1^{N_{\mathcal{D}}}\right\}$. One possible method to estimate this value is by means of the LACKI [25]. It yields the minimum L that is consistent with the observed dataset. This constant is obtained by making the following computation:

$$L = \max\left(\frac{\left\|\tilde{y}_i - \tilde{y}_j\right\| - \eta}{\left\|w_i - w_j\right\|},\ \forall i \neq j \in \mathbb{I}_1^{N_{\mathcal{D}}}\right) \tag{2.19}$$

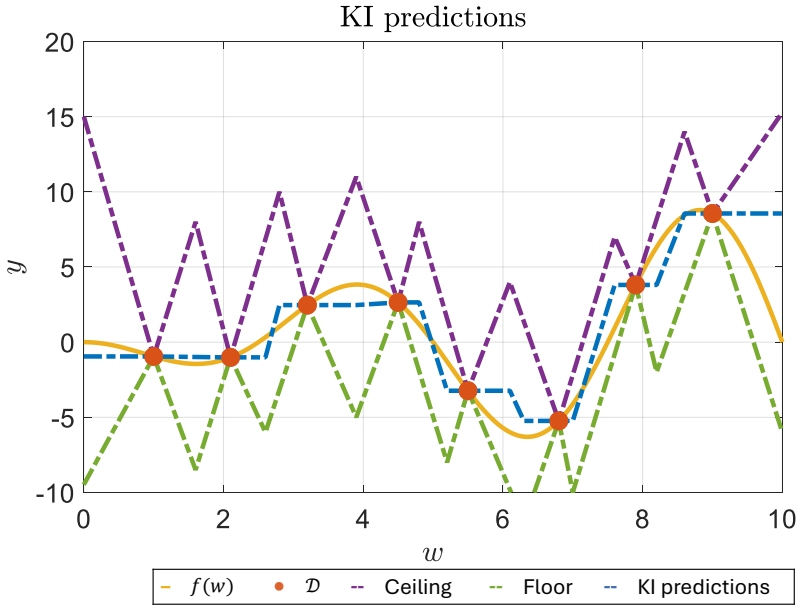


FIGURE 2.2: KI predictions. In yellow is represented the real function f from which a dataset is taken (i.e., the orange dots, $\mathcal{D}$). In purple and green are visible the upper and lower bounds, respectively, and the KI predictions $\hat{f}$ are shown as the blue dotted line.

where η is the regularization parameter used to smooth out the noise (i.e., avoiding overfitting). It is then required to have $\eta \geq 2\bar{\epsilon}$, where $\bar{\epsilon}$ is the noise level, because the bound on the uncertainty is a key value to design safe and robust predictive controllers.

KI properties KI has several properties that make it suitable for deriving robust predictive controllers. The first one is the learning simplicity, since the tuning process of KI only requires the computation of two hyperparameters, L and p. This learning step is usually done offline, before the system is operating, so computational complexity is generally not a key consideration at this stage. The prediction step is also simple, since KI just computes the upper and lower bounds functions, and their mean. On the other hand, its simplicity often requires large datasets to perform well enough (since it explicitly uses the training dataset $\mathcal{D}$ for prediction). This is related to the fact that the prediction error decreases as more data are considered to make predictions. Another important property is the consistency, due to the fact that the predictor in (2.18) is Hölder continuous, which implies that the functions derived from $\hat{f}$ are also Hölder continuous (including the cost function, which is a requirement for most of the robust MPC designs). Then, also the bound on the

uncertainty $\mu = \|y - \hat{y}\|$ is an important property, which depends on the real L_f, which means that there are some possibilities [83]:

- assume that the real Hölder constant is known,
- derive stochastic estimations of the real constant L_f and thus derive probabilistic estimations of the bound on the uncertainty,
- assume that the real μ is computed via validation (see Theorem 2.2).

2.2.2 CHoKI predictor

The objective of this section is to introduce an extended version of the aforementioned LACKI. In detail, the Componentwise Hölder Kinky Inference (CHoKI) [86] is considered, which exploits an enlarged version of continuity of the function, the so-called componentwise Hölder continuity, in Definition 2.7. This method considers matrices $\mathcal{L}$ and $\mathcal{P}$, instead of the Hölder constant L and exponent p (see (2.17)). This is done in order to be able to catch different variations of the output, according to the changes of each component of the input regressor. This is particularly useful in cases where a function may have sudden variations along one dimension of the input, while changing smoothly along other input dimensions.

Definition 2.7: Componentwise Hölder continuity

Given the matrices $\mathcal{L}$ and $\mathcal{P} \in \mathbb{R}^{n_y \times n_w}$, a function $f : \mathcal{W} \to \mathcal{Y}$ is componentwise $\mathcal{L}$-$\mathcal{P}$-Hölder continuous if $\forall w_1, w_2 \in \mathcal{W}$ and $\forall i \in \mathbb{I}_1^{n_y}$

$$|f(w_1) - f(w_2)| \leq \mathfrak{d}_{\mathcal{L}}^{\mathcal{P}}(|w_1 - w_2|) \tag{2.20}$$

where

$$\mathfrak{d}_{\mathcal{L}}^{\mathcal{P}}(w) := \left(a : \; a_i = \sum_{j=1}^{n_w} \mathcal{L}_{i,j} w_j^{\mathcal{P}_{i,j}}, \;\; \forall i \in \mathbb{I}_1^{n_y} \right). \tag{2.21}$$

The following theorem states the conditions under which Hölder continuity and componentwise Hölder continuity are equivalent.

Theorem 2.1

Let $f : \mathcal{W} \subseteq \mathbb{R}^{n_w} \to \mathcal{Y} \subseteq \mathbb{R}^{n_y}$.

1. *If f is Hölder continuous in $\mathcal{W}$, then f is componentwise Hölder continuous in $\mathcal{W}$.*
2. *If $\mathcal{W}$ is compact and f is componentwise Hölder continuous in $\mathcal{W}$, then f is Hölder continuous in $\mathcal{W}$.*

Then, assuming that f is Hölder continuous and given a dataset of inputs/outputs observations $\mathcal{D}$, the standard KI predictor in Definition 2.6 is slightly modified, to get the CHoKI predictor for a query $q \in \mathbb{R}^{n_w}$ as follows:

Definition 2.8: CHoKI predictor

Given a dataset of $N_\mathcal{D}$ input-output observations, $\mathcal{D} = \left\{ (\tilde{y}_i, w_i), i \in \mathbb{I}_1^{N_\mathcal{D}} \right\}$, the prediction of an unseen query point $q \in \mathbb{R}^{n_w}$ is computed as

$$\hat{\mathfrak{f}}(q; \Theta, \mathcal{D}) = \frac{1}{2} \min_{i=1,\dots,N_\mathcal{D}} \left(\tilde{y}_i + \mathfrak{d}_\mathcal{L}^\mathcal{P} \left(|q - w_i| \right) \right) + \frac{1}{2} \max_{i=1,\dots,N_\mathcal{D}} \left(\tilde{y}_i - \mathfrak{d}_\mathcal{L}^\mathcal{P} \left(|q - w_i| \right) \right), \tag{2.22}$$

where $\Theta = \{\mathcal{L}, \mathcal{P}\}$ and $\hat{f}$ is still componentwise $\mathcal{L}$-$\mathcal{P}$-Hölder continuous.

According to (2.22) it is possible to predict a new output $\hat{y}(k+1) = \hat{f}\big(w(k); \Theta, \mathcal{D}\big)$. Then, the prediction model can be formulated in a NARX state-space as follows:

$$\begin{cases} \hat{x}(k+1) = \hat{F}\big(x(k), u_1(k), u_2(k)\big) \\ \hat{y}(k) = M\hat{x}(k) \end{cases} \tag{2.23}$$

where $\hat{F}\big(x(k), u_1(k), u_2(k)\big) = \big(\hat{\mathfrak{f}}(x(k), u_1(k), u_2(k)), y(k), \dots, y(k - n_a + 1), u_1(k), \dots, u_1(k - n_b + 1), u_2(k), \dots, u_2(k - n_c + 1)\big)$ and $M = [I_{n_y}, 0, \dots, 0]$.

As with the LACKI, if the matrices $\mathcal{L}$ and $\mathcal{P}$ are unknown a priori, they can be estimated by exploiting the available data. To do that, differently from before, an optimization problem is solved offline, subject to a constraint that ensures their consistency with the samples of the dataset, and splitting the dataset $\mathcal{D}$ into two disjoint datasets: $\mathcal{D}_{\text{train}}$ for the estimation and $\mathcal{D}_{\text{test}}$ for the validation. The optimization problem to find $\Theta = \{\mathcal{L}, \mathcal{P}\}$ is designed as follows:

$$\Theta = \arg\min_\Theta g\left(\Theta, \mathcal{D}_{\text{train}}, \mathcal{D}_{\text{test}}\right) + \tau_\mathcal{L} \left\| \mathcal{L} - \mathcal{L}_0 \right\|_1 \tag{2.24a}$$

$$\text{s.t. } |\tilde{y}_i - \tilde{y}_j| \leq \mathfrak{d}_\mathcal{L}^\mathcal{P}\left(|w_i - w_j|\right), \quad \forall w_i, w_j \in \mathcal{W}_\mathcal{D},\ w_i \neq w_j \tag{2.24b}$$

$$0 < \mathcal{P}_{ij} \leq 1,\ \mathcal{L}_{ij} > 0, \quad i \in \mathbb{I}_1^{n_y},\ j \in \mathbb{I}_1^{n_w}, \tag{2.24c}$$

where $\mathcal{W}_{\mathcal{D}}$ represents the input data points in $\mathcal{D}$. $\tau_{\mathcal{L}} \geq 0$ is a design regularization hyper-parameter used to ensure boundedness of $\mathcal{L}$, and $\mathcal{L}_0$ stands for any possible prior guess of $\mathcal{L}$. The cost function $g(\Theta, \mathcal{D}_{\text{train}}, \mathcal{D}_{\text{test}})$ to be minimized, which evaluates the performance of the prediction over the dataset $\mathcal{D}_{\text{test}}$, is defined as:

$$g(\Theta, \mathcal{D}_{\text{train}}, \mathcal{D}_{\text{test}}) = \frac{1}{N_{\mathcal{D}_{\text{test}}}} \sum_{i=1}^{N_{\mathcal{D}_{\text{test}}}} \left\| \hat{\mathfrak{f}}(w_i; \Theta, \mathcal{D}_{\text{train}}) - \tilde{y}_i \right\|^2, \tag{2.25}$$

being $\hat{\mathfrak{f}}(w_i; \Theta, \mathcal{D}_{\text{train}})$ the predictions made with the CHoKI in (2.22) (computed with the data in $\mathcal{D}_{\text{train}}$), which are compared to $\tilde{y}_i$, that are the measured values of the noisy dataset $\mathcal{D}_{\text{test}}$.

Remark 2.1

The regularization term of the cost function, $\tau_{\mathcal{L}} \|\mathcal{L} - \mathcal{L}_0\|_1$ in (2.24), prevents the problem from overfitting the noise, while ensuring boundedness of $\mathcal{L}$. If $\mathcal{D}_{\text{test}}$ is separated from $\mathcal{D}$ and such that g is bounded for all $\mathcal{L}$, then it can be removed, setting $\tau_{\mathcal{L}} = 0$.

Theorem 2.2: Prediction error boundedness

Let $\Theta = (\mathcal{L}, \mathcal{P})$ be obtained as the solution of (2.24) for a dataset $\mathcal{D}$ with $\eta \geq 2\bar{\mathfrak{e}}$, and assume that the function f satisfies the componentwise Hölder condition (2.20) for the pair $(\mathcal{L}_f, \mathcal{P})$ in $\mathcal{W}$. Then, $\mathcal{L}$ is bounded and

$$\left| f(w) - \hat{\mathfrak{f}}(w; \Theta, \mathcal{D}) \right| \leq \mathfrak{d}^{\mathcal{P}}_{\mathcal{L}_f + \mathcal{L}}(\mathcal{R}_{\mathcal{D}}) + \frac{\eta}{2} + \bar{\mathfrak{e}}, \tag{2.26}$$

where $\mathcal{R}_{\mathcal{D}} = \max_{w \in \mathcal{W}} \min_{w_j \in \mathcal{W}_{\mathcal{D}}} (|w_j - w|)$ measures the maximum radius between a possible query and the dataset.

Based on the Theorem 2.2, it can be derived that the prediction error is bounded for all queries q in a compact space $\mathcal{W}$. Besides, it proves that as more observations are added to the dataset, the prediction error decreases, vanishing up to $\frac{\eta}{2} + \bar{\mathfrak{e}}$ for infinitely dense datasets, when $\mathcal{R}_{\mathcal{D}} \to 0$.

2.3 CHoKI-based MPC

This section presents the design of an MPC based on the CHoKI learning method, ensuring closed-loop constraint satisfaction and ISS, as proved in [83], to build robust learning-based MPC for nonlinear constrained systems.

First, the following problem setting is considered, in which the state-space model used in the MPC, with sampling time k, is given by:

$$\begin{cases} \hat{x}(j+1|k) = \hat{F}\left(x(j|k), u(k+j); \theta_{KI}, \mathcal{D}\right) \\ \hat{y}(j|k) = M\hat{x}(j|k) \end{cases} \tag{2.27}$$

where the predicted state

$$\begin{aligned} \hat{x}(j|k) = \big(&\hat{y}(j|k), ..., \hat{y}(1|k), y(k), ..., y(k+j-n_a), \\ &u(k+j-1), ..., u(k), ..., u(k+j-n_b)\big), \end{aligned} \tag{2.28}$$

includes past measurements y if $n_a \geq j$ (or $n_b > j$ for the u), or only estimated future values $\hat{y}$ otherwise. There are also possible constraints on the outputs $y \in \mathcal{Y}$, and on the inputs $u \in \mathcal{U}$.

The model $\hat{F}(\cdot)$ is:

$$\begin{aligned} \hat{F}(\hat{x}(j|k), u(k+j); \theta_{KI}, \mathcal{D}) = \big(&\hat{\mathfrak{f}}(\hat{x}(j|k), u(k+j); \theta_{KI}, \mathcal{D}), \hat{y}(j|k), ..., y(k), ..., \\ &y(k+j-n_a+1), u(k+j), ..., u(k+j-n_b+1)\big), \end{aligned} \tag{2.29}$$

where $\hat{\mathfrak{f}}$ is the prediction function, and the output matrix is $M = [I_{n_y}, 0, ..., 0] \in \mathbb{R}^{n_x \times n_x}$, being I_{n_y} the identity matrix.

MPC based on KI The general optimization problem $P_{N_p}(x(k); \mathcal{D})$ can be written as

$$\min_{\mathbf{u}} \quad V_{N_p}(x(k), \mathbf{u}) = \sum_{i=0}^{N_p-1} \ell\big(\hat{x}(i|k), u(k+i)\big) + \lambda V_f\big(\hat{x}(N_p|k)\big) \tag{2.30a}$$

$$\text{subject to} \quad \hat{x}(0|k) = x(k) \tag{2.30b}$$

$$\hat{x}(j+1|k) = \hat{F}\big(\hat{x}(j|k), u(k+j)\big),\ j \in \mathbb{I}_0^{N_p-1} \tag{2.30c}$$

$$\hat{y}(j|k) = M\hat{x}(j|k),\ j \in \mathbb{I}_0^{N_p-1} \tag{2.30d}$$

$$u(k+j) \in \mathcal{U},\ j \in \mathbb{I}_0^{N_p-1}, \tag{2.30e}$$

$$\hat{y}(j|k) \in \mathcal{Y}_j,\ j \in \mathbb{I}_0^{N_p-1}, \tag{2.30f}$$

where $\lambda \geq 1$ is the weighting parameter for the terminal cost V_f, and $\hat{F}$ and M define the state-space model. $\mathcal{Y}_j$ represents the tightened constraints, because the controller is learning-based, and thus it means that it has to deal with possible system perturbations and uncertainty given by the KI method. Therefore, in the design phase, the prediction error bounds need to be considered to tighten the problem constraints and guarantee closed-loop constraint satisfaction, and the ISS property [83].

Remark 2.2

Note that the prediction function $\hat{\mathfrak{f}}$ could be the LACKI one, or the CHoKI one (i.e., in (2.22)*) to obtain the CHoKI-based MPC. The same reasoning holds for the state-space model $\hat{F}, M$.*

Note that this problem is nonlinear, non-convex, and non-differentiable. Its ingredients are required to meet the following assumptions:

Assumption 2.4

1. *The stage cost function $\ell(\cdot,\cdot)$ is a Hölder continuous positive definite function such that $\ell(\cdot,\cdot) \geq \alpha_y(\|x\|_{\mathcal{X}})$ for a certain $\mathcal{K}$-function α_y, and its Hölder parameters are L_x and p_x.*
2. *There exists a control law $u = \kappa_f(x)$, a function V_f and a level set*

$$\Omega_\gamma = \{x : V_f(x) \leq \gamma\} \subseteq \mathbb{R}^{n_x} \tag{2.31}$$

for some $\gamma > 0$ such that for all $x \in \Omega_\gamma$ the following conditions hold:

(a) V_f is a Hölder continuous positive definite function, with Hölder constants denoted L_{V_f}, p_{V_f}. Given two $\mathcal{K}$-functions α_f, β_f,

$$\begin{aligned} \alpha_f(\|x\|_{\mathcal{X}}) \leq V_f(x) \leq & \beta_f(\|x\|_{\mathcal{X}}), \\ V_f(\hat{F}(x, \kappa_f(x))) - V_f(x) \leq & -\ell(x, \kappa_f(x)). \end{aligned}$$

(b) $\kappa_f(x) \in \mathcal{U}$, $Mx \in \mathcal{Y}_{N_p}$.

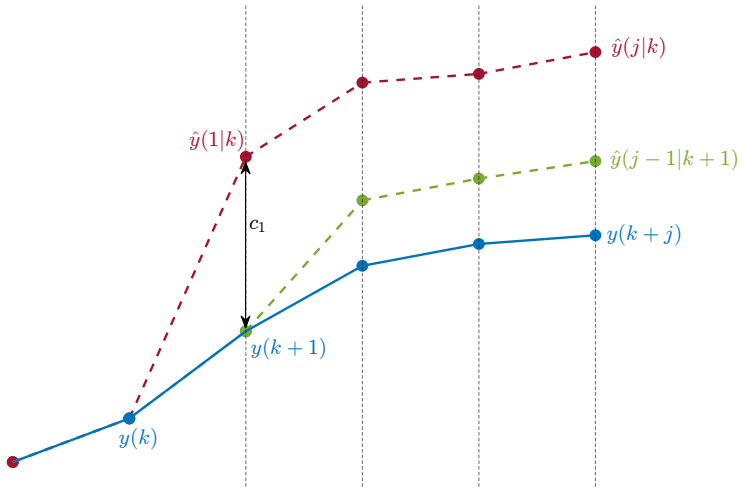


FIGURE 2.3: Propagation of the KI prediction error. The blue line represents the real values $y(\cdot)$, the red and green dotted lines represent the predictions $\hat{y}$ made at different times. c_1 is the one-step-ahead prediction error.

Assumption 2.5

The bound μ is such that the set $\Upsilon = \{x : \ell(x,0) \leq \nu(\mu)\}$ is contained in Ω_γ. The positive constants λ and ϕ are such that $\lambda \geq 1$ and $\ell(x,0) > \phi$ for all $x \notin \Omega_\gamma$, with

$$\nu(c_1) = \sum_{j=1}^{N_p} L_x r_j^{p_x} + \lambda L_{V_f} r_{N_p+1}^{p_{V_f}}, \tag{2.32}$$

whose components are defined in [83, Section 3].

2.3.1 Reachability sets and tightening of the constraints

As said, dealing with a KI learning-based approach means taking into consideration the prediction uncertainty, which is done by tightening the constraints. In Figure 2.3, the propagation of the prediction error is shown. The proposed robust MPC is based on nominal predictions and tightened constraints. To guarantee robustness, a bound on the propagation of the prediction error is calculated from the [83, Lemma 3.1].

The main advantages of using the CHoKI prediction method instead of the standard KI method are that the obtained prediction error bounds are smaller and the reachable sets are estimated more accurately. This also leads to a significant reduction in the conservatism of the tightened constraints.

Assumption 2.6

The prediction horizon N_p and estimation error bound μ are such that the set $\mathcal{Y}_{N_p}$ is non-empty.

According to Assumption 2.6, the prediction horizon value depends on the output constraints, which are computed as follows.

In order to implement the control problem (2.30), it is necessary to calculate the reachability sets $\mathcal{R}_j$ for the CHoKI predictor. The set of tightened output constraints $\mathcal{Y}_j$ in (2.30f) is given by

$$\mathcal{Y}_j = \mathcal{Y}_{j-1} \ominus \mathcal{R}_j, \;\; j \in \mathbb{I}_1^{N_p} \tag{2.33}$$

where $\mathcal{R}_j$ are the reachability sets that account for the possible errors in the nominal predictions and $\mathcal{Y}_0 = \mathcal{Y}$. To compute $\mathcal{R}_j$, the starting point is to consider a sequence of future inputs $u(k+1)$ and $c_1 \in \mathbb{R}^{n_y}$, such that

$$|y(k+1) - \hat{y}(1|k)| \leq c_1. \tag{2.34}$$

The difference between a prediction made at time step $k+j$ based on the measurement at step k, and the prediction made at step k based on the measurement at step $k+1$, for a given sequence of control inputs, is bounded by the sets

$$|\hat{y}(j|k) - \hat{y}(j-1|k+1)| \in \mathcal{M}_j \subseteq \mathbb{R}^{n_y}, \tag{2.35a}$$

$$|\hat{w}(j|k) - \hat{w}(j-1|k+1)| \in \mathcal{G}_j \subseteq \mathbb{R}^{n_w}. \tag{2.35b}$$

The sets $\mathcal{M}$ and $\mathcal{G}$ can be calculated from the equations

$$\mathcal{M}_j = \mathbb{B}\left(\mathfrak{d}_{\mathcal{L}}^{\mathcal{P}}(\mathcal{G}_{j-1})\right), \tag{2.36a}$$

$$\mathcal{G}_j = \mathcal{M}_j \times \cdots \times \mathcal{M}_{\sigma(j)} \times \{0\} \times \cdots \times \{0\}, \tag{2.36b}$$

with $\sigma(j) = \max(1, j - n_a)$ and $\mathcal{M}_1 = \mathbb{B}(c_1)$. The set $\mathcal{R}_j$ is defined as

$$\mathcal{R}_j = \{y : |y| \in \mathcal{M}_j\}, \tag{2.37}$$

for all $j \in \mathbb{I}_1^N$ and with $\mathcal{M}_1 = \mathbb{B}(\mu)$.

In [86] is also shown that $c_j \in \mathbb{R}^{n_y}$ and $d_j \in \mathbb{R}^{n_w}$ are such that $\mathcal{M}_j = \mathbb{B}(c_j)$ and $\mathcal{G}_j = \mathbb{B}(d_j)$. Then, the sets $\mathcal{M}_j$ and $\mathcal{G}_j$ can be computed using the recursion

$$c_j = \mathfrak{d}_{\mathcal{L}}^{\mathcal{P}}\left(d_{j-1}\right), \tag{2.38a}$$

$$d_j = (c_j, \ldots, c_{\sigma(j)}, 0, \ldots, 0), \tag{2.38b}$$

and $\mathcal{R}_j = \mathcal{B}(c_j)$.

2.3.2 Stabilizing conditions for CHoKI-based MPC

Some additional assumptions are required to recover the safe-by-design properties of the controller in (2.30) based on KI.

Assumption 2.7

It is assumed that for L, p, and $\mathcal{D}$, a bound on the error between the estimated output and the real output is known, denoted $\mu \in \mathbb{R}^{n_y}$, such that

$$\left|\hat{\mathfrak{f}}_i(x, u; \theta, \mathcal{D}) - f_i(x, u)\right| \leq \mu_i, \tag{2.39}$$

for all $i \in \mathbb{I}_1^{n_y}$, $x \in \mathcal{Y}^{n_a+1} \times \mathcal{U}^{n_b}$, and $u \in \mathcal{U}$.

Assumption 2.8

The stage cost function $\ell(x, u)$ is a componentwise Hölder continuous, positive definite function such that $\ell(x, u) \geq \alpha(\|x\|)$, where α is a $\mathcal{K}$-function. Given $\mathcal{P}$, its Hölder constant is $\mathcal{L}_\ell$.

Assumption 2.9

V_f is a componentwise Hölder continuous, positive definite function, with Hölder parameters $\mathcal{L}_f, \mathcal{P}$, such Assumption 2.8 holds.

Assumption 2.10

Assumption 2.5 holds redefining ν as

$$\nu(\mu) = \sum_{j=1}^{N_p} \left\| \mathfrak{d}_{\mathcal{L}_\ell}^{\mathcal{P}}(\mathcal{G}_j) \right\|_\infty + \lambda \left\| \mathfrak{d}_{\mathcal{L}_f}^{\mathcal{P}}(\mathcal{G}_{N_p+1}) \right\|_\infty. \tag{2.40}$$

Then, the following theorem can be derived, to ensure ISS (proved in [83]).

Theorem 2.3: ISS stability

Suppose that assumptions 2.7-2.10 hold for the optimization problem $P_N(\cdot)$ obtained with the CHoKI predictor. Let $\kappa_{N_p}(x)$ be the control law derived from the solution of $P_N(\cdot)$ applied using a receding horizon policy. Then, for any $x(0) \in \Gamma$, the system to be controlled by the control law $u(k) = \kappa_{N_p}(x(k))$ is input-to-state stable with respect to the estimation error, $x(k) \in \Gamma$, and the constraints are always satisfied, i.e., $y(k) \in \mathcal{Y}, \forall k$.

2.4 MPC for AP case

The MPC is already used in the glucose management context, where it is implemented as a control algorithm in the AP. It has been studied and tested a lot, due to its ability to anticipate undesired glucose variation and to compute the amount of insulin injections, respecting all the imposed constraints. There are several approaches that show how the MPC structure can be adapted and improved, some examples are in [22, 46, 53, 56, 66, 80, 107, 113]. Among the many, the zone MPC is one of the most widely used, for the reasons explained below [47]. In addition to the zone structure, the pulsatile approach is also analysed [1, 49, 50], due to the fact that commercial insulin pumps administer insulin by means of micro boluses (that is, as a pulsatile injection) and this approach shows to have physiological benefits [2]. There also exist some MPC that manage both insulin and glucacon (i.e., dual-hormon control) [18]. Some of these studies have also been tested in clinical trials, as described in [36, 45], or [37], which is the first wearable AP outpatient study based on MPC. This highlights the importance of considering clinical experience during MPC design [116]. It has also been implemented in the AP on the market (i.e., Control IQ algorithm in Tandem t:slimX2, and Diabeloop Generation 1 algorithm in Accu-Check Insight [39]).

In all of them, the main features to consider are the model describing the system, the cost function to be minimized, and the constraints. In the AP case, the model can be the physiological one, describing the patients' insulin-glucose relationship, or it can be derived from input-output data. Specifically, the manipulated variable (i.e., the control action) is the insulin amount, which has to be greater than zero and lower

than a maximal injectable value $u_{\min} \leq u_t \leq u_{\max}$, and the controlled variable (i.e., the BG value) has to be inside an admissible range $y_{\min} \leq y \leq y_{\max}$.

Zone MPC In previous sections, it was stated that the objective of control is to match the reference value perfectly (i.e., the target y_t or the artificial variable y_s in the case of robust control). However, in many applications, the optimal value of the controlled output is not given by a specific set-point, but it has to remain inside a range with specified limits [48]. For example, the desired zone can be defined as

$$y_{\min} \leq y \leq y_{\max} \tag{2.41}$$

In those cases, the zone control strategy can be implemented. In particular, the zone MPC modifies the structure of the cost function in order to penalize the cases in which the system leaves the desired zone (i.e., when y goes below $y_{\min}$ or above $y_{\max}$).

Note that this zone strategy is particularly useful in the case of the BG management, given that the control aim is precisely to keep it within the euglycemic range [107].

MPC cost function The case shown in [2] is examined, since, for the case study proven in this thesis, it is taken as a reference in the choice of the cost function. Specifically, many considerations need to be made to analyse this system. In the MPC, there is usually a specific output setpoint, but in the field of diabetes, the control goal is to maintain the BG inside the euglycemic range. Therefore, it makes more sense to create a target range as the setpoint. Another important factor to consider is that hypoglycemic and hyperglycemic cases are not equally dangerous. Thus, inside the cost function, the two conditions will be divided and different weights will be assigned, allowing hypoglycemic events to be penalised more severely.

In particular, the optimization problem is the following:

$$\begin{aligned}
\min_{\mathbf{u},u_a,y_a} \quad & V_{N_p}(\hat{x},\hat{u}_1,\mathcal{Y}_s^{\text{Tar}};\mathbf{u},u_a,y_a) && \text{(2.42a)}\\
\text{subject to} \quad & x(0)=\hat{x},\ \ u_1(0)=\hat{u}_1, && \text{(2.42b)}\\
& x(j+1)=A^d x(j)+B_{u_t}^d u_t(j)+B_{u_1}^d u_1(j)+E^d,\ \ j\in\mathbb{I}_0^{N_p-1}, && \text{(2.42c)}\\
& u_t(j)\in\mathcal{U},\ \ j\in\mathbb{I}_0^{N_p-1}, && \text{(2.42d)}\\
& \tilde{C}x(j)\in\tilde{C}\mathcal{X}(k),\ \ j\in\mathbb{I}_0^{N_p}, && \text{(2.42e)}\\
& u_1(j)=0,\ \ j\in\mathbb{I}_1^{N_p}, && \text{(2.42f)}
\end{aligned}$$

$$\tilde{C}x(N_p) = x_a, \tag{2.42g}$$

$$y_a = x_{1,a}, \tag{2.42h}$$

$$x_a = \tilde{C}A^d x_a + \tilde{C}B^d u_a + \tilde{C}E^d. \tag{2.42i}$$

Where k is the current time, N_p is the control horizon, $u_t(j)$ is the insulin infusion and $u_1(j)$ is the rate of carbohydrate intake, $\left(A^d, B^d_{u_t}, B^d_{u_1}, E^d\right)$ are the matrices of the known model, (u_a, y_a) is the artificial reference (i.e., an equilibrium and it is an optimization variable), $\mathcal{Y}_s^{\text{Tar}}$ is the target equilibrium (i.e., glucose safety interval), $x(N) = x_a$ is the terminal constraint, $x_a = f(x_a, u_a, 0)$ forces (u_a, y_a) to be a feasible equilibrium for the system and $u_1(j) = 0$ since no future meals are considered. The cost function is:

$$V_N\left(\hat{x}, \hat{r}, \mathcal{Y}_s^{\text{Tar}}; u_t, u_a, y_a\right) = V_{\text{dyn}}\left(\hat{x}, \hat{r}; u_t, u_a, y_a\right) + V_s\left(\mathcal{Y}_s^{\text{Tar}}; u_a, y_a\right) \tag{2.43}$$

with

$$V_{dyn}(\hat{x}, \hat{r}; u_t, u_a, y_a) = \sum_{j=0}^{N_p-1} \|Cx(j) - y_a\|_Q^2 + \|u_t(j) - u_a\|_R^2, \tag{2.44a}$$

$$V_s\left(\mathcal{Y}_s^{\text{Tar}}; u_a, y_a\right) = p_{\text{hyper}}\delta_{\text{hyper}}^2 + p_{\text{hypo}}\delta_{\text{hypo}}^2. \tag{2.44b}$$

Where $Q > 0, R > 0$ are the two weights to penalize the glucose mismatch with the reference value and the insulin injections, respectively. Note that setting $R > Q$ means to give more importance to the second factor, and, therefore, the solution of the optimization problem will not choose to give too much insulin (i.e., obtaining a more conservative controller). The stationary cost V_s is added to account for the fact that hypoglycemia is more dangerous than hyperglycemia, setting $p_{\text{hyper}} \ll p_{\text{hypo}}$, in order to penalize the hypoglycemic events more. δ_{hypo} and δ_{hyper} are additional optimization variables, constrained to be positive, allowing to obtain $\mathcal{Y}_s^{\min} - \delta_{\text{hypo}} \leq y_a \leq \mathcal{Y}_s^{\max} + \delta_{\text{hyper}}$, where $\mathcal{Y}_s^{\min}, \mathcal{Y}_s^{\max}$ are the limits values of $\mathcal{Y}_s^{\text{Tar}}$.

Data-driven MPC in AP case The aforementioned MPC employs a linear state-space model to make BG predictions. But there exists the possibility to implement a data-driven learning method instead of the model, to try to obtain more precise and customized BG predictions.

In particular, the management of the BG values in T1DM patients is a challenging control, since the system is nonlinear, complex, with exogenous disturbances (such

as, meals, physical activity or stress), and requires a safe and customised control strategy, which complicate the modeling of the BG-insulin behavior. Indeed, physiological models offer a detailed description of metabolic processes, but they could be too complex and difficult to personalise, and are also sensitive to uncertainties, such as due to S_I, lifestyle choices or hormonal variations. On the other hand, data-driven models allow for learning directly from patient data, being able to capture individual BG-insulin behavior, without the need to explicitly model each physiological process. This enables greater personalization and adaptability of the controller, thus improving the accuracy of the BG management [58].

Several studies in the literature demonstrate the feasibility and effectiveness of data-driven MPC in regulating BG levels, exploiting various techniques to model the input-output dynamics of the BG-insulin system. For example, a Neural Network (NN) can be used for modeling the complex, time-varying and patient-specific dynamics of BG regulation, as in [4], or in [10], where the use of recurrent NN is presented, or in [40], where a hybrid NN is proposed, combining convolutional and recurrent layers. As alternatives, several works exploit nonparametric regression techniques, such as Gaussian process regression or kernel methods, which offer probabilistic predictions and can explicitly model prediction uncertainty. For example, in [97] an MPC based on Gaussian processes is presented, while in [95] and [93] kernel-based methods are implemented.

For this reason, this thesis proposes the use of the CHoKI method to learn the relationship between patients' BG-insulin, and this learning technique is used to make BG predictions within the MPC, which will then calculate the optimal insulin dose.

CHAPTER 3

CHoKI-based MPC applied to the glucose management in T1DM patients

The aim of this chapter is to present the results of the implementation of the proposed CHoKI-based MPC to the virtual T1DM adult patients of the UVA/Padova simulator, with the aim of trying to drive the BG values in the euglycemic safe range, by injecting the optimal insulin amount. Specifically, this is focused on the computation of basal insulin injections, leaving bolus management to the simulator.

The results were published in [108], [110] and [111].

3.1 T1DM patient system

T1DM patient can be represented as a continuous-time sampled system, described by an a priori unknown discrete-time model, whose measured output is $y(k) \in \mathbb{R}^{n_y}$ and whose input is $u(k) \in \mathbb{R}^{n_u}$. Specifically, in the problem under study, there is one output (i.e., $n_y = 1$) which is the glucose level, in mg/dL, and there are two inputs (i.e., $n_u = 2$), a manipulable variable (that is, u_2, the insulin in pmol) and a measurable disturbance (that is, u_1, the meal in g of carbohydrates). A sampling time of 5 minutes is considered.

The measured output can be modeled as a NARX model, with the following state-space representation:

$$y(k+1) = f\big(x(k), u_1(k), u_2(k)\big) + e(k), \tag{3.1}$$

where $e(k) \in \mathbb{R}^{n_y}$ is the process noise and the regression state $x \in \mathbb{R}^{n_x}$ is

$$x(k) = \big(y(k), \dots, y(k-n_a), u_1(k-1), \dots, u_1(k-n_b), u_2(k-1), \dots, u_2(k-n_c)\big), \tag{3.2}$$

where $n_a \in \mathbb{N}_0$ is the memory horizon for the glucose values, $n_b \in \mathbb{N}_0$ for the meals and $n_c \in \mathbb{N}_0$ for the insulin injections. The arguments of f are then aggregated in $w = (x, u_1, u_2) \in \mathbb{R}^{n_w}$ so that it is possible to build a dataset of $N_{\mathcal{D}}$ observations, denoted $\mathcal{D} = \{(y_{k+1}, w_k)\}$, for $k = 1, \dots, N_{\mathcal{D}} - 1$.

According to (2.22) it is possible to predict a new output $\hat{y}(k+1) = \hat{f}\big(w(k); \Theta, \mathcal{D}\big)$. Then, the prediction model can be formulated in state-space as follows:

$$\begin{cases} \hat{x}(k+1) = \hat{F}\big(x(k), u_1(k), u_2(k)\big) \\ \hat{y}(k) = M\hat{x}(k) \end{cases} \tag{3.3}$$

where $M = [I_{n_y}, 0, \dots, 0]$ and

$$\hat{F}\big(x(k), u_1(k), u_2(k)\big) = \Big(\hat{f}(x(k), u_1(k), u_2(k)), y(k), \dots, y(k-n_a+1), \\ u_1(k), \dots, u_1(k-n_b+1), u_2(k), \dots, u_2(k-n_c+1)\Big).$$

3.2 Design of CHoKI-based MPC for glucose management

The primary objective is to design a customized controller that can calculate the amount of basal insulin required to maintain BG levels within the euglycemic range, with a particular focus on preventing hypoglycemic events due to their short-term risks. In the section, the steps followed to obtain and test the proposed controller are detailed.

The first step in building the proposed controller is to design and to train the CHoKI learning method. To achieve this, data collection and model identification are critical.

3.2.1 Data collection for CHoKI identification

In order to compute the BG predictions made with the CHoKI needed inside the MPC, as in (2.22), the first thing to do is to identify the CHoKI parameters $\Theta =$

$\{\mathcal{L}, \mathcal{P}\}$. In this case, to obtain a customized model, these parameters have to be estimated by learning the behavior of a specific subject, and therefore, data from the patients is required. The data collection phase is critical, since the quality of the generated dataset will affect the performance of the CHoKI predictor and, consequently, the functioning of the controller. This is done through the in-silico adult T1DM patients of the UVA/Padova simulator. For each of them, several simulations were conducted, varying the initial glycemic condition, the basal insulin quantity (from 0 to 500 pmol), and the carbohydrate amount of the meals. The boluses were given 20 min after the meal starting time to help the CHoKI to learn the correct effect that boluses and meals have on the BG levels. This was done to avoid possible identification issues due to the different dynamics that insulin and carbohydrates have on the BG variations. All these simulations were set to obtain an appropriate distribution of the points in the space, looking at the input-output representation. Then, the available predefined virtual typical commercial CGM was selected as a sensor, with auto-regressive noise with inverse Johnson transform distribution. The noise on the virtual pump is normally distributed with a mean of 0 pmol and a standard deviation of 0.1. An error with a normal distribution with a standard deviation equal to 30% of the meal amount is also added to the carbohydrate estimate, for the bolus computation.

The data collection performed is represented in Figure 3.1, showing the days of simulation of the subject Adult 1 as an example, where in the upper graph the BG trend is reported (i.e., the purple line is for the BG values and the green zone is the euglycemic range), in the middle graph the basal insulin is visible, and in the lower one are shown both the meals (i.e., the orange line) and the boluses (i.e., the blue dots).

3.2.2 Identification of CHoKI parameters

Once the data are collected for all the subjects, the next step is to identify the CHoKI parameters, then test and validate whether the learned model can make correct predictions of the system behavior.

At this point, only the relationship between BG, meals, and basal insulin is considered, as the aim of the proposed controller is to manage basal insulin injections automatically, while meal boluses are delivered manually (assuming they are a function of meals). The CHoKI requires the data to be in the right NARX shape (as in (3.2));

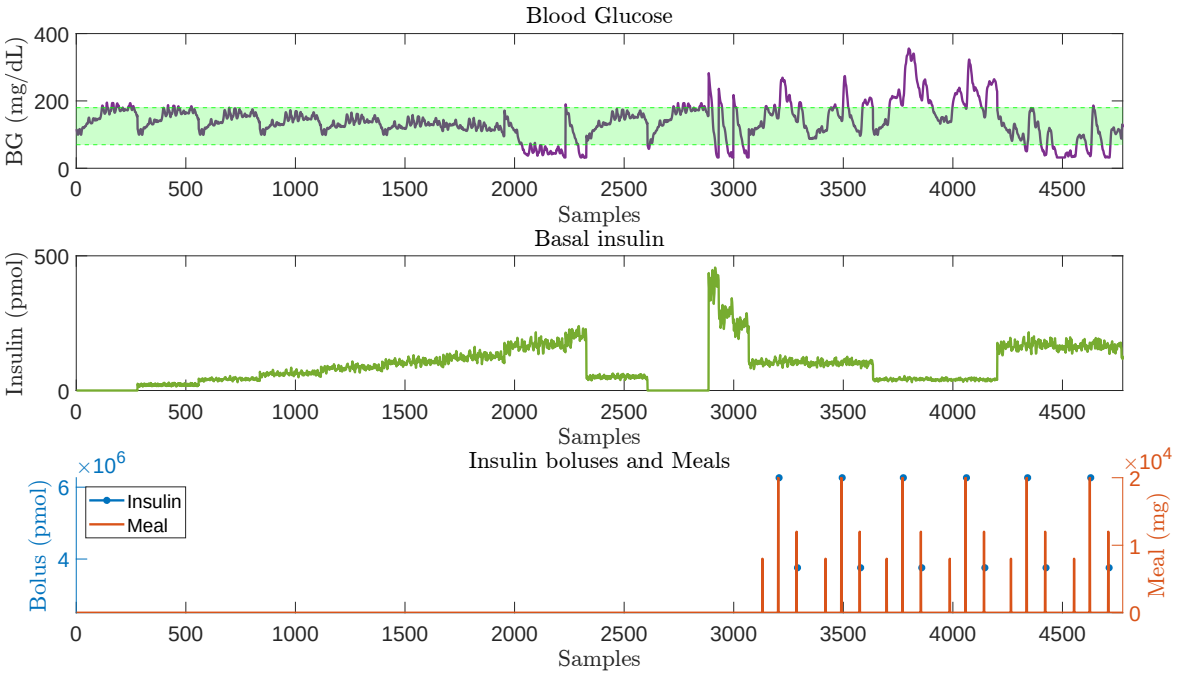


FIGURE 3.1: Example of the data collection for Adult 1. In the upper graph, the purple line represents the BG trend and the green zone the euglycemic range. In the middle graph, the green line represents the basal insulin injections. The lower graph shows the bolus insulin as the blue dots (given 20 min after the meal) and the meals as the orange line. Sampling time is 5 min.

thus, the model orders n_a, n_b, n_c have to be identified, and this is done through a cross-validation procedure.

In particular, many combinations of model orders were tested, with $n_a, n_b, n_c \in \mathbb{I}_1^{15}$. The Root Mean Squared Error (RMSE) among the 1-step ahead predictions and the actual values was measured on the dataset described above (as in (3.5)). The selected orders were chosen based on the lowest RMSEs, but a trade-off with model complexity was also considered to avoid the risk of overfitting (i.e., when a complex model perfectly describes the relationships on the training dataset, but fails in the generalization with new data). The resulting orders are $n_a = 5$, $n_b = 9$ and $n_c = 3$, recalling that each sampling time 5 min long.

To obtain the predictions employing (2.22), the hyperparameters $\Theta = \{\mathcal{L}, \mathcal{P}\}$ must be estimated according to (2.24). We assumed to have $\mathcal{P} = \mathbb{1}_{n_y \times n_w}$ and thus the optimization problem is solved to obtain just the values of the matrix $\mathcal{L}$. In this case, only three values are estimated:

- $L_a \in \mathbb{R}$ for the glucose part,
- $L_b \in \mathbb{R}$ for the meals,
- $L_c \in \mathbb{R}$ for the insulin.

Therefore, $\mathcal{L}$ contains those three values repeated to reach the right dimension (i.e., the regressor length), thus

$$\mathcal{L} = [L_a \mathbb{1}_{n_a}; L_b \mathbb{1}_{n_b}; L_c \mathbb{1}_{n_c}] \in \mathbb{R}^{n_y \times n_w}. \tag{3.4}$$

Some a priori knowledge was utilized in order to set the constraints of the optimization problem: as the $\mathcal{L}$ initial value the Lipschitz constant L was exploited, which is obtained from the LACKI method (i.e., with (2.19)), based on the Hölder continuity property [85]. The upper and lower bounds for L_a, L_b, L_c were set as $[10; 10; 10]$ and $[0; 0.9; 0.09]$, respectively, thanks to previous analyses.

