UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DEPARTMENT OF INFORMATION ENGINEERING

MASTER COURSE IN
BIOMEDICAL ENGINEERING

Design of closed-loop algorithms and of a decision support app for glucose clamp experiments

Supervisor
Prof. Del Favero Simone

Student
Tubaro Giovanni

Coadvisor
Prof. Cappon Giacomo
Prof. Schiavon Michele

ACADEMIC YEAR 2023-24

Graduation Date 11/04/2024

Abstract

Glucose clamps are experimental techniques used to assets insulin action and glucose utilization. By "clamping" the blood glucose of the patient to a desired level, insulin sensitivity and β-cell functioning can be investigated. In this work we present an approach to modulate glucose infusion rate (GIR) through a semi-automated clamp technique, serving as a decision support system to clinicians.

We make two key contributions. Firstly, we investigate the application of control algorithms, specifically Proportional-Integral-Derivative (PID) controllers and Model Predictive Control (MPC), in the context of glucose clamp experiments. Using MATLAB and Simulink simulations, the two approaches were evaluated in both ideal and realistic scenarios, considering measurement variations and uncertainties typical in clinical environments. The results obtained were promising, demonstrating the efficacy of our semi-automated clamp technique in achieving targeted blood glucose (BG) levels. Both the Proportional-Integral-Derivative (PID) control algorithm and the Model Predictive Control (MPC) approach effectively modulated the glucose infusion rate (GIR), leading to close adherence to predefined clamp BG targets.

The second contribution is the design of a mobile application, Glucas 2.0, to support clinical researchers conducting glucose clamps experiments. This app, which enables the possibility to carry out more than one experiment at the time, provides suggestion of glucose infusion rates (GIR) generated by a PID controller.

Contents

Contents **1**

1 Introduction **3**

2 Computational Methods and Theory **7**
2.1 PID . . . 8
2.2 MPC . . . 9
2.2.1 MPC with Integral Action . . . 13

3 MATLAB and Simulink **18**
3.1 Type 1 Diabetic Simulator . . . 19
3.2 PID controller . . . 21
3.3 MPC controller . . . 23
3.3.1 Full Increment Velocity Form . . . 23
3.3.2 Unmeasured Disturbance Estimator . . . 26
3.4 Reference Profiles . . . 30
3.5 Performance metrics . . . 35

4 Results **36**
4.1 Noise-free Scenario . . . 36
4.1.1 PID controller . . . 36
4.1.2 MPC Controller with Full Increment Velocity Form . . . 47
4.1.3 MPC Controller with the Unmeasured Disturbance Estimator Approach . . . 59
4.2 Realistic Scenario . . . 70
4.2.1 PID controller . . . 70
4.2.2 MPC controller with Full Increment Velocity Form . . . 80
4.2.3 MPC controller with the Unmeasured Disturbance Estimator approach . . . 89
4.2.4 Comments on the results . . . 98

5 Comparing PID and MPC Performance **99**
5.1 Noise-free Scenario . . . 99
5.1.1 Hyperglycemic clamp . . . 99

5.1.2 Euglycemic clamp . . . 100
5.1.3 Hypoglycemic clamp . . . 101
5.1.4 ISO-IV clamp . . . 102
5.2 Realistic Scenario . . . 105
5.2.1 Hyperglycemic Clamp . . . 105
5.2.2 Euglycemic Clamp . . . 105
5.2.3 Hypoglycemic Clamp . . . 106
5.2.4 ISO-IV Clamp . . . 107
5.3 Final Comparison . . . 109

6 Mobile Application **110**
6.1 Introduction . . . 110
6.2 Dart and Flutter . . . 111
6.3 Project Structure . . . 111
6.3.1 UI . . . 111
6.3.2 Navigation . . . 112
6.3.3 State Management and Data Persistence . . . 112
6.4 App Workflow . . . 114

7 Conclusions and Future Developments **124**

Bibliography **126**

List of Figures **130**

List of Tables **138**

Chapter 1

Introduction

Experimental techniques such as the glucose clamp methodology have been instrumental in advancing our comprehension of glucose metabolism and insulin sensitivity. The glucose clamp, introduced by DeFronzo et al. [4] in 1979, has since become a cornerstone in the modeling of metabolic systems, enabling precise assessment of insulin action and glucose utilization in vivo. These clamps serve not only to elucidate fundamental physiological mechanisms but also to model complex biological systems, including the intricate dynamics of diabetes.