The optimization problem in (2.24) was implemented in MATLAB by exploiting the `fmincon` function. For each patient, once the $\mathcal{L}$ is found, the model is validated on a new dataset, to verify its ability to predict future BG values, comparing them with the real outputs. The best and worst cases are reported in Figure 3.2. Specifically, this shows the comparison between the BG predictions made with the CHoKI approach (i.e., $\hat{y}$ in red) and the measured values (i.e., y_{val} in blue). The upper graph

Adult	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	Median
RMSE	9.28	7.59	7.38	5.98	9.21	7.65	15.21	5.84	9.32	7.17	7.62 [7.17, 9.28]
GoF	78.59	77.65	72.97	84.67	74.92	82.23	72.23	86.14	82.41	77.61	78.12 [74.92, 82.41]

TABLE 3.1: RMSE and GoF values obtained in the validation of the CHoKI learning method. Last column represents the median and the $[25, 75]^{\text{th}}$ percentiles.

represents the best case (Adult 8) and the lower one the worst case (Adult 9). To evaluate the performances on all the virtual patients, the outcomes of the validation are reported in Table 3.1, where the RMSE and the Goodness of Fitting (GoF) are shown, comparing the predictions and the measured data of the unseen validation 4-day dataset. Specifically,

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N_{\text{val}}} (y_i - \hat{y}_i)^2}{N_{\text{val}}}}, \quad GoF = 100 \left(1 - \frac{\|y_i - \hat{y}_i\|}{\|y_i - \bar{y}_i\|}\right), \tag{3.5}$$

where N_{val} is the number of data in the validation dataset, y_i are the measured values, $\hat{y}_i$ are the predictions of the next BG sample and $\bar{y}_i$ is the mean value. For each subject, the resulting $\mathcal{L}$, the u_{ref} (coming from the daily basal insulin in Table 1.3) and the L are reported in Table 3.2.

Further analysis was also carried out, starting with a fixed regressor and varying the input values, to ensure that the CHoKI strategy had correctly learned the effect of each input on the output. The fixed regressor has BG set to 120 mg/dL, no meals, and constant basal insulin equal to the reference value u_{ref} of each patient, obtained from the standard therapy provided by the simulator, i.e., $w_{\text{fix}} = [120 \cdot \mathbb{1}_{n_a};\ 0 \cdot \mathbb{1}_{n_b};\ u_{\text{ref}} \cdot \mathbb{1}_{n_c}]'$. As an example, the results obtained from subject Adult 1 are displayed in Figure 3.3: it can be seen how the glucose trend (blue lines) decreases when the amount of the basal insulin injections increases (in (C)), it rises when there is carbohydrate ingestion (in (D)), and it reaches a sort of equilibrium when u_{ref} is injected (in (B)). This holds for all virtual patients considered.

3.2.3 Computation of the reachable sets and of the prediction and control horizons

In the proposed control problem, an *a posteriori* analysis showed that extreme deviations from nominal predictions are highly unlikely. Then, the value representing the 90^{th} and 95^{th} percentiles of the uncertainty distribution is used as μ (i.e., where $c_1 = \mu$ in (2.34)) instead of the maximum error (see Table 3.2).

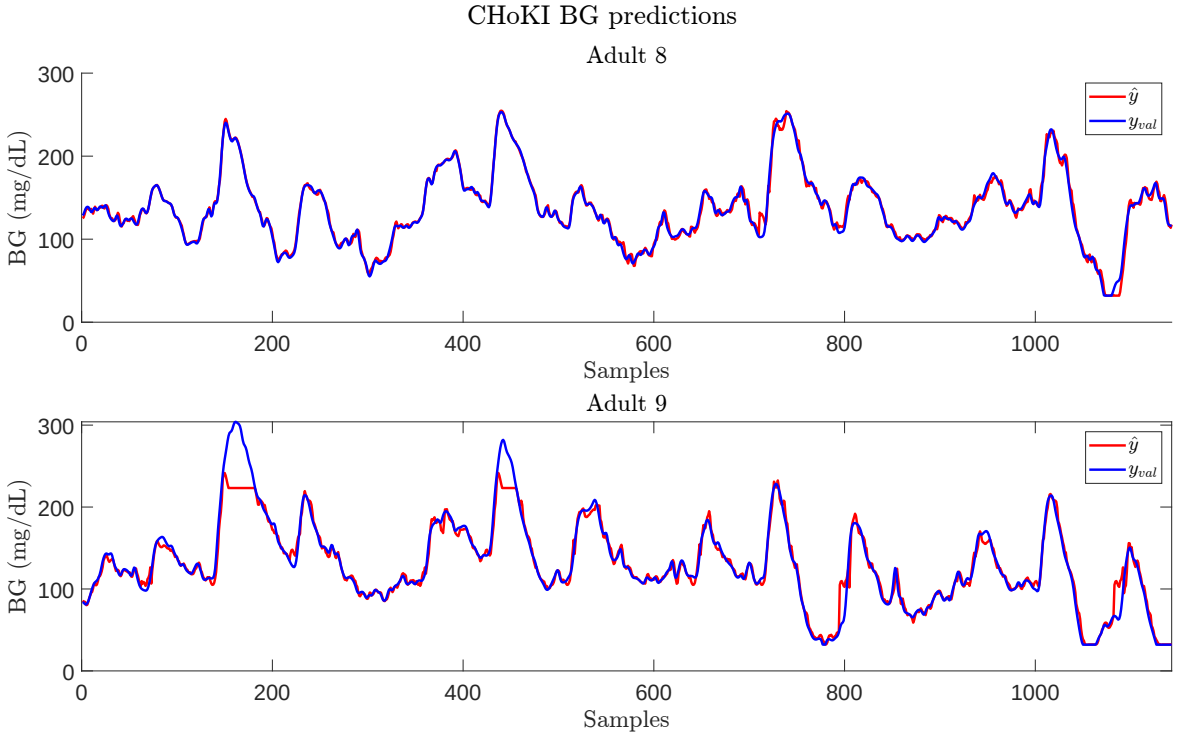


FIGURE 3.2: The upper graph represents the best case (Adult 8), while the lower one represents the worst case (Adult 9). The red lines are for the BG predictions $\hat{y}$ made with the CHoKI approach, and in blue the measured values y_{val} in the validation dataset.

Adult	u_{ref} (pmol)	$N_{\mathcal{D}}$	L (LACKI)	$[L_a; L_b; L_c]$ (CHoKI)	μ (95^{th}) (mg/dL)	μ (90^{th}) (mg/dL)	N_c (95^{th})	N_c (90^{th})	ϵ	Q
# 1	122.38	4775	3.46	[0.74; 5.46; 0.29]	20.37	14.83	2	2	10	1
# 2	134.89	4950	3.277	[4.89; 3.96; 0.09]	15.5	10.19	1	2	20	1
# 3	149.97	4990	3.076	[0.71; 5.45; 0.09]	15.99	9.29	2	3	10	1
# 4	95.07	4768	3.207	[0.87; 9.94; 0.13]	11.71	8.18	2	3	10	1
# 5	91.83	4156	6.563	[0.84; 5.52; 0.44]	21.13	13.91	2	2	5	1
# 6	190.22	5339	3.405	[4.72; 3.52; 0.09]	16.8	11.27	1	1	1	1
# 7	124.92	4803	4.096	[9.80; 0.9; 1.39]	22.84	11.89	1	1	10	1000
# 8	105.83	4703	2.582	[1.08; 5.84; 0.09]	11.13	7.8	2	3	1	100
# 9	94.59	3976	3.72	[1.13; 4.09; 0.09]	16.91	11.63	2	2	1	100
# 10	124.86	4966	3.294	[3; 2; 0.09]	16.02	10.1	1	1	20	1

TABLE 3.2: MPC settings

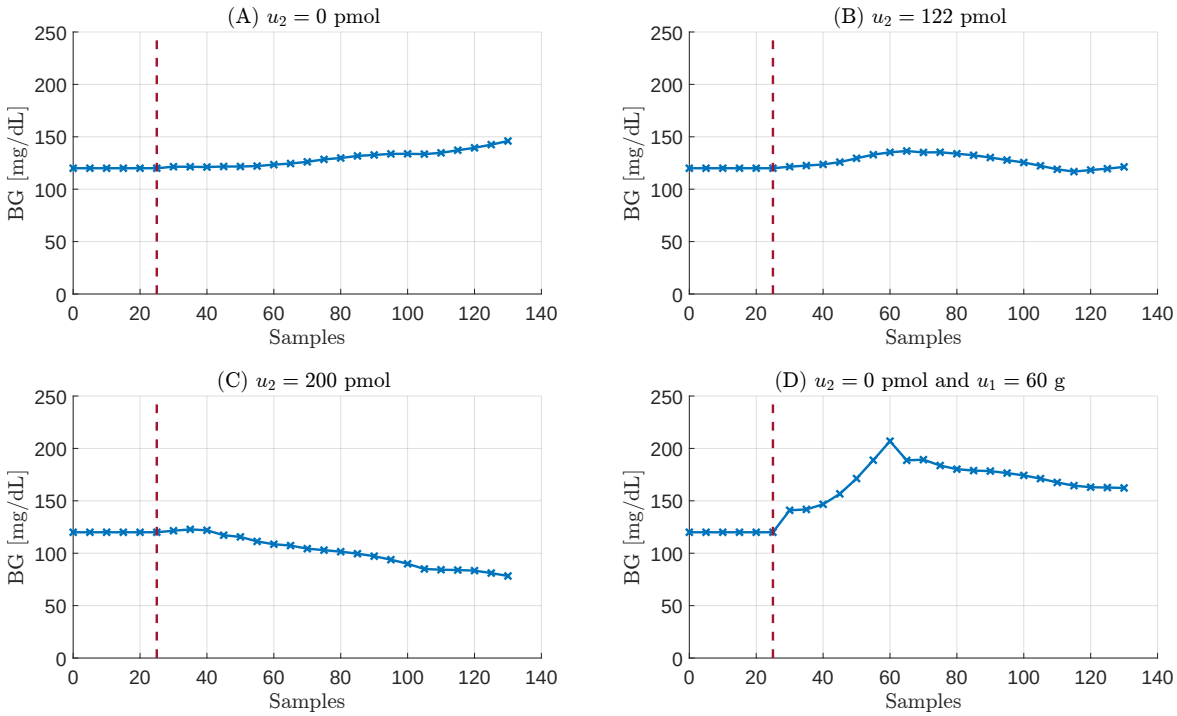


FIGURE 3.3: Example of the testing the CHoKI for Adult 1. The blue lines represent the BG predicted trends. Before the red dotted line there is BG value of the fixed regressor w_{fix}, while after the BG CHoKI predictions start. As input, in (A) there is no insulin, in (B) the basal insulin is equal to the $u_{\text{ref}} = 122$ pmol for the displayed subject, in (C) giving higher basal insulin (i.e., $u_2 = 200$ pmol), and (D) with a meal of 60 g of carbohydrates and no insulin.

Remark 3.1

To deal with the infeasibility of possible solutions outside the 90^{th} *or* 95^{th} *percentile region, some slack variables* $\boldsymbol{\delta} = \{\delta_{\min}, \delta_{\max}\}$ *are added to the optimization problem. Therefore the constraints on the glucose become* $\hat{y}(j|k) \in \mathcal{Y}_{j,\boldsymbol{\delta}}$, $\forall j \in \mathbb{I}_1^N$, *with*

$$\mathcal{Y}_{j,\boldsymbol{\delta}} = \{y : y_{\min}(j) - \delta_{\min}(j) \leq y \leq y_{\max}(j) + \delta_{\max}(j)\}, \tag{3.6}$$

where $y_{\min}$ *and* $y_{\max}$ *are the extreme values of* $\mathcal{Y}_j$ *from* (2.33).

As described in Section 2.3.1, the tightened constraints $\mathcal{Y}_i$ are obtained only once and offline, by computing the reachable sets $\mathcal{R}_j$. Then, the maximum prediction horizon N_p is such that Assumption 2.6 holds. Thus, in this Section, for each in-silico patient, the N_p is computed. This is done by considering the maximum possible value that makes it possible to have a non-empty set of constraints $\mathcal{Y}_{N_p}$, but it also takes into account the need to have reasonable ranges according to the system under control.

A possible drawback of this CHoKI technique is that the constraints could shrink fast, causing a short horizon, and being too conservative. Therefore, a prediction horizon N_p longer than the control horizon N_c is considered to increase the domain of attraction and the predictive ability of the controller, thus $N_p > N_c$. Specifically, the prediction horizon is the length of time for which the MPC computes the best control output (using the model to make predictions); while the control horizon (which is a portion of N_p) is the length of time for which the control signal is computed.

In order to apply this approach, a local control law for making the predictions from N_c to N_p must be established. The chosen one is the following:

$$u = K_{LQR}(\overline{x} - x) + \overline{u}, \tag{3.7}$$

with $u = (u_1, u_2)$ and where $K_{LQR} \in \mathbb{R}^{n_u \times n_x}$ is the control gain of a Linear Quadratic Regulator (LQR) and $(\overline{x}, \overline{u})$ is an equilibrium point around which the system $\hat{F}(x, u)$ is linearized. In particular, $\overline{x}$ is constructed as per (3.2), using $\overline{y} = 120\,\text{mg/dL}$ of glucose, and $\overline{u} = (0, u_{\text{ref}})$. Matrices $A \in \mathbb{R}^{n_x \times n_x}$ and $B \in \mathbb{R}^{n_x \times n_u}$ of the linearized model $x(k+1) = Ax(k) + Bu(k)$, are calculated numerically from the input-output data using the CHoKI model. In this way, each element $A(j,i)$ and $B(j,i)$ is obtained

by considering that

$$A(j,i) = \frac{\partial \hat{F}_j}{\partial x_i} = \frac{\hat{F}_j(\overline{x}_i + \epsilon) - \hat{F}_j(\overline{x}_i - \epsilon)}{2\epsilon}, \quad B(j,i) = \frac{\partial \hat{F}_j}{\partial u_i}, \tag{3.8}$$

where ϵ can be different for each subject (see Table 3.2). Note that

$$A(1,1) = \frac{\partial y_{k+1}}{\partial y_k}, \quad B(1,1) = \frac{\partial y_{k+1}}{\partial u_{1,k}}.$$

Linear Quadratic Regulator The Linear Quadratic Regulator (LQR) is an infinite horizon, unconstrained, optimal control technique used to minimize a quadratic cost function [15]. Considering a linear system $x(k+1) = Ax(k) + Bu(k)$ with a quadratic cost function, with weights Q, R, defined as

$$V(k+1) = V(k) - \frac{1}{2}\left(x(k)'Qx(k) + u(k)'Ru(k)\right). \tag{3.9}$$

An optimal control sequence K_{LQR} can be defined, such that:

$$u(k) = -K_{LQR}x(k), \tag{3.10}$$

where

$$\begin{aligned} K_{LQR} &= -(R + B'PB)^{-1}B'PA \\ P &= (A + BK_{LQR})'P(A + BK_{LQR}) + K'_{LQR}RK_{LQR} + Q. \end{aligned} \tag{3.11}$$

Thus, $u = K_{LQR}(\overline{x} - x) + \overline{u}$ is the optimal control law of the unconstrained LQR, where $(\overline{x}, \overline{u})$ is the equilibrium point.

3.2.4 Optimization problem

The MPC optimization problem is set as follows:

$$\min_{u_2, y_a, \delta_{\text{hyper}}, \delta_{\text{hypo}}, \boldsymbol{\delta}} V_N(\hat{x}, u; \Theta, \mathcal{D}) \tag{3.12a}$$

s.t.
$$\hat{x}(0|k) = x(k) \tag{3.12b}$$
$$\hat{x}(j+1|k) = \hat{F}(\hat{x}(j|k), u_1(j), u_2(j)), \;\; j \in \mathbb{I}_0^{N_c-1} \tag{3.12c}$$
$$\hat{x}(j+1|k) = \hat{F}(\hat{x}(j|k), K(\bar{x} - x(j)) + \bar{u}), \;\; j \in \mathbb{I}_{N_c}^{N_p-1} \tag{3.12d}$$
$$\hat{y}(j|k) = M\hat{x}(j|k), \; u_2(j) \in \mathcal{U}, \;\; j \in \mathbb{I}_0^{N_p-1} \tag{3.12e}$$
$$\hat{y}(j|k) \in \mathcal{Y}_{j,\boldsymbol{\delta}}, \;\; j \in \mathbb{I}_0^{N_c-1} \tag{3.12f}$$

$$\hat{y}(j|k) \in \mathcal{Y}_{N_c,\boldsymbol{\delta}}, \;\; j \in \mathbb{I}_{N_c}^{N_p-1} \tag{3.12g}$$

$$u_1(j) = 0, \;\; j \in \mathbb{I}_1^{N_p-1} \tag{3.12h}$$

$$70 - \delta_{\text{hypo}} \leq y_a \leq 140 + \delta_{\text{hyper}} \tag{3.12i}$$

$$\delta_{\text{hyper}} \geq 0, \; \delta_{\text{hypo}} \geq 0 \tag{3.12j}$$

$$\delta_{\min}(j) \geq 0, \; \delta_{\max}(j) \geq 0, \;\; j \in \mathbb{I}_0^{N_p-1} \tag{3.12k}$$

where (3.12h) is used since the meals are not predictable, and $\mathcal{Y}_{j,\boldsymbol{\delta}}$ comes from (3.6). The tightened constraints are computed as explained in the previous section, for all the subjects. This tightening implicitly defines the length of the control horizon N_c, which may be different for each virtual patient (divided for the 95$^{\text{th}}$ and 90$^{\text{th}}$ percentiles cases, see Table 3.2). As for the prediction horizon, it is set to $N_p = 12$ for all subjects, determining 60 minutes of predictions.

The cost functional $V_N(\hat{x}, u; \Theta, \mathcal{D})$ is constructed as the sum of different cost functions:

$$V_N(\hat{x}, u; \Theta, \mathcal{D}) = V_{N_c} + V_{N_p} + V_s + \lambda V_P + V_\delta, \tag{3.13}$$

which are now briefly detailed.

The first term V_{N_c} is given by the summation of the stage cost along the control horizon N_c:

$$V_{N_c} = \sum_{j=0}^{N_c-1} \|\hat{y}(j|k) - y_a\|_Q^2 + \|u_2(j) - u_{\text{ref}}\|_R^2, \tag{3.14}$$

where the insulin reference value u_{ref} is the constant basal insulin value for the chosen virtual patient of the UVA/Padova simulator. Similarly, the cost from N_c to the prediction horizon $N_p - 1$ is given by

$$V_{N_p} = \sum_{j=N_c}^{N_p-1} \|\hat{y}(j|k) - y_a\|_Q^2. \tag{3.15}$$

The set-point y_a is given by an auxiliary optimization variable, constrained to belong to the interval $[70, 140]$ and necessary for the implementation of the MPC in a zone control fashion. In addition, some slack variables δ_{hypo} and δ_{hyper} are added to the previous constraint, leading to the stationary cost given by

$$V_s = p_{\text{hyper}}\delta_{\text{hyper}}^2 + p_{\text{hypo}}\delta_{\text{hypo}}^2. \tag{3.16}$$

Such a cost is built in an asymmetric fashion, by taking the constant weights such

that $p_{\text{hypo}} > p_{\text{hyper}}$, to represent the fact that hypoglycemia is more dangerous than hyperglycemia [2].

The terminal cost V_P is used to guarantee the MPC stability and to penalize the difference between the last state $\hat{x}(N_p|k)$ and the reference state x_{ref}. It is weighted by a factor $\lambda > 0$, since no terminal constraint is considered [72]. It is defined as follows:

$$V_P = \|\hat{x}(N_p|k) - x_{\text{ref}}\|_P^2, \tag{3.17}$$

where P is the solution to the Riccati equation, given the LQR control gain K, the reference state x_{ref} contains the set point y_a as BG, no meals and u_{ref} as insulin.

The cost V_δ is added, to penalize the slack optimization variables $\boldsymbol{\delta} = \{\delta_{\min}, \delta_{\max}\}$, added in the constraints (3.6):

$$V_\delta = \sum_{j=1}^{N_P} \left(\delta_{\min}(j)^2 p_{\min} + \delta_{\max}(j)^2 p_{\max}\right). \tag{3.18}$$

The chosen weights are: $R = 10$, $p_{\text{hypo}} = 1 \cdot 10^7$, $p_{\text{hyper}} = 1 \cdot 10^6$, $p_{\min} = 1 \cdot 10^7$, $p_{\max} = 1 \cdot 10^6$, $\lambda = 10$ and P comes from the solution of the LQR for the linearized system around the reference point, to guarantee stability. Q are reported in Table 3.2. Note that in the case of R being greater than Q, it means having a more conservative controller.

Simulation results

The MPC optimization problem in (3.12) was tested as a control algorithm on the virtual adult T1DM patients of the UVA/Padova simulator.

Simulations with 3 meals per day. To evaluate its performance, three-day simulations were designed, with the same noise setting as in the data collection, and with three meals a day (40 g of carbohydrates at 06:00, 100 g at 12:00, and 60 g at 19:00, with a duration of 15 min) and the relative boluses, whose amount is computed by the bolus calculator of the simulator, and given 20 min after the meal starts. The results were published in [108].

Remark 3.2

The 90^{th} *and* 95^{th} *percentile cases mean that the value representing the* 90^{th} *and* 95^{th} *percentiles of the uncertainty distribution is used as* c_1 *in* (2.34)*, instead of the maximum error.*

The results of the simulations of the analyzed virtual patients are displayed in Figure 3.4. In particular, the two upper graphs show the BG trends and the basal insulin injections computed by the proposed personalized MPC for the 90^{th} percentile case. While the two lower graphs report the BG and basal insulin values for the 95^{th} percentile case. The BG values are mainly inside the euglycemic range (that is, between 70 and 180 mg/dL, the green zone in the figure), except for some peaks caused by the increase in glucose due to carbohydrate ingestion, especially for the virtual patient Adult 7. The main goal was to reduce hypoglycemic events due to their short-term danger. Such a result can be seen to have been achieved, except for Adult 7, who experienced a hypoglycemic peak before dinner on the third day in both cases (and an additional one on the first day in the 95^{th} percentile case).

Important tools for evaluating AP performance are CVGA and GRI (detailed in Section 1.3.4), which are reported in Figure 3.5(A) and Figure 3.5(B), respectively. Each dot on these graphs represents a specific subject in the 90^{th} percentile case, while the squares represent the 95^{th} percentile case. The CVGA results, with the minimum BG value as x-coordinate and the maximum BG value as y-coordinate, show that the worst-condition cases are mainly inside the safe zones. Some exceptions are in the C zone: Adults 6 and 9 for the 90^{th} and 95^{th} percentile cases, and Adult 10 for 95^{th} case. The problems with controlling Adult 7 are also shown here, since it is located in Zone D. This is also confirmed by examining the GRI results, since the Adult 7 is in the Zone C, and Adults 8 and 9 are in the Zone B. As all points lie on the y-axis, meaning that the greatest risk factor is the hyperglycemic one. This emphasises that the main objective of avoiding hypoglycemia is met. Once again, the position of Adult 7 highlights what has just been said. Patients Adults 7,8 and 9 are the ones that have more hyperglycemic issues, even though the Q inside the cost function (i.e., in (3.14)) are higher, trying to reach the glucose target y_a faster.

Finally, the TIRs can also be assessed to represent the percentages of time a patient spends in each specific BG range (detailed in Section 1.3.4). The results are displayed in Figure 3.5(C) and detailed in Table 3.3, where, for each subject, the bar on the left is for the 90^{th} percentile, while the bar on the right is for the 95^{th} percentile. With

Adult	90^{th} case					95^{th} case				
	< 54 mg/dL	54-70 mg/dL	70-180 mg/dL	180-250 mg/dL	> 250 mg/dL	< 54 mg/dL	54-70 mg/dL	70-180 mg/dL	180-250 mg/dL	> 250 mg/dL
# 1	0%	0%	81%	17%	2%	0%	0%	78%	19%	2%
# 2	0%	0%	89%	11%	0%	0%	0%	91%	9%	0%
# 3	0%	0%	78%	22%	0%	0%	0%	84%	16%	0%
# 4	0%	0%	80%	17%	3%	0%	0%	83%	16%	1%
# 5	0%	0%	83%	15%	2%	0%	0%	80%	18%	2%
# 6	0%	0%	87%	10%	2%	0%	3%	85%	10%	2%
# 7	2%	1%	62%	26%	9%	2%	2%	64%	24%	8%
# 8	0%	0%	63%	30%	7%	0%	0%	61%	32%	7%
# 9	0%	0%	64%	28%	8%	0%	0%	68%	24%	8%
# 10	0%	0%	95%	5%	0%	0%	0%	95%	5%	0%

TABLE 3.3: TIRs percentages for the 90^{th} and 95^{th} cases

the proposed controllers, the TIRs ADA requirements are mostly satisfied, since the subjects never enter into the hypoglycemic ranges (except for the virtual Adult 6 in the 95^{th} percentile case, but still satisfying the TIRs specifications) and they generally stay between 70 and 180 mg/dL for more than 70% of the simulation time. A slight exception occurs for subjects Adults 8 and 9, who are a bit under 70% and thus also a bit higher in the two hyperglycemia ranges. Another exception is for Adult 7, due to higher times in both the hypoglycemia and hyperglycemia ranges, as expected from the previous analyses.

The obtained results are promising, especially if compared to the ones obtained through the standard therapy provided by the simulator, with constant and continuous basal insulin injection (u_{ref}) and with the same simulation settings. This is visible looking at the CVGA represented in Figure 3.6(A), where, for example, Adults 5 and 9 are in the Lower D zone, while with the proposed CHoKI-based MPC controller, the same subjects are in the Upper B zone and on the border between Upper B and Upper C zones (Figure 3.5(A)), respectively. The same improvements can also be seen by comparing with the GRI results, displayed in Figure 3.6(B), and with the TIRs results, displayed in Figure 3.6(C) and detailed in Table 3.4.

The outcomes for the 90^{th} and 95^{th} percentiles cases are quite similar, but considering the less conservative case with the 90^{th} error percentile, it allows for increasing the prediction horizon N_c slightly. Looking at the results, the greater complexity and variability of Adult 7 make it more difficult to find an accurate controller, thus requiring additional analysis.

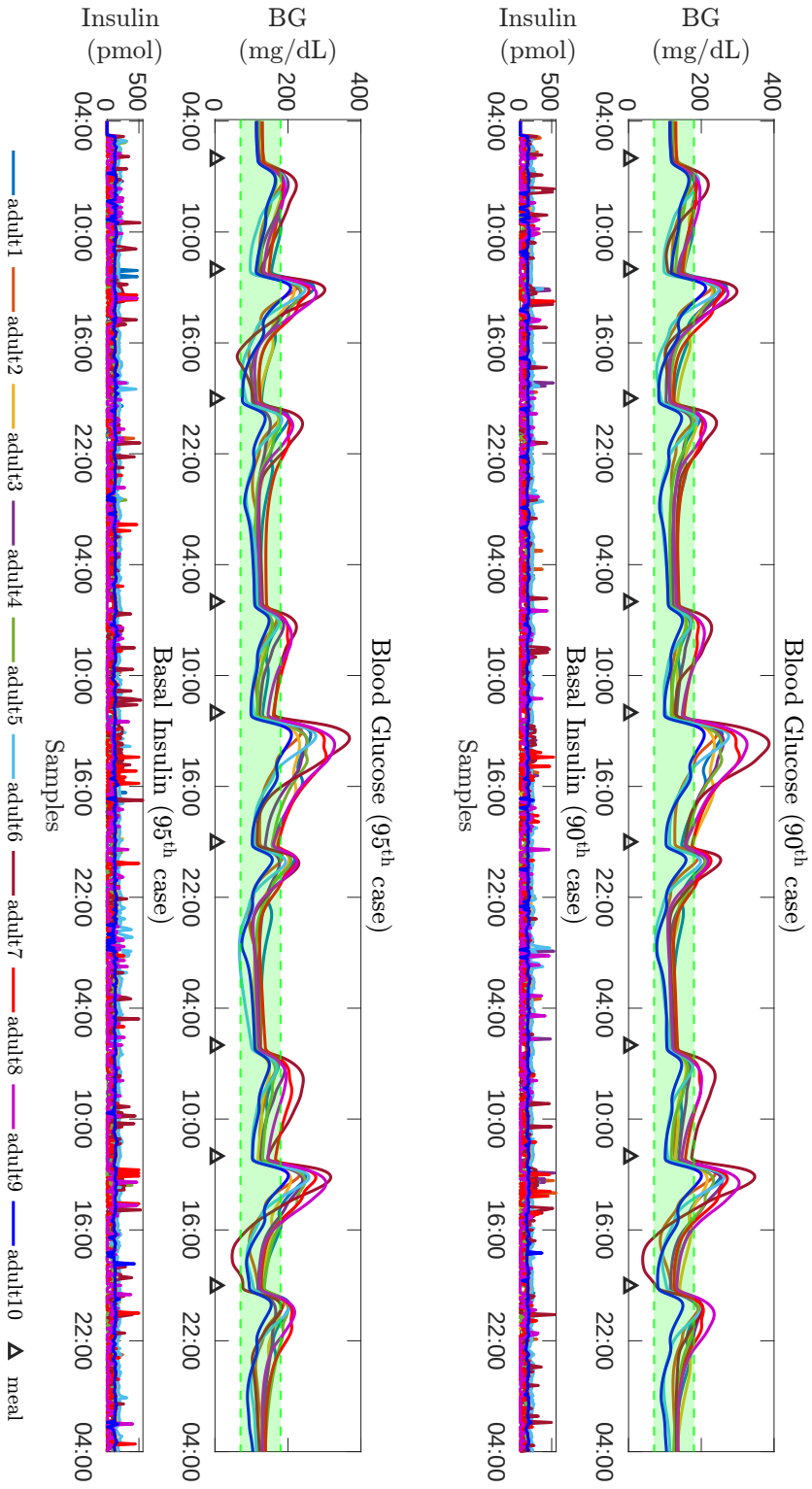


FIGURE 3.4: Upper plots: the first one represents the BG trends of all patients, in the 90th case. The green zone represents the safe range. The second one represents the basal insulin injections computed by the proposed MPC. Lower plots: the BG and basal insulin values for the 95th case.

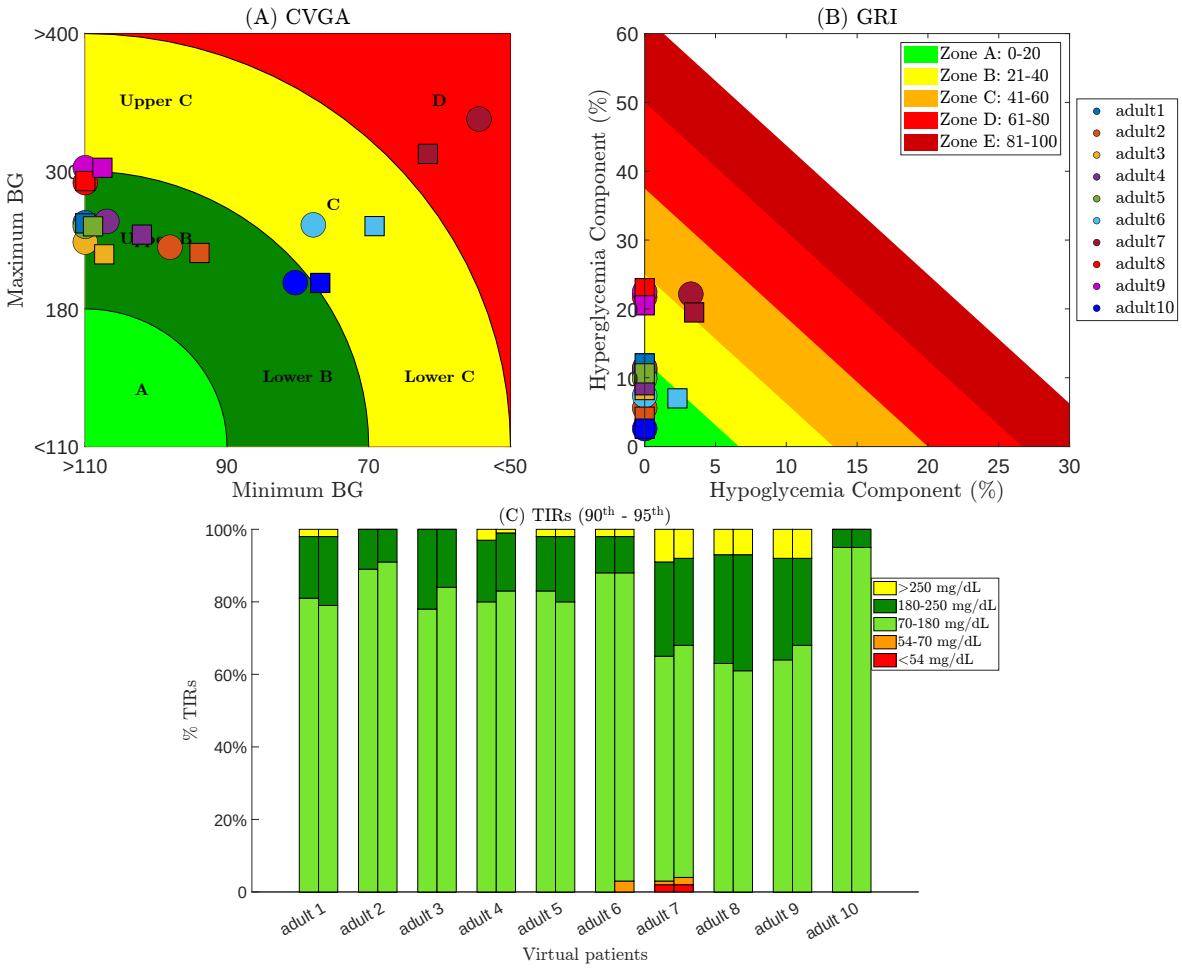


FIGURE 3.5: Results of the proposed controllers application to the simulator. (A) and (B) shows the CVGA and GRI results, respectively, where the dots are for the 90^{th} case, and the squares for the 95^{th} case. (C) shows the TIRs results, where the left columns are the 90^{th} case, and the right columns are the 95^{th} case.

Adult	<54 mg/dL	54-70 mg/dL	70-180 mg/dL	180-250 mg/dL	>250 mg/dL
# 1	0%	0%	92%	8%	0%
# 2	0%	2%	91%	7%	0%
# 3	0%	0%	94%	6%	0%
# 4	0%	2%	86%	12%	0%
# 5	14%	7%	75%	4%	0%
# 6	0%	2%	87%	10%	1%
# 7	18%	4%	64%	12%	2%
# 8	0%	0%	77%	18%	5%
# 9	6%	7%	78%	8%	1%
# 10	0%	0%	91%	9%	0%

TABLE 3.4: TIRs percentages of the constant basal insulin therapy provided by the simulator.

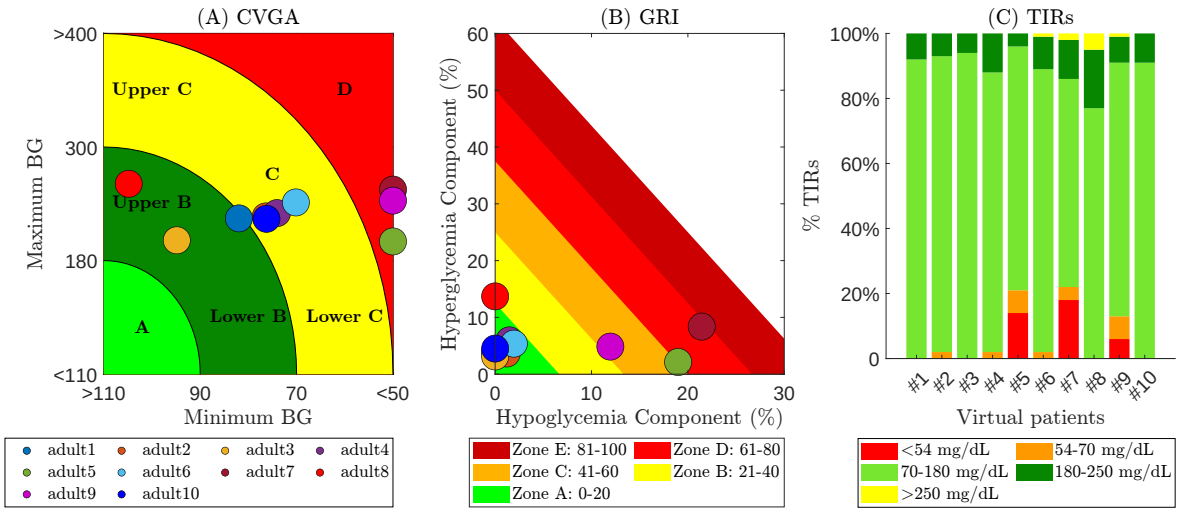


FIGURE 3.6: Results of the constant basal insulin therapy provided by the simulator, where each dot represents a specific virtual patient of the simulator. (A) represent the CVGA, (B) represents the GRI, and (D) shows the TIRs.

Simulations with different meals per day, considering the 90^{th} percentile. As visible in the previous results, the outcomes considering the 90^{th} percentile allow for a small enlargement of the control horizon. Therefore, in this section, the controller with 90^{th} percentile for the constraints computation is also tested with higher variability in the meals. Specifically, three days were simulated, with the following three meals per day: (i) 20 g at 06:00, 90 g at 12:00, and 30 g at 19:00 for the first day; (ii) 30 g at 07:00, 80 g at 12:30, and 50 g at 20:00 for the second day; and (iii) 40 g at 06:30, 100 g at 13:00, and 60 g at 19:30 for the last day. The meals were announced, and the postprandial boluses were computed by the bolus calculator of the simulator and injected 20 min after the meal's start. All the devices have the same noise setting as in the data acquisition stage, including the carbohydrate estimation error with a normal distribution with a standard deviation equal to 30% of the meal amount, for the bolus computation. The results were published in [110].

The BG results coming from the basal insulin injections computed by the proposed controller are shown in Figure 3.7. As for the previous case, the BG values are mainly inside the euglycemic range, and they increase after the meals (i.e., depicted as the black triangles), reaching also 300 mg/dL, which makes sense considering the delay in the bolus injection. Adult 7 is the most complicated one to be controlled, as visible from the presence of both hyper- and hypoglycemic events. The CVGA and GRI results are reported in Figure 3.8. The TIRs results are detailed in Table 3.5. These confirm the issues in the BG management for Adult 7, which is in the zone D for the

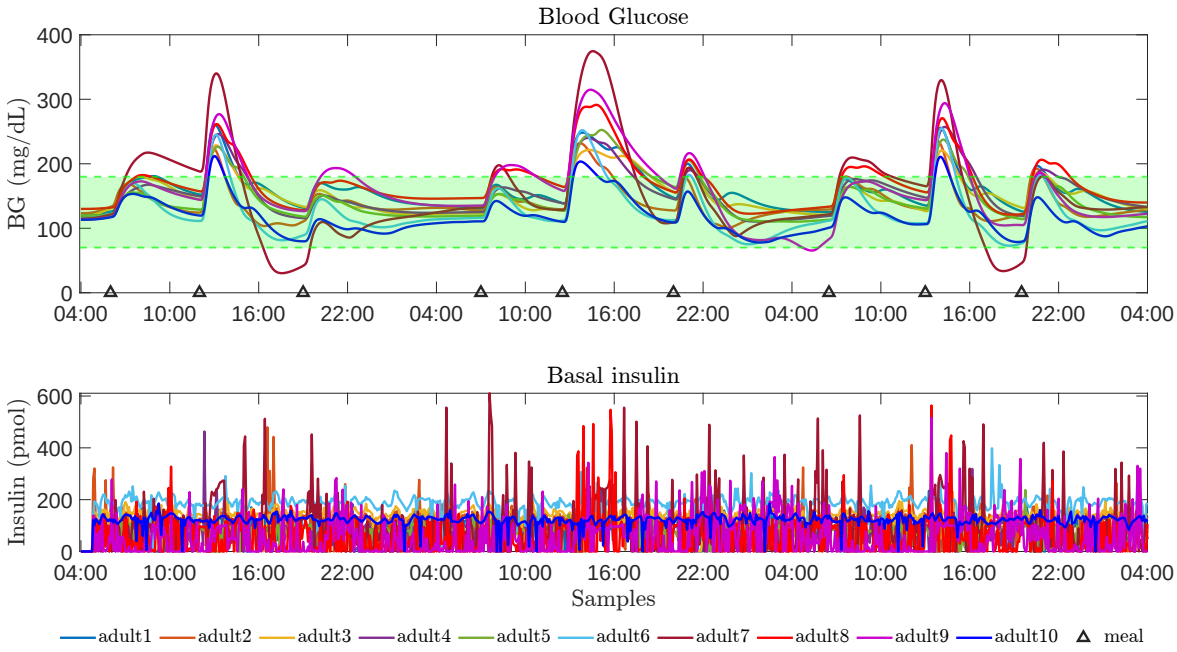


FIGURE 3.7: BG and basal insulin results of the 3-day simulation (with different meals). The upper plot displays BG trends for all patients. The green zone represents the safe range, and the black triangles depict meals. The lower plot shows basal insulin injections computed by the proposed MPC.

CVGA and in the zone C for the GRI (with a value of 53.87). Looking at Figure 3.8(B), excluding Adult 7, the other in-silico patients are in the optimal zone A (except for Adult 8 and 9, who are in the zone B), and lie on the y-axis, which means that the higher risk component in the hyperglycemic one, which was the main goal. While in Figure 3.8(A), the subjects are in zone B (except for Adults 6 and 9, who are in zone C, since they went close to 70 mg/dL and also have some hyperglycemic peaks). Moreover, here the majority of the patients lie on the y-axis, showing the absence of hypoglycemic issues.

3.3 Including Insulin On Board in the controller

Starting from the optimization problem (3.12), an additional constraint on the maximum insulin amount was added.

The Insulin On Board (IOB) represents the current insulin still active in the body, whose amount is influenced by the patient's metabolism and by the Insulin Action

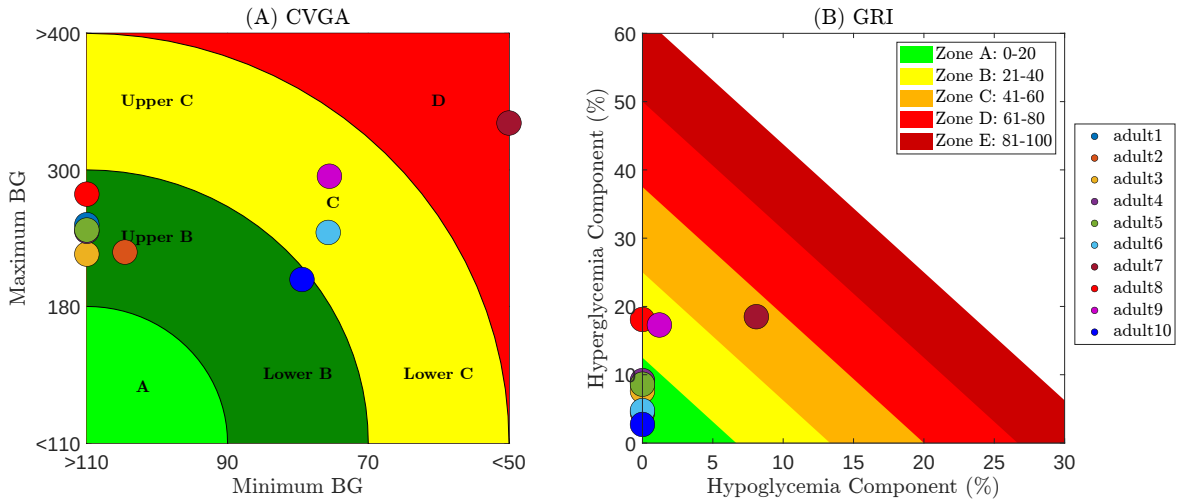


FIGURE 3.8: CVGA and GRI results of the 3-day simulation (with different meals).

Adult	< 54 mg/dL	54-70 mg/dL	70-180 mg/dL	180-250 mg/dL	> 250 mg/dL
# 1	0%	0%	84%	15%	1%
# 2	0%	0%	91%	9%	0%
# 3	0%	0%	85%	15%	0%
# 4	0%	0%	83%	16%	1%
# 5	0%	0%	83%	16%	1%
# 6	0%	0%	91%	8%	1%
# 7	7%	1%	64%	19%	9%
# 8	0%	0%	70%	24%	6%
# 9	0%	2%	72%	18%	8%
# 10	0%	0%	95%	5%	0%

TABLE 3.5: TIRs percentages of the 3-day simulation (with different meals).

Duration (IAD). This dynamic value can be considered as an upper constraint in the MPC optimization problem, to limit the maximum value of basal insulin. This is done with the aim of reducing the risk of hypoglycemic events, since the computation of the basal insulin correction amounts now also takes into consideration the insulin still active from the previous injections.

The IOB cannot be measured. Thus, at each sampling time k, it can be estimated by analysing the residuals of the previous bolus insulin administrations, which means having:

$$\mathrm{IOB}(k) = \sum_{i=1}^{n_{\mathrm{IOB}}} \alpha(k-i) u_{\mathrm{bol}}(k-i), \tag{3.19}$$

in which the vector of weights, denoted by α, represents the insulin action curve, while u_{bol} contains the previous insulin bolus injections. n_{IOB} represents the IAD, considered with a sampling time of 5 min. In this case, looking at the data coming from the UVA/Padova simulator, it is chosen to be $n_{\mathrm{IOB}} = 72$, which means to have a duration of 6 h [70].

The value of the upper limit for the calculation of the basal dose is $u_2^{\max}$, and it is derived from

$$u_2^{\max}(k+j) = \begin{cases} u_2^{\lim} - \mathrm{IOB}(k+j) & \text{if } u_2^{\lim} > \mathrm{IOB}(k+j) \\ u_{\mathrm{ref}} & \text{otherwise} \end{cases} \tag{3.20}$$

where k is the sampling time and $j \in \mathbb{I}_0^{N_p-1}$ [41]. The maximum amount of basal insulin that can be delivered has been set equal to $u_2^{\lim} = 500\,\mathrm{pmol}$. u_{ref} is the basal insulin reference value, provided by the standard therapy of the simulator (coming from the daily basal insulin in Table 1.3). The IOB varies at each step along the prediction horizon (i.e., with j), and thus the estimations have to decrease according to the insulin action curve, without considering possible new boluses, since we consider that the meals are unpredictable.

Thus, the new set for the basal insulin amount u_2 is

$$\mathcal{U}_2 = \{u_2 : 0 \leq u_2 \leq u_2^{\max}\}. \tag{3.21}$$

Remark 3.3

To prevent possible infeasibilities during MPC resolution, N_p slack variables δ_u were included in the optimization problem. These were added to the upper bound $u_2^{\max}$ in equation (3.21)*, obtaining, $\forall j \in \mathbb{I}_0^{N_p-1}$:*

$$\mathcal{U}_2' = \{u_2(j) : 0 \leq u_2(j) \leq u_2^{\max}(j) + \delta_u(j)\}. \tag{3.22}$$

Optimization problem including IOB

The MPC optimization problem, which also takes into consideration the IOB for the insulin constraint, is set as follows:

$$\min_{u_2, y_a, \delta_{\text{hyper}}, \delta_{\text{hypo}}, \boldsymbol{\delta}, \delta_u} V_N(\hat{x}, u; \Theta, \mathcal{D}) \tag{3.23a}$$

$$\text{s.t.} \quad \hat{x}(0|k) = x(k) \tag{3.23b}$$

$$\hat{x}(j+1|k) = \hat{F}\big(\hat{x}(j|k), u_1(j), u_2(j)\big),\ j \in \mathbb{I}_0^{N_c-1} \tag{3.23c}$$

$$\hat{x}(j+1|k) = \hat{F}\big(\hat{x}(j|k), K(\bar{x} - x(j)) + \bar{u}\big),\ j \in \mathbb{I}_{N_c}^{N_p-1} \tag{3.23d}$$

$$\hat{y}(j|k) = M\hat{x}(j|k),\ j \in \mathbb{I}_0^{N_p-1} \tag{3.23e}$$

$$u_2(j) \in \mathcal{U}_2',\ j \in \mathbb{I}_0^{N_p-1}, \tag{3.23f}$$

$$\hat{y}(j|k) \in \mathcal{Y}_{j,\boldsymbol{\delta}},\ j \in \mathbb{I}_0^{N_c-1} \tag{3.23g}$$

$$\hat{y}(j|k) \in \mathcal{Y}_{N_c,\boldsymbol{\delta}},\ j \in \mathbb{I}_{N_c}^{N_p-1} \tag{3.23h}$$

$$u_1(j) = 0,\ j \in \mathbb{I}_1^{N_p-1} \tag{3.23i}$$

$$70 - \delta_{\text{hypo}} \leq y_a \leq 140 + \delta_{\text{hyper}} \tag{3.23j}$$

$$\delta_{\text{hyper}} \geq 0,\ \delta_{\text{hypo}} \geq 0 \tag{3.23k}$$

$$\delta_{\min}(j) \geq 0,\ \delta_{\max}(j) \geq 0,\ j \in \mathbb{I}_0^{N_p-1} \tag{3.23l}$$

$$\delta_u(j) \geq 0,\ j \in \mathbb{I}_0^{N_p-1} \tag{3.23m}$$

where the differences with respect to the optimization problem in (3.12) are the following:

- the new insulin constraint in (3.23f), which takes into consideration the IOB, coming from (3.22);
- the constraint (3.23m) on the slack variables δ_u which are included in the control action constraints (3.23f), that have to be positive;

- the cost function $V_N(\hat{x}, u; \Theta, \mathcal{D})$ to be minimized, which is build as

$$V_N(\hat{x}, u; \Theta, \mathcal{D}) = V_{N_c} + V_{N_p} + V_s + \lambda V_P + V_\delta + V_u, \tag{3.24}$$

  where the additional last component V_u is the cost to penalise the slack variables δ_u:

$$V_u = \sum_{j=1}^{N_P} \|\delta_u(j)\|_{p_u}^2, \tag{3.25}$$

  with $p_u = 1 \cdot 10^7$.

The upper limit of basal insulin in (3.20) requires the IOB value. Thus, the weight vector α needs to be identified in order to obtain an accurate IOB estimation. We have tested two possibilities: a linear one and an exponential one. Then, these two estimations are compared with the measured IOB values, which can be read from the UVA/Padova simulator. For the reachable sets computation, in both cases, only the 90^{th} percentile is considered as μ, since it was the case that allows for a small increase in the prediction horizon.

3.3.1 Linear IOB estimation

For the linear IOB estimation, the weights of the insulin action curve α are obtained from $((IAD - t_{\text{bol}})/IAD)$, where $IAD = 6$ h and t_{bol} is the time passed from the previous bolus. This generates a vector of values between $[0, 1]$, to represent the decreasing weight of the bolus over time. In Figure 3.9 an example of the IOB estimation for virtual Adult 1 is reported. This shows that (3.19) approximates quite well the measured values (blue line) and the choice of IAD equal to 6 h is appropriate. The initial IOB value after a bolus could be different between the two curves. This is because the estimated IOB is based on the value of the bolus calculated for the meal; on the other hand, the measured IOB is based on the value that is actually injected, which may vary from the calculated value due to pump noise, and the noise in the meal estimation in the bolus computation.

Simulation results

The proposed customized optimization problem in (3.23), with the linear IOB estimation, was tested on the virtual adult T1DM patients of the simulator. These results are reported in [110].

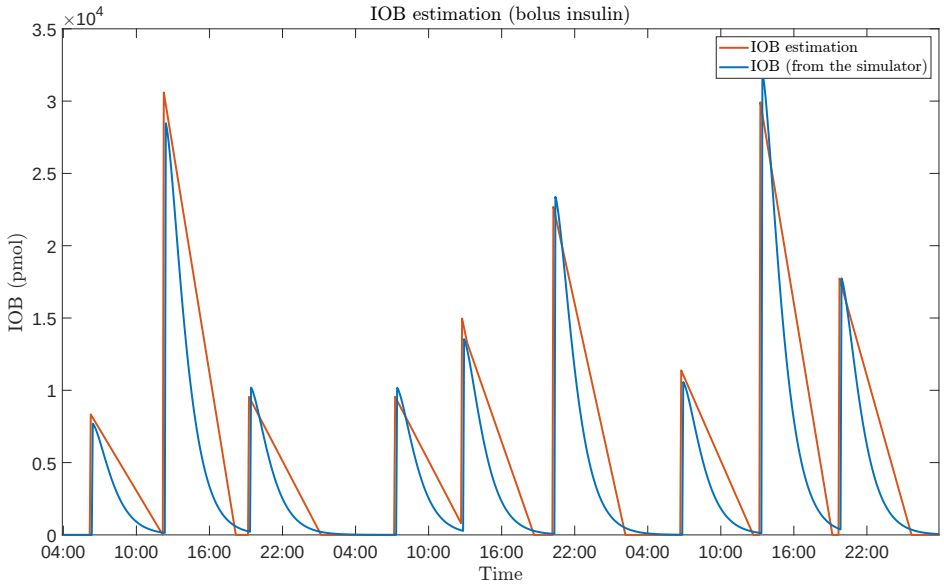


FIGURE 3.9: The orange line is for the estimated IOB, while the blue line is the measured IOB of patient Adult 1, coming from the simulator.

Three days were simulated, that are set as in the previous case (i.e., as the one reported in Figure 3.7), with the same meals, bolus settings, and noise definition.

The results of the simulations for all patients are shown in Figure 3.10. The top graph displays the BG trends caused by the insulin injections depicted in the bottom graph, which vary based on the patients' CHoKI parameters and datasets. The primary objective is to reduce the frequency and the severity of hypoglycemic events, which are very dangerous in the short term, and it can be seen that such a result is achieved, as no values below the euglycemic range are visible. There are some hyperglycemic peaks, in particular for the Adult 7 (i.e., the dark red one), who goes above 300 mg/dL, with a solution that is still feasible thanks to the slack variables in the constraints. This is due to the MPC setup, which ends up being over-conservative.

An additional useful tool is the GRI, whose risk components are represented in Figure 3.11(B). Each dot on the graph represents a specific subject, and specifically, Adult 7 is in Zone D, Adults 8 and 9 are in Zone C, Adult 5 is in Zone A, and the others are in Zone B. All the subjects lie on the y-axis, this is because the proposed controller is designed to avoid hypoglycemia, which is why the only risk component is the hyperglycemic one. Also the TIRs results are detailed in Table 3.6.

To have a more complete analysis, the CVGA can also be assessed. In Figure 3.11(A),

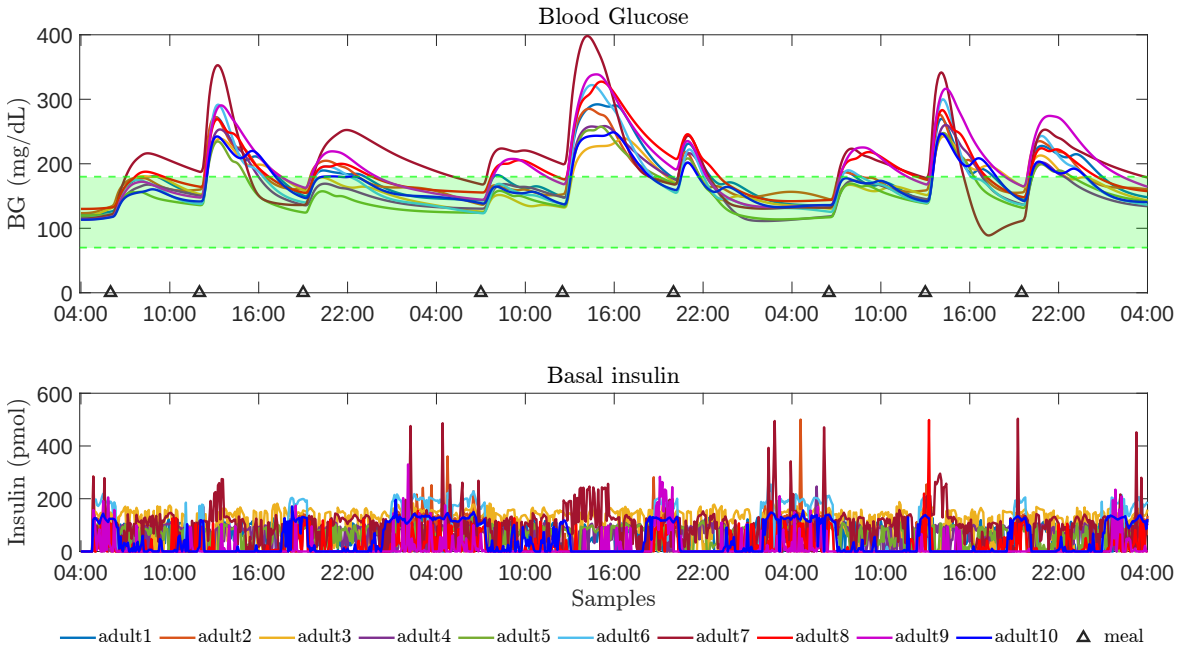


FIGURE 3.10: Proposed controller with IOB constraint (with linear IOB estimation) applied to the virtual patients. Upper plot: BG trends of all patients, with the green zone for the safe range and the black triangles for the meal times. Lower plot: basal insulin injections.

each dot on the graph describes a specific subject. This tool allows for the analysis of the worst condition, with the minimum and maximum BG value. All the subjects are in the safe zone B, except for Adults 7, 8 and 9, who are in Zone C. This demonstrates how including the IOB constraint resolves the issues related to hypoglycemic events, increasing the minimum BG value, even at the cost of also increasing the maximum value.

To make a further analysis, the results obtained are compared with those performed before, with the same simulation settings, but with the controller without the IOB constraint (i.e., the ones reported in Figure 3.7). In Figure 3.12 this comparison is shown, by considering the mean and standard deviation of the BG values of the virtual patients. The blue dotted line represents the mean values of the cases obtained with the MPC with IOB constraints and the blue area displays their standard deviations, while the cases without IOB are in red. This indicates that including the IOB in the constraints means having a more conservative controller, since the BG level is higher in the cases with the IOB safety constraints. This is due to the fact that the controller does not manage post-prandial boluses. As a result, since the bolus amount (computed by the bolus calculator of the simulator) is based only

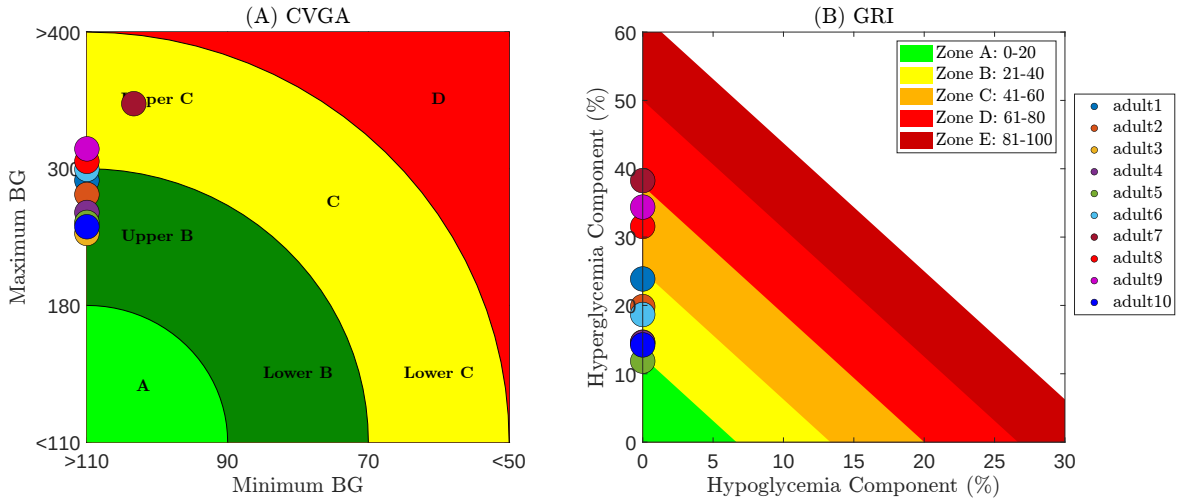


FIGURE 3.11: CVGA and GRI results of the 3-day simulation with the controller with linear IOB constraint. Each dot represents a specific patient. (A) shows the CVGA, with minimum BG values on the x-axis and maximum BG values on the y-axis. (B) shows the GRI, with the hypoglycemic risk component on the x-axis and hyperglycemic risk component on the y-axis.

Adult	<54 mg/dL	54-70 mg/dL	70-180 mg/dL	180-250 mg/dL	>250 mg/dL
# 1	0%	0%	60%	32%	8%
# 2	0%	0%	67%	26%	7%
# 3	0%	0%	71%	29%	0%
# 4	0%	0%	76%	19%	5%
# 5	0%	0%	78%	20%	2%
# 6	0%	0%	71%	21%	8%
# 7	0%	0%	35%	53%	12%
# 8	0%	0%	47%	43%	10%
# 9	0%	0%	47%	38%	15%
# 10	0%	0%	72%	28%	0%

TABLE 3.6: TIRs percentages of the 3-day simulation with the controller with linear IOB constraint.

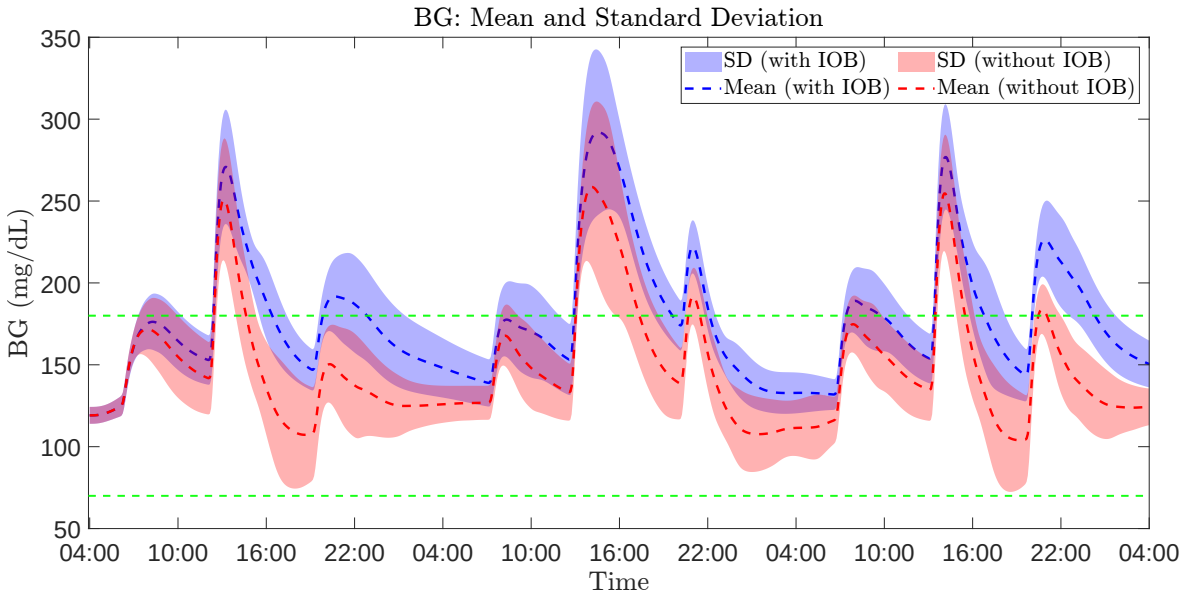


FIGURE 3.12: Comparison of the BG values: the mean and the standard deviation of the simulations performed exploiting the MPC controller with IOB constraints are represented in blue, and without them are in red.

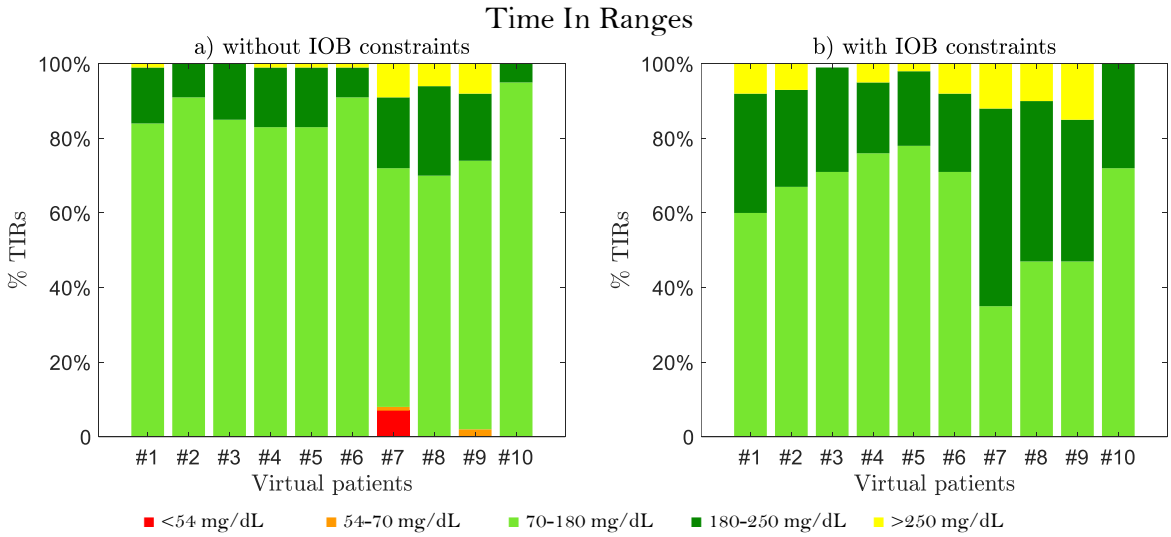


FIGURE 3.13: TIRs results of the simulations performed with (graph on the right) and without the IOB constraints (graph on the left). Each bar represents a specific subject, with the percentages of time spent in each range.

on the meal amount, even though the BG value is high at mealtime, the possible basal corrections are limited by the IOB constraint. The conservative results are a consequence of the linear function used to estimate the IOB. In fact, as shown in Figure 3.9, the IOB is always overestimated.

Another important tool for assessing AP performance is the TIRs. The proposed controller ensures that the requirements for the hypoglycemic ranges are always met, which is the main objective. However, the controller permits the subjects to stay a little longer in the two hyperglycemic ranges, which also means that they stay within the 70-180 mg/dL range for less than 70% of the simulation time. The results are displayed in Figure 3.13, where, for each subject, the graph on the left is for the cases without the IOB, while the one on the right is for the cases with the IOB constraints.

The current controller may be too conservative, especially for Adult 7, as it keeps the virtual patient in a hyperglycemic state for an extended period. Consequently, further analysis is necessary to improve control, especially considering the patient's high variability and complex response to insulin.

3.3.2 Exponential IOB estimation

The results obtained with the linear weight decreasing from 1 to 0 tested in the previous section turned out to be too conservative, due to the overestimation of the IOB values. Therefore, here the weights α of the insulin action curve (i.e., for (3.19)) are computed by exploiting the data coming from the UVA/Padova simulator. Specifically, to get the IOB decreasing behavior, the following sum of two exponential functions is considered: $\gamma_m = a_m e^{b_m \beta} + c_m e^{d_m \beta}$, where the parameters a_m, b_m, c_m, d_m have to be estimated from the data (β, γ), where β represents the time and γ is the IOB value. The estimated parameters were obtained considering the IOB curves of all the analysed patients, and using the mean value, since with real patients in not possible to read the IOB value; therefore, this average approximation is used for all. The results are: $a_m = 8.2 \cdot 10^5$, $b_m = -0.08$, $c_m = -1.3 \cdot 10^6$ and $d_m = -0.11$, visible in Figure 3.14. After that, to get the weights $\alpha \in \mathbb{R}^{n_{\text{IOB}}}$ that have to be multiplied to u_{bol}, the values γ_m are normalized between 0 and 1, to follow the real amount of the IOB behavior.

Figure 3.15 displays an example of the IOB estimation for virtual Adult 1. This shows that (3.19) approximates quite well the real values and the choice of IAD

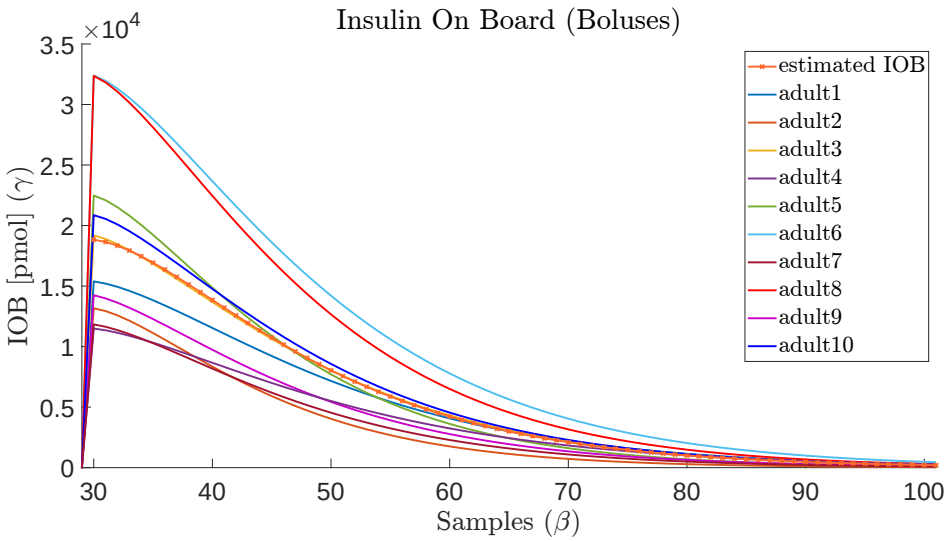


FIGURE 3.14: Each line represents the real IOB of a specific patient, coming from the simulator. The orange line is for the estimated IOB.

equal to 6 h is appropriate. After a bolus, the initial IOB value might differ between the two curves. This is because the estimated IOB value is based on the calculated value of the bolus for the meal. While the real IOB value is based on the value really injected, which may vary from the calculated value due to pump noise.

Simulation results

The optimization problem in (3.23) with the exponential IOB estimation was also tested on the virtual patients of the simulator, considering the same simulation settings as those coming from the linear IOB estimation aforementioned. These results were reported in [111].

In Figure 3.16, the upper part shows the BG trends of the virtual patients, resulting from the insulin injections reported in the lower part. The BG values mainly fall within the euglycemic range (the green zone), and they rise after the meals (indicated by black triangles). The controller was designed with the IOB constraint to avoid hypoglycemic events, and this result is achieved, even though the obtained controller ends up being a bit conservative.

To evaluate the quality of closed-loop glucose control, as in the previous case, the TIRs, CVGA, and GRI tools are considered. In Table 3.7, the TIRs results indicate that the proposed controller tends to be conservative, allowing a bit of hyperglycemia but avoiding hypoglycemia: although the percentage of time spent in

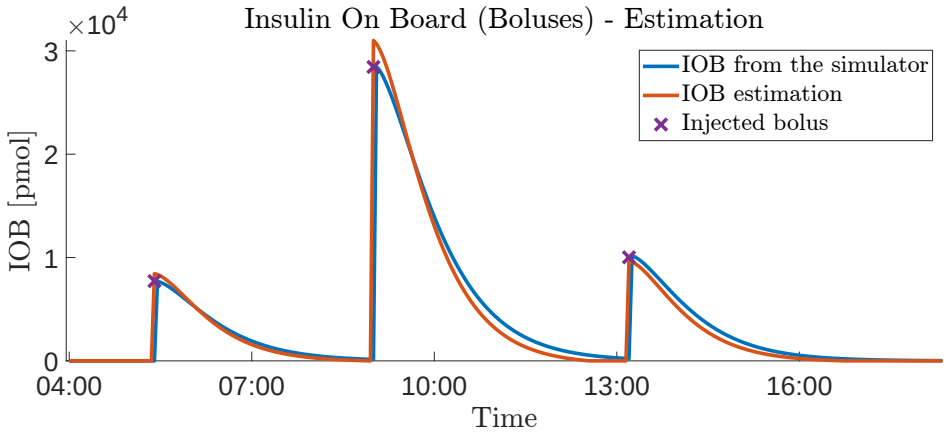


FIGURE 3.15: The orange line represents the estimated IOB, the blue one the IOB computed by the bolus calculator of the simulator, and the injected boluses are the purple crosses. This is an example of the virtual patient Adult 1.

hyperglycemic ranges is slightly higher than expected, the patients never enter the hypoglycemic ranges. As shown with the CVGA in Figure 3.17(A), most patients are inside the safe zones, except for virtual Adults 7, 8 and 9 who are in the Upper C zone due to their high BG maximum values. The GRI is shown in Figure 3.17(B), where all the dots lie on the y-axis due to the controller design: the aim is to avoid hypoglycemic events, thus the higher risk component is the hyperglycemic one.

This controller results in being too conservative for Adult 7, who remains in a hyperglycemic state for too long, even though the MPC is more aggressive due to the $Q = 1000$. Thus, additional analysis is required to enhance the control, also due to the patient's high variability and complex response to insulin.

The comparison with the previous case, with the linear weights for IOB estimation, is reported in Figure 3.18, which shows the mean and standard deviation of the BG values of the virtual subjects in both cases. The controller proposed here shows less conservative results. Figure 3.19 shows that the IOB values computed with the linear weights (green line) overestimate the real values (blue line), while the exponential weights allow better estimates of the IOB values (orange line).

The IOB additional constraint (both in the linear and in the exponential cases) decreases the performance apart from the problematic patient Adult 7. This is because the main issue in BG management was related to hypoglycemic events. Thus, obtaining a more conservative controller helps in avoiding them, even though the subject enters hyperglycemic ranges.

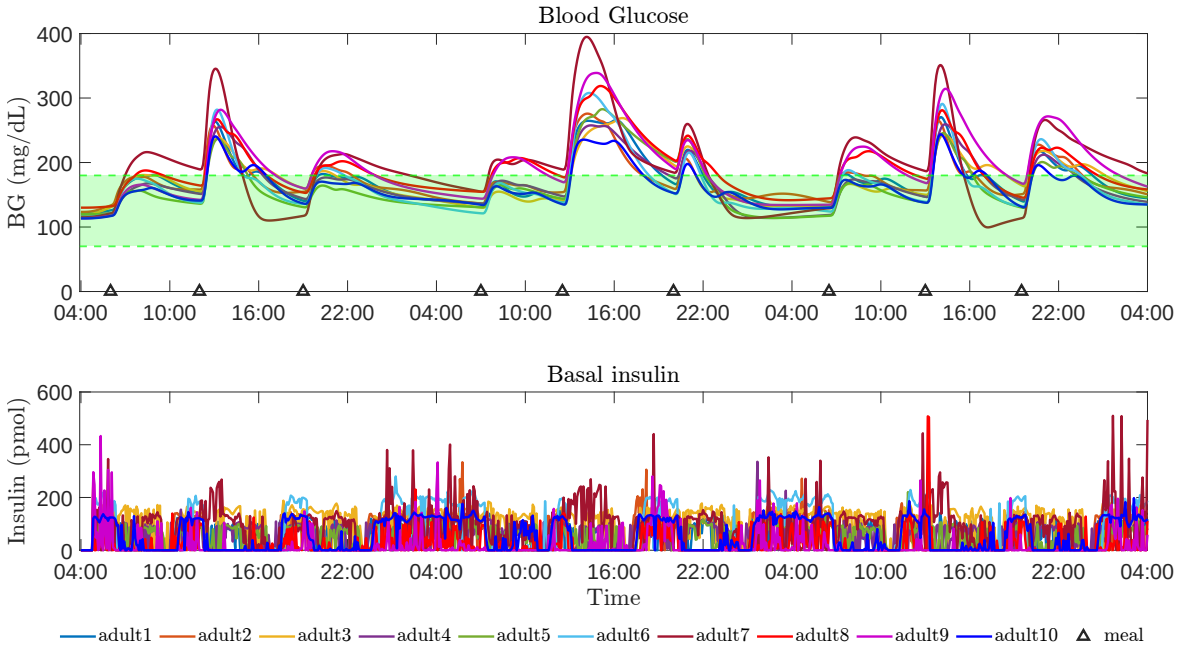


FIGURE 3.16: Proposed controller with IOB constraint (with exponential IOB estimation) applied to the virtual patients. Upper plot: BG trends of all patients, with the green zone for the safe range and the black triangles for the meal times. Lower plot: basal insulin injections.

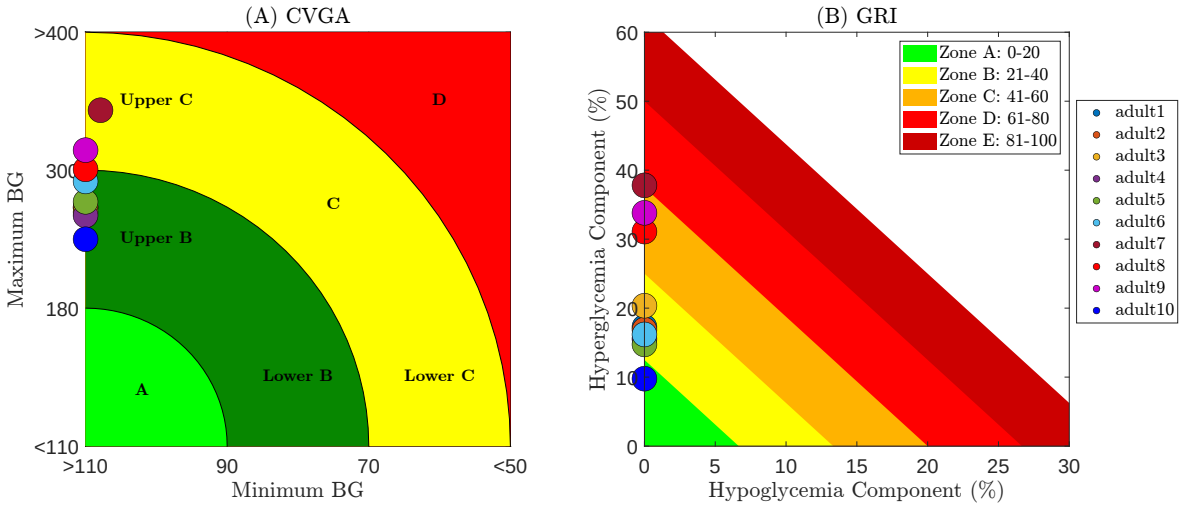


FIGURE 3.17: CVGA and GRI results of the 3-day simulation with the controller with exponential IOB constraint. Each dot represents a specific patient. (A) shows the CVGA, with minimum BG values on the x-axis and maximum BG values on the y-axis. (B) shows the GRI, with the hypoglycemic risk component on the x-axis and hyperglycemic risk component on the y-axis.

Adult	< 54 mg/dL	54-70 mg/dL	70-180 mg/dL	180-250 mg/dL	> 250 mg/dL
# 1	0%	0%	72%	21%	7%
# 2	0%	0%	71%	24%	5%
# 3	0%	0%	64%	32%	4%
# 4	0%	0%	74%	21%	5%
# 5	0%	0%	75%	20%	5%
# 6	0%	0%	75%	18%	7%
# 7	0%	0%	37%	51%	12%
# 8	0%	0%	47%	44%	9%
# 9	0%	0%	47%	39%	14%
# 10	0%	0%	80%	20%	0%

TABLE 3.7: TIRs percentages of the 3-day simulation with the controller with exponential IOB constraint.

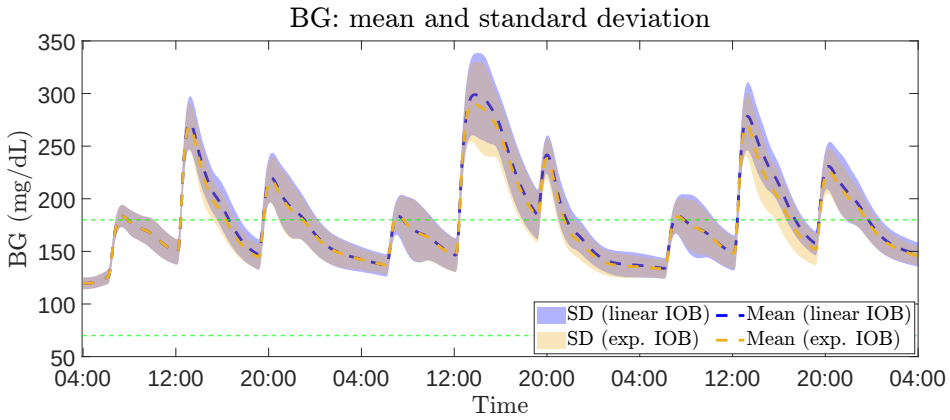


FIGURE 3.18: Comparison of the BG: the simulations with linear IOB are in blue, and the exponential ones in yellow.

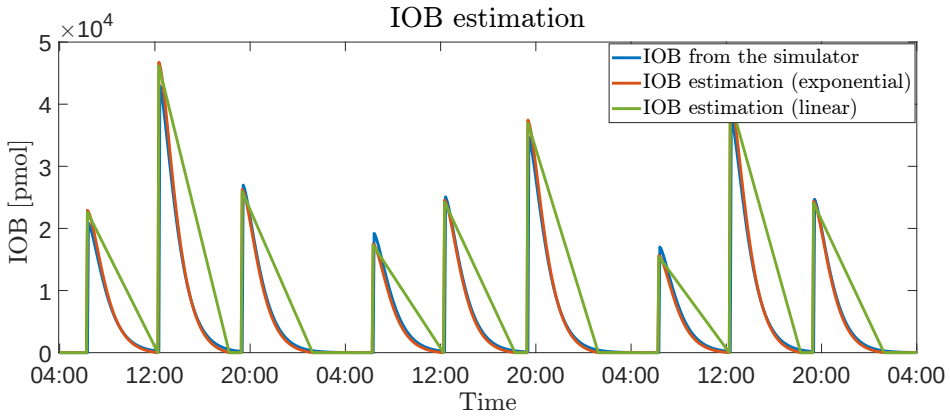


FIGURE 3.19: The estimated boluses IOB with exponential weights are represented as the orange line, the one with linear weights is the green line, and the one of the simulator is in blue (Adult 10 example).

3.4 Simulations with Physical Activity and Insulin Sensitivity variability

In this section, the optimization problem in (3.23) with linear IOB estimation is tested under more challenging conditions, including in the simulations Physical Activity (PA) sessions and Insulin Sensitivity (S_I) circadian variability. The results were published in [110].

3.4.1 Insulin Sensitivity

The proposed CHoKI-based MPC with the IOB constraints is tested on the same virtual patients, but with variations in the Insulin Sensitivity (S_I), thanks to the availability of the circadian variability option in the updated version of the UVA/Padova simulator (DMMS.R, version 1.3). S_I refers to how responsive the cells are to insulin, in terms of enhancing glucose uptake and reducing endogenous glucose production [119]. S_I (often referred as its opposite, insulin resistance) is not static, and it can vary in the subject inter- and intra-day. Factors such as circadian rhythms, physical activity, stress, or hormonal variations affect the S_I modifications [104].

In the commercial version 1.3 of the UVA/Padova simulator, the implemented S_I variability is the same for all the in-silico patients and follows a circadian variability. Specifically, it is obtained thanks to the circadian coefficient that follows the shape shown in Figure 3.20.

Simulation results

The simulations are performed with the same settings as in the previous cases (as in Section 3.3). The results of the application of the proposed controller are presented in Figure 3.21, where the upper part shows the BG trends that are obtained thanks to the insulin injections displayed in the lower graph. The cases reported here derive from the attempt to control a more complex system due to the variability of S_I, which affects the relationship between glucose, insulin, and carbohydrates. This leads to more fluctuations in glucose levels, as depicted in the upper part of Figure 3.21. This is evident when looking at Adult 7, who worsens compared to the case without variability (as shown in Figure 3.10), both because the hyperglycemia peak is higher,

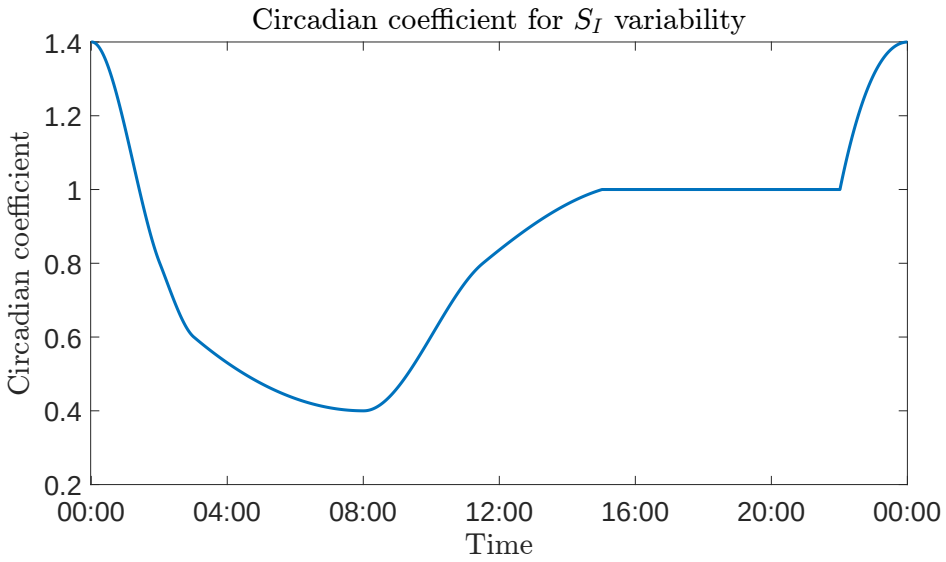


FIGURE 3.20: Circadian variation of the coefficient representing the insulin sensitivity variability in the UVA/Padova simulator.

but also due to greater fluctuations in glucose level, since the patient goes also in hypoglycemia (but > 54 mg/dL).

To better evaluate the performance of the proposed controller in managing the variations in S_I, the CVGA, GRI, and TIRs results are considered as well. Figure 3.22(A) shows the CVGA outcomes, where Adult 7 is in the D zone, Adult 5, 6 and 9 are in the Upper C zone, while the others are in the Upper B zone. Figure 3.22(B) displays the GRI values, where Adult 7 is in Zone D, while Adult 4, 5 and 6 are in Zone B, and the remaining ones are in Zone C. TIRs outcomes are reported in 3.8

These results are quite promising, since they demonstrate that the proposed controller can still prevent hypoglycemic events while remaining conservative, even in the presence of insulin variability. However, the variations in insulin sensitivity affect the ability of the CHoKI learning method, even if the outcomes are similar to the previous case, leading to hyperglycemic events (see Adult 7).

3.4.2 Physical Activity

PA has many benefits for T1DM patients, including enhancements in cardiorespiratory fitness, in body composition, in achieving glycemic goals, and in the overall quality of life. On the other hand, this is one of the many factors that affect the relationship between insulin and glucose levels, which can cause issues for the patient,

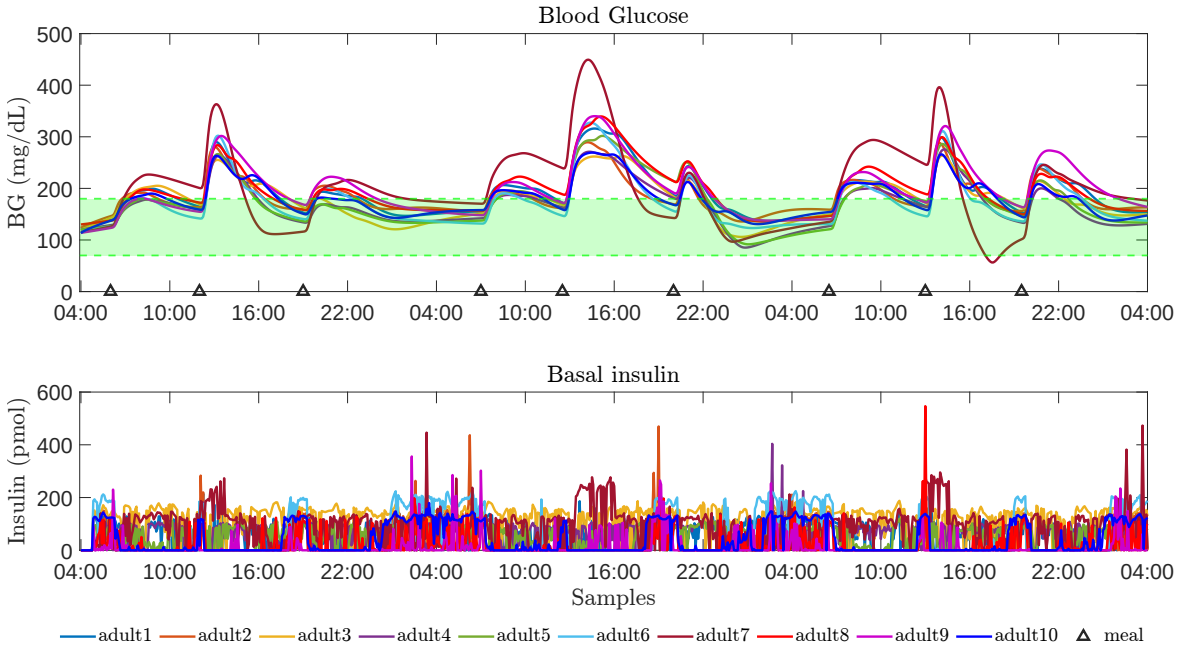


FIGURE 3.21: Proposed controller applied to the virtual patients with varying insulin sensitivity. Upper plot: BG trends of all patients, with the green zone for the safe range and the black triangles for the meal times. Lower plot: basal insulin injections.

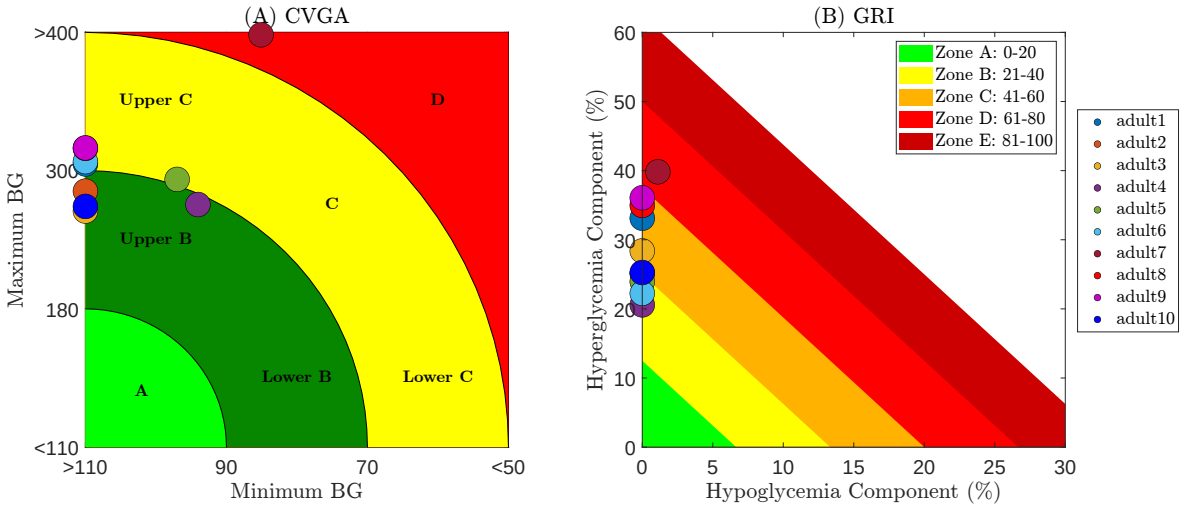


FIGURE 3.22: Results of the simulations performed with the IOB constraints, applied to the varying insulin sensitivity case: (A) CVGA and (B) GRI.