While the glucose clamp technique provides valuable insights into insulin action and glucose regulation, its application in clinical research and practice is hampered by several limitations. First and foremost, traditional glucose clamp studies are resource-intensive and laborious, requiring specialized equipment, skilled personnel, and extensive participant monitoring. In addition to that, the complexity of clamp procedures and the variability in individual responses can introduce bias and limit the reproducibility of results.

In recent years, there has been a growing interest in leveraging computational approaches and advanced technologies to enhance the efficiency and accuracy of glucose clamp experiments. Mathematical modeling techniques, such as compartmental models and physiological-based models, have been employed to simulate glucose-insulin dynamics and optimize clamp protocols. These modeling approaches enable researchers to predict glucose clamp outcomes, design personalized clamp protocols, and investigate the underlying mechanisms of insulin resistance and β-cell function.

In glucose clamp experiments the glucose regulation mechanism is perturbed, and blood glucose concentration is forced to remain at a desired level (achieved by "clamping" blood glucose concentration) through intravenous administration of glucose. There are three main types of glucose clamps that serve different research purposes:

Hyperglycemic Clamp In hyperglycemic clamps, blood glucose (BG) levels are elevated above normale range (up to 230-240 mg/dl). This enables to quantify β-cell sensitivity to glucose and examine the early and late phase

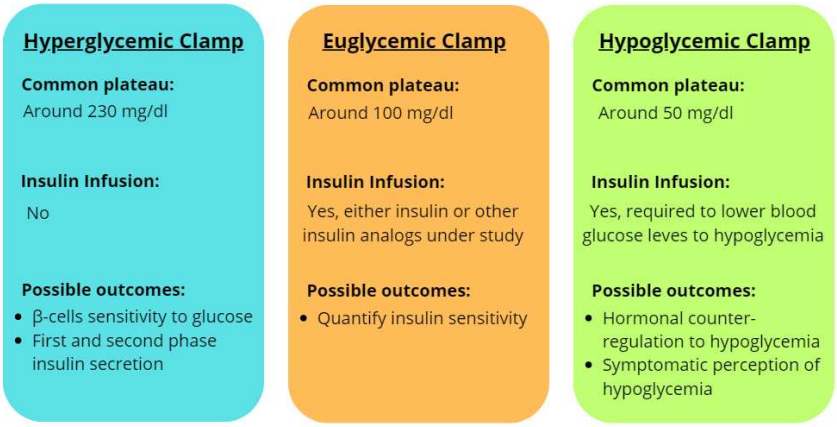


Figure 1.1: Common protocol of glycemic clamp experiments including possible outcomes.

of insulin secretion.

Euglycemic Clamp Euglycemic clamps involve maintaining blood glucose at a constant, normal level (100 mg/dl) despite the introduction of additional insulin. By measuring the amount of glucose required to counterbalance the insulin-induced increase in glucose consumption, researchers can quantify insulin sensitivity.

Hypoglycemic Clamp In hypoglycemic clamps blood glucose is decreased (typically by extra-insulin administration) and maintained in hypoglycemia (50 mg/dl) to facilitate the study of hormonal counter-regulation, cognitive functions during hypoglycemia, and the perception of hypoglycemic awareness.