Adult	< 54 mg/dL	54-70 mg/dL	70-180 mg/dL	180-250 mg/dL	> 250 mg/dL
# 1	0%	0%	44%	46%	10%
# 2	0%	0%	57%	36%	7%
# 3	0%	0%	51%	42%	7%
# 4	0%	0%	66%	27%	7%
# 5	0%	0%	62%	28%	10%
# 6	0%	0%	64%	27%	9%
# 7	0%	1%	41%	36%	22%
# 8	0%	0%	43%	45%	12%
# 9	0%	0%	43%	41%	15%
# 10	0%	0%	56%	37%	6%

TABLE 3.8: TIRs percentages of the simulations performed with the IOB constraints, applied to the varying insulin sensitivity case.

primarily due to the risk of hypoglycemia [5, 101]. This is why, in this section, the PA is included in the analysis.

The simulations are set up in the same way as described in the previous section, with the S_I variability, but also adding training sessions. PA is not announced and, thus, it is considered as an external disturbance. It is involved in the analysis to evaluate its impact on BG of in-silico patient, and to verify the performance of the proposed controller in managing these variations.

Specifically, the UVA/Padova allows to add PA by setting its starting time, duration, and intensity level (there are four possible levels, expressed as a percentage of the maximum rate of oxygen consumption, $\%VO_2^{max}$: 0%, 25%, 50%, and 65%). In this case, PA was performed on the first and third days of the simulation, at different times. This is done to alternate between a day of exercise and a day of rest, avoiding more than two days without PA, as recommended by the ADA [29]. To reproduce a PA session, including warm-up, high-intensity resistance bursts, aerobic exercise, and cool-down phases, the following plan is implemented: light intensity for 10 min (25% VO_2^{max}), high intensity for 10 min (65% VO_2^{max}), moderate intensity for 20 min (50% VO_2^{max}), and light intensity for 5 min (25% VO_2^{max}) [71].

Simulation results

The simulations results of the proposed controller application to the T1DM in-silico adult patients are reported in Figure 3.23. Where in the lower graph there are the basal insulin injections, while in the upper graph there are the BG values, the meals

Adult	< 54 mg/dL	54-70 mg/dL	70-180 mg/dL	180-250 mg/dL	> 250 mg/dL
# 1	0%	0%	43%	47%	10%
# 2	0%	0%	57%	36%	7%
# 3	0%	0%	52%	40%	7%
# 4	0%	0%	66%	26%	7%
# 5	0%	0%	62%	28%	9%
# 6	0%	0%	65%	27%	9%
# 7	1%	1%	41%	39%	17%
# 8	0%	0%	43%	45%	12%
# 9	0%	0%	44%	41%	15%
# 10	0%	0%	56%	37%	6%

TABLE 3.9: TIRs percentages of the simulations performed with the IOB linear constraint, applied to the varying insulin sensitivity case with PA.

(depicted with black triangles), and the PA, represented as horizontal lines with different colors according to the intensity level (i.e., light blue for light intensity, yellow for moderate intensity, and pink for high intensity). Figure 3.24(A) shows the CVGA outcomes, while Figure 3.24(B) shows the GRI results.

Adding PA introduces more variability in the scenario, leading to increased BG oscillation. This is particularly evident in Adult 7, who goes in both hyper- and hypoglycemia ranges. In fact, in the CVGA, its point is in the C zone, especially due to hyperglycemic issues. This is also confirmed by looking at the GRI, which is equal to 64.28, thus in the D zone. The controller is also able to manage this disturbance, since the outcomes are similar to the previous cases, being conservative. This can be seen in Table 3.9, where the TIRs results show that all the virtual patients stay in hyperglycemic ranges for high percentages of the time and never in hypoglycemia (except for Adult 7).

There exist some guidelines for hypoglycemia prevention, based on carbohydrate requirements and glucose concentration before exercise [101]. These suggestions to the patient include consuming carbohydrates before or during PA, including delaying the session if BG is too low, and continuously monitoring glycemia.

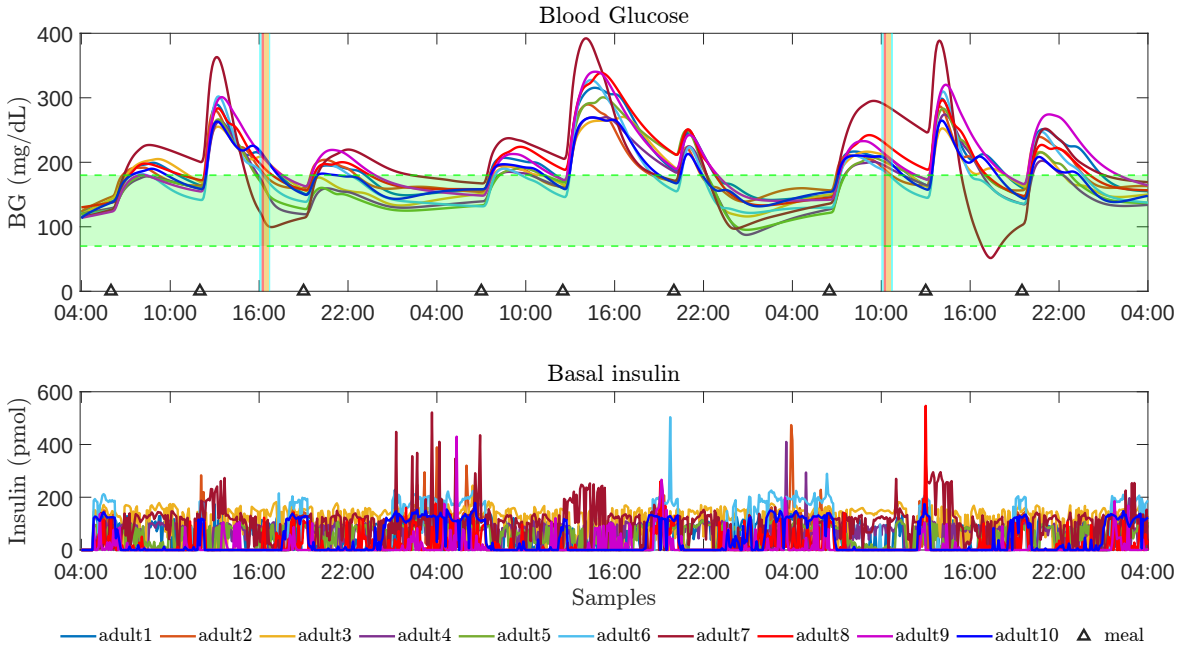


FIGURE 3.23: Proposed controller applied to the virtual patients with varying S_I and PA. Upper plot: BG trends of all patients, with the green zone for the safe range and the black triangles for the meal times. The horizontal lines depict the PA, considering light blue for the light intensity, yellow for the moderate intensity, and red for the high intensity PA. Lower plot: basal insulin injections.

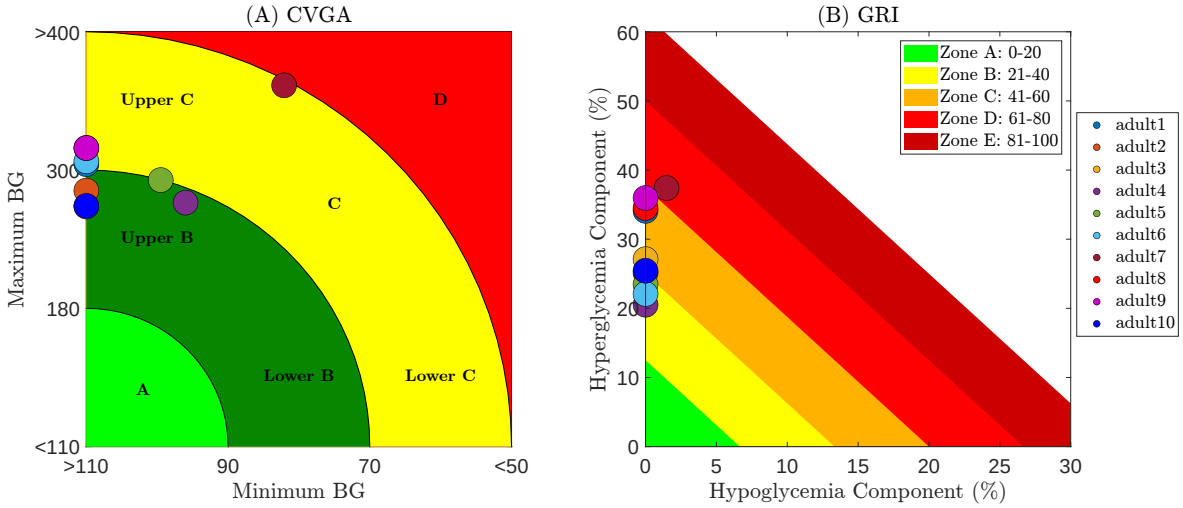


FIGURE 3.24: Results of the simulations performed with the IOB linear constraint, applied to the varying insulin sensitivity case with PA: (A) CVGA and (B) GRI.

3.5 Hybrid Modeling of the Insulin-Glucose System

This section describes another way of modeling the glucose-insulin-carbohydrate system, by proposing a hybrid modeling approach, combining a linear model with a data-driven one, used to capture all the nonlinearities. The results were published in [112].

Specifically, as stated in Chapter 2, advanced control algorithms often rely on a model of the system to compute optimal insulin injections. However, the insulin-glucose behavior varies significantly inter- and intra-patient, which complicates the identification of a general and accurate model. In particular, among the many factors, the variation over the day of the S_I complicates more the model prediction performances. That is why, in this section, the data-driven CHoKI learning method is used to model the nonlinearities, which means to consider the mismatch between the glucose predictions made with a linear model and the real values.

This is done to merge the benefits of considering a linear model, simpler to be implemented in a control algorithm and with physiological meaning [32, 56, 79], with a data-driven part [110, 120], to consider the nonlinearities and to personalize the model further.

Also in this case, the data collection for the model identification and the testing of the proposed controller have been performed on the 10 in-silico T1DM adult patients of the UVA/Padova simulator.

The main goal is to obtain better predictions with respect to those obtained with the linear model in [2], thanks to the CHoKI corrections. The aim is also to compare the outcomes with the results proposed in the previous sections, in which the data-based CHoKI method was used inside the MPC to obtain directly the BG predictions, to see if there are any improvements.

3.5.1 Problem statement

As aforementioned, the AP control problem consists of computing the appropriate insulin infusion $u_t(t)$, measured in U/min, injected to maintain the BG concentration within the desired values. As a disturbance, $u_1(t)$ is the rate of carbohydrate

intake (g/min). The glucose level is the output of the system (i.e., $y(t) = G$) and can be measured through the CGM. It evolves according to a nonlinear function of several factors, such that

$$\frac{dx(t)}{dt} = f\big(x(t), u_t(t), u_1(t)\big), \tag{3.26}$$

where

$$x(t) = \begin{bmatrix} x_1(t) & x_2(t) & x_3(t) & x_4(t) & x_5(t) \end{bmatrix}, \tag{3.27}$$

where, according to the compartmental physiological model [103],

- x_1 is the BG concentration G (i.e., $y(t) = x_1(t)$) (mg/dL),
- x_2 is the insulin delivery rate in plasma Q_i (U/min),
- x_3 is the insulin delivery rate in the subcutaneous compartment $Q_{i_{\text{sub}}}$ (U/min),
- x_4 is the rate of carbohydrate absorption from the gut Q_g (g/min),
- x_5 is the glucose delivery rate from the stomach Q_{sto} (g/min).

The objective is to learn the nonlinearities of the real system using data-based algorithms. To do so, $f(\cdot)$ is decomposed into its linear trend $f_L(\cdot)$ and the deviation of the linear model, with $\tilde{f}(\cdot)$, such that:

$$\frac{dx(t)}{dt} = f\big(x(t), u_t(t), u_1(t)\big) = f_L\big(x(t), u_t(t), u_1(t)\big) + \tilde{f}\big(x(t), u_t(t), u_1(t)\big) \tag{3.28}$$

which are described in the following.

Remark 3.4

Note that, differently from the previous cases, here there is u_t and not u_2 because it represents the total amount of insulin, meaning both the basal and the bolus amounts (i.e., $u_t(t) = u_2(t) + u_{\text{bol}}(t)$), and its unit of measurement is U/min, *not* pmol *as before.*

Linear Model

This section briefly summarizes the linear model $f_L(\cdot)$, which is detailed in [2]. The physiological long-term minimal model based on Ruan's proposal [103] is considered.

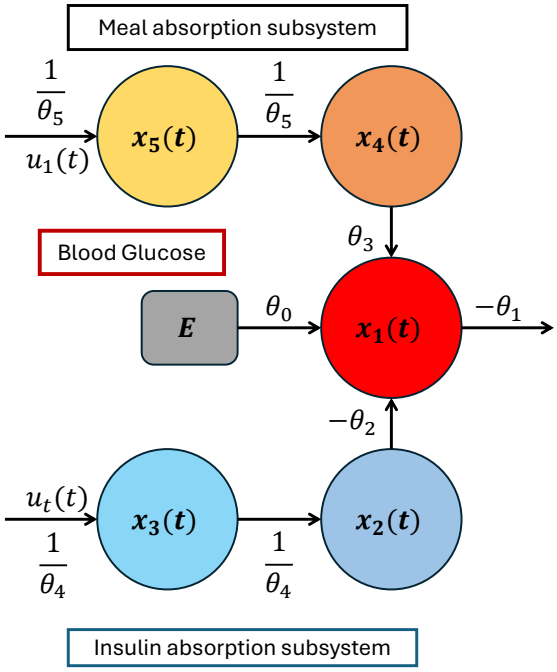


FIGURE 3.25: Scheme of the linear compartmental model

This is a compartmental model, whose structure is reported in Figure 3.25, with the following subsystems:

- Meal absorption subsystem:

$$\frac{dQ_g(t)}{dt} = -\frac{1}{\theta_5}Q_g(t) + \frac{1}{\theta_5}Q_{\text{sto}}(t) \tag{3.29a}$$

$$\frac{dQ_{\text{sto}}(t)}{dt} = -\frac{1}{\theta_5}Q_{\text{sto}}(t) + \frac{1}{\theta_5}u_1(t) \tag{3.29b}$$

- Blood glucose:

$$\frac{dG(t)}{dt} = \theta_0 - \theta_1 G(t) - \theta_2 Q_i(t) + \theta_3 Q_g(t) \tag{3.30}$$

- Insulin absorption subsystem:

$$\frac{dQ_i(t)}{dt} = -\frac{1}{\theta_4}Q_i(t) + \frac{1}{\theta_4}Q_{i_{\text{sub}}}(t) \tag{3.31a}$$

$$\frac{dQ_{i_{\text{sub}}}(t)}{dt} = -\frac{1}{\theta_4}Q_{i_{\text{sub}}}(t) + \frac{1}{\theta_4}u_t(t) \tag{3.31b}$$

Where $Q_g(t)$ represents the rate of carbohydrate absorption from the gut (in g/min), $Q_{\text{sto}}(t)$ stands for the glucose delivery rate from the stomach (in g/min), $G(t)$ is the blood glucose concentration (in mg/dL), $Q_i(t)$ is the insulin delivery rate in plasma (in U/min), and $Q_{i_{\text{sub}}}$ stands for the insulin delivery rate in the subcutaneous compartment (in U/min). And with the following parameters:

- θ_0 is the endogenous glucose production at basal levels (in mg/(dL · min)),
- θ_2 is the insulin sensitivity (in mg/(dL · U)),
- θ_3 is the carbohydrate ratio (in mg/(dL · g)),
- θ_4 is the time-to-maximum effective insulin concentration (in min),
- θ_5 is time-of-maximum appearance rate of glucose in gut (in min).

Then, the following continuous-time state space model is studied:

$$\begin{cases} \dot{x}(t) = Ax(t) + B_{u_t}u_t(t) + B_{u_1}u_1(t) + E, \;\; x(0) = x_0 \\ y(t) = Cx(t). \end{cases} \tag{3.32}$$

The matrices of the model in (3.32) are given by:

$$\begin{aligned} A &= \begin{pmatrix} -\theta_1 & -\theta_2 & 0 & \theta_3 & 0 \\ 0 & -\frac{1}{\theta_4} & \frac{1}{\theta_4} & 0 & 0 \\ 0 & 0 & -\frac{1}{\theta_4} & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{\theta_5} & \frac{1}{\theta_5} \\ 0 & 0 & 0 & 0 & -\frac{1}{\theta_5} \end{pmatrix}, \; B_{u_t} = \begin{pmatrix} 0 \\ 0 \\ \frac{1}{\theta_4} \\ 0 \\ 0 \end{pmatrix} \\ B_{u_1} &= \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{\theta_5} \end{pmatrix}, \; E = \begin{pmatrix} \theta_0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \; C = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}'. \end{aligned} \tag{3.33}$$

The parameters $\boldsymbol{\theta} = [\theta_0, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5]$ can be estimated with an identification technique based on Regularized Least-Squares (RLS), with the minimization of the following proposed objective function:

$$V_N(\theta) = \frac{\|y(k) - \hat{y}(k)\|^2}{\|y(k) - \bar{y}(k)\|^2} + \theta^0 \alpha \theta^{0\prime} \tag{3.34}$$

where $y(k)$ is the measured BG, $\bar{y}(k)$ is its average, $\hat{y}(k)$ is the model predicted output, and with the following regularization term:

$$\theta^0 = \left[\left(\theta_2 - CR^0\theta_3\right)\left(\theta_4 - \theta_4^0\right)\left(\theta_5 - \theta_5^0\right)\right],$$

with $\alpha = diag\left(1, 1, \frac{1}{(\theta_5^0)^2}\right)$, carbohydrate-to-insulin ratio CR^0, time-to-maximum effective insulin concentration θ_4^0, and time-to-maximum appearance rate of glucose in gut (or plasma) θ_5^0.

The minimization of $V_N(\theta)$ was performed using `fmincon` MATLAB routine, with personalized initial parameters given by: $\theta_0(0) = \theta_1(0)G_b + \theta_2 U_b$, $\theta_1(0) = \theta_1^0$, $\theta_2(0) = CF$ (i.e., the correction factor), $\theta_3(0) = RF$ (i.e., the carbohydrate factor), $\theta_4(0) = \theta_4^0$ and $\theta_5 = \theta_5^0$, with $\theta_1^0 = 0.004$ min and $G_b = y(1)$. The parameter CR_0 is established according to the CR value of each patient, while θ_5^0 is set to 40 min for all patients. Moreover, since an accurate estimation of θ_4 is crucial to avoid insulin stacking problems, θ_4^0 is determined by minimizing the sum of squared residuals (Least-Squares (LS) method) between the estimated IOB and the real value of the simulator. The IOB estimation is obtained by computing

$$IOB(t) = \theta_4 \left(x_2(t) + x_3(t)\right).$$

By sampling the continuous-time solution of the system at times $t = kT$, the following discrete-time system is obtained:

$$x((k+1)T) = A^d x(kT) + B_{u_t}^d u_t(kT) + B_{u_1}^d u_1(kT) + E^d, \tag{3.35}$$

whose matrices are obtained with the Euler forward discretization, by multiplying the continuous-time matrices by the sampling time T.

Note that in what follows, T is omitted for the sake of conciseness.

Hybrid modeling with CHoKI correction factor

As already stated, the idea is to find a correction factor that has to be added to the BG predictions made with the linear model in (3.32) to obtain more precise values. This is achieved by implementing a data-driven learning method, specifically, the CHoKI, in order to capture all the nonlinearities.

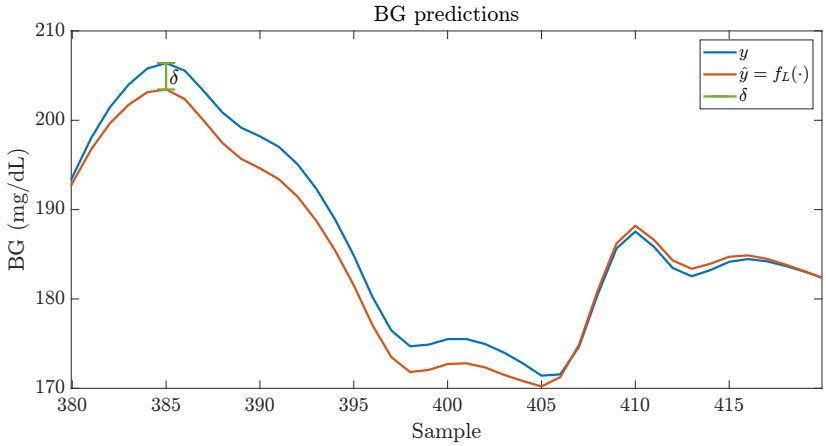


FIGURE 3.26: In blue, the measured BG measurements are represented, the predictions made with the linear model f_L are plotted in orange. In green is highlighted the model mismatch $\delta(k) = y(k) - \hat{y}(k)$.

The theory of the CHoKI learning method has been detailed in Section 2.2.2, and in Section 3.2.2 this has been exploited for making BG predictions, based on the past inputs and the current condition (i.e., with the values of BG, basal insulin, and meals). However, in this section, the goal of the learning method is slightly different, since it has to estimate all the nonlinearities, which means to compute the difference between the linear model predictions and the real BG values. In this section, the design of the data-driven CHoKI is detailed to this aim. Note that in the following, to differentiate from the previous case, the superscript h represents the hybrid model case.

Given a dataset of historical trajectories $\{x, u_t, u_1\}$, the linear model f_L is applied, and δ is computed as the mismatch between the predictions, denoted $\hat{y}$, and the real glucose values y, i.e.,

$$\delta(k) = y(k) - \hat{y}(k). \tag{3.36}$$

Figure 3.26 shows a graphical representation, where the real values y are represented by the blue line and the predictions $\hat{y}$ made with the linear model are represented by the orange line. The model mismatch δ is highlighted in green.

Then, similar in spirit to learning the nonlinear system as in [85], but tailored in this case to learn only the dynamical deviation from the linear system, we assume that a given deviation $\delta(k)$ is a function of past terms, such that

$$\hat{\delta}(k+1) = \tilde{f}(w^h(k)), \tag{3.37}$$

with the regressor w^h that grouped the inputs as

$$w^h(k) = \big(\delta(k), \dots, \delta(k - n_a^h), u_t(k-1), \dots, u_t(k - n_b^h), u_1(k-1), \dots, u_1(k - n_c^h)\big). \tag{3.38}$$

Here, $n_a^h, n_b^h, n_c^h \in \mathbb{N}_0$ are the memory horizons for the mismatch δ, insulin u_t and meals u_1, respectively. A dataset of $N_{\mathcal{D}^h}$ observations, denoted $\mathcal{D}^h = \{(\delta(k+1), w^h(k))\}$, for $k = 1, \dots, N_{\mathcal{D}^h}$ is collected.

Notice that this is an output-feedback approach, and hence we are only able to predict the output (that is, x_1), maintaining the rest of the state as given by the linear system. Thus, the final hybrid model to make glucose predictions is

$$\begin{cases} x(k+1) &= A^d x(k) + B_{u_t}^d u_t(k) + B_{u_1}^d u_1(k) + E^d + C^d \tilde{f}\big(w^h(k)\big), \\ y(k) &= C^d x(k). \end{cases} \tag{3.39}$$

3.5.2 Hybrid modeling identification

In the previous sections, the theory and the procedure for the model identification are detailed, while in this section, the training phase and the testing of the proposed hybrid model are reported. The first step is the data collection performed on the UVA/Padova simulator, followed by the identification of the linear model parameters, also needed to be able to estimate the CHoKI correction factor, ending with the testing of the proposed model, comparing its results with other methodologies.

Data collection for hybrid model identification

The first step needed to identify the proposed model in (3.39) is the data collection, which is carried out by exploiting the in silico T1DM adult patients of the UVA/Padova simulator. The work [93] suggests a possible setting for the data collection, which allows for good outcomes in the model identification. In particular, two datasets were proposed: one for training over four days (whose setting details are in Table 3.10, looking at the first 4 days) and another for testing over three days (whose details are in Table 3.11). In this case, to help the identification process of the two models, 5 additional days were simulated, built as detailed in Table 3.10 (see from day 5 to day 9). All the datasets consider circadian variability of S_I, thanks to its implementation in the simulator, as stated in [119]. The data collection has been

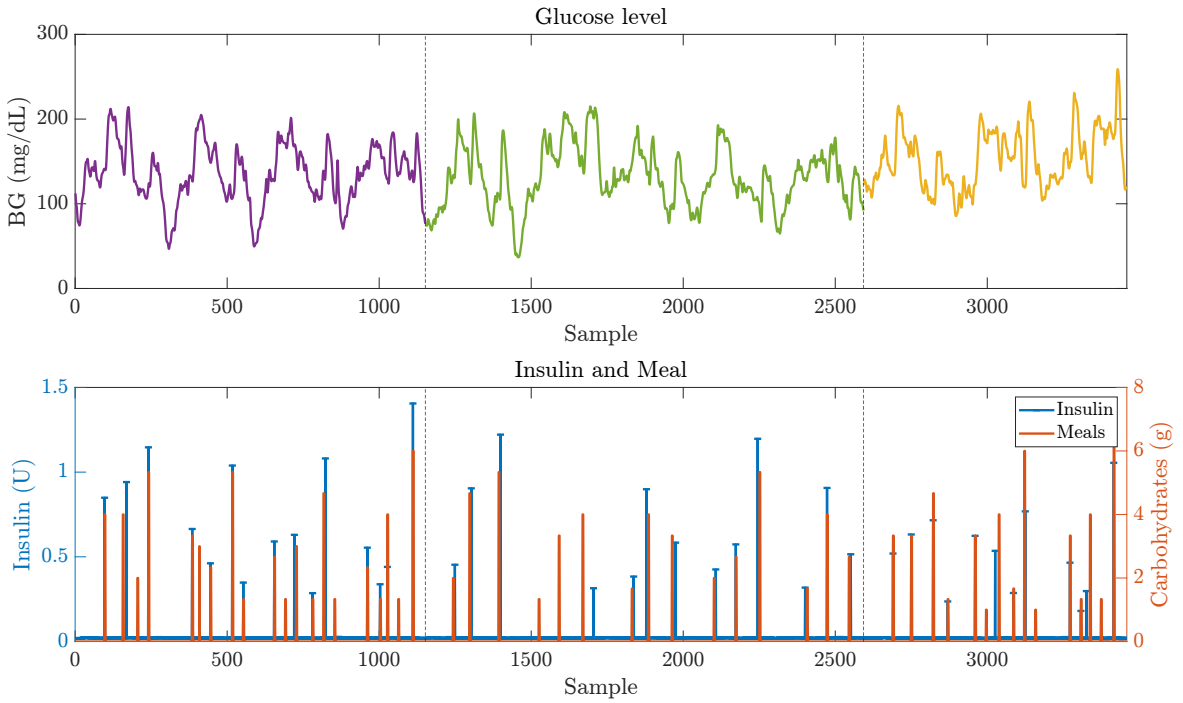


FIGURE 3.27: In the upper graph, the BG trends are reported: in purple the dataset used for the identification of the linear model, in green the one used for the identification of the CHoKI components, and in yellow the one used to test the proposed model. In the lower graph, the insulin injections (both basal and boluses) are in blue, and the meals are in orange. Example of Adult 1.

performed with the same noise settings (that is, on the insulin pump, CGM sensor, and on the meal amounts) as the ones detailed in Section 3.2.1.

The simulation setting is the same for all the in-silico patients, but, as an example, the datasets collected for virtual Adult 1 are reported in Figure 3.27. In the upper part, the BG level of the first 4 days of the training is in purple, the one of the additional 5 days is in green, and the one of the 3 days for testing is in yellow, considering the sampling time equal to 5 min. While, in the lower graph, the insulin amounts (basal and boluses) are shown in blue, and the meals are in orange. This highlights the variations between insulin boluses and meals, changing the time gap between them, the amounts, and their presence/absence, allowing for greater variability. This generates variations in the BG values. Both these aspects help the identification procedure.

	Time (hh:MM)	CHO (g)	Duration (min)	Insulin Bolus
Day 1	08:00	60	15	On time
	13:00	60	15	At 14:00
	17:00	30	15	No bolus
	20:00	80	15	On time
Day 2	08:00	50	15	On time
	10:00	15	5	No bolus
	13:00	35	15	On time
	19:00	80	15	On time
	22:00	20	15	On time
Day 3	06:30	40	15	On time
	09:30	20	15	No bolus
	12:30	45	15	At 12:00 for 50 g
	17:00	20	15	On time
	20:00	70	15	Bolus at 20:30
	23:00	20	15	No bolus
Day 4	08:00	35	15	On time
	11:30	20	15	On time
	13:30	60	15	At 13:30 for 30 g
	16:30	20	15	No bolus
	20:30	90	15	On time
Day 5	07:30	30	15	At 08:00
	12:00	70	15	At 12:30
	20:00	80	15	At 20:30
Day 6	07:00	20	15	No bolus
	12:30	50	15	No bolus
	19:00	60	15	No bolus
	22:00	/	/	At 22:00 for 20 g
Day 7	08:30	25	15	At 09:00
	13:00	60	15	At 12:30
	19:30	50	15	At 20:30
Day 8	07:00	30	15	At 07:30
	13:00	40	15	On time
	19:30	80	15	At 19:00
Day 9	08:30	25	15	At 08:00
	14:00	60	15	On time
	20:00	40	15	At 20:30

TABLE 3.10: Identification protocol for the training dataset. The first 4 days come from [93].

Linear model and CHoKI parameters identification

The identification of the linear model $f_L(\cdot)$ was done considering the first 4 days of the training dataset (i.e., the purple one in Figure 3.27), considering 3 days for the training and the last one for the testing, as explained in Section 3.5.1. The values of the $\boldsymbol{\theta}$ parameters that were obtained are reported in Table 3.12, for each patient. These values are comparable with those obtained in [2], which means that the identification phase was correctly performed.

The following step is the computation of the CHoKI components $(\mathcal{L}^h, \mathcal{P}^h)$ for obtaining $\tilde{f}(\cdot)$, with the aim of being able to assess the model error, by catching the nonlinearities, including, for example, the variations in the S_I. First, the values of the orders n_a^h, n_b^h, n_c^h needed for the regressor w^h were chosen with a validation procedure (by considering $n_a^h, n_b^h, n_c^h \in \mathbb{I}_1^{15}$), selecting the combination that returned the lowest RMSE between the predictions $\hat{\delta}$ and the real values δ, also considering a trade-off with the model complexity. The final values are $n_a^h = 14$ for mismatch values δ, $n_b^h = 6$ for the insulin u_t, and $n_c^h = 2$ for the meals u_1.

Then, assuming $\mathcal{P}^h = 1$, the components of $\mathcal{L}^h$ were estimated starting by applying the linear model previously identified on the 5-day training dataset (i.e., the green one in Figure 3.27), considering the first 4 for the training and the last one for the testing. The mean and standard deviation of the mismatch between the model

	Time (hh:MM)	CHO (g)	Duration (min)	Insulin Bolus
Day 1	08:00	50	15	Bolus on time
	13:00	50	15	Bolus on time
	19:00	70	15	Bolus on time
	23:00	20	15	Bolus on time
Day 2	06:30	50	15	Bolus on time
	09:30	15	15	No bolus
	13:00	60	15	Bolus at 12:00 for 50 g
	17:00	25	15	Bolus on time
	20:00	90	15	Bolus at 20:15 for 70 g
	23:00	15	15	No bolus
Day 3	08:30	50	15	Bolus on time
	11:30	20	15	Bolus on time
	14:00	60	15	Bolus at 13:00 for 30 g
	17:00	20	15	No bolus
	20:30	100	15	Bolus on time

TABLE 3.11: Test protocol for individualized models, from [93].

Patient	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
$\theta_0(\mathrm{mg}/(\mathrm{dL}\cdot\mathrm{min}))$	2.300	2.299	2.106	1.923	1.454	2.262	2.300	1.401	2.243	2.009
$\theta_1(/\mathrm{min})$	0.0068	0.0035	0.0053	0.0051	0.0038	0.0062	0.0034	0.0054	0.0053	0.0074
$\theta_2(\mathrm{mg}/(\mathrm{dL}\cdot\mathrm{U}))$	66.410	83.379	53.547	79.291	59.507	46.873	83.709	33.639	100.11	52.622
$\theta_3(\mathrm{mg}/(\mathrm{dL}\cdot\mathrm{g}))$	3.466	3.708	3.679	4.024	4.419	5.211	4.614	3.826	5.066	3.821
$\theta_4(\mathrm{min})$	56.004	40.011	52.207	59.505	45.507	52.506	47.505	50.007	50.507	50.506
$\theta_5(\mathrm{min})$	34.177	27.508	34.348	35.5992	35.636	33.906	31.800	35.279	34.355	34.793
$[L_a^h; L_b^h; L_c^h]$	[0.587 14.99 0.00001]	[0.681 1.887 3.539]	[0.921 0.021 0.197]	[0.514 5.588 0.708]	[0.756 0.318 2.409]	[0.932 1.888 1.779]	[0.505 0.399 1.060]	[0.649 1.198 0.821]	[0.698 1.295 1.744]	[0.661 0.00002 2.452]

TABLE 3.12: Estimated parameters $\boldsymbol{\theta}$ and $\mathcal{L}^h$ of 10 in-silico adults

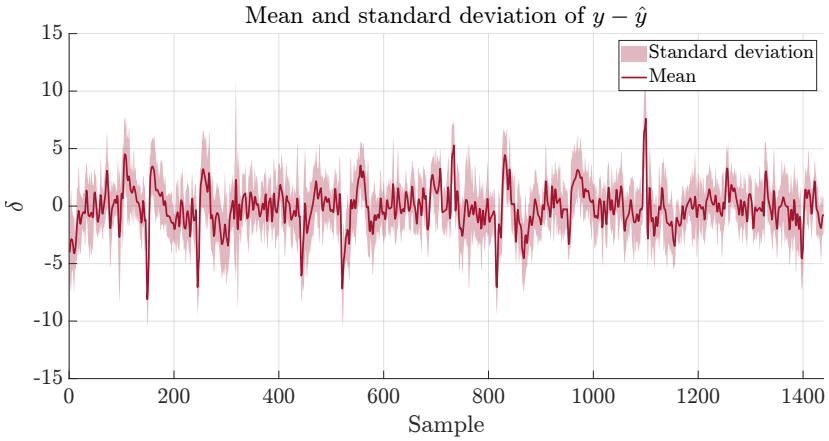


FIGURE 3.28: Mean and standard deviation of the mismatch δ between model predictions $\hat{y}$ and measured glucose values y, for the 10 virtual patients (considering the 5-day dataset, the green one in Figure 3.27).

predictions and the measured values of the 10 in silico patients are reported in Figure 3.28 (i.e., $y - \hat{y}$, where $\hat{y}$ is obtained from the personalised f_L applied on the 5-day training dataset). These values are used to create the dataset $\mathcal{D}^h$, by building $w^h(k)$ as per (3.38) and with $\delta(k+1)$. To learn the effect that each input has on the output, only three values in $\mathcal{L}^h$ were used:

- $L_a^h \in \mathbb{R}$ for the δ (i.e., $\delta = y - \hat{y}$),
- $L_b^h \in \mathbb{R}$ for the insulin u_t,
- $L_c^h \in \mathbb{R}$ for the meals u_1.

The values of the identified CHoKI parameter $\mathcal{L}^h = [L_a^h \mathbb{1}_{n_a^h}; L_b^h \mathbb{1}_{n_b^h}; L_c^h \mathbb{1}_{n_c^h}] \in \mathbb{R}^{n_{w^h}}$, that are obtained by following the procedure in Section 3.5.1, are reported in Table 3.12, for each subject.

3.5.3 Identification results

Once the two components $f_L(\cdot)$ and $\tilde{f}(\cdot)$ are identified, the model $f = f_L + \tilde{f}$ can be applied to the testing dataset (i.e., the yellow one in Figure 3.27, considering as dataset $\mathcal{D}^h$ for the CHoKI the training dataset employed before for the CHoKI identification). In Figure 3.29 the boxplots for the mean absolute error (i.e., $|y - \hat{y}|$) are reported considering all the 10 virtual patients of the simulator, making a comparison of the outcomes of the different models: the linear one (f_L), the linear model with the CHoKI correction ($f_L + \tilde{f}$), and the zero-order hold. With the zero-order hold

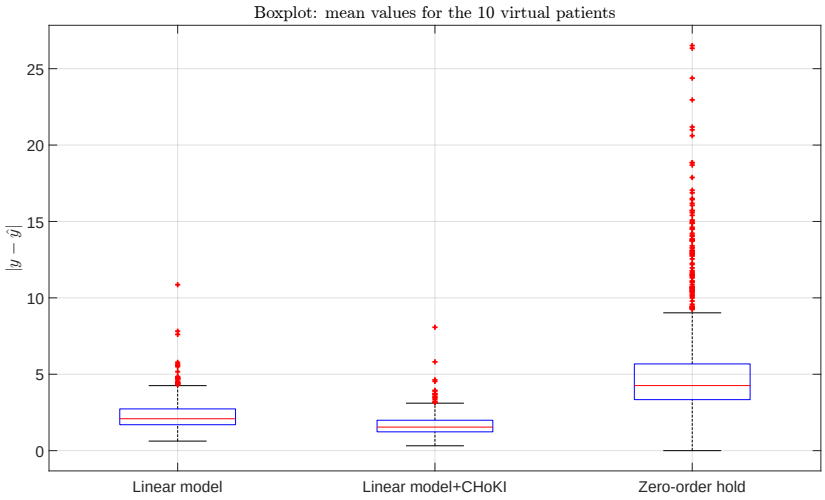


FIGURE 3.29: Boxplot of the mean absolute error between the measured BG values and the predictions, made with different models: the linear model, the linear model with the CHoKI correction, and the zero-order hold.

Adult	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	Median
$RMSE_m$	2.912	2.903	2.831	2.696	2.661	3.077	3.423	2.912	3.081	3.113	2.91 [2.83, 3.08]
$RMSE_{mc}$	2.109	2.199	2.199	2.069	1.869	2.308	2.441	2.070	2.267	2.229	2.20 [2.07, 2.27]
$RMSE_z$	7.389	6.609	6.104	6.885	5.817	7.799	8.599	7.017	7.601	6.891	6.95 [6.61, 7.60]
GoF_m	91.43	90.02	92.01	92.75	93.64	91.34	94.76	92.73	92.44	90.85	92.23 [91.34, 92.75]
GoF_{mc}	93.79	92.44	93.79	94.44	95.53	93.51	96.31	96.83	94.44	93.44	94.12 [93.51, 95.53]
GoF_z	78.25	77.29	82.76	81.48	86.10	78.05	86.84	82.84	81.36	79.74	81.42 [78.25, 82.84]

TABLE 3.13: RMSE and GoF (on the testing dataset, 1-step-ahead predictions). Last column represents the median and the $[25, 75]^{th}$ percentiles.

method, the predictions are made by considering $\hat{y}(k+1) = y(k)$, which means to take the previous measured value of the BG. This shows that the predictions made with the CHoKI corrections are much more accurate than the ones made with the zero-order hold, but these are also better than the predictions made only with the linear model. This means that the δ estimated with the CHoKI are a quite good approximation of the real errors. In Table 3.13 are reported the GoF and the RMSE to compare the three models on the testing dataset, considering the 1-step-ahead predictions, for all the patients and their median and the $[25, 75]^{th}$ percentiles values. Specifically, $(RMSE_m, GoF_m)$ are for the linear model, $(RMSE_{mc}, GoF_{mc})$ for the model with the CHoKI, and $(RMSE_z, GoF_z)$ for the zero-hold. These values confirm what was stated before, since the highest GoF and the lowest RMSE are the ones for the model with the CHoKI corrections.

To further evaluate the performance of the proposed model (3.39), a comparison among its predictions, the ones obtained with the linear model [2], and the ones

with only the CHoKI method, as shown in Section 3.2.2, has been performed. Specifically, the models are tested on a new 1-day simulation dataset, which is done without meals or boluses to have a fair comparison, since the CHoKI learning method in Section 3.2.2 was focused only on the computation of the basal insulin (that is, before the BG predictions were based on u_2, while the hybrid modeling on u_t). It has just the standard continuous basal insulin therapy provided by the simulator (i.e., the daily basal insulin amount in Table 1.3), with a noise on the insulin pump, with a Gaussian distribution, with 0 pmol mean and 0.1 standard deviation (with the minimum insulin value of 0 pmol). In Figure 3.30(A) and Figure 3.30(B), there are the comparisons of the BG levels of the best and worst cases, respectively, as an example. In detail, in blue there are the measured BG values y, in red the predictions made with the linear model f_L, in yellow the ones made with the linear model and the CHoKI correction $f_L + \tilde{f}$, and in purple the predictions made with the CHoKI (as in Section 3.2.2). The predictions are quite accurate in both cases, and this is confirmed by checking the absolute mean error of all 10 in silico patients, whose boxplots are reported in Figure 3.31, to compare the performance of the three models. This shows that the proposal has a lower mean absolute error than the linear model and the CHoKI model, which was the first goal: trying to obtain more accurate glucose predictions thanks to the CHoKI correction.

The proposed model was also tested on a more challenging scenario, with another 2-day dataset, which includes PA sessions [71]. Specifically, it has continuous basal insulin, with the same noise in the insulin pump as before. The settings of this dataset are reported in Table 3.14, with the meals, the boluses, and the PA sessions. The results of the glucose trends for the best and worst cases are reported in Figure 3.32(A) and Figure 3.32(B), respectively. As before, the measured BG values y are plotted in blue, the predictions made with the linear model f_L are plotted in red, the ones made with the linear model and the CHoKI correction $f_L + \tilde{f}$ are plotted in yellow, and the gray areas represent the exercise sessions. Also in this scenario, the predictions are quite good, which is confirmed by looking at the boxplots with the absolute errors for all the virtual patients, which are shown in Figure 3.33. The model corrected with the CHoKI makes better predictions than the linear model and the zero-order hold.

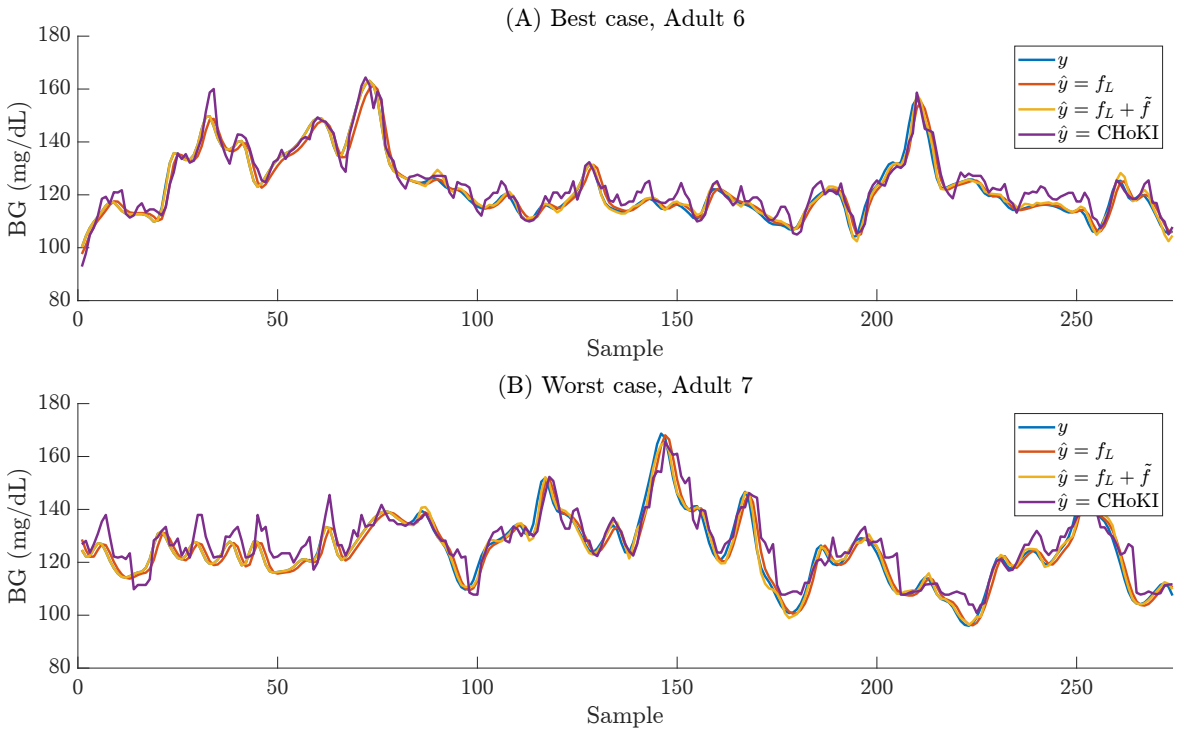


FIGURE 3.30: Comparison of the BG predictions: in red the ones made with the linear model f_L, in yellow the ones of the linear model and the CHoKI $f_L + \tilde{f}$, in purple the ones made just with the CHoKI, and in blue the measured values y. (A) Best case, Adult 6 and (B) Worst case, Adult 7.