Another type of clamp we wanted to explore is the ISO-IV clamp. These studies consist in a two day experiment. On the first day, the patient takes a Oral Glucose Tolerance Test (OGTT) ingesting a fixed quantity of glucose (in our study we analyzed both the 40g and 75g scenarios). The next day, the patient is subjected to an isoglycemic infusion (ISO-IV) of glucose, together with a bolus of subcutaneous insulin.

These types of clamp are useful to assest the incretin effect. The incretin effect is the increase in insulin response to nutrient ingestion compared to the response to glycemia alone. It is crucial in understanding how our bodies handle ingested carbohydrates and regulate glucose levels. It refers to the increased insulin response triggered by food intake compared to just the blood sugar levels alone. This response, mediated by factors in the gastrointestinal tract, is essential for keeping blood sugar levels stable after meals. Measuring the incretin

effect provides valuable insights into diabetes and other metabolic disorders related to mealtime glucose control. It also helps us understand how the gut and pancreas interact to regulate glucose levels. Accurate measurement of the incretin effect is important for research and clinical studies, as it can guide the development of new treatments for conditions affecting glucose metabolism.

As depicted by Aulinger and D'Alessio [18], with (almost) identical glucose doses administered (one via OGTT and one via ISO-IV) (Fig. 1.2), the insulin response exhibits a stark contrast (Fig. 1.3). The incretin effect manifests as a more than twofold increase in the insulin response to oral glucose ingestion (OGTT) compared to intravenous administration (ISO-IV).

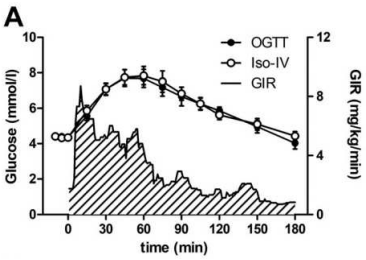


Figure 1.2: Glucose response for OGTT vs ISO-IV, taken from [19].

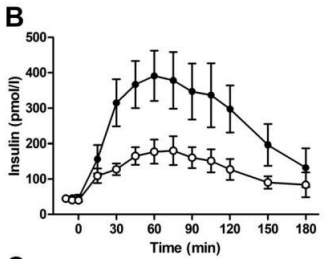


Figure 1.3: Insulin response for OGTT vs ISO-IV , taken from [19].

There are two main approaches to modulate the glucose infusion rate (GIR). In non-computerized clamps, GIR adjustments are made based on frequent BG measurements and clinical judgment. This task is far from trivial, and the expertise of the staff significantly impacts the quality of control achieved. Conversely, in fully automated clamps, BG levels are continuously monitored using a glucose sensor, which feeds data into a closed-loop algorithm. This algorithm then dynamically adjusts GIR in response to BG levels and predefined targets. In these clamp systems, glucose infusion occurs through an automatic actuator, eliminating the necessity for user intervention. The aim of automated clamps is to minimize variability in BG control stemming from human operators, thereby reducing control deviation overall. Nonetheless, their implementation necessitates additional specialized hardware to integrate automatic BG measurements and actuation of the GIR, resources that are frequently lacking. Moreover, complete automation presents regulatory hurdles in light of the Medical Device Regulation.

In this work we propose a third approach to modulate glucose infusion rate. We developed a semi automated clamp technique intended to serve as a decision support system rather than a fully automated clamp system. This hybrid approach allows for enhanced control over BG levels while still leveraging the expertise of clinical staff. Crucial to this decision support system is the design of control algorithms to modulate blood glucose in glucose clamps. In this study, we developed both a Proportional-Integral-Derivative (PID) control algorithm

and a Model Predictive Control (MPC) approach for adjusting the GIR to attain predefined clamp BG levels.

Additionally, we have designed a mobile application as part of the decision support system for clinical researchers conducting glucose clamp experiments. This app aims to help investigators by suggesting the optimal GIR to achieve pre-determined BG levels. Users retain the ability to intervene with the suggested control action, thus integrating clinical reasoning into the decision-making process.

Chapter 2

Computational Methods and Theory

In this section, we outline the design of the PID and MPC algorithm employed to offer GIR suggestions. To achieve this, we conceptualize BG control in glucose clamp experiments as a reference tracking problem in a closed loop strategy.

Closed-loop controllers, also known as feedback control systems, are essential components across various engineering and scientific disciplines, offering precise regulation and stabilization in dynamic environments. Unlike open-loop systems that operate without feedback, closed-loop controllers continuously monitor system output and adjust control actions based on measured feedback signals, aiming to maintain desired set points or track reference trajectories. In Fig. 2.1, we can observe a general closed-loop controller scheme. The controller uses the outputs of the system $y(t)$ (measured by the sensor), together with with the reference value $r(t)$, to generate the control inputs $u(t)$. These inputs are then applied to the system, which produces the outputs, and the process continues. Additionally, there may be disturbances acting on the system.

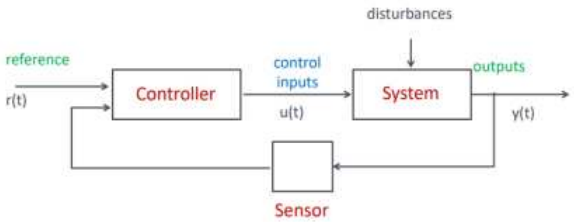


Figure 2.1: General closed-loop controller architecture, taken from [32].

In this framework, we will work with discrete variables, and we will assume uniform sampling, defining $T_s = \Delta_k = t_k - t_{k-1}$. Congruent with standard glucose clamp experiments, in our control algorithms we considered a sampling

time of 5 minutes.

Let $y(k) \in \mathbb{R}$ (mg/dl) be the patient BG levels at time t_k, the measured output of the process we want to control. Let $y_0(k) \in \mathbb{R}$ (mg/dl) be the desired glycemic reference trajectory ($r(t)$ in Fig.2.1). We aim to track y_0 as closely as possible.

The variable we can manipulate, the control input, is the glucose infusion rate $u(k) \in \mathbb{R}_0^+$ (g/min). This quantity must be greater than or equal to zero, leading to a strongly asymmetric effect on the admissible control actions.

2.1 PID

The first control technique we used is Proportional-Integral-Derivative control, where the computed GIR is the sum of three components:

$$u_{PID}(k) = u_P(k) + u_I(k) + u_D(k). \tag{2.1}$$

We define the tracking error as $e(k) = y_0(k) - y(k)$.

The first term is called 'proportional action', and it is proportional to $e(k)$, so the control action is:

$$u_P(k) = K_P \cdot e(k) \tag{2.2}$$

higher K_p, higher the control aggressiveness.

The second term is called 'integral action' and it is proportional to the saturated version of the integral of the tracking error $I(k)$. We have, in continuous time:

$$u_I(t) = K_I \cdot \int_0^t e(t)dt \tag{2.3}$$

which in discrete time it becomes:

$$u_I(k) = K_I \cdot I(k) \tag{2.4}$$

where $I(k)$ is the discrete integral of the error and it is defined as:

$$I(k)_{unsat} = I(k-1) + e(k) \cdot T_s \tag{2.5}$$

The saturated version is:

$$I(k) = max(I_{min}, I(k)_{unsat}) \tag{2.6}$$

and

$$I(k) = min(I_{max}, I((k)_{unsat}) \tag{2.7}$$

This approach is referred to as an 'Anti-wind-up scheme', implemented to prevent the accumulation of large quantities in the integral memory. In our scenario, we set $I_{\min} = 0$ $\frac{mg \cdot min}{dl}$ and $I_{\max} = 10000$ $\frac{mg \cdot min}{dl}$ We selected these values because in our problem, an accumulation of negative integral error is justifiable, particularly in euglycemic and hypoglycemic clamps, as the patient's

blood glucose (BG) y tends to be greater than the desired reference y_0 for the majority of the experiment. In PID control, the integral action plays a vital role as it enables the system to achieve zero-error tracking in steady-state conditions.