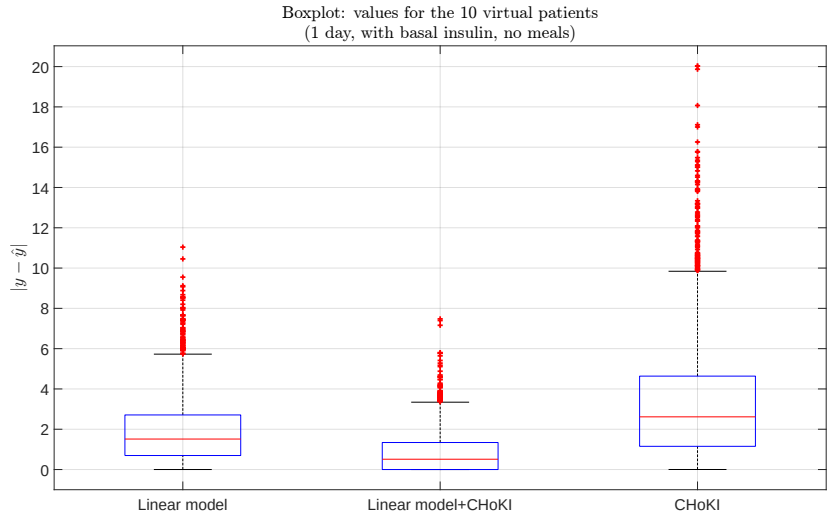


FIGURE 3.31: Boxplot of the mean absolute error between the measured BG values and the predictions, made with different models: the linear model, the linear model with the CHoKI correction, and the CHoKI. The comparison is made on 1 day, with basal insulin and no meals.

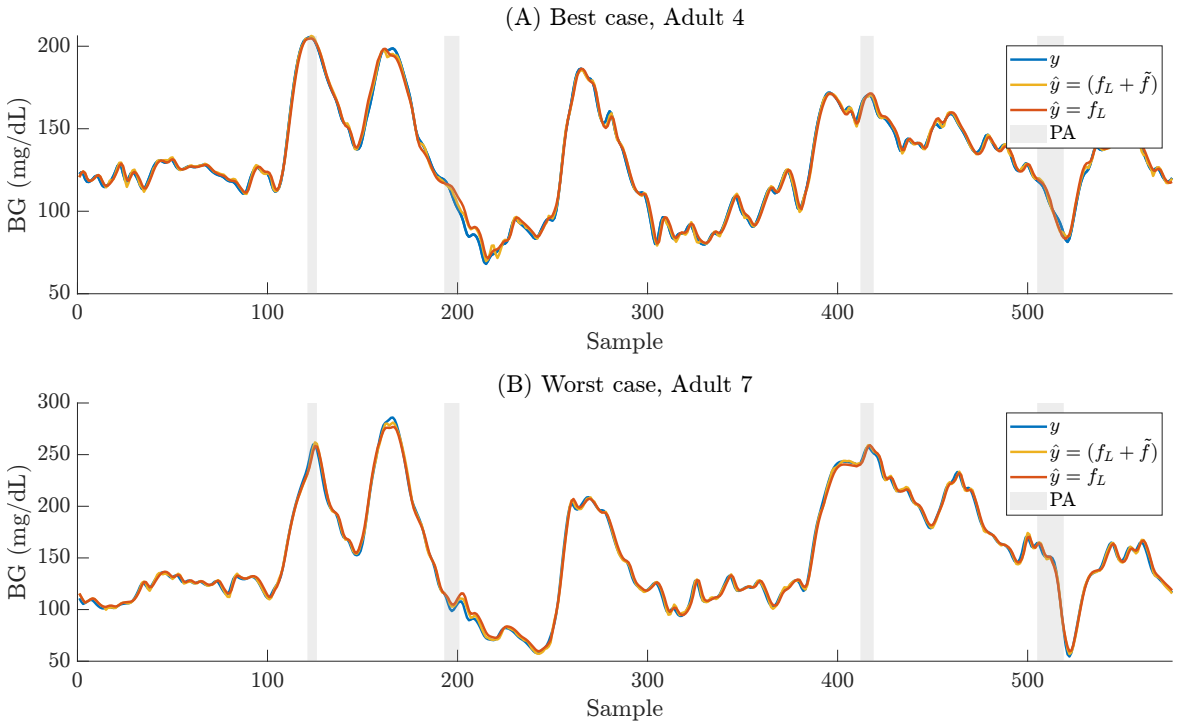


FIGURE 3.32: Comparison of the BG predictions: in red the ones of the linear model f_L, in yellow the one of the linear model plus the CHoKI component $f_L + \tilde{f}$, and in blue the measured values y. Considering the 2-day dataset with PA sessions represented as the gray areas. (A) Best case, with Adult 4, and (B) Worst case, with Adult 7.

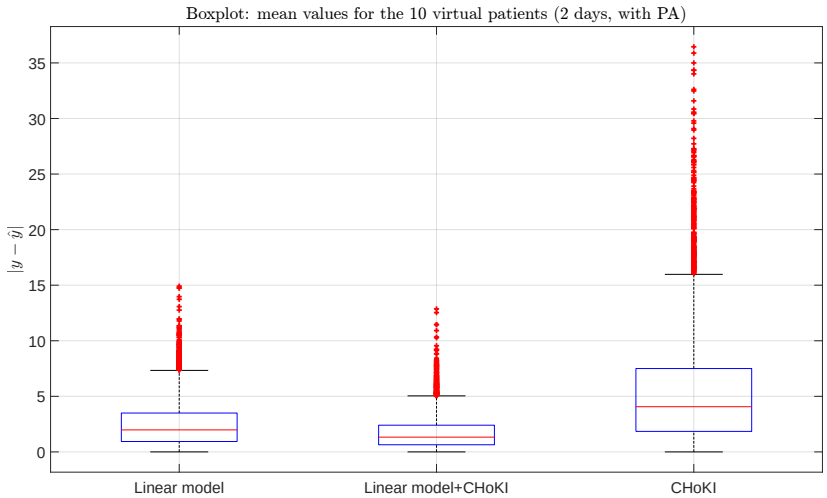


FIGURE 3.33: Boxplot of the mean absolute error between the measured values and the BG predictions: the linear model, the linear model with the CHoKI, and the zero-order hold. Considering the 2-day with PA sessions.

	Day 1				Day 2			
Meals / Bolus	Time (hh:MM)	CHO (g)	Duration (min)	Insulin Bolus	Time (hh:MM)	CHO (g)	Duration (min)	Insulin Bolus
	08:30	60	15	Bolus ok	07:30	50	15	Bolus ok
	12:00	60	15	Bolus at 13:00	10:00	15	15	No bolus
	16:00	30	15	No bolus	13:00	35	15	Bolus ok
	20:30	80	15	Bolus ok	19:00	80	15	Bolus at 18:30
					22:00	20	15	Bolus ok
Physical Activity	Time (hh:MM)	Intensity	Duration (min)		Time (hh:MM)	Intensity	Duration (min)	
	10:00	Moderate	30		10:15	Light	10	
	16:00	Light	10		10:25	Intense	10	
	16:10	Intense	10		10:35	Moderate	20	
	16:20	Moderate	20		10:55	Light	5	
	16:40	Light	5		18:00	Light	60	

TABLE 3.14: Setup of a new 2-day dataset, with physical activity

3.6 Conclusions

In this chapter, new customized MPC algorithms based on the CHoKI learning method were proposed to be used in the AP for managing basal insulin in T1DM patients, with the aim of maintaining their BG levels inside the safe range. The whole system was then tested on the 10 virtual adult T1DM patients of the UVA/Padova simulator. The results are quite promising, especially when compared to those obtained through standard therapy provided by the simulator, with constant basal insulin injections. The main outcome is that all the proposed controllers reduce the (more dangerous) hypoglycemic events, maintaining the patients inside the euglycemic zone (i.e., 70-180 mg/dL) most of the time.

In addition, in the MPC optimization problem, a new constraint on the insulin upper limit was included by considering the IOB. This helped to prevent the subjects from entering the hypoglycemic range, since it takes into account the insulin from previous boluses that is still active in the patient. The IOB estimation was done in two ways: with a linear weight and with an exponential one. The exponential weights in the IOB estimation perform better than the linear one; however, the controller might still be too conservative, allowing some hyperglycemic events.

The proposed controllers were also tested on virtual patients with circadian variability in insulin sensitivity and also considering physical activity, to evaluate the performance of the proposal under more challenging scenarios.

The outcomes end up being conservative, however, this conservativeness is a limitation of the CHoKI learning method [86]. Therefore, future work will focus on

finding new methods to reduce this limitation, such as the probabilistic constraints tested in the next chapter.

Moreover, as mentioned in Section 3.5, the objective was to analyse the outcomes of combining the advantages of a linear model, which is easier to implement in a control algorithm and has physiological meaning, with a data-driven part that captures nonlinearities and enables further personalisation of the model. The results showed that the data-based nonlinear corrections, made with the CHoKI learning method, added to the linear model, allow for obtaining more accurate BG predictions. This is verified by comparing the proposed model results with the linear model [2] and with the previous data-based algorithms in Section 3.2.2.

Additional analyses were performed to compare the BG predictions and the measured values, by performing simulations on the UVA/Padova simulator, adding variability in insulin sensitivity and considering physical activity, to evaluate the performance under more challenging scenarios. To further test the proposed model, a possible future work could be to try to implement this prediction model inside a model predictive control framework.

CHAPTER 4

CHoKI-based MPC with probabilistic constraints

A possible drawback of the CHoKI-based MPC technique is that the deterministic bounds obtained may end up being so conservative that they render an infeasible setup. In particular, the propagation of prediction error typical of this approach tends to tighten constraints rapidly, resulting in a very short control horizon. For this reason, in Chapter 3 (detailed in Section 3.2.3), a truncated bound on the prediction error was used to obtain practical controllers. However, in this chapter, a stochastic description of uncertainties and chance constraints is proposed as a potential solution to the problem of overconservatism. Specifically, to account for the model-system mismatch, the approach proposed in [78] is employed, since it allows for a small probability of violating the constraints, enabling the extension of the control horizon. The results of this CHoKI-based MPC with probabilistic constraints were published in [109].

4.1 Chance constraints

In this section, a brief description of the chance constraints theory applied to the stochastic MPC is reported [78]. This method allows for a small probability of constraint violation, taking into consideration the probabilistic description of the uncertainty. This could face the infeasibilities due to possible values outside the range of the tightened constraints. This means having less conservative tightening, which leads to an increased region of attraction without changing the prediction horizon.

As in [78], the predicted state trajectory $\hat{x}(j|k)$ of the system can be decomposed into its deterministic (nominal evolution, $\hat{z} \in \mathbb{R}^{n_x}$) and stochastic ($\hat{e}_x \in \mathbb{R}^{n_x}$) components:

$$\hat{x}(j|k) = \hat{z}(j|k) + \hat{e}_x(j|k), \quad j \in \mathbb{I}_0^{N_c-1}, \tag{4.1}$$

given a finite control horizon N_c and with sampling time k.

The state constraints $x(j|k) \in \mathcal{X}$ can be written in a probabilistic way, as

$$\mathbb{P}\{x(j|k) \in \mathcal{X}\} \geq 1 - \varepsilon. \tag{4.2}$$

Which means that the state constraints are satisfied with a probability higher than $1 - \varepsilon$, since ε represents the probability of violating the i-th state constraint. Equation (4.2) can be rewritten componentwise as

$$\mathbb{P}\{A_x(i)x(j|k) \leq b_x(i)\} \geq 1 - \varepsilon(i), \tag{4.3}$$

with $i \in \mathbb{I}_1^{p_c}$, where p_c is the number of linear constraints that defines the set $\mathcal{X}$, and $j \in \mathbb{I}_1^{N_c}$. These chance constraints (4.3) are satisfied if the nominal system satisfies the constraints $z(j|k) \in \tilde{\mathcal{X}}_j$, with

$$\tilde{\mathcal{X}}_j = \{z \in \mathbb{R}^{n_x} | A_x z \leq \eta_j\}, \quad j \in \mathbb{I}_1^{N_c}. \tag{4.4}$$

Where η_j is given, for $i \in \mathbb{I}_1^{p_c}$, by

$$\eta(j,i) = \max_{\eta} \eta \tag{4.5a}$$

$$\text{s.t. } \mathbb{P}_k\{\eta \leq b_x(i) - A_x(i)e_x(j|k)\} \geq 1 - \varepsilon(i) \tag{4.5b}$$

$A_x \in \mathbb{R}^{p_c \times n_x}$ and $b_x \in \mathbb{R}^{p_c}$ are defined such that $x(j|k) \in \mathcal{X}$ is equivalent to $A_x x(j|k) \leq b_x$. $\varepsilon(i)$ is the probability of violating the linear state constraint i ($\varepsilon \in [0,1]^{p_c}$) [38].

In [78, Section V], a sampling approach is proposed to solve the optimization problem in (4.5). In particular, this can be done because sampling approaches enable the direct use of error measurements (i.e., the difference between the measured values and the predictions), rather than determining a probability density function. This is done by following the Proposition 4.1 (i.e., [78, Proposition 5])

Proposition 4.1

Let N_s and r be chosen according to (4.7). *Let q_{1-r/N_s} be the $(1-r/N_s)$-quantile of the set $\{A_x(i){e_x}^l(j|k)\}_{l=1,\dots,N_s}$ with ${e_x}^i(j|k)\ e_{l|k}^{(i)} = \sum_{j=1}^{l} A_{cl}^{j-1} B_w w_j^{(i)}$ independently chosen samples. Then, with confidence $1-\beta$,*

$$\eta_j(i) = b_x(i) - q_{1-r/N_s} \tag{4.6}$$

solves (4.5) *with $\varepsilon \in [\varepsilon_l, \varepsilon_u]$.*

This is because the optimization problem (4.5) can be solved to the desired accuracy by drawing a sufficiently large number N_s of samples, and the constraints do not hold for a fixed number r of samples,

$$r \leq \varepsilon_u N_s - \sqrt{2\varepsilon_u N_s \ln\frac{1}{\beta}}, \quad r \geq \varepsilon_l N_s - 1 + \sqrt{3\varepsilon_l N_s \ln\frac{2}{\beta}}, \tag{4.7}$$

where N_s is the total number of samples. The satisfaction of the chance constraints holds with a confidence $1-\beta$, and $\varepsilon_l, \varepsilon_u$ are the lower and upper bounds. Thus, the solution to the sampled program is equal to the chance-constrained one with confidence $1-\beta$.

4.2 Chance constraints with the CHoKI for the AP case

As in (4.1), the system describing a T1DM patient can be split into its deterministic and stochastic components (see (3.1)). Then, as in the previous chapter, the output constraints have to be tightened to face the error propagation in the predictions. Their calculation is done once and offline, to obtain $y_{\min}$ and $y_{\max}$ (that is, to get η_j from (4.6)), given the evolution of the uncertainty, to have

$$\mathcal{Y}_j' = \{y : y_{\min}(j) \leq y \leq y_{\max}(j)\}. \tag{4.8}$$

The propagation of the error is obtained by computing c_j and d_j at each step, as reported in Section 2.3.1, in (2.38). The main difference is that, since in this case the constraints tightening is based on the probability distribution of the errors, the computation is done considering all the uncertainty values, to be able to get the distribution at each time step. Once these are obtained, each mean is subtracted from

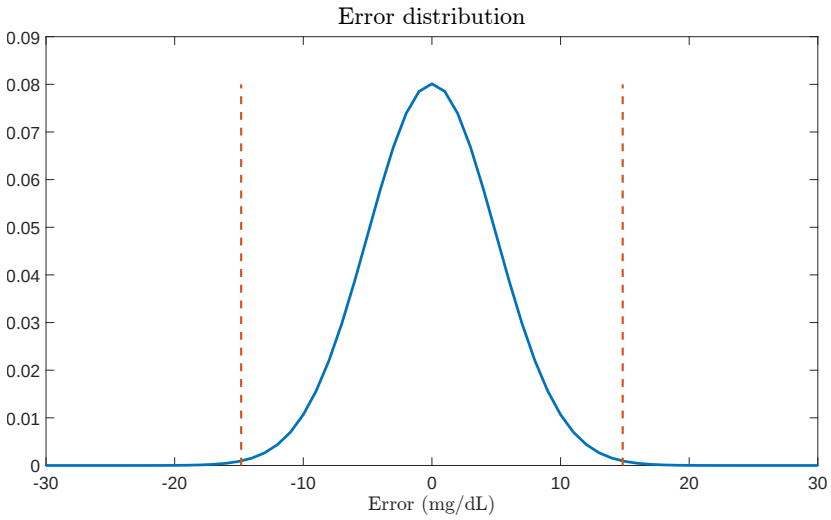


FIGURE 4.1: Fit of Gaussian distribution over the prediction error at the first prediction step (i.e., blue line), for the virtual patient adult 1, truncated at the value corresponding to the 90$^{\text{th}}$ percentile (i.e., red dashed lines).

all the values to fit a normal distribution with zero mean (since in the computation of c_j and d_j the propagation of the error is in absolute value) and this is then truncated at the value corresponding to the 90$^{\text{th}}$ percentile (with $\varepsilon = 0.1$, because of the 10% probability of constraints violation). Then, the $(1 - r/N_s)$-quantile is computed on the truncated uncertainty distribution obtained through the CHoKI, at each step. The entire procedure is repeated for all the considered in-silico patients, with $N_s = 1143$. The values of the 90$^{\text{th}}$ percentiles at the first prediction step are the μ reported in Table 3.2. Note that the satisfaction of the chance constraints holds with a confidence $1 - \beta = 95\%$ (that is, $\varepsilon_l = 0.095$ and $\varepsilon_u = 0.105$), since the constraints are obtained directly from the data, with the sampling approach.

As an example, the computation of the constraints for the virtual patient Adult 1 is reported. The error distribution at the first prediction step is analysed, which is then truncated considering the value corresponding to the 90$^{\text{th}}$ percentile (i.e., $\mu = 14.83$ mg/dL). The errors are assumed to fit a normal distribution with zero mean, reported in Figure 4.1. Obviously, the values of the prediction error increase as the number of step-ahead predictions considered increases, due to the uncertainty propagation. Moreover, the number of samples that do not satisfy the constraints is $r = 121$, which represents 10.6% of the data. It is a bit higher than ε due to the chosen sampling approach, which exploits the data, but allowing to increase its applicability.

Adult	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
N_c	3	1	3	3	3	1	1	3	2	1

TABLE 4.1: Control horizon N_c in the stochastic MPC case

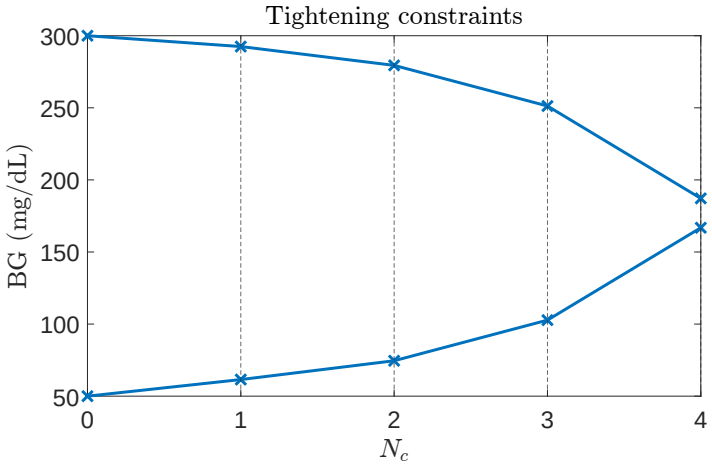


FIGURE 4.2: Tightening of the chance constraints $\mathcal{Y}_j'$ on the BG levels. Adult 1 example.

The values of the tightened constraints also affect the choice of the control horizons N_c (reported in Table 4.1), for each virtual subject. While the prediction horizon is set to $N_p = 12$ for all (to reach 60 min of predictions), for the same reasoning explained in Section 2.3.1. A graphical representation is reported in Figure 4.2, where the tightening chance constraints of Adult 1 are shown as an example. This illustrates how the constraints are shrunk at each time sample (starting from [50 300] mg/dL), allowing also to choose the control horizon $N_c = 3$, to satisfy Assumption 2.6, and still maintaining a reasonable range.

4.3 Optimization problem

For the formulation proposed in this chapter, the MPC optimization problem for the BG management is set as follows:

$$\min_{u_2, y_a, \delta_{\text{hyper}}, \delta_{\text{hypo}}} \quad V_N^s(\hat{x}, u; \Theta, \mathcal{D}) \tag{4.9a}$$

$$\text{s.t.} \quad \hat{x}(0|k) = x(k) \tag{4.9b}$$

$$\hat{x}(j+1|k) = \hat{F}\big(\hat{x}(j|k), u_1(j), u_2(j)\big), \quad j \in \mathbb{I}_0^{N_c-1} \tag{4.9c}$$

$$\hat{x}(j+1|k) = \hat{F}\big(\hat{x}(j|k), K(\bar{x} - x(j)) + \bar{u}\big), \quad j \in \mathbb{I}_{N_c}^{N_p-1} \tag{4.9d}$$

$$\hat{y}(j|k) = M\hat{x}(j|k),\ u_2(j) \in \mathcal{U}, \quad j \in \mathbb{I}_0^{N_p-1} \tag{4.9e}$$

$$\hat{y}(j|k) \in \mathcal{Y}'_j, \quad j \in \mathbb{I}_0^{N_c-1} \tag{4.9f}$$

$$\hat{y}(j|k) \in \mathcal{Y}'_{N_c}, \quad j \in \mathbb{I}_{N_c}^{N_p-1} \tag{4.9g}$$

$$u_1(j) = 0, \quad j \in \mathbb{I}_1^{N_p-1} \tag{4.9h}$$

$$70 - \delta_{\text{hypo}} \leq y_a \leq 140 + \delta_{\text{hyper}} \tag{4.9i}$$

$$\delta_{\text{hyper}} \geq 0,\ \delta_{\text{hypo}} \geq 0 \tag{4.9j}$$

Where the differences with the previous optimization problem in (3.12) are the following:

- the use of the chance constraints on the output, which means to have (4.9f) instead of (3.12f), and (4.9g) instead of (3.12g).
- The absence of (3.12k), since the chance constraints are now considered, and therefore the slack variables $\boldsymbol{\delta} = \{\delta_{\min}, \delta_{\max}\}$ are no longer needed to enlarge the insulin constraints to face feasibility issues.
- As a consequence of the previous point, in the cost function $V_N^s(\hat{x}, u; \Theta, \mathcal{D})$, there is no longer the component V_δ as in (3.18), that was present to penalize the slack variables in the cost (3.13).

Thus, the cost function is built in this way:

$$V_N^s(\hat{x}, u; \Theta, \mathcal{D}) = V_{N_c} + V_{N_p} + V_s + \lambda V_P,$$

Where the stage cost V_{N_c}, along the control horizon, and the cost V_{N_p} from N_c to $N_p - 1$, are defined as

$$V_{N_c} = \sum_{j=0}^{N_c-1} \|\hat{y}(j|k) - y_a\|_Q^2 + \|u_2(j) - u_{\text{ref}}\|_R^2,$$

$$V_{N_p} = \sum_{j=N_c}^{N_p-1} \|\hat{y}(j|k) - y_a\|_Q^2,$$

which are the same as those detailed in (3.14) and (3.15), respectively. The stationary cost V_s is considered to penalize the slack variables $\delta_{\text{hypo}}, \delta_{\text{hyper}}$ added to the constraints of y_a in (4.9i), built as in (3.16):

$$V_s = p_{\text{hyper}}\delta_{\text{hyper}}^2 + p_{\text{hypo}}\delta_{\text{hypo}}^2.$$

The last component is the terminal cost V_P, weighted by a factor $\lambda > 0$, as the one defined in (3.17):

$$V_P = \left\| \hat{x}(N_p|k) - x_{\text{ref}} \right\|_P^2,$$

where P is the solution to the Riccati equation, given the LQR control gain K, and this is weighted by a factor $\lambda > 0$, since no terminal constraint is considered [72]. Moreover, several combinations of the weights were tested, and the definitive ones are the following: $R = 10$, $p_{\text{hypo}} = 1 \cdot 10^7$, $p_{\text{hyper}} = 1 \cdot 10^6$, $\lambda = 10$ and $Q = 1$, except for Adults 8 and 9, for whom $Q = 100$.

As in the previous chapter, the aim is to create a customised controller. This means that, for each analysed virtual patient, a personalised optimization problem is set with corresponding values (i.e., the CHoKI parameters and the MPC settings).

4.4 Simulation results

In this section, the proposed customized MPC with stochastic constraints is applied as the control algorithm for the virtual AP used for the in-silico T1DM adults in the UVA/Padova simulator. The controller is tested through three-day simulations, with the following 15 min duration meals each day: 40 g of carbohydrates at 06:00, 100 g at 12:00 and 60 g at 19:00; the boluses, computed by the bolus calculator of the simulator, are injected 20 min after the meal starts. The simulation settings are the same as those used for data collection (including the ones for the noise), which are the same as the ones detailed in Section 3.2.

The results of the simulations are displayed in Figure 4.3, where the upper graph represents the BG outcomes, while the lower graph shows the basal insulin injections computed by the proposed stochastic MPC. The BG values are mainly inside the euglycemic range (i.e., the green zone), except for some higher values that are caused by the ingestion of carbohydrates (note that the meals occur when there are the black triangles in the upper graph). The exception is Adult 7, who experiences both hyperglycemic and hypoglycemic events, which confirms the control issues discussed in the previous chapter.

Looking only at the BG trends may not be enough, and indeed other important tools are considered for the evaluation of the quality of the insulin-glucose management: CVGA, GRI, and TIR. The CVGA results are shown in Figure 4.4(A). The dots representing the virtual adult patients are mainly located within safe zone B, except for

Adult 6, who is in zone C, and Adult 7, who is in zone D, highlighting the control issues once again. The GRI results are shown in Figure 4.4(B), where all the dots lie on the x-axis, representing that the higher risk component is the hyperglycemic one (which is a good result, since the main goal of the proposal was to reduce the risk of hypoglycemic events). The subjects are mainly inside the optimal zone A, except for Adult 9, who is in zone B, due to the higher time spent in hyperglycemic ranges (as visible in Figure 4.3). The position of Adult 7 in zone C reinforces the need for further analysis. The TIRs results are reported in the upper part of Table 4.2. The TIRs requirements (detailed in 1.3.4) are mostly satisfied since the subjects stay between 70 and 180 mg/dL for more than 70% of the simulation time, according to the ADA specifications (apart from Adults 7 and 8 who also have TIR slightly lower than 70%). An exception occurs for Adults 7, 8 and 9, who are slightly more than 5% over 250 mg/dL. However, they always meet the time requirements for the hypoglycemic ranges, which was the main goal because of its dangerousness.

These results are compared with the continuous basal therapy provided by the simulator, whose results are visualised in Figure 3.6 (with the CVGA in (A) and GRI in (B). While the TIRs are reported in the lower part of Table 4.2 (i.e., the same as the ones shown in 3.6(C)). The benefit of the proposal, particularly in the management of hypoglycemic conditions, is clear. For example, looking at the CVGA, the virtual patients Adults 5 and 9 pass from the Lower D zone to the Upper B zone, which means that the minimum BG value is increased. This is also confirmed by looking at the comparison of the BG mean values of all the in-silico patients plotted in Figure 4.5. The red area (i.e., the one representing the BG with the stochastic MPC) goes higher in the hyperglycemic range, but it never enters the hypoglycemic range (while the blue one, for the continuous therapy, goes below 70 mg/dL), which was the main goal.

4.4.1 Comparison with the results obtained with MPC without chance constraints

To further test these outcomes, the results of the application of the proposed stochastic MPC are compared with those of the previously presented MPC with the slack in the constraints, considering the case of 90^{th} percentile, described in Chapter 3.

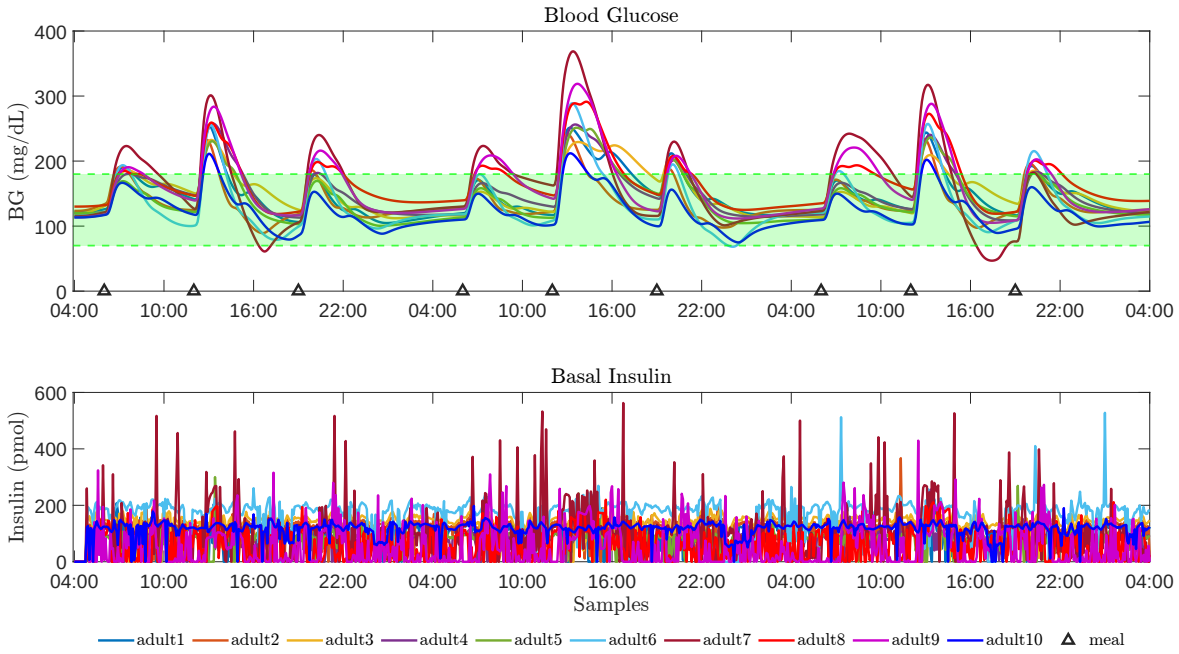


FIGURE 4.3: BG and basal insulin results of the 3-day simulation. The upper plot displays BG trends for all patients. The green zone represents the safe range, and the black triangles depict meals. The lower plot shows basal insulin injections computed by the proposed MPC with stochastic constraints.

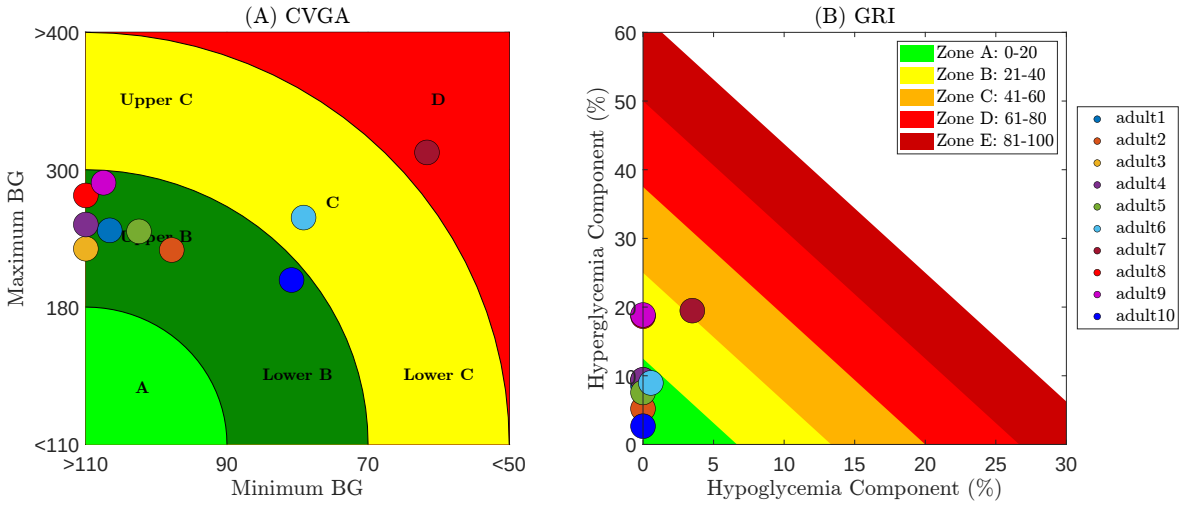


FIGURE 4.4: (A) shows the CVGA results, while (B) shows GRI results of the 3-day simulation, applying the proposed stochastic MPC to the virtual patients of the simulator.

	Adult	< 54 mg/dL	54-70 mg/dL	70-180 mg/dL	180-250 mg/dL	> 250 mg/dL
Proposed stochastic controller	# 1	0%	0%	84%	15%	1%
	# 2	0%	0%	90%	10%	0%
	# 3	0%	0%	81%	19%	0%
	# 4	0%	0%	83%	15%	2%
	# 5	0%	0%	85%	14%	1%
	# 6	0%	1%	84%	12%	3%
	# 7	2%	1%	65%	24%	8%
	# 8	0%	0%	69%	25%	6%
	# 9	0%	0%	70%	23%	7%
	# 10	0%	0%	95%	5%	0%
Constant insulin therapy	# 1	0%	0%	92%	8%	0%
	# 2	0%	2%	91%	7%	0%
	# 3	0%	0%	94%	6%	0%
	# 4	0%	2%	86%	12%	0%
	# 5	14%	7%	75%	4%	0%
	# 6	0%	2%	87%	10%	1%
	# 7	18%	4%	64%	12%	2%
	# 8	0%	0%	77%	18%	5%
	# 9	6%	7%	78%	8%	1%
	# 10	0%	0%	91%	9%	0%

TABLE 4.2: TIRs percentages

Comparison with the results obtained with MPC with slack on the constraints Specifically, they are compared with the BG-insulin results reported in the two upper graphs in Figure 3.4, with the CVGA in Figure 3.5(A) (looking at the dots), with the GRI in Figure 3.5(B) (looking at the dots), and with the TIRs represented in Figure 3.5(C) (looking, for each subject, at the left-hand columns).

In Figure 4.6, the outcomes of Adult 1 are shown as an example. In particular, the BG is plotted in the upper graph, and the basal insulin amounts are plotted in the lower graph. The results of the implementation of the MPC with slack on the BG constraints are represented in blue, while the ones of the MPC with probabilistic constraints are represented in orange. This shows that the use of the stochastic MPC allows a reduction in the BG levels, increasing the TIR. Moreover, the implementation of the chance constraints enables an increase of the control horizon, from $N_c = 2$ (in Table 3.2) to $N_c = 3$ (in Table 4.1). However, this is not a significant advantage, as the increase achieved in the horizon is small, due to the conservative strategy of the CHoKI tightening restrictions.

Figure 4.7 shows the comparison between the mean and standard deviation of the BG values of all the considered 10 in-silico patients. The blue area represents the outcomes of the previous chapter, while the red area represents the outcomes of the

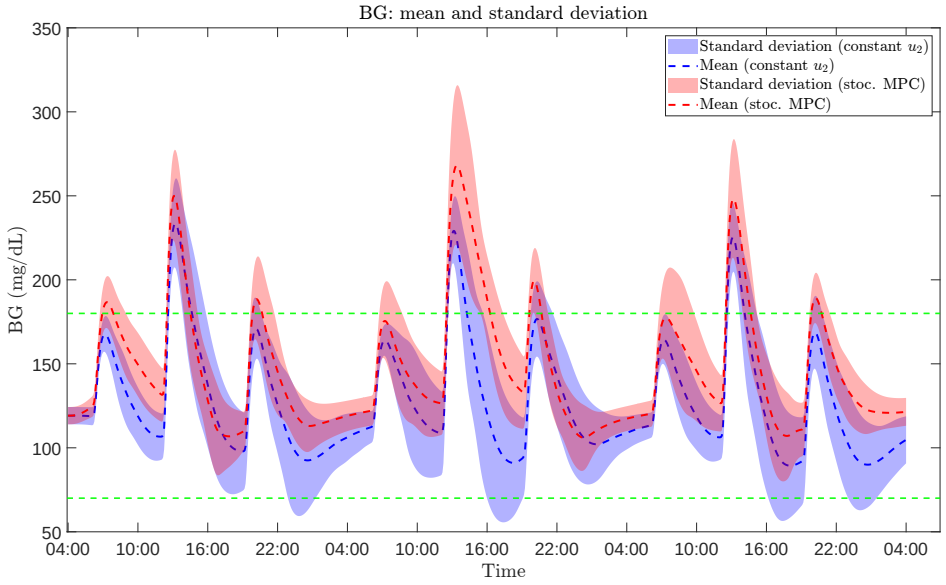


FIGURE 4.5: Comparison of the BG values: the mean and the standard deviation of the simulations performed exploiting the stochastic MPC controller are represented in red, while the outcomes exploiting the continuous basal insulin therapy of the simulator are drawn in blue.

MPC with chance constraints. The use of stochastic constraints allows for an average reduction in BG levels, reducing the TARs, which is also visible in the average TIRs values reported in Figure 4.8.

To simplify the comparison, the results in Figure 3.5 and those in Figure 4.4 are reported in a single image, in Figure 4.9 (where the dots represent the outcomes of the MPC with the chance constraints, while the squares represent the ones with the slack in the constraints). The results are similar, even though the ones of the stochastic case are slightly improved.

Comparison with the results obtained with MPC with slack on the constraints and with IOB In Section 3.3, an additional constraint that considers the IOB was added to the optimization problem (3.23), and the exponential IOB estimation is detailed in Section 3.3.2. This constraint is also added to the stochastic MPC, by including in the optimization problem (4.9) the new insulin constraint (3.23f), the constraints on the slack variable (3.23m) used in the insulin limitation, and the cost component (3.25).

This is then applied to the in-silico adult T1DM patients of the UVA/Padova simulator, with the same simulation setting detailed in the Section 3.3.2, in order to enable

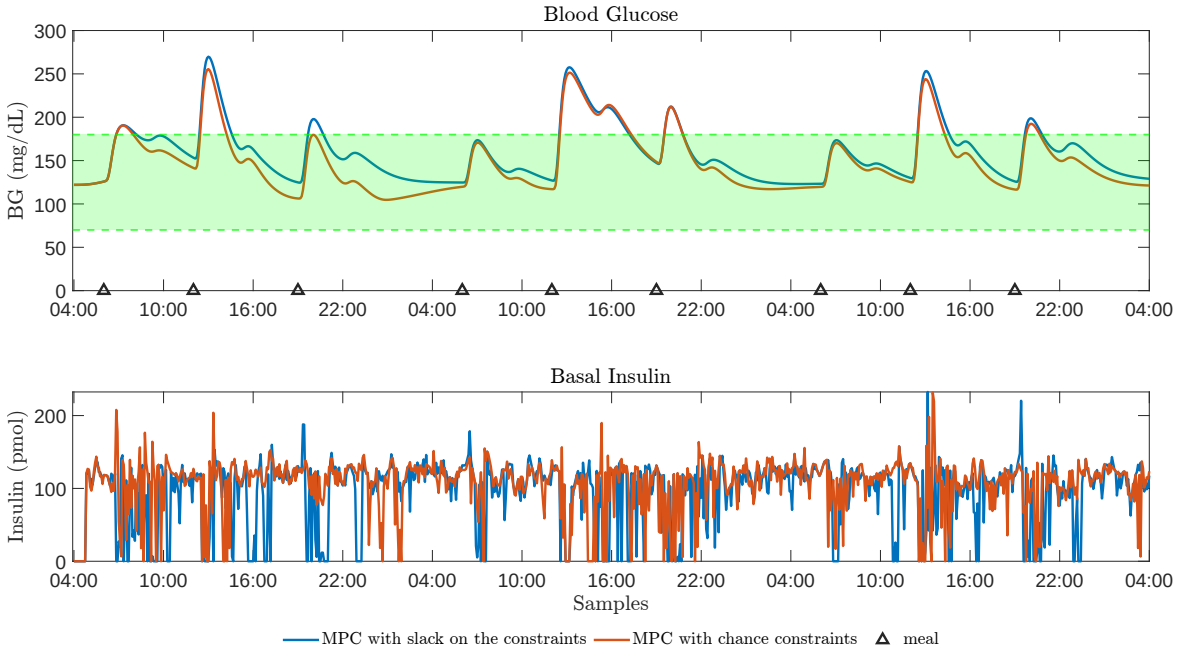


FIGURE 4.6: The upper graph represents the BG level, where the outcomes of the MPC with the slack on the BG constraints are drawn in blue, and the one with the chance constraints are drawn in orange. The lower graph represents the basal insulin injections in the two cases. Adult 1 example.

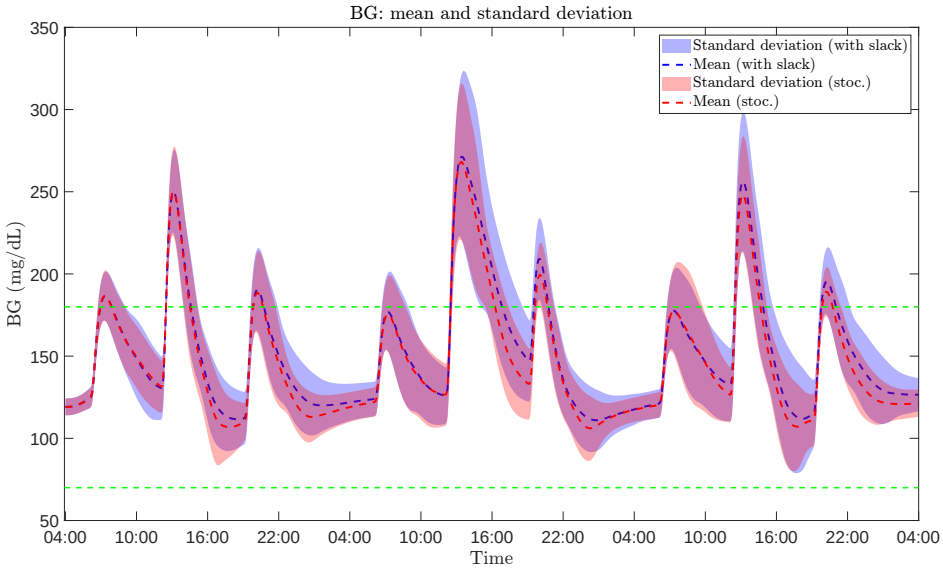


FIGURE 4.7: Comparison of the BG values: the mean and the standard deviation of the simulations performed exploiting the stochastic MPC controller are represented in red, while the outcomes exploiting the slack on the constraints are represented in blue.

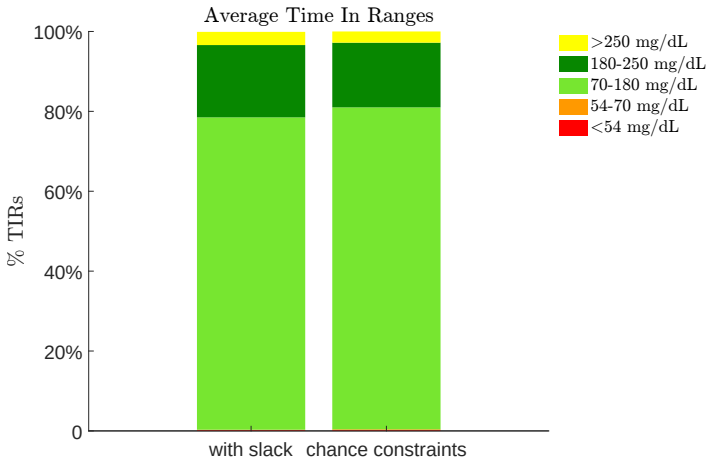


FIGURE 4.8: Comparison of the average TIRs values: the mean for the simulations performed exploiting the stochastic MPC controller is on the right, while the mean of the outcomes exploiting the slack on the constraints is on the left.