The third term is called 'Derivative Action' and it is proportional to the derivative of the tracking error. We have, in continuous time:

$$u_D(t) = K_D \cdot \frac{de(t)}{dt} \tag{2.8}$$

which in discrete time it becomes:

$$u_D(k) = K_D \cdot \frac{e(k) - e(k-1)}{T_s} \tag{2.9}$$

The derivative action takes into account the error trend, aiding in the prevention of overshoot. To mitigate large control actions resulting from sudden changes in the reference signal, we calculate the derivative of the output instead of the derivative of the error. Thus, we have:

$$u_D(k) = -K_D \cdot \frac{y(k) - y(k-1)}{T_s} \tag{2.10}$$

The three parameters K_P $[\frac{g \cdot dl}{mg \cdot min}]$, K_I $[\frac{g \cdot dl}{mg \cdot min^2}]$, and K_D $[\frac{g \cdot dl}{mg}]$ are employed to adjust the weighting of the various control actions.

Although it is a relatively simple technique, the PID approach is widely employed across diverse control problems and consistently delivers strong performance. The general PID architecture is displayed in Fig.2.2.

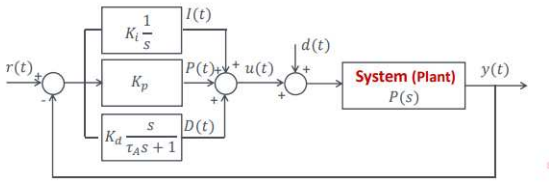


Figure 2.2: General PID controller architecture, taken from [32].

2.2 MPC

Another technique we employed is Model Predictive Control (MPC). Operating primarily in the digital framework, MPC is utilized to "re-formulate" the control problem as an optimization problem. The MPC algorithm computes the optimal control action for tracking the reference. An effective analogy for describing MPC reasoning is likening it to a chess player. In a match, the chess player can anticipate the impact of their moves on the game. Therefore, each time it is their turn:

1. They explore all possible sequences of N moves and choose the most effective one after evaluating their impact on the game.

2. They make the first move of the sequence and await the opponent's reaction.

3. Upon their next turn, if the opponent's reaction is unexpected (not considered in their predictions), the chess player will recompute a new optimal sequence of N moves.

4. They once again apply the first move and wait for the opponent's reaction.

This strategy is repeated until the game concludes.

Similar to the chess player, MPC utilizes a model of the system to predict the impact of a sequence of N control actions. Thus, at each step, the MPC:

1. Evaluates all possible sequences of N control actions and selects the optimal one based on their anticipated effect on the system over the next N steps.

2. Implements only the first control action and awaits the system's response (a new measurement).

3. Upon receiving the new measurement, the MPC updates or recomputes the optimal sequence of control actions for the next N steps.

4. It then applies the first control action once again.

This iterative process continues throughout the control task.

In MPC, N is referred to as the 'Prediction Horizon', indicating the span into the 'future' that the controller considers. Had the controller executed all N control actions in the initial step without pausing for measurements, this method would be termed 'open-loop'. What distinguishes the MPC strategy as 'closed-loop' is that, following the application of the first control action, the controller awaits the new measurement and recalculates the entire sequence. In Fig.2.3 we can see the general closed-loop scheme for reference tracking of a MPC controller.