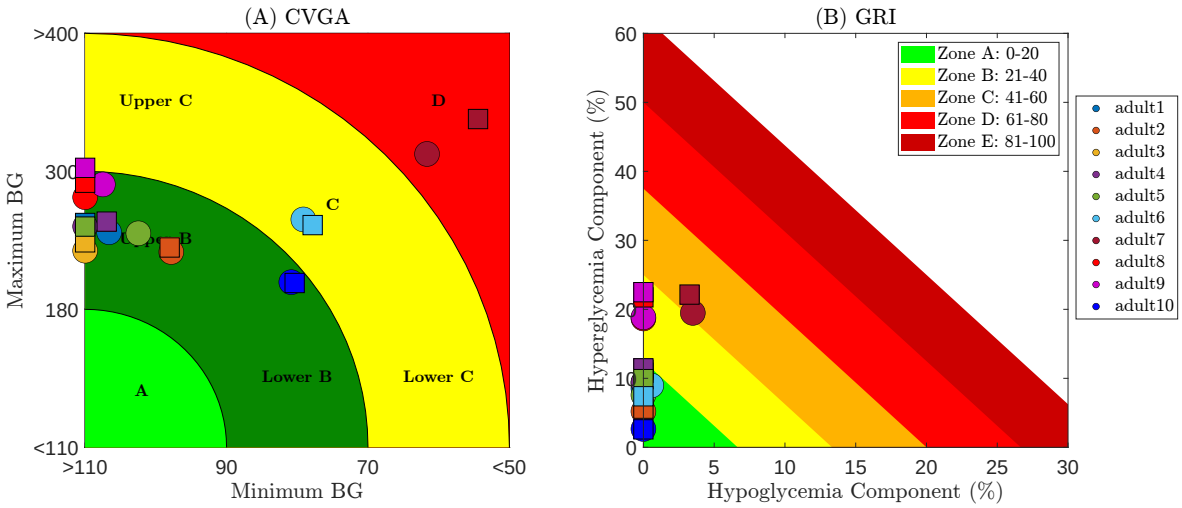


FIGURE 4.9: (A) and (B) shows the CVGA and GRI results, respectively. The squares represent the case with the slack, and the dots represent the stochastic case.

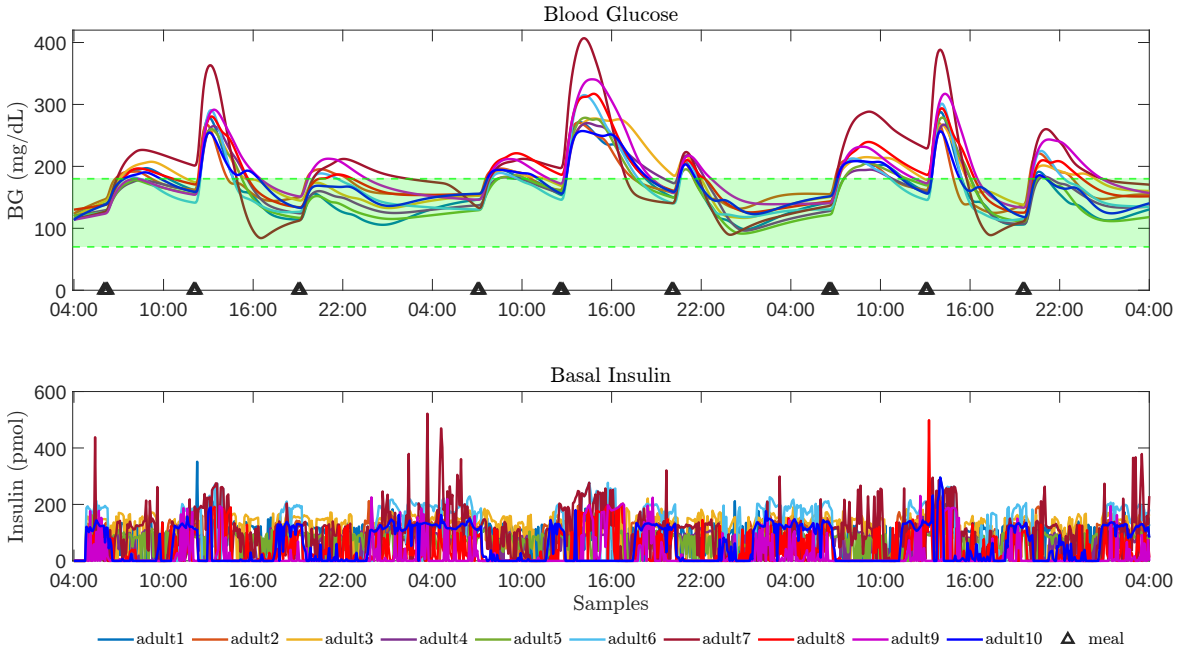


FIGURE 4.10: BG and basal insulin results with the stochastic MPC, with the simulation setting as in Section 3.3.2. The upper plot displays BG trends for all patients. The green zone represents the safe range, and the black triangles depict meals. The lower plot shows basal insulin injections.

the comparison with those results (i.e., considering 3-day simulations with circadian variability in S_I and different meals per day). The BG and insulin outcomes are reported in Figure 4.10. The results are quite good, as there are no episodes of hypoglycemia; however, this comes at the cost of high peaks of hyperglycemia after meals. This can be due to the constraints on the IOB, which make the controller more conservative, limiting the maximum values of insulin action.

As in the previous case, it is useful to observe the mean and standard deviation of patients' BG levels, comparing the results of the two proposed controllers with (depicted in red) and without (depicted in blue) stochastic constraints, as shown in Figure 4.11. This confirms that the values remain too much in the hyperglycemic ranges, but the use of stochastic MPC actually manages to increase the times in TIR and decrease those in TAR, as can be seen in Figure 4.12, which shows the average TIRs values. It can be seen that in the right-hand column (the one with stochastic MPC), the percentages improve slightly compared to the left-hand column (the one with slack on the constraints, reported in Table 3.8).

These results are also compared to the ones represented in Figure 3.21. And to ease the comparison, the CVGA and GRI results in Figure 3.22 are represented as squares

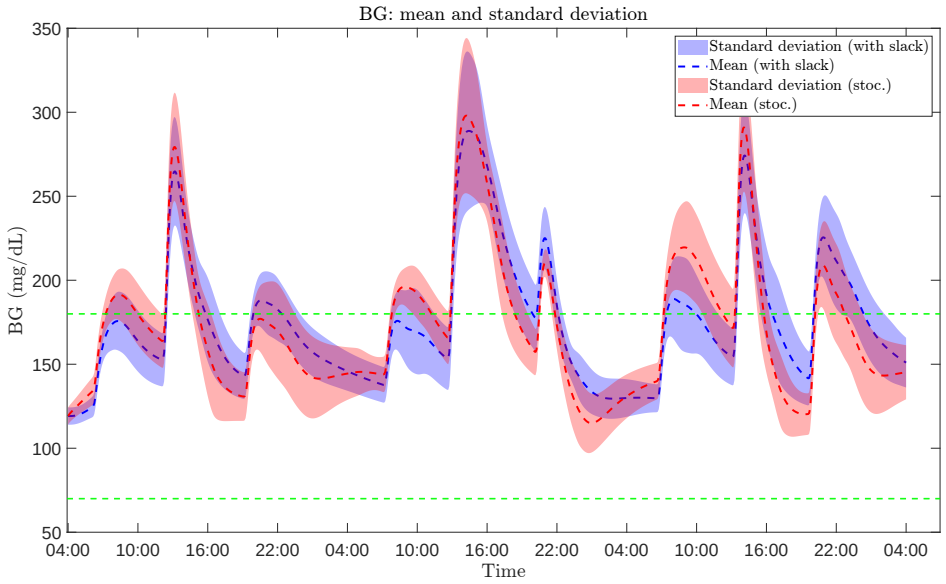


FIGURE 4.11: Comparison of the BG values: the mean and the standard deviation of the simulations performed exploiting the stochastic MPC controller are represented in red, while the outcomes exploiting the slack on the constraints are represented in blue. Considering the simulations with circadian variability in S_I and different meals per day.

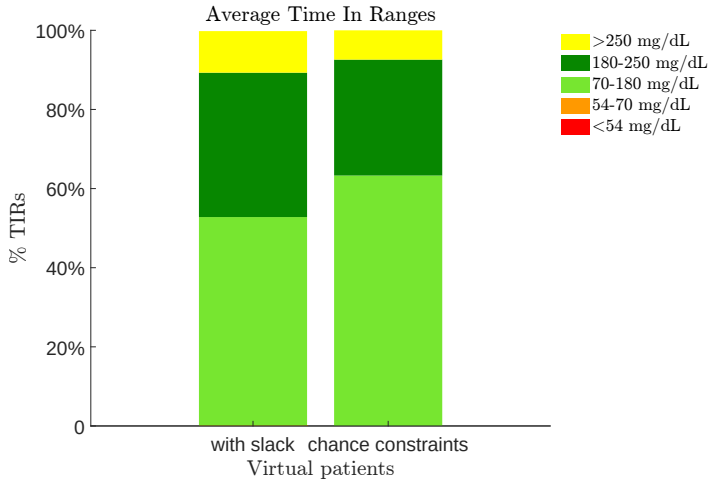


FIGURE 4.12: Comparison of the average TIRs values: the mean for the simulations performed exploiting the stochastic MPC controller is on the right, while the mean of the outcomes exploiting the slack on the constraints is on the left.

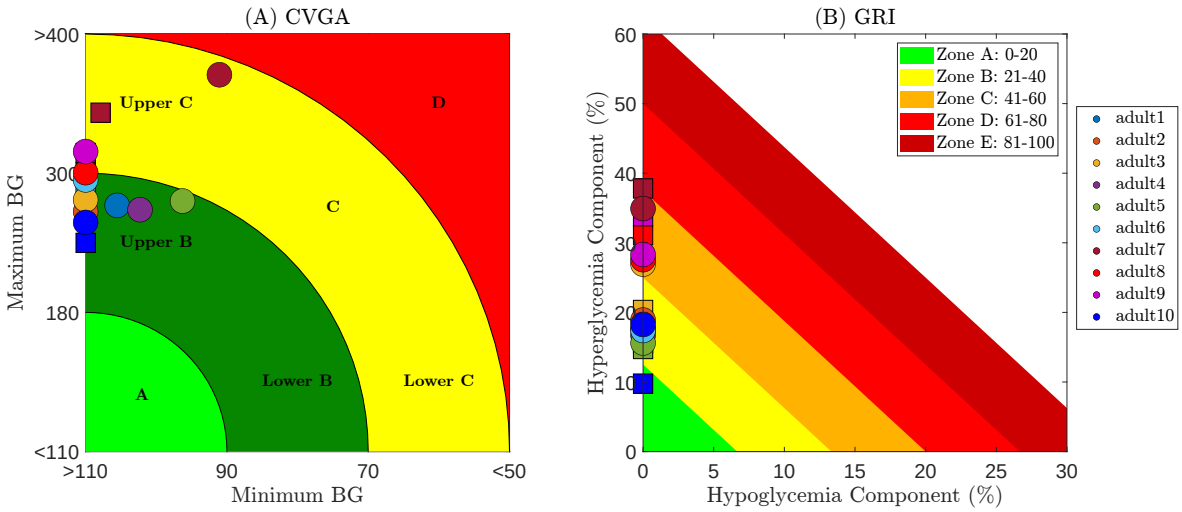


FIGURE 4.13: (A) and (B) shows the CVGA and GRI results, respectively. The squares represent the case with the slack, and the dots represent the stochastic case.

in Figure 4.13, while the dots are for the previously described stochastic MPC. The results are actually quite similar. Both tools confirm the issues in the hyperglycemia management. Looking at the GRI in (B), all the points are on the y-axis, highlighting that the highest risk is the hyperglycemic components, but that have high values; in fact, the points are in zones C and D. While looking at the CVGA in (A), all the points are in zone B except for Adults 7 and 9, which are in zone C, but still quite high due to the maximum BG value.

4.5 Conclusions

In this chapter, a CHoKI-based MPC with stochastic constraints tightening was proposed as a control algorithm of the AP, with the aim of driving and maintaining the T1DM patients' BG values inside the euglycemic range, updating the basal insulin injections amount every 5 min. The additional IOB constraint is also considered. The application of the proposed controllers to the virtual patients of the UVA/Padova simulator was presented. The results are quite promising, and the main outcome is that it reduces the hypoglycemic events, which is fundamental due to their dangerousness.

The comparisons with the constant basal insulin therapy and with the MPC versions without stochastic constraints detailed in Chapter 3 are performed. Specifically, when compared to the standard therapy provided by the simulator, there is

a significant improvement in BG management, as the proposal eliminates hypoglycemic events. While, there is a slight improvement also when compared to the MPC with slack on constraints. Not only because the control horizons increase a little, but also because of the increases in the TIR percentages and the decreases in the TARs ones.

Although the proposed strategy works, in practice, the widening of the constraints has not been achieved as much as expected. This is due to the CHoKI tightening constraints approach, which remains too conservative, even in the case of probabilistic constraints.

Therefore, future work will focus on studying the recursive feasibility and stability of the proposed controller. It will also involve searching for alternative methods of managing prediction uncertainty, in order to try to extend the horizons of control further.

CHAPTER 5

Online learning robust CHoKI-based MPC

The aim of this chapter is to describe how to improve the CHoKI learning strategy online, which is done by updating the dataset used by this method to enhance system predictions.

As already stated before, T1DM is a disease that varies significantly inter- and intra-patients, and also over time. The idea developed in this chapter is based on finding a possible solution to face this aspect. The controllers proposed in previous chapters were based on the CHoKI learning method, in which the parameters $\mathcal{L}, \mathcal{P}$ were estimated for each subject only once, and the used dataset $\mathcal{D}$ was fixed. This means that the method is static, while the idea here is to develop an approach that is able to capture all the variations over time. For this reason, a possible suggestion is to update the dataset with more recent data (thanks to the availability of new information over time, when the controller is applied to the system), which allows for learning about the patient's glucose behavior and being able to follow possible changes in the subject.

Briefly, the concept is to start by following the work in [84], in which new data may be included in the database used for the KI learning method online. To do that, an exploration and exploitation policy is proposed. Specifically, the *exploration* is used to measure how far from the known workspace the system is allowed to move. The *exploitation* is used to address the fact that obtaining a large dataset may not be a computationally efficient strategy. In addition, in [87], a *double prediction* approach is proposed. On the one hand, a safe prediction is computed, using KI methods, considering the identification dataset selected offline, which guarantees the safety of the

controlled system (that is, following what was explained in Chapter 2). On the other hand, the controller also benefits from the use of a second online learning-based prediction, since new measurements become available incrementally over time. Sufficient conditions for robust stability and constraint satisfaction are given in [87].

The two works mentioned above are both based on the LACKI learning method, while here these are extended to cope with the CHoKI approach.

5.1 Online LACKI updating strategy

In this section, a brief summary of the general theory behind the proposed online approach is shown, considering the LACKI case, and assuming $p = 1$ [84, 87].

5.1.1 Exploration and Exploitation strategy

Exploration The exploration strategy is used to measure how far from the known workspace the system is allowed to move. In particular, this is used to stay close to observed data, in a safe region where there is enough information to guarantee an upper bound on the worst-case prediction error. This safe region $\mathcal{W}_r$ is defined as:

$$\mathcal{W}_r = \left\{ w : \min \left(\|w - w_i\| \right) \leq \tau_r, \ \forall i \in \mathbb{I}_1^{N_\mathcal{D}} \right\}, \tag{5.1}$$

for a certain threshold $\tau_r \geq 0$, and recalling that w is the regressor and $\mathcal{D} = \{y_{k+1}, w_k\}$, for $k = 1, \ldots, N_\mathcal{D} - 1$.

The prediction error μ is related to the threshold distance τ_r, since it is bounded by $\mu = L_f \tau_r + 2\bar{\epsilon}$. This is based on the true Lipschitz constant L_f, which is assumed to be known and equal to the estimated constant L. This is based on the KI predictor in (2.18), with LACKI estimation of the Lipschitz constant [84].

Exploitation The exploitation is used to address the fact that obtaining a large dataset may not be a computationally efficient strategy. Based on Theorem 2.2, it is possible to state that the prediction error $\left| f(w) - \hat{f}(w; \Theta, \mathcal{D}) \right|$ vanishes when the density of the dataset becomes infinite, up to $2\bar{\epsilon}$. However, in practice, adding data points to the model increases computation times. Therefore, it may not be good to include every new observed data point. For this reason, the implementation of an exploitation policy is proposed, which adds only informative data points, the ones that are not close to data points already present in the dataset.

This can be seen as establishing a new threshold for the distance. This means that a new data point (y, w) is not informative if it belongs to the well-known region $\mathcal{W}_t$, which is defined as:

$$\mathcal{W}_t = \left\{ w : \min \left(\|w - w_i\| \right) \leq \tau_t,\ \forall i \in \mathbb{I}_1^{N_\mathcal{D}} \right\}, \tag{5.2}$$

for a threshold $0 \leq \tau_t \leq \tau_r$, and it has to be appropriately chosen, according to the information provided by the fact that (y, w) is added to $\mathcal{D}$.

Based on the well-known region, the update policy can be written as follows:

$$\mathcal{D}(k+1) = \begin{cases} \mathcal{D}(k) & \text{if } w(k) \in \mathcal{W}_t(k) \\ \mathcal{D}(k) \cup (y(k+1), w(k)) & \text{if } w(k) \notin \mathcal{W}_t(k) \end{cases} \tag{5.3}$$

meaning that, at time $k+1$, the new point $(y(k+1), w(k))$ is added to the dataset $\mathcal{D}(k)$ (i.e., starting from the initial dataset $\mathcal{D}(0)$) if it does not belong to $\mathcal{W}_t$ [84].

5.1.2 MPC formulation with double prediction approach

As stated in Chapter 2, the KI learning methods require a dataset to be implemented. This initial dataset $\mathcal{D}(0)$ is available in the design phase, performed offline, and this can be obtained via specific experiments or by using historical data. Then, once the controller has been designed and applied to the system, new measurements $\tilde{y}(k)$ become available during its execution. This allows the dataset to be updated up to the current time step k, producing $\mathcal{D}(k)$. Similar to what is done in [7], the proposed controller will use two different prediction models: one to guarantee safety and one to improve the performance.

The first is the safe model $\hat{y}_s$, defined as:

$$\hat{y}_s(k+1) = \hat{\mathfrak{f}}_s\left(x(k), u(k)\right) = \hat{\mathfrak{f}}\left(x(k), u(k); L_\mathcal{D}, \mathcal{D}(0)\right), \tag{5.4}$$

where $\hat{\mathfrak{f}}(\tilde{w}; L_\mathcal{D}, \mathcal{D})$ is the LACKI predictor, obtained applying KI with $\mathcal{D}(0)$ and $L_\mathcal{D}$ obtained using the LACKI method. The state-space version is $\hat{x}_s(k+1) = \hat{F}_s\left(x(k), u(k)\right)$.

As before, to prove robust stability, it is assumed that a guaranteed bound of the estimation error is determined (that is, the prediction error bound is obtained in

practice via validation tests), as stated in the following assumption.

Assumption 5.1

The prediction error of the safe model $e_s(\cdot) \in \mathbb{R}^{n_y}$, which depends on the dataset $\mathcal{D}(0)$ and the estimated Lipschitz constant $L_{\mathcal{D}}$, is bounded by some known $\mu \in \mathbb{R}^{n_y}$. That is, for all admissible (x, u),

$$e_s(k) = \left| y(k+1) - \hat{\mathfrak{f}}_s(w(k); L_{\mathcal{D}}, \mathcal{D}(0)) \right| \leq \mu. \tag{5.5}$$

According to Assumption 5.1, the real output $f(w)$ lies in a ball centered in $\hat{\mathfrak{f}}_s(w(k))$ and width μ, which is defined as the set

$$\mathcal{Y}_s(x, u) = \{\mathfrak{f}_s(x, u) \oplus \mathcal{B}(\mu)\}. \tag{5.6}$$

The second is the online model $\hat{y}_p$, defined as:

$$\hat{y}_p(k+1) = \hat{\mathfrak{f}}_p(x(k), u(k)) = \text{Proj}_{\mathcal{Y}_s(x(k),u(k))}\left(\hat{\mathfrak{f}}(x(k), u(k); L_{\mathcal{D}}, \mathcal{D}(0))\right), \tag{5.7}$$

that provides the prediction with the updated dataset $\mathcal{D}(k)$, as long as it is in contained in $\mathcal{Y}_s$. If not, a guaranteed prediction is obtained by projection, according to what was aforementioned [87].

Optimization problem The general formulation of an MPC optimization problem $P_N^{\text{on}}(x(k); L, \mathcal{D}(0), \mathcal{D}(k))$ with a double prediction approach is the following:

$$\min_u V_N^{\text{on}}(k) = \sum_{j=0}^{N-1} \ell(\hat{y}_p(j|k), u(j)) + \lambda V_f(\hat{x}_p(N|k)) \tag{5.8a}$$

$$\text{s.t.} \quad \hat{x}_p(0|k) = x(k) \tag{5.8b}$$

$$\hat{x}_p(j+1|k) = \hat{F}_p(\hat{x}_p(j|k), u(j)),\ j \in \mathbb{I}_0^{N-1} \tag{5.8c}$$

$$\hat{y}_p(j|k) = M\hat{x}_p(j|k),\ j \in \mathbb{I}_0^{N-1} \tag{5.8d}$$

$$\hat{x}_s(0|k) = x(k) \tag{5.8e}$$

$$\hat{x}_s = \hat{F}_s(\hat{x}_s(j|k), u(j)),\ j \in \mathbb{I}_0^{N-1} \tag{5.8f}$$

$$\hat{y}_s(j|k) = M\hat{x}_s(j|k),\ j \in \mathbb{I}_1^{N} \tag{5.8g}$$

$$u(j) \in \mathcal{U},\ j \in \mathbb{I}_0^{N-1} \tag{5.8h}$$

$$\hat{y}_s(j|k) \in \mathcal{Y}_j,\ j \in \mathbb{I}_0^{N-1} \tag{5.8i}$$

where the subscript p is for the prediction part considering $\mathcal{D}(k)$ (i.e., equations (5.8b)-(5.8d)), and the subscript s is for the safe model considering $\mathcal{D}(0)$ (i.e., equations (5.8e)-(5.8i)) [7, 87]. Thus, the upper part of (5.8) is related to the online model in (5.7), focused on improving the performance, while the lower part is related to the offline model in (5.4), to guarantee safety.

Assumption 5.2

1. *The stage cost function $\ell(y,u)$ is a continuous positive definite function $\forall y \in \mathcal{Y}$ and $u \in \mathcal{U}$ such that $\ell(y,u) \geq \alpha_y(\|y\|) + \alpha_u(\|u\|)$ and*

$$|\ell(y_1,u) - \ell(y_2,u)| \leq \theta_\ell(\|(y_1,u) - (y_2,u)\|), \tag{5.9}$$

   for certain $\mathcal{K}$-functions $\theta_\ell, \alpha_y, \alpha_u$.
2. *There exists a local control law $u = \kappa_f(x)$, a terminal cost function V_f and a set $\Omega_\gamma = \{x : V_f(x) \leq \gamma\} \subseteq \mathbb{R}^{n_x}$, for $\gamma > 0$ such that for all $x \in \Omega_\gamma$ the following conditions holds:*
   - $$\kappa_f(x) \in \mathcal{U}, \tag{5.10a}$$
   $$Mx \oplus \mathcal{B}(a_N) \subseteq \mathcal{Y}_N, \tag{5.10b}$$
   where a_j is given by the recursion $a_{j+1} = L\|a_j\| + \mu_p$, with $a_1 = \mu_p$.
   - *V_f is a continuous positive definite function such that for certain $\mathcal{K}$-functions $\alpha_f, \beta_f, \theta_f$, $|V_f(x_1) - V_f(x_2)| \leq \theta_f(\|x_1 - x_2\|)$ and*

$$\alpha_f(\|x\|) \leq V_f(x) \leq \beta_f(\|x\|), \tag{5.11a}$$

$$V_f\left(\hat{F}_p(x,\kappa_f(x))\right) - V_f(x) \leq \ell(Mx,\kappa_f(x)). \tag{5.11b}$$

Assumption 5.3

The set

$$\mathrm{Y} = \{x : \ell(Mx,0) \leq \theta_a(2\mu_p)\} \tag{5.12}$$

is contained in Ω_γ.

5.2 Updating criteria policy

Another possibility to evaluate the addition of a new data point to the dataset is by considering the satisfaction of some conditions [87]. Specifically, as before, at each sampling time instant $k+1$ a new data point is obtained $(y(k+1), w(k))$, and it is added to the dataset $\bar{\mathcal{D}}(k)$,

$$\bar{\mathcal{D}}(k+1) = \mathcal{D}(k) \cup (y(k+1), w(k)),$$

if the following three conditions are true.

1. *Condition 1.* The current data point $(y(k+1), w(k))$ must not be close to any point already contained in $\mathcal{D}(k)$, for a given threshold $\tau \geq 0$. This rule prevents $\mathcal{D}$ from becoming too large, which would make the method computationally expensive. A reasonable estimate for τ is the noise level $\bar{e}$. This condition can be written as:

$$\min_{w_i \in \mathcal{W}_{\mathcal{D}(k)}} \|w(k) - w_i\| > \tau. \tag{5.13}$$

   This is the same as considering the well-known region $\mathcal{W}_t$ in (5.2), by setting $\tau = \tau_t$.

2. *Condition 2.* Adding the new data point must not increase the cost calculated with the shifted input sequence. This condition is satisfied if the following equation is true:

$$\bar{V}_N^{\text{on}}\big(x(k), \bar{\mathbf{u}}; \bar{\mathcal{D}}(k)\big) \leq \bar{V}_N^{\text{on}}\big(x(k), \bar{\mathbf{u}}; \mathcal{D}(k-1)\big). \tag{5.14}$$

   where the shifted sequence $\bar{\mathbf{u}}$ is given by:

$$\bar{u}(j|k) = u^*(j+1|k-1),\ j \in \mathbb{I}_0^{N-2},$$

$$\bar{u}(N|k) = \kappa_f(\hat{x}_p(N|k-1)).$$

3. *Condition 3.* The new candidate $(y(k+1), w(k))$ must be consistent with the Lipschitz constant obtained before. Which means that it has to satisfy:

$$\max_{i \in \mathbb{I}_1^{N_{\mathcal{D}(k)}}} \frac{|y(k) - y_i| - 2\bar{e}}{\|w(k) - w_i\|} \leq L_{\mathcal{D}} \tag{5.15}$$

Remark 5.1

In the proposed closed-loop algorithm, at each time sample k, the new system information $(y(k+1), w(k))$ is read, and the latest data point is added to the dataset, only if it satisfies the updating criteria policy. The optimization problem is then solved, and the first element of the optimal output is applied to the system.

In Theorem 5.1 (proved in [87, Chapter 5]) is stated that the proposed control algorithm (described in Remark 5.1) guarantees that the closed-loop system is ISS.

Theorem 5.1

Consider that Assumptions 5.1-5.3 hold, and let $\kappa_{MPC}(x)$ be the control law derived from the solution of $P_N^{\text{on}}(k)$ applied using this algorithm. Then, for any feasible state $x(0) \in \Gamma$, the system controlled by the control law $u(k) = \kappa_{MPC}(x(k))$ is ISS w.r.t. the estimation error $e_s(k)$, and the constraints are fulfilled along the operation, i.e., $y(k) \in \mathcal{Y}$ and $u(k) \in \mathcal{U}, \forall k$.

5.2.1 Updating criteria with CHoKI

In the previous section, the updating criteria were defined based on the LACKI approach, while here the conditions (5.13)-(5.15) are substituted with their version that considers the CHoKI method.

First, recall that the value of the CHoKI predictor $\hat{f}(q; \Theta, \mathcal{D})$ for a query $q \in \mathbb{R}^{n_w}$ is computed as in (2.22):

$$\hat{f}(q;\Theta,\mathcal{D}) = \frac{1}{2} \min_{i=1,\dots,N_{\mathcal{D}}} \left(\tilde{y}_i + \mathfrak{d}_{\mathcal{L}}^{\mathcal{P}} \left(|q - w_i| \right) \right) + \frac{1}{2} \max_{i=1,\dots,N_{\mathcal{D}}} \left(\tilde{y}_i - \mathfrak{d}_{\mathcal{L}}^{\mathcal{P}} \left(|q - w_i| \right) \right),$$

where $\Theta = \{\mathcal{L}, \mathcal{P}\}$ and $\mathfrak{d}_{\mathcal{L}}^{\mathcal{P}}(w) := \left(a : a_i = \sum_{j=1}^{n_w} \mathcal{L}_{i,j} w_j^{\mathcal{P}_{i,j}}, \forall i \in \mathbb{I}_1^{n_y} \right)$, assuming that f is Hölder continuous and given a dataset $\mathcal{D}$ of $N_{\mathcal{D}}$ observations.

Thus, the updating criteria become the following:

1. *Condition 1.* The condition (5.13) now becomes the following:

$$\exists j \in \mathbb{I}_1^{n_w} : \min_{w_i \in \mathcal{W}_{\mathcal{D}(k)}} \left| w_j(k) - w_{i,j} \right| > \tau_j, \;\; \tau \in \mathbb{R}^{n_w}. \tag{5.16}$$

Which means that this condition is true if the distance between each component $j \in \mathbb{I}_1^{n_w}$ of $w_j(k)$ and $w_{i,j}$ is lower than τ, at least for one of the regressor components.

2. *Condition 2.* The condition (5.14) is basically the same, with

$$\bar{V}_N^{\text{on}}\big(x(k), \bar{\mathbf{u}}; \bar{\mathcal{D}}(k)\big) \leq \bar{V}_N^{\text{on}}\big(x(k), \bar{\mathbf{u}}; \bar{\mathcal{D}}(k-1)\big). \tag{5.17}$$

to ensure that the cost function does not increase when adding the new data point.

3. *Condition 3.* The condition (5.15) now considers the CHoKI computation of the matrix $\mathcal{L}$ (i.e., solving the optimization problem in (2.24)). Thus, it can be written as:

$$\max_{i \in \mathbb{I}_1^{N_{\mathcal{D}}(k)}} \frac{|y(k) - y_i| - 2\bar{e}}{\mathfrak{d}_{\mathcal{L}}^{\mathcal{P}}(|w(k) - w_i|)} \leq \mathbb{1}_{n_y}, \tag{5.18}$$

where, differently from before, on the right-hand side of the equation, there is the vector of ones, denoted by $\mathbb{1}_{n_y}$, since the $\mathcal{L}$ value is already included in the denominator on the left-hand side.

5.2.2 Updating criteria for the T1DM patient case

The definition of the insulin-glucose system of the patient is the same as the one detailed in the Chapter 3.

The three conditions (5.16)-(5.18) are here applied to the management of the BG case, and therefore they need to be properly defined. Specifically,

1. *Condition 1.* The first condition (5.16) could be considered satisfied if one of the components of the regressor satisfies the condition, thus when the distance between the two points in one dimension is greater than the component of the τ which corresponds to it. The τ is made of different values depending of the memory horizons n_a, n_b, n_c, built as

$$\tau = [\tau_a \mathbb{1}_{n_a}; \tau_b \mathbb{1}_{n_b}; \tau_c \mathbb{1}_{n_c}] \in \mathbb{R}^{n_w} \tag{5.19}$$

where τ_a refers to the error in the BG sensor measurements, τ_b refers to a consistent variation in meal amount, and τ_c refers to the error in the insulin pump injections. A posteriori analyses showed that possible reasonable values are the following: $\tau_a = 0.5$ mg/dL, $\tau_b = 999$ mg, and $\tau_c = 1$ pmol. Figure 5.1

represents the well-known regions $\mathcal{W}_t$ from (5.2) (based on the values of τ_t, plotting only BG and insulin).

2. *Condition 2.* The second condition (5.17) is computed considering as $\bar{\mathbf{u}}$ the optimal control action computed by the MPC in the previous time step.

3. *Condition 3.* The third condition (5.18) is needed to avoid the updating of the $\mathcal{L}$, since it verifies that the new data point $(y(k+1), w(k))$ is consistent with the estimated Lipschitz matrix.

Optimization problem Thus, the optimization problem $P_N^{\text{on}}(x(k); \mathcal{L}, \mathcal{D}(0), \mathcal{D}(k))$ becomes the one in (5.20). This is obtained by considering the previously proposed optimization problem in (3.23), but adapting it with the shape of the online CHoKI (i.e., with the optimization problem in (5.8)).

$$\min_{u_2, y_a, \delta_{\text{hyper}}, \delta_{\text{hypo}}, \boldsymbol{\delta}, \delta_u} V_N^{\text{on}}(\hat{x}, u; \Theta, \mathcal{D}) \tag{5.20a}$$

s.t.

$$\hat{x}_p(0|k) = x(k) \tag{5.20b}$$

$$\hat{x}_p(j+1|k) = \hat{F}_p\big(\hat{x}_p(j|k), u_1(j), u_2(j)\big),\ j \in \mathbb{I}_0^{N_c-1} \tag{5.20c}$$

$$\hat{x}_p(j+1|k) = \hat{F}_p\big(\hat{x}_p(j|k), K(\bar{x} - x_p(j)) + \bar{u}\big),\ j \in \mathbb{I}_{N_c}^{N_p-1} \tag{5.20d}$$

$$\hat{y}_p(j|k) = M\hat{x}_p(j|k),\ j \in \mathbb{I}_0^{N_p-1} \tag{5.20e}$$

$$\hat{x}_s(0|k) = x(k) \tag{5.20f}$$

$$\hat{x}_s(j+1|k) = \hat{F}_s\big(\hat{x}_s(j|k), u_1(j), u_2(j)\big),\ j \in \mathbb{I}_0^{N_c-1} \tag{5.20g}$$

$$\hat{x}_s(j+1|k) = \hat{F}_s\big(\hat{x}_s(j|k), K(\bar{x} - x_s(j)) + \bar{u}\big),\ j \in \mathbb{I}_{N_c}^{N_p-1} \tag{5.20h}$$

$$\hat{y}_s(j|k) = M\hat{x}_s(j|k),\ j \in \mathbb{I}_0^{N_p-1} \tag{5.20i}$$

$$u_2(j) \in \mathcal{U}_2',\ j \in \mathbb{I}_0^{N_p-1}, \tag{5.20j}$$

$$\hat{y}(j|k) \in \mathcal{Y}_{j,\boldsymbol{\delta}},\ j \in \mathbb{I}_0^{N_c-1} \tag{5.20k}$$

$$\hat{y}(j|k) \in \mathcal{Y}_{N_c,\boldsymbol{\delta}},\ j \in \mathbb{I}_{N_c}^{N_p-1} \tag{5.20l}$$

$$u_1(j) = 0,\ j \in \mathbb{I}_1^{N_p-1} \tag{5.20m}$$

$$70 - \delta_{\text{hypo}} \leq y_a \leq 140 + \delta_{\text{hyper}} \tag{5.20n}$$

$$\delta_{\text{hyper}} \geq 0,\ \delta_{\text{hypo}} \geq 0 \tag{5.20o}$$

$$\delta_{\min}(j) \geq 0,\ \delta_{\max}(j) \geq 0,\ j \in \mathbb{I}_0^{N_p-1} \tag{5.20p}$$

$$\delta_u(j) \geq 0,\ j \in \mathbb{I}_0^{N_p-1} \tag{5.20q}$$

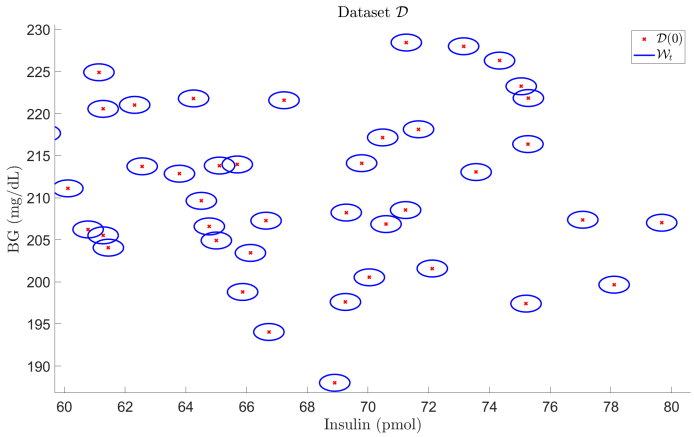


FIGURE 5.1: Input-output representation of the dataset $\mathcal{D}(0)$ (for Adult 1). On the x-axis, there is basal insulin, and on the y-axis, there are BG values. The red crosses represent the original dataset $\mathcal{D}(0)$ and the blue circles represent the well-known area $\mathcal{W}_t$.

Here, from (5.20a) to (5.20e), the online part (that is, with the subscript p) is reported, focused on improving the prediction performances, by considering the online updated dataset $\mathcal{D}(k)$ and thus the online model (i.e., with (5.20c), (5.20d), (5.20e)). Conversely, from (5.20f) to (5.20q), the part used to guarantee safety (that is, with the subscript s) is reported. Where the initial dataset built offline $\mathcal{D}(0)$, and thus the safe model (i.e., (5.20g), (5.20h), (5.20i)) are considered. Also, all the constraints are included in this part, i.e., from (5.20j) to (5.20q). (5.20b) and (5.20f) guarantee that the initial state is the same with both models.

5.3 Simulation Results

In this chapter, the initial dataset $\mathcal{D}(0)$ is made of 500 data points per patient, performing the k-means clustering on the initial dataset $\mathcal{D}$ used in Section 3.3.2. Specifically, the rows of the original dataset $\mathcal{D}$ were randomly shuffled, and the k-means clustering was performed to divide the data points into 500 clusters. Then, for each cluster, the one point closest to the cluster centroid was selected, to obtain the 500 most representative data points [124].

The CHoKI parameters are the same as the ones in Chapter 3. The simulation settings for the UVA/Padova simulator are the ones detailed in Section 3.3.2, which are also reported here for the sake of completeness. Three days were simulated, with circadian variability of S_I, and with the following announced meals per day: (i) 20 g

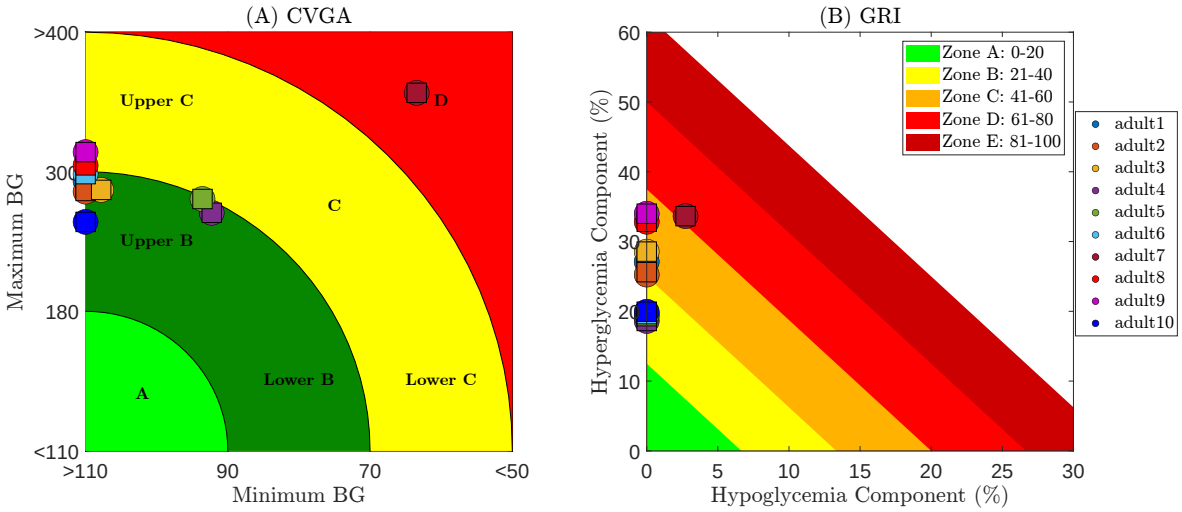


FIGURE 5.2: (A) CVGA and (B) GRI results of the 3-day simulation results. The dots represent the cases with dataset $\mathcal{D}(0)$, while the squares represent the cases with online dataset $\mathcal{D}(k)$ updating.

at 06:00, 90 g at 12:00, and 30 g at 19:00 for the first day; (ii) 30 g at 07:00, 80 g at 12:30, and 50 g at 20:00 for the second day; and (iii) 40 g at 06:30, 100 g at 13:00, and 60 g at 19:30 for the last day. The meal bolus amounts were computed by the bolus calculator of the simulator and were injected 20 min after the meal's start. All the devices have the same noise setting as in the data collection stage (detailed in Section 3.2.1), including the carbohydrate estimation error with a normal distribution with a standard deviation equal to 30% of the meal amount, for the bolus computation.

The simulations were performed considering the updating of the dataset $\mathcal{D}(k)$ and also by maintaining the fixed dataset $\mathcal{D}(0)$, to make the comparison. Figure 5.2 shows the CVGA and GRI results, for comparing the cases with dataset $\mathcal{D}(0)$ (depicted as dots) with the cases with online dataset $\mathcal{D}(k)$ updating (depicted as squares). The outcomes are very similar, as shown also in Figure 5.3, which represents BG mean and standard deviation of the in-silico patients (in blue the cases with $\mathcal{D}(0)$, and in red the cases with $\mathcal{D}(k)$), which are basically overlapping.

The results are then compared with the ones shown in Figure 3.16 (where the BG and basal insulin values are plotted), with the GRI and CVGA in Figure 3.17, and TIR in Table 3.7. In that case, the dataset $\mathcal{D}$ used inside the CHoKI is made of $N_{\mathcal{D}}$ data, the same used for the identification of the model (i.e., $N_{\mathcal{D}}$ reported in Table 3.2). To have a comparison, the mean BG values are shown in Figure 5.4. The red area displays the cases with online dataset $\mathcal{D}(k)$ updating, while the blue area displays the cases

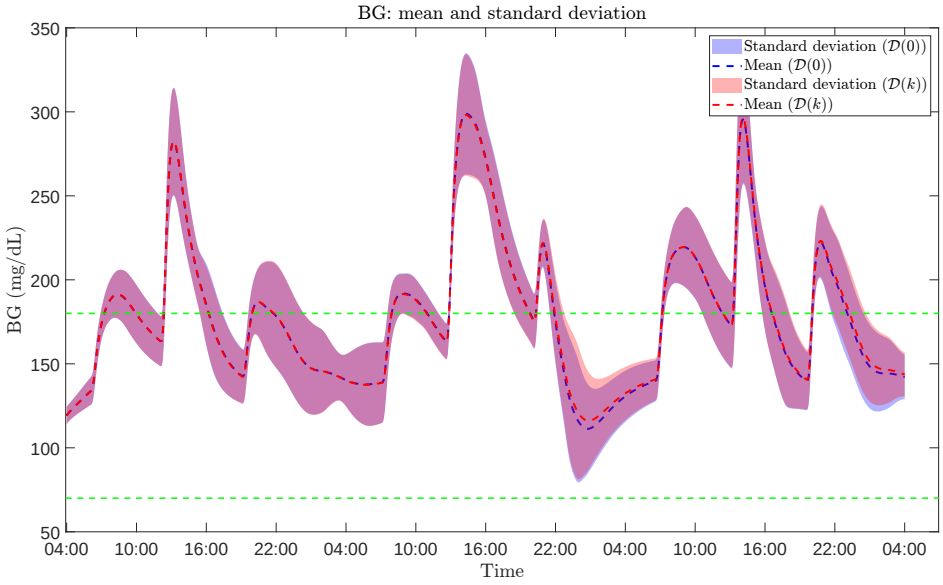


FIGURE 5.3: Mean and standard deviation of the BG values. The cases with dataset $\mathcal{D}(0)$ are represented in blue, while the cases with online updating dataset $\mathcal{D}(k)$ are represented in red.

Adult	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	mean value
Time with updating $\mathcal{D}(k)$	40.3	30.9	45.2	46.3	45.9	38.6	33.9	38.9	34.2	35.4	38.9
Time with $\mathcal{D}(0)$ with dimension $N_{\mathcal{D}}$	227.5	252.3	274.2	278.3	246.7	245.1	219.8	283.6	201.5	233.5	246.2

TABLE 5.1: Solving MPC optimization problem times (in seconds). The first line represents the mean times for solving the MPC at each step with the proposed online dataset updating, while the second row represents the mean times for the ones presented in Chapter 3. Recalling the sampling time of 5 min (300 s).

without dataset updating (and with the initial dataset $\mathcal{D}(0)$ with dimension $N_{\mathcal{D}}$). Also in this comparison, the results are similar, even though the red area (the one for the online updating) goes a bit more in hyperglycemic conditions.

The main outcome here is that decreasing the dimension of the initial dataset allows for a huge reduction in the time required for solving the optimization problem. In Table 5.1, the mean values for solving the MPC at each sampling time are reported, where the first row is for the online updating CHoKI-based MPC, while the second row is for the results shown in Section 3.3.2. In both cases, the times are less than the sampling time of 300 s (i.e., 5 min), but the online version is much faster, even though the update procedure is added, moving from an average value of 246.2 s to 38.9 s.

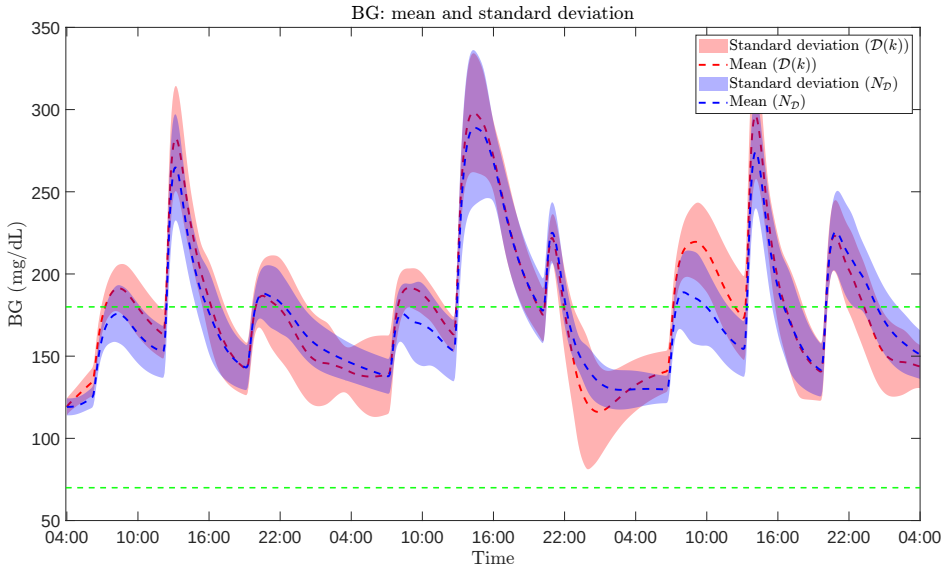


FIGURE 5.4: Mean and standard deviation of the BG values. The cases with online updating dataset $\mathcal{D}(k)$ are drawn in red, while the cases shown in Figure 3.16 are drawn in blue.

5.3.1 Results of 14-day simulations, with bolus 20 min after the meal

The outcomes presented in the previous section are related to simulations with a 3-day duration, but the dataset increment in this period may not be enough to see an improvement in the control. Therefore, in this section, simulations of 14 days are proposed. In particular, an improvement in the prediction model is also proposed, with the boluses given 20 min before the meals. This requires to make a new data collection, a new identification model, and then the application to the in-silico patients in the UVA/Padova simulator.

Data collection and model identification

The data collection was done on the in-silico patients of the UVA/Padova simulator, performing the simulations with the same noise settings as in Chapter 3. The main difference is that, in this case, the boluses are given 20 min before the meals, trying to mimic the standard therapy. As before, in the bolus calculation, a standard deviation of the error of 30% of the meal size is considered. Moreover, the circadian variability of the S_I is activated. The data collection is made of 4 days, with the following meals: (i) in the first day, 30 g at 07:00, 80 g at 12:30, 50 g at 19:00; (ii) in the second day, 40

Adult	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
$[L_a; L_b; L_c]$	[0.8159 0.0045 0.0045]	[0.9164 0.0013 0.0030]	[1.3952 0.0010 0.0082]	[1.2143 0.0010 0.0010]	[1.1593 0.0012 0.0048]	[1.3480 0.0010 0.0010]	[9.9607 0.2778 0.0692]	[0.8700 0.0010 0.0010]	[2.8462 0.0075 0.0045]	[0.9041 0.0023 0.0014]
N_c	3	3	2	3	3	2	1	3	3	3
μ	9.1	9.27	8.98	7.94	8.96	9.13	12.31	10.27	9.66	10.81

TABLE 5.2: Identified values of Lipschitz matrix $\mathcal{L} = [L_a \mathbb{1}_{n_a}; L_b \mathbb{1}_{n_b}; L_c \mathbb{1}_{n_c}]$, control horizons N_c, and 90$^{\text{th}}$ percentile of the validation error μ. For the case with S_I and boluses injected 20 min before the meals.

g at 08:00, 60 g at 13:30, 40 g at 19:30; (iii) in the third day, 20 g at 06:30, 70 g at 13:00, 60 g at 20:30; and (iv) in the fourth day, 10 g at 06:00, 100 g at 12:15, 30 g at 20:00.

The Lipschitz matrices $\mathcal{L}$ are identified as detailed in Section 3.2.2, and the final results are reported in Table 5.2 (recall that $\mathcal{L} = [L_a \mathbb{1}_{n_a}; L_b \mathbb{1}_{n_b}; L_c \mathbb{1}_{n_c}] \in \mathbb{R}^{n_y \times n_w}$). As a consequence, also the control horizons N_c may change, according to the 90$^{\text{th}}$ percentile of the error (i.e., μ) in the validation, which are reported in Table 5.2. The validation dataset was built as the one in Chapter 3, but moving the boluses 20 min before the meals, and adding the circadian variability of S_I.

Simulation results

Once the CHoKI parameters and MPC settings are defined, the controllers can be applied to the in-silico patients of the UVA/Padova simulator. As in the previous section, the initial dataset $\mathcal{D}(0)$ is made of 500 points, selected with the k-means clustering, from the 4-day dataset used in the identification phase. To see the improvements of including new data in the dataset, thanks to the online updating procedure, 14-day simulations were performed (i.e., by implementing the MPC optimization problem (5.20)). The meals for the 14-days are the following: on Monday, 20 g at 06:00, 90 g at 12:00, 30 g at 19:00; on Tuesday 30 g at 07:00, 80 g at 12:30, 50 g at 20:00; on Wednesay 40 g at 06:30, 100 g at 13:00, 60 g at 19:30; on Thursday, 35 g at 07:00, 75 g at 12:00, 25 g at 19:00; on Friday, 15 g at 06:30, 125 g at 12:30, 20 g at 20:30; on Saturday, 30 g at 08:30, 60 g at 13:30, 45 g at 20:00; and on Sunday, 20 g at 09:00, 45 g at 13:30, 65 g at 19:30.

The BG and basal insulin amounts obtained with the application of the MPC online updating CHoKI are reported in Figure 5.5. The BG values are mainly inside the safe range, except for some hyperglycemic events after the meals, which confirms what was also seen in the results of the previous chapters, even though the bolus given

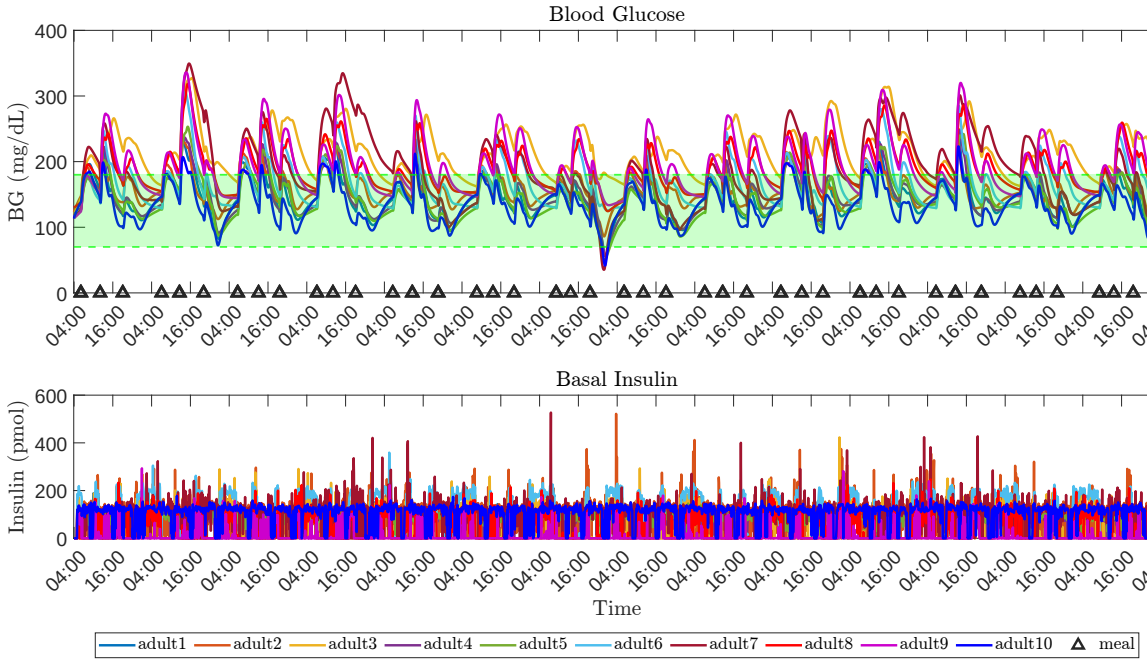


FIGURE 5.5: The upper plot displays BG trends for all patients, the green zone represents the safe range, and the black triangles depict meals. The lower plot shows basal insulin injections computed by the proposed MPC. These are the results of 14-day simulations with online CHoKI updating.

before the meal helps by slightly reducing the peaks. These results can be compared with those obtained without updating the dataset (i.e., by maintaining $\mathcal{D}(0)$), whose average BG values are plotted in Figure 5.6. There, the outcomes obtained with $\mathcal{D}(0)$ are represented in blue, while the ones with the updated dataset $\mathcal{D}(k)$ are represented in red. On average, the results are similar, the use of $\mathcal{D}(k)$ allows for a slight reduction in the TAR in the last days. It is also possible to look at the CVGA and GRI results to make the comparison, which are reported in Figure 5.7, where the dots represent the $\mathcal{D}(0)$ cases and the squares represent the $\mathcal{D}(k)$ cases. The results of the two cases are quite similar. For the CVGA all the patients are in the safe zone B, except for Adult 7, who is in zone C (even with the use of the online strategy, the dark red square is closer to zone B). The GRI results show that the hypoglycemic risk component is basically zero for all the patients (which was the main goal), but the hyperglycemic risk components tend to be too high for the Adults 3, 7, 8, and 9. This is linked to the conservatism of the CHoKI and the overly restrictive constraint on the IOB (which were discussed in the previous chapters).

To analyse the updating criteria, as an example, the data of Adult 1 are reported. In Figure 5.8 the input-output representation is reported (considering the BG and

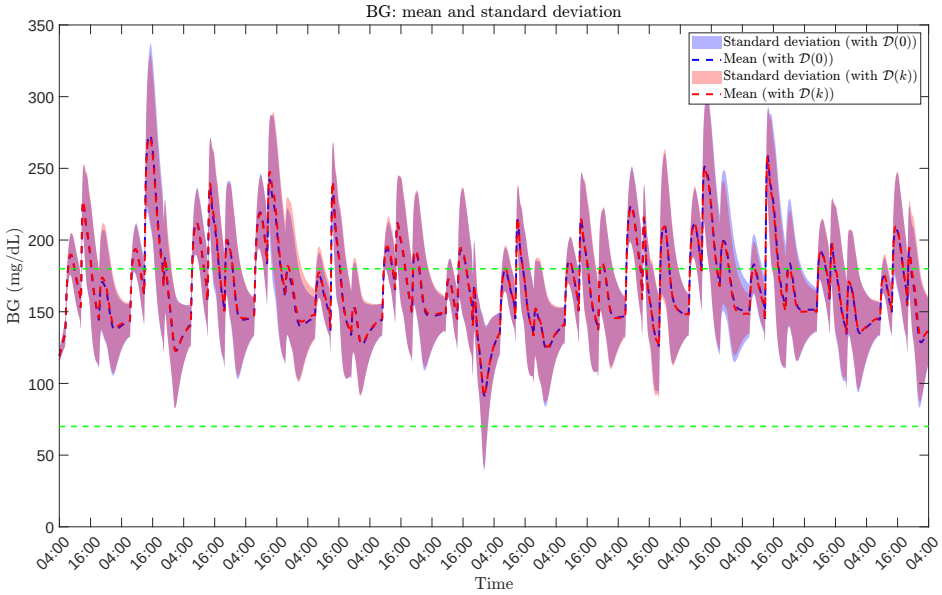


FIGURE 5.6: Mean values of the BG values. The red area represents the outcomes with online updating of $\mathcal{D}(k)$, while the blue area represents the outcomes with $\mathcal{D}(0)$.

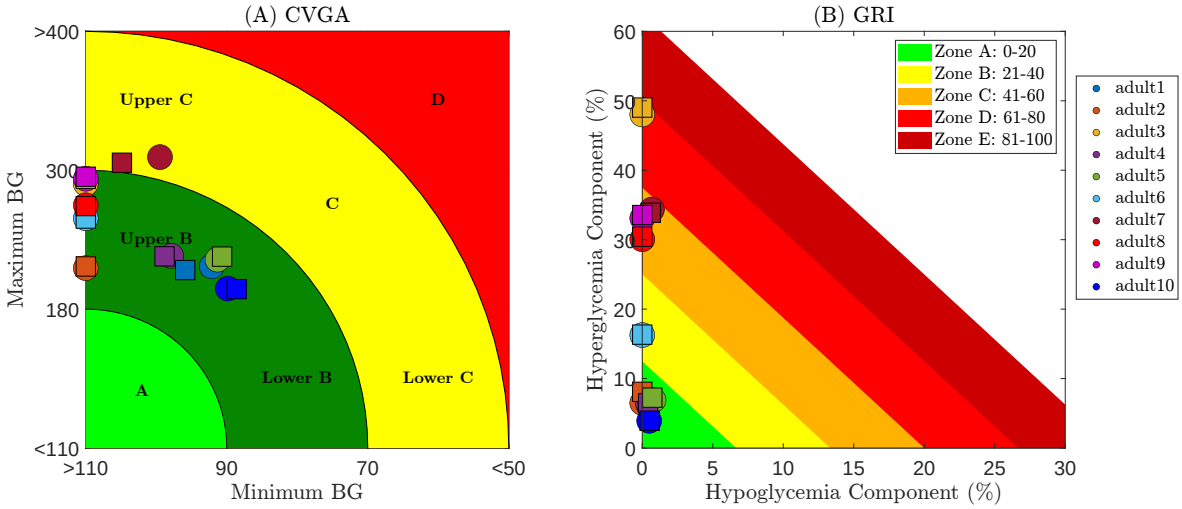


FIGURE 5.7: (A) CVGA and GRI results. The dots represent the simulations without dataset updating (i.e., with $\mathcal{D}(0)$), while the squares represent the cases with $\mathcal{D}(k)$ updating.

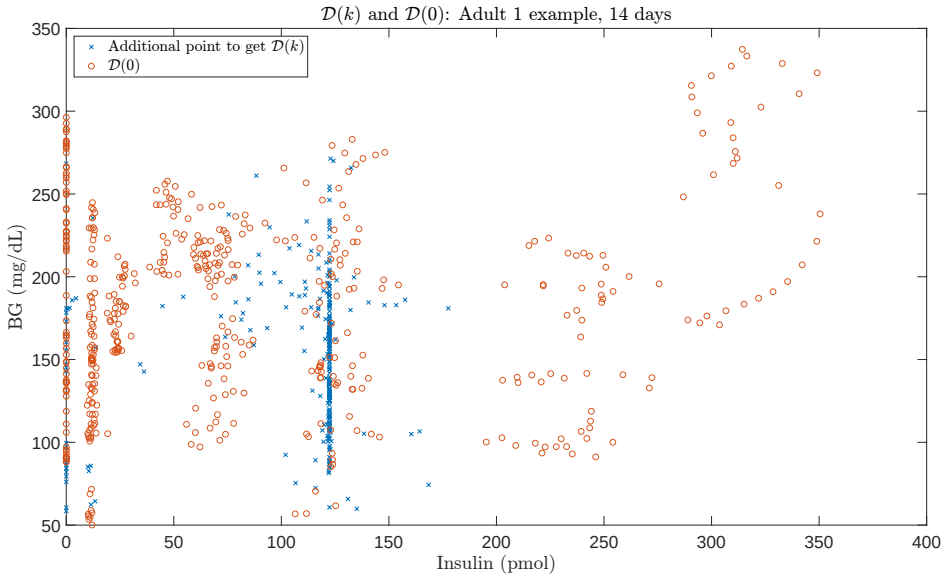


FIGURE 5.8: Input-output representation of the updating strategy (for Adult 1). The basal insulin is represented on the x-axis, while the BG values are represented on the y-axis. The orange dots belong to the original dataset $\mathcal{D}(0)$ (with 500 data points), while the blue crosses are the additional data points included during a simulation of 14 days (i.e., $\mathcal{D}(k)$ contains 787 points).

insulin), to show the outcomes of the updating strategy, where the orange dots are the data points of the initial dataset $\mathcal{D}(0)$, and the included data points are the blue crosses. This allows to obtain the dataset $\mathcal{D}(k)$. To visualise all the components of the new data points (y, w) Figure 5.9 shows the BG, meals (CHO) and basal insulin values of both $\mathcal{D}(0)$ and all the added points to reach $\mathcal{D}(k)$.

Moreover, in Figure 5.10 the BG and basal insulin values of Adult 1 are reported, for the 14-day simulations with 4 different controllers: in blue the case without updating the dataset (i.e., with $\mathcal{D}(0)$), in orange the case with online updating $\mathcal{D}(k)$, in purple the case without updating the dataset, but taking from the beginning the entire dataset obtained with the previous online updating (i.e., considering the $\mathcal{D}(k)$ obtained in the orange case, and using it as $\mathcal{D}(0)$ without updating it), and in yellow using constant basal insulin therapy. The results of the CHoKI-based MPC cases (i.e., blue, orange, and purple cases) are better than the constant basal insulin therapy. The purple case is the best one, according to Theorem 2.2, since the prediction error decreases when increasing the density of data.

The same analysis is now shown for patient Adult 7 to determine whether online updates to the dataset improve the identification of subject variations, which was the main issue in previous control attempts discussed in past chapters. In Figure 5.11,

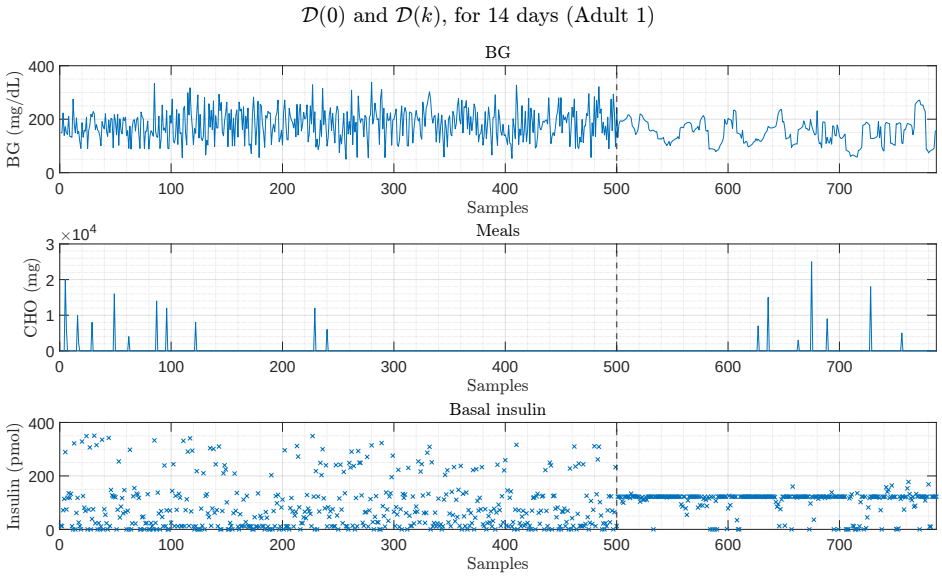


FIGURE 5.9: Updating dataset approach example. The upper graph represents the BG values, the middle one shows the meal amounts, and the lower one shows the basal insulin injections. The dashed vertical black line divides the $\mathcal{D}(0)$ (on the left, with 500 data points) from the additional points to get $\mathcal{D}(k)$ (on the right). Example for Adult 1, with 14 days of simulation.

the BG and basal insulin outcomes of the four control strategies are represented. According to what was also said for Adult 1, by increasing the data density, the prediction error decreases and, as a result, the controller also improves, as can be seen by looking at the purple line (i.e., the one obtained using the dataset reached at the end of the 14-day online updating $\mathcal{D}(0) = \mathcal{D}(k)$). Figure 5.12 shows the TIRs percentages in the four cases. This confirms the too high percentages in the hyperglycemic ranges (the TARs), but the hypoglycemic events are reduced, and updating the dataset online helps to control Adult 7, because it can better adapt to its variations.

5.4 Conclusions

Due to the complexity of the T1DM disease, with all the variations over time (such as, S_I variability, circadian rhythms, physical activity sessions, patient growth, stressful situations, ...), an online learning robust CHoKI-based MPC is proposed in this chapter. This is obtained thanks to the fact that new measurements become available incrementally over time when the controller is applied to the system. Thus, an updating criterion is defined, with the aim of selecting only the most useful data

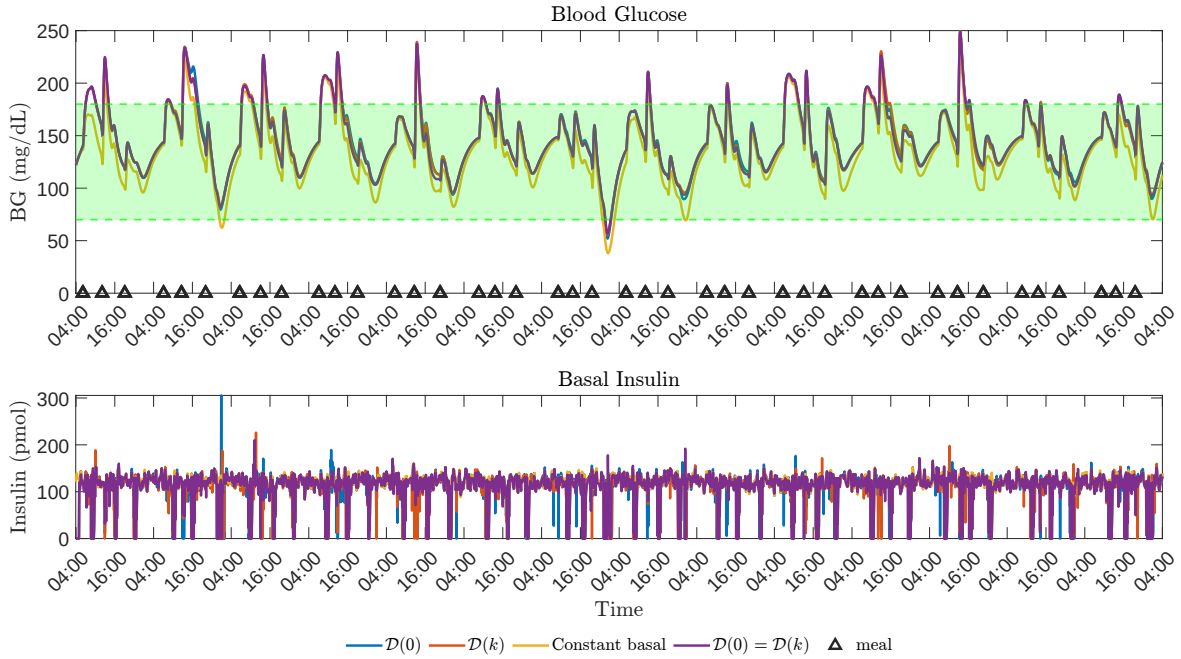


FIGURE 5.10: Adult 1 case. The BG values are plotted in the upper graph, while the basal insulin amounts are plotted in the lower graph. The case without updating the dataset $\mathcal{D}(0)$ is drawn in blue, the case with online updating $\mathcal{D}(k)$ is drawn in orange, the case without updating the dataset but using the one obtained at the end of the 14-day simulation with online updating (i.e., using the dataset obtained in the orange case as $\mathcal{D}(0) = \mathcal{D}(k)$ without updating it) is drawn in purple, and the case using constant basal insulin therapy is drawn in yellow.

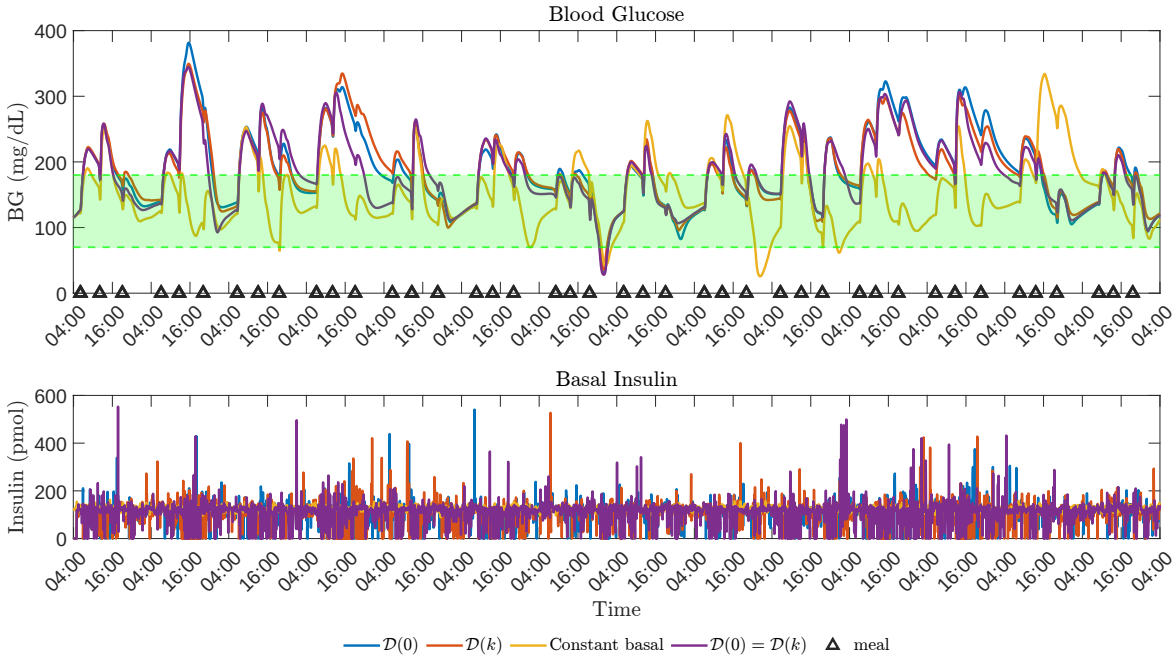


FIGURE 5.11: Adult 7 case. The BG values are plotted in the upper graph, while the basal insulin amounts are plotted in the lower graph. The case without updating the dataset $\mathcal{D}(0)$ is drawn in blue, the case with online updating $\mathcal{D}(k)$ is drawn in orange, the case without updating the dataset but using the one obtained at the end of the 14-day simulation with online updating (i.e., using the dataset obtained in the orange case as $\mathcal{D}(0) = \mathcal{D}(k)$ without updating it) is drawn in purple, and the case using constant basal insulin therapy is drawn in yellow.

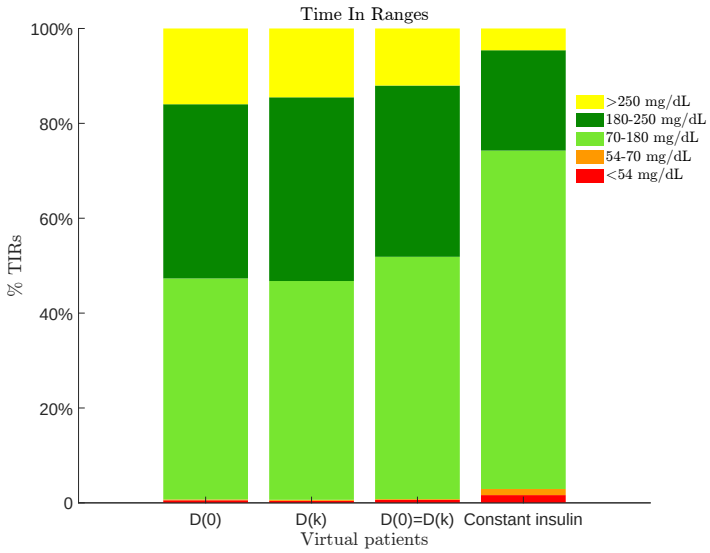


FIGURE 5.12: Adult 7 case. Mean values of the TIRs percentages of the virtual patients. The first column represents the cases without updating the dataset $\mathcal{D}(0)$, the second one represents the cases with online updating $\mathcal{D}(k)$, the third column represents the cases without updating the dataset but using the one obtained at the end of the 14-day simulation with online updating it ($\mathcal{D}(0) = \mathcal{D}(k)$), and the last column represents the cases using constant basal insulin therapy.

to add to the dataset used in the CHoKI, in order to avoid overloading the dataset and consequently lengthening resolution times. A double prediction approach is then implemented, merging the use of an offline model to guarantee safety, with the online model used to improve the prediction performance.

The customized controllers were then tested on the virtual adult patients of the UVA/Padova simulator. The results are quite satisfactory, as the controllers perform well even when working with a smaller amount of data in the initial CHoKI dataset. Moreover, it allows for a reduction in the MPC solving time with respect to the controllers proposed in Chapter 3.

However, the values of τ need to be studied further in order to achieve the right trade-off between adding information and keeping the time required to find the MPC solution low. In addition, more analyses of the well-known region are required, to better define the exploration policy. Also, updating the dataset online did not reduce the conservativeness of the CHoKI method, which still tends to have short control horizons. The limitation due to the conservatism of the IOB constraint remains in this case, too, leading to excessive levels of hypoglycemia.

The most promising result concerns the BG control of Adult 7. As discussed in previous chapters, this virtual patient exhibited significant variability, making it difficult to identify an effective control strategy, which was confirmed by the application of all previously proposed MPC schemes. The main motivation behind the online data update introduced in this chapter was to learn and adapt to the patient's dynamics over time. The outcomes for the most challenging case of Adult 7 are promising, since this online approach demonstrates improved BG management.

In order to improve the prediction performances without further increasing computational costs, in future works, it may be possible to consider the implementation of a strategy to eliminate older (i.e., with some possible forgetting factors) or irrelevant data after major changes in the patient's condition.

CHAPTER 6

Conclusions and future developments

This chapter summarises the content and contributions of this thesis, and shows some possible future research lines that could be developed based on the results presented in this thesis.

6.1 Contribution of the thesis

This thesis goal was to design a customized control algorithm for regulating the BG in T1DM adult patients with AP. Specifically, the goal was to compute the basal insulin in T1DM patients, to maintain their BG levels inside the safe range. To this aim, the MPC was implemented as a control strategy. It required a model of the system to work, but the complexity of the diabetic disease, which varies inter- and intra-patients, made it difficult to identify an accurate and general model. That is why, in this thesis, a data-driven learning method was implemented to identify the insulin-glucose behavior directly from the patients' data. Specifically, the CHoKI learning method was the selected one. The proposed customized controllers were tested on the in-silico adult T1DM patients of the UVA/Padova simulator, which is FDA-accepted for preclinical trials.

Chapter 2 detailed the MPC theory, discussing its structure and the various possible versions, as well as stability. It was also presented the data-driven MPC, both in general and in the glucose management case. Then, the CHoKI learning method was detailed, ending with the description of the CHoKI-based MPC.

Chapter 3 proposed several versions of the CHoKI-based MPC. Specifically, the first proposal involved the tightening of the constraints to address the evolution of the CHoKI prediction errors, and the introduction of slack variables to prevent feasibility issues. The hypo- and hyperglycemia were weighed differently in the cost function to highlight the danger of hypoglycemic events, since the main goal was to avoid this condition. The results were quite promising, especially when compared to those obtained through standard therapy provided by the simulator, with constant basal insulin injections. The main outcome was that all the proposed controllers reduce hypoglycemic events (the more dangerous kind), keeping patients within the euglycemic zone (i.e., 70-180 mg/dL) most of the time. Then, an additional constraint was included on the basal insulin computation, by considering the IOB, to further reduce the risk of hypoglycemia. Since it can not be measured, two possible estimation ways were proposed: one with a linear decreasing weight and one with an exponential one. The exponential weights in the IOB estimation performed better than the linear one; however, the controller might still be too conservative, allowing some hyperglycemic events. The proposed controllers have also been tested on more challenging scenarios. In particular, including circadian variability in S_I and PA sessions in the simulations on the UVA/Padova simulator The outcomes ended up being conservative, however, this conservativeness was a limitation of the CHoKI learning method.

At the end of the chapter, a hybrid model was proposed, with the aim of combining the advantages of a linear model, which is easier to implement in a control algorithm and which has physiological meaning, with a data-driven part that captures nonlinearities and enables further personalisation of the prediction model. The results showed that the data-based nonlinear corrections, made with the CHoKI learning method, added to the linear model, allow for obtaining more accurate BG predictions.

In Chapter 4, a CHoKI-based MPC with stochastic constraints tightening was proposed as a control algorithm of the AP. This was done to deal with the fact that the propagation of prediction error typical of the CHoKI approach tended to tighten constraints rapidly, resulting in very short control horizons. Specifically, this meant to implement the chance constraints, which allowed for a violation of them with a small probability. The results of the application to the UVA/Padova simulator were quite promising, and the main outcome was that it reduced the hypoglycemic

events. The comparisons with the MPC versions without stochastic constraints detailed in Chapter 3 were performed.

Chapter 5 presented the online updating of the CHoKI, which was proposed to deal with all the variations over time of the T1DM patients (such as S_I variability, circadian rhythms, physical activity sessions, patient growth, stressful situations, ...). The idea was that by applying MPC to the system, new data became available; therefore, this updated data can be used within the CHoKI to improve BG predictions, as it adapts to the changes in the patient. Thus, an updating criterion was defined, with the aim of selecting only the most useful data to add to the CHoKI dataset, to avoid an increase in the resolution times of the optimization problem. A double prediction approach was then implemented, merging the use of an offline model to guarantee safety, with the online model used to improve the prediction performance. The customized controllers were then tested on the virtual adult patients of the UVA/Padova simulator. The results were quite satisfactory, as the controllers performed well even when working with a smaller amount of data in the initial CHoKI dataset. However, updating the dataset online did not reduce the conservativeness of the CHoKI method, which still tends to have short control horizons. The most promising result concerned the BG control of the most challenging virtual patient, since the online updating approach proposed was able to learn and to adapt to the patient's dynamics over time.

6.1.1 Summary

This section provides a brief summary of the various controllers presented, using in-silico patient Adult 1 as an example to facilitate understanding. Figure 6.1 shows a comparison between the first four controllers proposed:

- The first controller, shown in blue, is the one shown in Figure 3.7, where there are slack variables on the constraints, even though the 90^{th} percentile was used to tighten them.
- Then, to try to avoid the risk of hypoglycemic events (due to their dangerousness), an additional constraint was included in the optimization problem. This takes into account the insulin that has been previously injected and that is still active (i.e., IOB), in order to limit the maximum amount of insulin to be administered. The yellow line shows the result obtained with the linear estimation

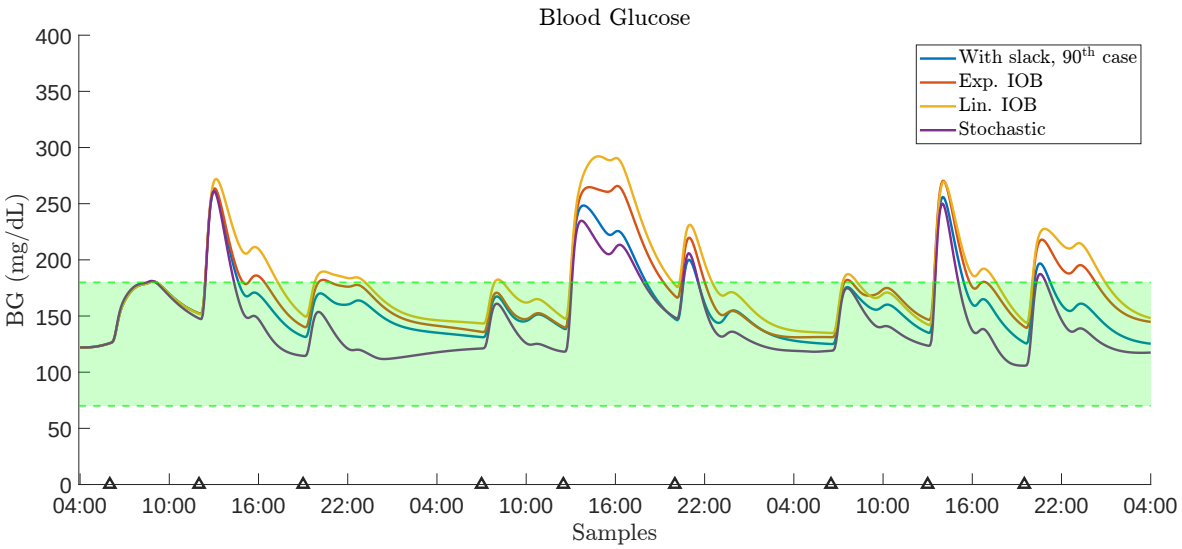


FIGURE 6.1: BG comparison. Adult 1 example. The case with slack variables on the constraints (with the 90th percentile considered) is drawn in blue. The case with linear IOB estimation is drawn in yellow, and the case with exponential IOB estimation is drawn in red. The case with probabilistic constraints is drawn in purple.

of the IOB (from the Figure 3.10), while the red line shows the result obtained with the exponential estimation of the IOB (from the Figure 3.16).

- Then, stochastic constraints were considered in order to try to extend the prediction horizon. The result is shown in purple (from the Figure 4.3).

The additional restrictions on the IOB appear to be too conservative, resulting in higher BG values, but ensuring that hypoglycemia is avoided. While the MPC with stochastic constraints allows for a slightly less conservative controller, thus achieving better BG management.

The controllers were also tested with circadian variability of the insulin sensitivity. Figure 6.2 shows the controller with the insulin constraint based on the linear IOB estimation in blue, the version with the CHoKI that updates online in red, and the one with stochastic constraints and with exponential IOB estimation in purple. This demonstrates that the CHoKI-based MPC can also manage this disturbance, and that online updating of the CHoKI dataset, which learns the patient's behavior, helps to improve BG outcomes. As in previous examples, the controller that manages BG better is the least conservative one, which is the one with stochastic constraints.

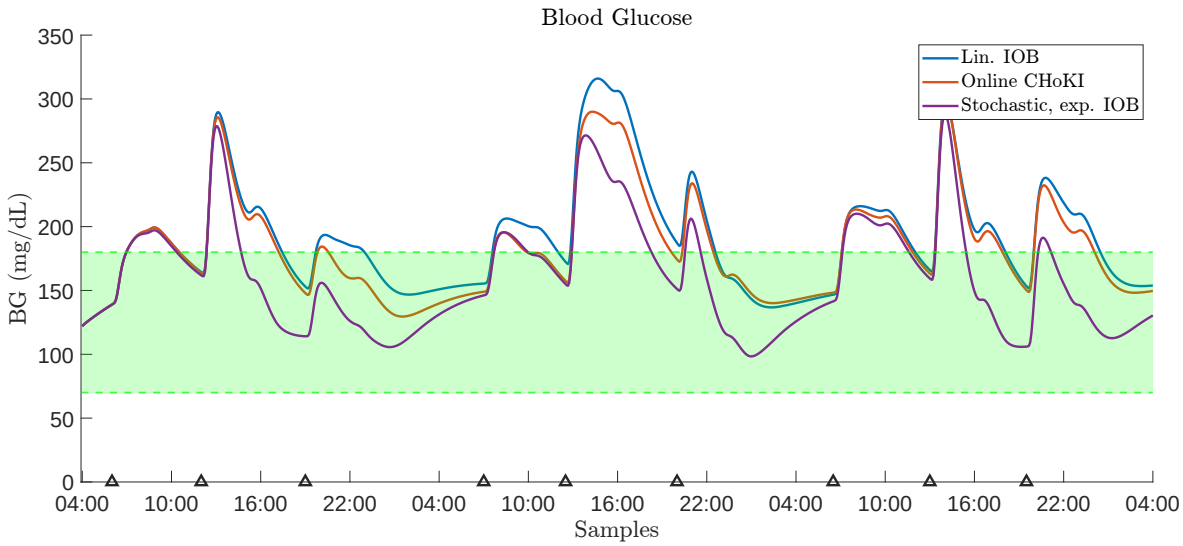


FIGURE 6.2: BG comparison. Adult 1 example. The case with linear IOB estimation is drawn in blue, the case with online CHoKI update is drawn in red, and the case with stochastic constraints and exponential IOB estimation is drawn in purple.

6.2 Future developments

In this section, some possible lines of research deriving from this thesis are presented.

- The control horizons are quite short due to the evolution of the CHoKI prediction errors. The use of chance constraints helps this, but the search for another way of dealing with the uncertainty of the model is still required.
- The CHoKI learning method and the hybrid model (i.e., linear model with CHoKI corrections) are proposed to make BG predictions. It could also be useful to compare the glucose predictions with other types of data-driven models.
- The hybrid prediction model needs to be implemented inside an MPC framework, to test the improvements on the accuracy BG predictions also in the closed-loop control.
- Further analyses on the stochastic MPC controller with CHoKI predictions are required, to study its recursive feasibility and stability.
- The online updating of the CHoKI inside the MPC is proposed in this thesis, and it could be useful to study other approaches to adapt the MPC to the patients' changing over time.

- The proposed controllers show some issues in the BG management of in-silico Adult 7, due to the greater complexity and variability of this subject. Thus, additional analyses are required.
- The proposed controllers are focused on computing the basal insulin injections, leaving the control of the boluses to the simulator (which is done by implementing the standard bolus formulation, based on the meal amount and the CR). Therefore, future developments are related to the research of more complex techniques to include the insulin boluses in the MPC control actions.
- Comparing the proposed MPCs with other types of data-driven MPCs can be very useful, to see the possible improvements that can be made.

APPENDIX A

Simulator model

In this Appendix, the equations of the glucose model in the UVA/Padova simulator [33] are defined, from Section 1.3.5.

Glucose rate of appearance:

$$\begin{cases} Q_{\text{sto}}(t) = Q_{\text{sto1}}(t) + Q_{\text{sto2}}(t) \\ Q_{\text{sto}}(t) = 0 \\ \dot{Q}_{\text{sto1}}(t) = -k_{\text{gri}} \cdot Q_{\text{sto1}}(t) + D \cdot \delta(t) \\ Q_{\text{sto1}}(t) = 0 \\ \dot{Q}_{\text{sto2}}(t) = -k_{\text{empt}}(Q_{\text{sto}}) \cdot Q_{\text{sto2}}(t) + k_{\text{gri}} \cdot Q_{\text{sto1}}(t) \\ Q_{\text{sto2}}(t) = 0 \\ \dot{Q}_{\text{gut}}(t) = -k_{\text{abs}} \cdot Q_{\text{gut}}(t) + k_{\text{empt}}(Q_{\text{sto}}) \cdot Q_{\text{sto2}}(t) \\ Q_{gut}(t) = 0 \\ Ra(t) = \frac{f \cdot k_{\text{abs}} \cdot Q_{\text{gut}}(t)}{BW} \\ Ra(0) = 0 \end{cases} \tag{A.1}$$

with

$$k_{\text{empt}}(Q_{\text{sto}}) = k_{\text{min}} + \frac{k_{\text{max}} - k_{\text{min}}}{2} \cdot \{\tanh[\alpha(Q_{\text{sto}} - b \cdot D)] - \tanh[\beta(Q_{\text{sto}} - c \cdot D)] + 2\} \tag{A.2}$$

Endogenous glucose production:

$$EGP(t) = k_{\text{p1}} - k_{\text{p2}} \cdot G_p(t) - k_{\text{p3}} \cdot X^L(t) + \xi \cdot X^H(t) \tag{A.3}$$

$$\begin{aligned} \dot{X}^L(t) &= -k_i \cdot [X^L(t) - I'(t)] \\ X^L(0) &= I_b \end{aligned} \tag{A.4}$$

$$\begin{aligned} \dot{I}'(t) &= -k_i \cdot [I'(t) - I'(t)] \\ I'(0) &= I_b \end{aligned} \tag{A.5}$$

$$\begin{aligned} \dot{X}^H(t) &= -k_H \cdot X^H(t) + k_H \cdot \max[(H(t) - H_b), 0] \\ X^H(0) &= 0 \end{aligned} \tag{A.6}$$

Where $H(t)$ is the plasma glucagon concentration, $X^H(t)$ is the delayed glucagon action on EGP, ξ represents the liver responsivity to glucagon, and $1/k_H$ stands for the delay between glucagon concentration and its action.

Glucose utilization The model of glucose kinetics is described by:

$$\begin{cases} \dot{G}_p(t) = EGP - U_{ii} - k_1 \cdot G_p(t) + k_2 \cdot G_t(t) \\ G_p(0) = G_{pb} \\ \dot{G}_t(t) = -U_{id}(t) + k_1 \cdot G_p(t) - k_2 \cdot G_t(t) \\ G_t(0) = G_{pb}\frac{k1}{k2} \end{cases} \tag{A.7}$$

where the amount of glucose in the plasma and the one in the tissues are represented as $G_p(t)$ and $G_t(t)$ respectively, and k_1 and k_2 are rate parameters.

The insulin-independent utilization U_{ii} by the brain and the erythrocytes is constant:

$$U_{ii}(t) = F_{\text{cns}}. \tag{A.8}$$

The insulin-dependent utilization is:

$$U_{id}(t) = \frac{[V_{m0} + V_{mx} \cdot X(t) \cdot (1 + r_1 \cdot risk)] \cdot G_t(t)}{K_{m0} + G_t(t)} \tag{A.9}$$

with the insulin action on glucose utilization $X(t)$, the rate parameters V_{m0}, V_{mx}, K_{m0}, p_{2U}, and with

$$\dot{X}(t) = -p_{2U} \cdot X(t) + p_{2U}[I(t) - I_b], \quad X(0) = 0. \tag{A.10}$$

Then, the *risk* is defined as:

$$risk = \begin{cases} 0 & \text{if } G \geq G_b \\ 10 \cdot [f(G)]^2 & \text{if } G_{th} \leq G < G_b \\ 10 \cdot [f(G_{th})]^2 & \text{if } G < G_{th} \end{cases} \tag{A.11}$$

where G_b is basal glucose, G_{th} is the hypoglycemic threshold (set at 60 mg/dL), $f(G) = \log\left(\frac{G}{G_b}\right)^{r_2}$, and r_1, r_2 are model parameters.