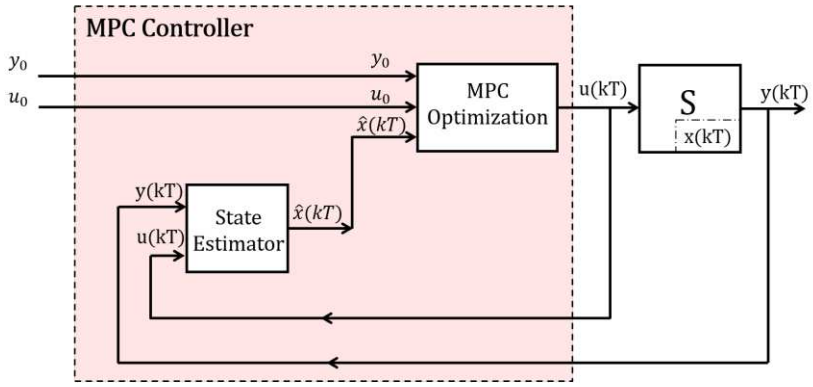


Figure 2.3: MPC for reference tracking controller scheme, taken from [32].

The key factors needed to describe a MPC algorithm are:

1. A mathematical model of the system to be controlled,
2. A cost function to optimise,
3. A set of constraints.

The Model Predictive Control framework offers considerable flexibility regarding model requirements for the algorithm. For our purposes, we opted to utilize a state-space model, which represents a highly versatile class of models.

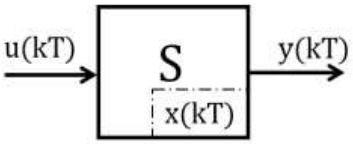


Figure 2.4: Discrete-time system, taken from [32].

Let us consider a discrete-time system S (Fig. 2.4), with m inputs $u(kT) \in \mathbf{R}^m$ and p outputs $y(kT) \in \mathbf{R}^p$. A general description for the system S is:

$$\begin{cases} x((k+1)T) = f(x(kT), u(kT)) \\ y(kT) = g(x(kT), u(kT)) \end{cases} \tag{2.11}$$

where $x \in \mathbf{R}^n$ is called State of the system, $f : \mathbb{R}^{n+m} \to \mathbb{R}^n$ and $g : \mathbb{R}^{n+m} \to \mathbb{R}^p$.

For our objectives, we concentrated on linear systems, which results in the following state model:

$$\begin{cases} x((k+1)T) = Ax(kT) + Bu(kT) + Md(kT) \\ y(kT) = Cx(kT) + Du(kT) \end{cases} \tag{2.12}$$

with $A \in \mathbf{R}^{n\times n}, B \in \mathbf{R}^{n\times m}, M \in \mathbf{R}^{n\times m_{dist}}, C \in \mathbf{R}^{p\times n}, D \in \mathbf{R}^{p\times m}$. Additionally, we will assume $D = 0$. $d(kT)$ is a possible disturbance acting on the system. Even if extremely simplified, in this work we chose to use the following monocompartimental model to for internal reasoning in our MPC. (Fig. 2.5). The Fig.2.6 illustrates the model used for MPC reasoning in the euglycemic and hypoglycemic clamp. As we can see there is an unknown disturbance $d(t)$ caused by the increased insulin level needed to lower BG. Since is unknown, we will not consider it in our MPC model, so we will set $d(kT)$=0;

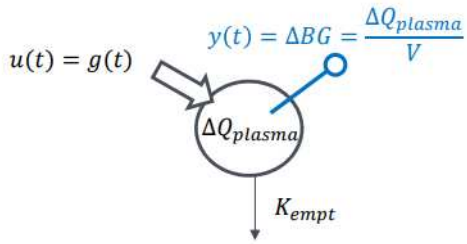


Figure 2.5: Monocompartimental model for MPC reasoning.

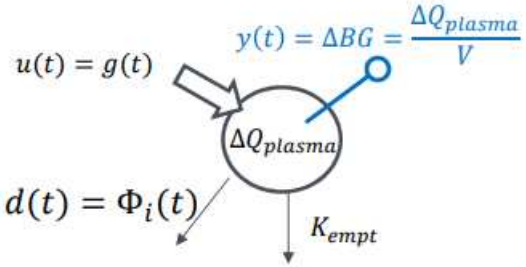


Figure 2.6: Monocompartimental model for MPC reasoning in euglycemic and hypoglycemic clamps.