Renal excretion:

$$E(t) = \begin{cases} k_{\text{e1}} \cdot [G_p(t) - k_{\text{e2}}] & \text{if } G_p(t) > k_{\text{e2}} \\ 0 & \text{if } G_p(t) \leq k_{\text{e2}} \end{cases} \tag{A.12}$$

Subcutaneous insulin kinetics:

$$R_{ai}(t) = k_{\text{a1}} \cdot I_{\text{sc1}}(t) + k_{\text{a2}} \cdot I_{\text{sc2}}(t) \tag{A.13}$$

with

$$\begin{cases} \dot{I}_{\text{sc1}}(t) = -(k_d + k_{\text{a1}}) \cdot I_{\text{sc1}}(t) + IIR(t) \\ I_{\text{sc1}}(0) = I_{\text{sc1ss}} \\ \dot{I}_{\text{sc2}}(t) = k_d \cdot I_{\text{sc1}}(t) - k_{\text{a2}} \cdot I_{\text{sc2}}(t) \\ I_{\text{sc2}}(0) = I_{\text{sc2ss}} \end{cases} \tag{A.14}$$

Subcutaneous glucose kinetics:

$$\dot{Q}_s(t) = -\frac{1}{T_s} \cdot G_s(t) + \frac{1}{T_s} \cdot G(t); \quad G_s(0) = G_b \tag{A.15}$$

Glucagon kinetics and secretion: Glucagon kinetics are described with a 1-compartmental linear model:

$$\dot{H}(t) = -n \cdot H(t) + SR_H(t) + Ra_H(t), \quad H(0) = H_b \tag{A.16}$$

with plasma hormone concentration $H(t)$, glucagon secretion $SR_H(t)$ (with its basal value SR_H^b), and the clearance rate n.

Glucagon secretion is described as a sum of two components:

$$SR_H(t) = SR_H^s(t) + SR_H^d(t) \tag{A.17}$$

with

$$\dot{SR}^s_H(t) = \begin{cases} -\rho \cdot \left[SR^s_H(t) - \max\left(\sigma_2 \cdot [Gth - G(t)] + SR^b_H, 0\right)\right] & \text{if } G(t) \geq G_b \\ -\rho \cdot \left[SR^s_H(t) - \max\left(\frac{\sigma \cdot [Gth - G(t)]}{I(t)+1} + SR^b_H, 0\right)\right] & \text{if } G(t) < G_b \end{cases} \tag{A.18}$$

with plasma glucose $G(t)$ and its basal value G_b, plasma insulin concentration $I(t)$ and its basal value I_b, α and α_2 represent the alpha-cell responsivity to glucose level, $1/\rho$ is for the delay between static glucagon secretion and plasma glucose (considering that the static secretion is stimulated when $G < G_b$ and inhibited when $G \geq G_b$).

$$SR^d_H(t) = \delta \cdot \max\left(-\frac{dG(t)}{dt}, 0\right) \tag{A.19}$$

with glucose rate of change $\frac{dG(t)}{dt}$, and δ that represents the alpha-cell responsivity to glucose rate of change. Model parameters σ and δ also reflect that the glucagon response decreases with the age of diabetes.

Subcutaneous glucagon kinetics:

$$\begin{cases} \dot{H}_{\text{sc1}}(t) = -(k_{\text{h1}} + k_{\text{h2}}) \cdot H_{\text{sc1}}(t) \\ H_{sc1}(0) = H_{sc1b} \\ \dot{H}_{\text{sc2}}(t) = k_{\text{h1}} \cdot H_{\text{sc1}}(t) - k_{\text{h3}} \cdot H_{\text{sc2}}(t) \\ H_{\text{sc2}}(0) = H_{\text{sc2b}} \end{cases} \tag{A.20}$$

$$Ra_H(t) = k_{\text{h3}} \cdot H_{\text{sc2}}(t) \tag{A.21}$$

Where k_{hi} are rate parameters describing subcutaneous glucagon kinetics, H_{inf} is the glucagon infusion rate, and H_{sc1} and H_{sc2} are glucagon concentrations in the subcutaneous space.

APPENDIX B

Definitions

Definition B.1: $\mathcal{K}$ function

A function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a $\mathcal{K}$-function if

- *it is continuous*
- *is it strictly increasing, i.e., if $a > b$, then $\alpha(a) > \alpha(b)$*
- *$\alpha(0) = 0$*

Definition B.2: $\mathcal{K}_\infty$ function

A function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a $\mathcal{K}_\infty$-function if it is a $\mathcal{K}$-function and $\alpha(a) \to \infty$ when $a \to \infty$ (unbounded)

Definition B.3: $\mathcal{KL}$ function

A function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{I}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a $\mathcal{KL}$-function if

- *the function $\beta(a, k)$ is a $\mathcal{K}$ in a for every fixed $k \geq 0$*
- *the function $\beta(a, k)$ is nonincreasing in k for every fixed $a \geq 0$, in such a way that $\beta(a, k) \to 0$ for $k \to \infty$*

Definition B.4: $\mathcal{PD}$ function

A function $V : \mathbb{R}^{nx} \to \mathbb{R}_{\geq 0}$ is locally positive definite ($\mathcal{PD}$) if it is continuous, $V(0) = 0$ and $V(x) > 0$ for every $x \neq 0$ in a neighborhood or the origin.

Property B.1: Main properties of $\mathcal{K}$, $\mathcal{K}_\infty$, and $\mathcal{KL}$ functions

Consider two $\mathcal{K}$ functions $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ defined in $[0, a)$, two $\mathcal{K}_\infty$ functions $\alpha_3(\cdot)$ and $\alpha_4(\cdot)$, and a $\mathcal{KL}$ function $\beta(\cdot, \cdot)$. Therefore:

- *the function $\alpha_1^{-1}(\cdot)$ is a $\mathcal{K}$ function defined in $[0, \alpha_1(a))$;*
- *the function $\alpha_3^{-1}(\cdot)$ is a $\mathcal{K}_\infty$ function;*
- *the composition of $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$, $\alpha_1 \circ \alpha_2$, is a $\mathcal{K}$ function;*
- *the composition of $\alpha_3(\cdot)$ and $\alpha_4(\cdot)$, $\alpha_3 \circ \alpha_4$, is a $\mathcal{K}_\infty$ function;*
- *the composition of a $\mathcal{K}$ function and a $\mathcal{KL}$ function, $\alpha_1 \circ \beta$, is a $\mathcal{KL}$ function;*
- *the function $\alpha_5(s) := \max\{\alpha_1(s), \alpha_2(s)\}$ is a $\mathcal{K}$ function;*
- *the function $\alpha_6(s) := \max\{\alpha_3(s), \alpha_4(s)\}$ is a $\mathcal{K}_\infty$ function;*
- *the function $\alpha_7(s) := \alpha_1(s) + \alpha_2(s)$ is a $\mathcal{K}$ function;*
- *the function $\alpha_8(s) := \alpha_3(s) + \alpha_4(s)$ is a $\mathcal{K}_\infty$ function;*
- *given a $\mathcal{K}_\infty$ function, $\alpha_3(\cdot)$, there exists a $\mathcal{K}_\infty$ function $\alpha_9(\cdot)$ such that:*
  - *$\alpha_9(s) \leq \alpha_3(s)$ for all $s > 0$;*
  - *the function $\alpha_{10}(s) := s - \alpha_9(s)$ is a $\mathcal{K}$ function, that is $\alpha_9(s) < s$ for all $s > 0$;*
- *if $\alpha_3(s) \leq \alpha_4(s)$ for all $s \geq 0$, then $\alpha_3^{-1}(s) \geq \alpha_4^{-1}(s)$;*
- *given $a, b \in \mathbb{R}_{\geq 0}$, $\alpha_1(a + b) \leq \alpha_1(2a) + \alpha_1(2b)$;*
- *given a $\mathcal{K}_\infty$ function, $\alpha_3(\cdot)$, there exist two $\mathcal{K}_\infty$ functions $\alpha_{11}(\cdot), \alpha_{12}(\cdot)$ such that $\alpha_3(rs) \leq \alpha_{11}(r)\alpha_{12}(s)$, for all $r, s \in \mathbb{R}_{\geq 0}$.*

Definition B.5: Normed vector space

A normed vector space is a couple $\mathcal{X}, \|\cdot\|$, where $\mathcal{X}$ is a vector space and $\|\cdot\| : \mathcal{X} \to \mathbb{R}$ is a real function, called norm, such that:

1. *$\|x\| \geq 0$, for all $x \in \mathcal{X}$; $\|x\| = 0$ iff $x = 0$*
2. *$\|\alpha x\| = |\alpha|\|x\|$, for all $x \in \mathcal{X}$, and all α*
3. *$\|x + y\| \leq \|x\| + \|y\|$, for all $x, y \in \mathcal{X}$ (triangle inequality)*

Definition B.6: p-norm

A general p-norm in $\mathbb{R}$ is defined as:

$$\|x\|_p = \sqrt[p]{|x_1|^p + |x_2|^p + ... + |x_n|^p} \tag{B.1}$$

Definition B.7: Convergence of a sequence

Let $\{x_n\}_1^\infty$ be a sequence of elements belonging to a normed vector space $(\mathcal{X}, \|\cdot\|)$. We say that such a sequence converges to the element $x_0 \in \mathcal{X}$ if $\|x_n - x_0\| \to 0$ for $n \to \infty$. That is, $\{x_n\}_1^\infty$ converges to x_0 if, for all $\epsilon > 0$, there exists and integer $N(\epsilon)$ such that

$$\|x_n - x_o\| < \epsilon, \quad \forall n \geq N(\epsilon) \tag{B.2}$$

Definition B.8: Continuity

Let $(\mathcal{X}, \|\cdot\|_\mathcal{X})$ and $(\mathcal{Y}, \|\cdot\|_\mathcal{Y})$ be two normed vector spaces. Let $f : \mathcal{X} \to \mathcal{Y}$ be a mapping from $\mathcal{X}$ to $\mathcal{Y}$. We state that f is continuous in $x_0 \in \mathcal{X}$ if for all $\epsilon > 0$ there exists a $\delta(\epsilon, x_0) > 0$ such that
if $\|x_0 - x\|_\mathcal{X} < \delta(\epsilon, x_0)$, then $\|f(x_0) - f(x)\|_\mathcal{Y} < \epsilon$.
f is continuous if it is continuous in all $x \in \mathcal{X}$
f is uniformly continuous if it is continuous and for all $\epsilon > 0$ there exists a $\delta(\epsilon) > 0$ such that
if $\|x_0 - x\|_\mathcal{X} < \delta(\epsilon)$, then $\|f(x_0) - f(x)\|_\mathcal{Y} < \epsilon$.

Definition B.9: Positively invariant set

A set $\Omega \subseteq \mathbb{R}^n$ is a positively invariant set for the autonomous system $x(k+1) = f(x(k))$ if

$$\forall x : x \in \Omega \implies f(x) \in \Omega.$$

Definition B.10: Robust positively invariant set (RPI)

A set $\Omega \subseteq \mathbb{R}^n$ is a robust positively invariant set for the system $x(k+1) = f(x(k), w(k))$, where $w \in \mathcal{W}$, if

$$\forall x : x \in \Omega \implies f(x, w) \in \Omega, \forall w \in \mathcal{W}.$$

Definition B.11: Stability

An autonomous system $x(k+1) = f(x(k))$ is stable at the origin if, for all $\delta > 0$, a constant $\epsilon = \epsilon(\delta)$ exists such that

$$\forall x(0) : \|x(0)\| \leq \epsilon \implies \|x(k)\| \leq \delta \; \forall k \geq 0.$$

Definition B.12: Attractivity

The system $x(k+1) = f(x(k))$ is attractive on a set $\mathcal{X}$ if $\lim_{k\to\infty} x(k) = 0$ for all $x(0) \in \mathcal{X}$.

Definition B.13: Asymptotic stability

An autonomous system $x(k+1) = f(x(k))$ is asymptotically stable at the origin if it is stable and there exists a constant $\epsilon > 0$ such that

$$\forall x(0) : \|x(0)\| \leq \epsilon \implies x(k) \to 0 \text{ for } k \to \infty.$$

Equivalently, the system $x(k+1) = f(x(k))$ is asymptotically stable at the origin if a $\mathcal{KL}$ function $\beta(\cdot,\cdot)$ exists such that

$$\|x(k)\| \leq \beta(\|x(0)\|, k)$$

for all $k \geq 0$ and all $x(0)$ that verify $\|x(0)\| \leq \epsilon$.
Alternatively, the $x(k+1) = f(x(k))$ is asymptotically stable at the origin if it is stable and attractive.

Definition B.14: Exponential stability

An autonomous system $x(k+1) = f(x(k))$ is exponentially stable at the origin if it is asymptotically stable and there exist constants $\epsilon > 0$, $a > 0$, and $\lambda \in [0,1)$ that verify

$$\|x(k)\| \leq a \cdot \|x(0)\| \cdot \lambda^k$$

for all $k \geq 0$ and all $x(0)$ such that $\|x(0)\| \leq \epsilon$.

Definition B.15: Lyapunov function

A function $V : \mathbb{R}^{nx} \to \mathbb{R}_{\geq 0}$ is said to be Lyapunov function for the system $x^+ = f(x)$ and set Ω if there exist $\mathcal{K}_\infty$-functions α_1 and α_2, and a $\mathcal{PD}$ function α_3, such that for ant $x \in \mathbb{R}^{nx}$,

$$V(x) \geq \alpha_1(|x|_\Omega), \tag{B.3a}$$
$$V(x) \leq \alpha_2(|x|_\Omega), \tag{B.3b}$$
$$V(f(x)) - V(x) \leq -\alpha_3(|x|_\Omega), \tag{B.3c}$$

Theorem B.1

Given an autonomous system $x(k+1) = f(x(k))$ with an equilibrium at the origin, if there exists an associated Lyapunov function, then the origin is a stable equilibrium of the system. If these conditions hold in $\mathbb{R}^n$, then the origin is globally stable.

Theorem B.2

Given an autonomous system $x(k+1) = f(x(k))$ with an equilibrium at the origin, if there exists an associated Lyapunov function such that $V(f(x)) - V(x) \leq -\alpha_3(\|x\|)$, where $\alpha_3(\cdot)$ is a $\mathcal{PD}$ function, then the origin is an asymptotically stable equilibrium of the system. If these conditions hold in $\mathbb{R}^n$, then the origin is globally asymptotically stable.

Theorem B.3

Given an autonomous system $x(k+1) = f(x(k))$ with an equilibrium at the origin, if there exists an associated Lyapunov function $V(x)$ and constants $a > 0$, $b > 0$, $c > 0$ and $\sigma \geq 1$ such that

$$a\|x\|^\sigma \leq V(x) \leq b\|x\|^\sigma$$

and

$$V(f(x)) - V(x) \leq -c\|x\|^\sigma,$$

then the origin is an exponentially stable equilibrium of the system. If these conditions hold in $\mathbb{R}^n$, then the origin is globally exponentially stable.

Definition B.16: Control Lyapunov function

A function $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ associated to a system $x(k+1) = f(x(k), u(k))$ is a control Lyapunov function in $\mathcal{X}$ if there exist functions $\alpha_1(\cdot), \alpha_2(\cdot), \alpha_3(\cdot) \in \mathcal{K}_\infty$ and a control law $\kappa(x)$ defined and feasible in $\mathcal{X}$ such that the following conditions hold for any $x \in \mathcal{X}$:

$$\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|)$$
$$V(f(x, \kappa(x))) - V(x) \leq -\alpha_3(\|x\|).$$

Definition B.17: Input-to-state stability

A system $x(k+1) = f(x(k), w(k))$ is Input-to-State Stability (ISS) with respect to w if there exist a $\mathcal{KL}$ function $\beta(\cdot,\cdot)$ and a $\mathcal{K}$ function $\alpha(\cdot)$ such that

$$\|x(k)\| \leq \beta(\|x(0)\|, k) + \sup_{j\in[0,k]} \alpha(\|w(j)\|).$$

Definition B.18: Input-to-state practical stability

A system $x(k+1) = f(x(k), w(k))$ is Input-to-State practically Stable (ISpS) with respect to w if there exist a $\mathcal{KL}$ function $\beta(\cdot,\cdot)$, a $\mathcal{K}$ function $\alpha(\cdot)$ and a constant $c > 0$ such that

$$\|x(k)\| \leq \beta(\|x(0)\|, k) + \sup_{j\in[0,k]} \alpha(\|w(j)\|) + c.$$

Definition B.19: Ground truth function

Given two Hilbert spaces, denoted $\mathcal{W} \subseteq \mathbb{R}^{n_w}$, and $\mathcal{Y} \subseteq \mathbb{R}^{n_y}$, and referred to as input and output space, respectively, the ground truth function is defined as the mapping $f : \mathcal{W} \to \mathcal{Y}$ such that $y = f(w)$.

Definition B.20: Prediction error

Given the ground truth function f and a predictor $\hat{f}$ for this function, the prediction error $d \in \mathbb{R}^{n_y}$ for a given query w is computed as $d(w) = |f(w) - \hat{f}(w; \theta, \mathcal{D})|$

Definition B.21: Worst-case prediction error

The maximum prediction error using the predictor $\hat{f}$, given the parameters θ and data set $\mathcal{D}$ is denoted μ, such that

$$|\hat{f}(q; \theta, \mathcal{D}) - f(q)| = d(q) \leq \mu \in \mathbb{R}^{n_y},\ \forall q \in \mathcal{W}.$$

Bibliography

[1] P. Abuin, A. Ferramosca, C. Toffanin, L. Magni, and A. H. González. Pulsatile Zone MPC with asymmetric stationary cost for artificial pancreas based on a non-standard IOB constraint. *Journal of Process Control*, 136:103191, 2024.

[2] P. Abuin, P. S. Rivadeneira, A. Ferramosca, and A. H. González. Artificial pancreas under stable pulsatile MPC: Improving the closed-loop performance. *Journal of Process Control*, 92:246–260, 2020.

[3] V. Adetola and M. Guay. Robust adaptive MPC for constrained uncertain nonlinear systems. *International Journal of Adaptive Control and Signal Processing*, 25(2):155–167, 2011.

[4] E. M. Aiello, M. Jaloli, and M. Cescon. Model Predictive Control (MPC) of an artificial pancreas with data-driven learning of multi-step-ahead blood glucose predictors. *Control Engineering Practice*, 144:105810, 2024.

[5] A. D. Association. Physical activity/exercise and diabetes. *Diabetes care*, 27(suppl_1):s58–s62, 2004.

[6] A. Aswani, P. Bouffard, and C. Tomlin. Extensions of learning-based model predictive control for real-time application to a quadrotor helicopter. In *2012 American Control Conference (ACC)*, pages 4661–4666. IEEE, 2012.

[7] A. Aswani, H. Gonzalez, S. S. Sastry, and C. Tomlin. Provably safe and robust learning-based model predictive control. *Automatica*, 49(5):1216–1226, 2013.

[8] M. A. Atkinson, G. S. Eisenbarth, and A. W. Michels. Type 1 diabetes. *The lancet*, 383(9911):69–82, 2014.

[9] I. D. Atlas. International diabetes federation. 11st, 2025.

[10] S. Bahremand, H. S. Ko, R. Balouchzadeh, H. Felix Lee, S. Park, and G. Kwon. Neural network-based model predictive control for type 1 diabetic rats on artificial pancreas system. *Medical & biological engineering & computing*, 57(1):177–191, 2019.

[11] V. Bellido, E. Aguilera, R. Cardona-Hernandez, G. Diaz-Soto, N. González Pérez de Villar, M. J. Picón-César, and F. J. Ampudia-Blasco. Expert recommendations for using time-in-range and other continuous glucose monitoring metrics to achieve patient-centered glycemic control in people with diabetes. *Journal of Diabetes Science and Technology*, 17(5):1326–1336, 2023.

[12] A. Bemporad. Control of constrained nonlinear systems via reference management. In *Proceedings of the 1997 American Control Conference (Cat. No. 97CH36041)*, volume 5, pages 3343–3347. IEEE, 1997.

[13] A. Bemporad, F. Borrelli, and M. Morari. Min-max control of constrained uncertain discrete-time linear systems. *IEEE Transactions on automatic control*, 48(9):1600–1606, 2003.

[14] A. Bemporad and M. Morari. Robust model predictive control: A survey. In *Robustness in identification and control*, pages 207–226. Springer, 2007.

[15] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

[16] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer. Data-driven tracking MPC for changing setpoints. *IFAC-PapersOnLine*, 53(2):6923–6930, 2020.

[17] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer. Linear tracking MPC for nonlinear systems—Part II: The data-driven case. *IEEE Transactions on Automatic Control*, 67(9):4406–4421, 2022.

[18] D. Boiroux, V. Bátora, M. Hagdrup, S. L. Wendt, N. K. Poulsen, H. Madsen, and J. B. Jørgensen. Adaptive model predictive control for a dual-hormone artificial pancreas. *Journal of Process Control*, 68:105–117, 2018.

[19] G. B. Bolli, A. Y. Cheng, and D. R. Owens. Insulin: evolution of insulin formulations and their application in clinical practice over 100 years. *Acta Diabetologica*, 59(9):1129–1144, 2022.

[20] N. D. Borella, A. Ferramosca, G. Castagna, S. Ippolito, S. Ceresoli, A. Taverna, B. Sonzogni, R. Trevisan, and G. Lepore. Comparison of the night-time effectiveness in achieving glycemic targets in adults with type 1 diabetes of three advanced hybryd closed-loop systems. *Acta Diabetologica*, pages 1–9, 2024.

[21] P. Bouffard, A. Aswani, and C. Tomlin. Learning-based model predictive control on a quadrotor: Onboard implementation and experimental results. In *2012 IEEE International Conference on Robotics and Automation*, pages 279–284. IEEE, 2012.

[22] M. Breton, A. Farret, D. Bruttomesso, S. Anderson, L. Magni, S. Patek, C. Dalla Man, J. Place, S. Demartini, S. Del Favero, et al. Fully integrated artificial pancreas in type 1 diabetes: modular closed-loop glucose control maintains near normoglycemia. *Diabetes*, 61(9):2230–2237, 2012.

[23] J. Buchanan, J. A. Zabinsky, C. Ferrara-Cook, S. Adi, and J. C. Wong. Comparison of insulin pump bolus calculators reveals wide variation in dose recommendations. *Journal of diabetes science and technology*, 15(6):1290–1296, 2021.

[24] J.-P. Calliess. *Conservative decision-making and inference in uncertain dynamical systems*. PhD thesis, University of Oxford, 2014.

[25] J.-P. Calliess, S. J. Roberts, C. E. Rasmussen, and J. Maciejowski. Lazily adapted constant kinky inference for nonparametric regression and model-reference adaptive control. *Automatica*, 122:109216, 2020.

[26] M. Canale, L. Fagiano, and M. C. Signorile. Nonlinear model predictive control from data: a set membership approach. *International Journal of Robust and Nonlinear Control*, 24(1):123–139, 2014.

[27] L. Chisci and G. Zappa. Dual mode predictive tracking of piecewise constant references for constrained linear systems. *International Journal of Control*, 76(1):61–72, 2003.

[28] C. Cobelli, E. Renard, and B. Kovatchev. Artificial pancreas: past, present, future. *Diabetes*, 60(11):2672–2682, 2011.

[29] S. R. Colberg, R. J. Sigal, J. E. Yardley, M. C. Riddell, D. W. Dunstan, P. C. Dempsey, E. S. Horton, K. Castorino, and D. F. Tate. Physical activity/exercise and diabetes: a position statement of the American Diabetes Association. *Diabetes care*, 39(11):2065, 2016.

[30] A. D. A. P. P. Committee. Standards of medical care in diabetes. *Diabetes Care*, 45(Supplement_1), 2022.

[31] V. M. Cunha and T. L. Santos. Robust nonlinear model predictive control based on nominal predictions with piecewise constant references and bounded disturbances. *International Journal of Robust and Nonlinear Control*, 32(6):3944–3968, 2022.

[32] C. Dalla Man, M. D. Breton, and C. Cobelli. Physical activity into the meal glucose—insulin model of type 1 diabetes: In silico studies. *J Diabetes Sci Technol.*, 3(1):56–67, 2009.

[33] C. Dalla 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*, 8(1):26–34, 2014.

[34] C. Dalla Man, R. A. Rizza, and C. Cobelli. Meal simulation model of the glucose-insulin system. *IEEE Transactions on biomedical engineering*, 54(10):1740–1749, 2007.

[35] T. Danne, M. Phillip, B. A. Buckingham, P. Jarosz-Chobot, B. Saboo, T. Urakami, T. Battelino, R. Hanas, and E. Codner. Ispad clinical practice consensus guidelines 2018: Insulin treatment in children and adolescents with diabetes. *Pediatric diabetes*, 19, 2018.

[36] E. Dassau, S. A. Brown, A. Basu, J. E. Pinsker, Y. C. Kudva, R. Gondhalekar, S. Patek, D. Lv, M. Schiavon, J. B. Lee, et al. Adjustment of open-loop settings to improve closed-loop results in type 1 diabetes: a multicenter randomized trial. *The Journal of Clinical Endocrinology & Metabolism*, 100(10):3878–3886, 2015.

[37] S. Del Favero, D. Bruttomesso, F. Di Palma, G. Lanzola, R. Visentin, A. Filippi, R. Scotton, C. Toffanin, M. Messori, S. Scarpellini, et al. First use of model predictive control in outpatient wearable artificial pancreas. *Diabetes care*, 37(5):1212–1215, 2014.

[38] A. D'Jorge, B. F. Santoro, A. Anderson, A. H. González, and A. Ferramosca. Stochastic model predictive control for tracking linear systems. *Optimal Control Applications and Methods*, 41(1):65–83, 2020.

[39] F. J. Doyle III and B. P. Kovatchev. Artificial pancreas: the first 20 years. *Journal of Diabetes Science and Technology*, 19(4):875–882, 2025.

[40] S. Dutta, T. Kushner, and S. Sankaranarayanan. Robust data-driven control of artificial pancreas systems using neural networks. In *International Conference on Computational Methods in Systems Biology*, pages 183–202. Springer, 2018.

[41] C. Ellingsen, E. Dassau, H. Zisser, B. Grosman, M. W. Percival, L. Jovanovič, and F. J. Doyle III. Safety constraints in an artificial pancreatic β cell: an implementation of model predictive control with insulin on board. *Journal of diabetes science and technology*, 3(3):536–544, 2009.

[42] A. Ferramosca, D. Limon, I. Alvarado, T. Alamo, and E. Camacho. MPC for tracking with optimal closed-loop performance. *Automatica*, 45(8):1975–1978, 2009.

[43] A. Ferramosca, D. Limon, A. H. González, I. Alvarado, and E. F. Camacho. Robust MPC for tracking zone regions based on nominal predictions. *Journal of Process Control*, 22(10):1966–1974, 2012.

[44] E. Garone, S. Di Cairano, and I. Kolmanovsky. Reference and command governors for systems with constraints: A survey on theory and applications. *Automatica*, 75:306–328, 2017.

[45] R. Gondhalekar, E. Dassau, and F. J. Doyle III. Periodic zone-MPC with asymmetric costs for outpatient-ready safety of an artificial pancreas to treat type 1 diabetes. *Automatica*, 71:237–246, 2016.

[46] R. Gondhalekar, E. Dassau, and F. J. Doyle III. Velocity-weighting & velocity-penalty MPC of an artificial pancreas: Improved safety & performance. *Automatica*, 91:105–117, 2018.

[47] R. Gondhalekar, E. Dassau, H. C. Zisser, and F. J. Doyle III. Periodic-zone model predictive control for diurnal closed-loop operation of an artificial pancreas, 2013.

[48] A. H. González and D. Odloak. A stable MPC with zone control. *Journal of Process Control*, 19(1):110–122, 2009.

[49] A. H. González, P. S. Rivadeneira, A. Ferramosca, N. Magdelaine, and C. H. Moog. Impulsive zone MPC for type I diabetic patients based on a long-term model. *IFAC-PapersOnLine*, 50(1):14729–14734, 2017.

[50] A. H. González, P. S. Rivadeneira, A. Ferramosca, N. Magdelaine, and C. H. Moog. Stable impulsive zone model predictive control for type 1 diabetic patients based on a long-term model. *Optimal Control Applications and Methods*, 41(6):2115–2136, 2020.

[51] A. K. J. Gradel, T. Porsgaard, J. Lykkesfeldt, T. Seested, S. Gram-Nielsen, N. R. Kristensen, and H. H. F. Refsgaard. Factors affecting the absorption of subcutaneously administered insulin: effect on variability. *Journal of diabetes research*, 2018(1):1205121, 2018.

[52] L. Grüne and J. Pannek. Nonlinear model predictive control. In *Nonlinear model predictive control: Theory and algorithms*, pages 45–69. Springer, 2016.

[53] I. Hajizadeh, M. Rashid, and A. Cinar. Plasma-insulin-cognizant adaptive model predictive control for artificial pancreas systems. *Journal of Process Control*, 77:97–113, 2019.

[54] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger. Learning-based model predictive control: Toward safe learning in control. *Annual Review of Control, Robotics, and Autonomous Systems*, 3:269–296, 2020.

[55] I. B. Hirsch. Introduction: history of glucose monitoring. 2018.

[56] R. Hovorka, V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. R. Pieber, H. C. Schaller, L. Schaupp, T. Vering, et al. Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes. *Physiological measurement*, 25(4):905, 2004.

[57] ISTAT. Number of individuals suffering from diabetes in Italy from 2010 to 2023 (in 1,000 people), 2024.

[58] P. G. Jacobs, P. Herrero, A. Facchinetti, J. Vehi, B. Kovatchev, M. D. Breton, A. Cinar, K. S. Nikita, F. J. Doyle, J. Bondia, et al. Artificial intelligence and machine learning for improving glycemic control in diabetes: best practices, pitfalls, and opportunities. *IEEE reviews in biomedical engineering*, 17:19–41, 2023.

[59] M. A. Johnson and M. H. Moradi. *PID control*. Springer, 2005.

[60] S. L. Kang, Y. N. Hwang, J. Y. Kwon, and S. M. Kim. Effectiveness and safety of a model predictive control (MPC) algorithm for an artificial pancreas system in outpatients with type 1 diabetes (T1D): systematic review and meta-analysis. *Diabetology & metabolic syndrome*, 14(1):187, 2022.

[61] E. C. Kerrigan and J. M. Maciejowski. Feedback min-max model predictive control using a single linear program: robust stability and the explicit solution. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 14(4):395–413, 2004.

[62] J. Kesavadev, B. Saboo, M. B. Krishna, and G. Krishnan. Evolution of insulin delivery devices: from syringes, pens, and pumps to DIY artificial pancreas. *Diabetes Therapy*, 11(6):1251–1269, 2020.

[63] D. C. Klonoff. Afrezza inhaled insulin: the fastest-acting FDA-approved insulin on the market has favorable properties, 2014.

[64] D. C. Klonoff, D. Ahn, and A. Drincic. Continuous glucose monitoring: a review of the technology and clinical use. *Diabetes Research and Clinical Practice*, 133:178–192, 2017.

[65] D. C. Klonoff, J. Wang, D. Rodbard, M. A. Kohn, C. Li, D. Liepmann, D. Kerr, D. Ahn, A. L. Peters, G. E. Umpierrez, et al. A glycemia risk index (GRI) of hypoglycemia and hyperglycemia for continuous glucose monitoring validated by clinician ratings. *Journal of diabetes science and technology*, page 19322968221085273, 2022.

[66] B. Kovatchev. Automated closed-loop control of diabetes: the artificial pancreas. *Bioelectronic Medicine*, 4(1):1–12, 2018.

[67] B. P. Kovatchev, M. Breton, C. Dalla Man, and C. Cobelli. In silico preclinical trials: a proof of concept in closed-loop control of type 1 diabetes, 2009.

[68] P. Krupa, J. Köhler, A. Ferramosca, I. Alvarado, M. N. Zeilinger, T. Alamo, and D. Limon. Model predictive control for tracking using artificial references: Fundamentals, recent results and practical implementation. In *2024 IEEE 63rd Conference on Decision and Control (CDC)*, pages 2977–2991. IEEE, 2024.

[69] M. Lazar, D. M. De La Peña, W. Heemels, and T. Alamo. Min-max nonlinear model predictive control with guaranteed input-tostate stability. In *17th Symposium on Mathematical Theory for Networks and Systems. Kyoto, Japan*, 2006.

[70] F. León-Vargas, F. Garelli, H. De Battista, and J. Vehí. Postprandial blood glucose control using a hybrid adaptive PD controller with insulin-on-board limitation. *Biomedical Signal Processing and Control*, 8(6):724–732, 2013.

[71] N. Licini, B. Sonzogni, P. Abuin, F. Previdi, A. H. Gonzalez, and A. Ferramosca. Artificial Pancreas under stable pulsatile Model Predictive Control: including the Physical Activity effect. In *63rd IEEE conference on decision and control*, 2024.

[72] D. Limon, T. Alamo, F. Salas, and E. F. Camacho. On the stability of constrained MPC without terminal constraint. *IEEE Trans. on Automatic Control*, 51(5):832–836, 2006.

[73] D. Limón, I. Alvarado, T. Alamo, and E. F. Camacho. MPC for tracking piecewise constant references for constrained linear systems. *Automatica*, 44(9):2382–2387, 2008.

[74] D. Limon, I. Alvarado, A. Ferramosca, T. Alamo, and E. F. Camacho. Enhanced robust NMPC based on nominal predictions. *IFAC Proceedings Volumes*, 43(14):220–225, 2010.

[75] D. Limon, A. Ferramosca, I. Alvarado, and T. Alamo. Nonlinear MPC for tracking piece-wise constant reference signals. *IEEE Transactions on Automatic Control*, 63(11):3735–3750, 2018.

[76] C. Liu, H. Li, J. Gao, and D. Xu. Robust self-triggered min–max model predictive control for discrete-time nonlinear systems. *Automatica*, 89:333–339, 2018.

[77] B. T. Lopez, J.-J. E. Slotine, and J. P. How. Dynamic tube MPC for nonlinear systems. In *2019 American Control Conference (ACC)*, pages 1655–1662. IEEE, 2019.

[78] M. Lorenzen, F. Dabbene, R. Tempo, and F. Allgöwer. Constraint-tightening and stability in stochastic model predictive control. *IEEE Transactions on Automatic Control*, 62(7):3165–3177, 2016.

[79] N. Magdelaine, L. Chaillous, I. Guilhem, J.-Y. Poirier, M. Krempf, C. H. Moog, and E. Le Carpentier. A long-term model of the glucose–insulin dynamics of type 1 diabetes. *IEEE Transactions on Biomedical Engineering*, 62(6):1546–1552, 2015.

[80] L. Magni, D. M. Raimondo, C. Dalla Man, G. De Nicolao, B. Kovatchev, and C. Cobelli. Model predictive control of glucose concentration in type I diabetic patients: An in silico trial. *Biomedical Signal Processing and Control*, 4(4):338–346, 2009.

[81] L. Magni, D. M. Raimondo, C. Della Man, M. Breton, S. Patek, G. D. Nicolao, C. Cobelli, and B. P. Kovatchev. Evaluating the efficacy of closed-loop glucose regulation via Control-Variability Grid Analysis. *Journal of Diabetes Science and Technology*, 2(4):630–635, 2008.

[82] L. Magni and R. Scattolini. On the solution of the tracking problem for non-linear systems with MPC. *International journal of systems science*, 36(8):477–484, 2005.

[83] J. M. Manzano. *Learning-based model predictive control for constrained nonlinear systems*. PhD thesis, Universidad de Sevilla, 2020.

[84] J. M. Manzano, J.-P. Calliess, D. Muñoz de la Peña, and D. Limon. Online learning robust MPC: an exploration-exploitation approach. *IFAC-PapersOnLine*, 53(2):5292–5297, 2020.

[85] J. M. Manzano, D. Limon, D. Muñoz de la Peña, and J.-P. Calliess. Robust learning-based MPC for nonlinear constrained systems. *Automatica*, 117, 2020.

[86] J. M. Manzano, D. Muñoz de la Peña, J.-P. Calliess, and D. Limon. Componentwise Hölder Inference for Robust Learning-Based MPC. *IEEE Trans. on Automatic Control*, 66(11):5577–5583, 2021.

[87] J. M. Manzano, D. Muñoz de la Peña, J.-P. Calliess, and D. Limon. Online learning constrained model predictive control based on double prediction. *International Journal of Robust and Nonlinear Control*, 31(18):8813–8829, 2021.

[88] A. Marcellusi, R. Viti, A. Mecozzi, and F. S. Mennini. The direct and indirect cost of diabetes in italy: a prevalence probabilistic approach. *The European Journal of Health Economics*, 17:139–147, 2016.

[89] R. Mauseth, I. B. Hirsch, J. Bollyky, R. Kircher, D. Matheson, S. Sanda, and C. Greenbaum. Use of a "fuzzy logic" controller in a closed-loop artificial pancreas. *Diabetes technology & therapeutics*, 15(8):628–633, 2013.

[90] D. Q. Mayne, E. C. Kerrigan, E. Van Wyk, and P. Falugi. Tube-based robust nonlinear model predictive control. *International journal of robust and nonlinear control*, 21(11):1341–1353, 2011.

[91] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

[92] D. Q. Mayne, M. M. Seron, and S. V. Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219–224, 2005.

[93] M. Messori, C. Toffanin, S. Del Favero, G. De Nicolao, C. Cobelli, and L. Magni. Model individualization for artificial pancreas. *Computer methods and programs in biomedicine*, 171:133–140, 2019.

[94] S. J. Moon, I. Jung, and C. Y. Park. Current Advances of Artificial Pancreas Systems: A Comprehensive Review of the Clinical Evidence. *Journal of Diabetes Science and Technology*, 45(6):831–839, 2021.

[95] N. Paoletti, K. S. Liu, H. Chen, S. A. Smolka, and S. Lin. Data-driven robust control for a closed-loop artificial pancreas. *IEEE/ACM transactions on computational biology and bioinformatics*, 17(6):1981–1993, 2019.

[96] M. Polver, D. Limon, F. Previdi, and A. Ferramosca. Robust tracking MPC for perturbed nonlinear systems. *IEEE Transactions on Automatic Control*, 2025.

[97] M. Polver, B. Sonzogni, M. Mazzoleni, F. Previdi, and A. Ferramosca. Artificial pancreas under a zone model predictive control based on gaussian process models: toward the personalization of the closed loop. *IFAC-PapersOnLine*, 56(2):9642–9647, 2023.

[98] Rawlings, J. Blake, and D. Q. Mayne. *Model predictive control: Theory and design*. Nob Hill Pub., 2009.

[99] F. Ribet, G. Stemme, and N. Roxhed. Real-time intradermal continuous glucose monitoring using a minimally invasive microneedle-based system. *Biomedical microdevices*, 20:1–10, 2018.

[100] R. Rickenbach, J. Köhler, A. Scampicchio, M. N. Zeilinger, and A. Carron. Active learning-based model predictive coverage control. *IEEE Transactions on Automatic Control*, 69(9):5931–5946, 2024.

[101] M. C. Riddell, I. W. Gallen, C. E. Smart, C. E. Taplin, P. Adolfsson, A. N. Lumb, A. Kowalski, R. Rabasa-Lhoret, R. J. McCrimmon, C. Hume, et al. Exercise management in type 1 diabetes: a consensus statement. *The lancet Diabetes & endocrinology*, 5(5):377–390, 2017.

[102] P. V. Röder, B. Wu, Y. Liu, and W. Han. Pancreatic regulation of glucose homeostasis. *Experimental & molecular medicine*, 48(3):e219–e219, 2016.

[103] Y. Ruan, M. E. Wilinska, H. Thabit, and R. Hovorka. Modeling day-to-day variability of glucose–insulin regulation over 12-week home use of closed-loop insulin delivery. *IEEE Transactions on Biomedical Engineering*, 64(6):1412–1419, 2016.

[104] A. Saad, C. Dalla Man, D. K. Nandy, J. A. Levine, A. E. Bharucha, R. A. Rizza, R. Basu, R. E. Carter, C. Cobelli, Y. C. Kudva, et al. Diurnal pattern to insulin secretion and insulin action in healthy individuals. *Diabetes*, 61(11):2691–2700, 2012.

[105] D. B. Sacks, M. Arnold, G. L. Bakris, D. E. Bruns, A. R. Horvath, Å. Lernmark, B. E. Metzger, D. M. Nathan, and M. S. Kirkman. Guidelines and recommendations for laboratory analysis in the diagnosis and management of diabetes mellitus. *Clinical chemistry*, 69(8):808–868, 2023.

[106] J.-L. Selam. Evolution of diabetes insulin delivery devices. *Journal of diabetes science and technology*, 4(3):505–513, 2010.

[107] D. Shi, E. Dassau, and F. J. Doyle. Adaptive zone model predictive control of artificial pancreas based on glucose-and velocity-dependent control penalties. *IEEE Trans. on Biomedical Engineering*, 66(4):1045–1054, 2018.

[108] B. Sonzogni, J. M. Manzano, M. Polver, F. Previdi, and A. Ferramosca. CHoKI-based MPC for blood glucose regulation in Artificial Pancreas. *IFAC-PapersOnLine*, 56(2):9672–9677, 2023.

[109] B. Sonzogni, J. M. Manzano, M. Polver, F. Previdi, and A. Ferramosca. CHoKI-Based MPC for Blood Glucose Regulation in Artificial Pancreas with Probabilistic Constraints. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 1619–1624. IEEE, 2023.

[110] B. Sonzogni, J. M. Manzano, M. Polver, F. Previdi, and A. Ferramosca. CHoKI-based MPC for blood glucose regulation in artificial Pancreas. *IFAC Journal of Systems and Control*, page 100294, 2025.

[111] B. Sonzogni, J. M. Manzano, F. Previdi, and A. Ferramosca. Insulin on Board safety constraint effect in a CHoKI-based MPC for Artificial Pancreas. *IFAC-PapersOnLine*, 58(24):257–262, 2024.

[112] B. Sonzogni, J. M. Manzano, F. Previdi, and A. Ferramosca. Hybrid Modeling of the Insulin-Glucose System: Combining Linear and Data-Driven Models for Artificial Pancreas. *IFAC-PapersOnLine*, 59(2):109–114, 2025.

[113] P. Soru, G. De Nicolao, C. Toffanin, C. Dalla Man, C. Cobelli, L. Magni, A. H. Consortium, et al. MPC based artificial pancreas: strategies for individualization and meal compensation. *Annual Reviews in Control*, 36(1):118–128, 2012.

[114] G. M. Steil. Algorithms for a closed-loop artificial pancreas: the case for proportional-integral-derivative control. *Journal of diabetes science and technology*, 7(6):1621–1631, 2013.

[115] The Epsilon Group. DMMS.R (Version 1.1) [Software]. Retrieved from https://tegvirginia.com/, 2016.

[116] C. Toffanin, M. Messori, F. Di Palma, G. De Nicolao, C. Cobelli, and L. Magni. Artificial pancreas: model predictive control design from clinical experience. *Journal of Diabetes Science and Technology*, 7(6):1470–1483, 2013.

[117] D. Ucieklak, S. Mrozinska, A. Wojnarska, M. T. Malecki, T. Klupa, and B. Matejko. Insulin-induced lipohypertrophy in patients with type 1 diabetes mellitus treated with an insulin pump. *International Journal of Endocrinology*, 2022(1):9169296, 2022.

[118] R. Valente, A. Coppola, C. M. Scandavini, A. Halimi, A. Magnusson, A. Lauro, I. Sotirova, U. Arnelo, and O. Franklin. Interactions between the exocrine and the endocrine pancreas. *Journal of Clinical Medicine*, 13(4):1179, 2024.

[119] R. Visentin, C. Dalla Man, Y. C. Kudva, A. Basu, and C. Cobelli. Circadian variability of insulin sensitivity: physiological input for in silico artificial pancreas. *Diabetes Technol Ther*, 17(1):1–7, 2015.

[120] A. Z. Woldaregay, E. Årsand, S. Walderhaug, D. Albers, L. Mamykina, T. Botsis, and G. Hartvigsen. Data-driven modeling and prediction of blood glucose

dynamics: Machine learning applications in type 1 diabetes. *Artificial intelligence in medicine*, 98:109–134, 2019.

[121] S. Yu, C. Maier, H. Chen, and F. Allgöwer. Tube MPC scheme based on robust control invariant set with application to Lipschitz nonlinear systems. *Systems & Control Letters*, 62(2):194–200, 2013.

[122] L. A. Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on systems, Man, and Cybernetics*, (1):28–44, 1973.

[123] M. N. Zeilinger, D. M. Raimondo, A. Domahidi, M. Morari, and C. N. Jones. On real-time robust model predictive control. *Automatica*, 50(3):683–694, 2014.

[124] Z.-H. Zhou. *Machine learning*. Springer nature, 2021.