This simple model describes how an input infusion $u(t) = g(t)$ $[g/min]$ affects the quantity of glucose in the plasma $\Delta Q_{plasma} = Q_{plasma} - Q_{eq}[mg]$.

K_{empt}	0.003 $[1/min]$
V	210 $[dl]$
BG_{eq}	patient specific, it represents BG at the beginning of the experiment (130-160 $[mg/dl]$)

Table 2.1: Model parameters.

The model's output is $\Delta BG = BG - BG_{eq} = \frac{\Delta Q_{plasma}}{V}[mg/dl]$. From this state-space model, we derived the following equations:

$$\begin{cases} \Delta\dot{Q}(t) = -K_{empt}\Delta Q(t) + 1000g(t) \\ \Delta BG(t) = \frac{\Delta Q(t)}{V} \end{cases} \tag{2.13}$$

We observe that this system is linear. Thus, we find $A = -K_{empt}$, $B = 1000$, $C = \frac{1}{V}$, and $D = 0$. Since this system is in continuous time, we will discretize the model in MATLAB. The parameters K_{empt} and V (Table 2.1) were estimated from experiments on patient #adult001 and used for all the population.

In addition to the model, MPC requires a cost function to minimize. We opted for a quadratic cost function for reference tracking, signifying our desire for the MPC to track the reference signal in the output y_0, while also ensuring that the input u_0 remains reasonably close to a predetermined value. The cost function is:

$$J = \sum_{i=0}^{N-1} \left[(\hat{y}(k+i) - y_0(k+i))^T Q (\hat{y}(k+i) - y_0(k+i)) + (u(k+i) - u_0(k+i))^T R (u(k+i) - u_0(k+i)) \right] \tag{2.14}$$

Where $\hat{y}(k+i)$ represents the predicted output at time $k+i$ in response to the previous control action $u(k+i-1)$, and $u(k+i)$ denotes the control action planned for time $k+i$. Since we are dealing with a SISO system (Single Input Single Output), we have $Q, R \in \mathbf{R}$, so we can rewrite 2.14 as:

$$J = \sum_{i=0}^{N-1} \left[\frac{Q}{R} (\hat{y}(k+i) - y_0(k+i))^2 + (u(k+i) - u_0(k+i))^2 \right] \tag{2.15}$$

The parameters Q and R regulate control aggressiveness. Specifically, if $Q >> R$, the controller will exhibit more aggressiveness because deviating from y_0 will incur a higher cost. On the other hand, if $Q << R$, the controller will tend to apply control actions closer to u_0, thereby reducing aggressiveness.

The final key element is the set of constraints. MPC can as a matter of fact incorporate constraints on both the input and state of the system. In our problem, we have only one constraint on the control action: it must be greater than or equal to 0 (since we cannot remove glucose once it has been injected). Therefore, our set of constraints is simplified to $u(k) \geq 0 \ \forall k$.

2.2.1 MPC with Integral Action

As we have seen in Section 2.2, the MPC controller at each step attempts to minimize the cost function J, continuously trying to find a trade off between

following output and input references y_0 and u_0. In contrast to PID control, where zero tracking error in steady state is achieved through the integral action, the MPC strategy reviewed so far does not garantee to eliminate the offset as the controller continuously aims to find balance between the two components in J. There are several techniques available to incorporate the integral action into our MPC controller. We will explore two of these methods: the Full Increment Velocity Form and the Unmeasured Disturbance Estimator approach.

2.2.1.1 Full Increment Velocity Form

The idea under this technique is to penalize $\delta u(k)$ in J instead of $u(k)$. By penalizing $\delta u(k) = u(k) - u(k-1)$ the controller can deliver a constant control action without penalty. The way this is done is considering the variation of all variables. Starting from the system model

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) + Md(k) \\ y(k) = Cx(k) \end{cases} \tag{2.16}$$

we consider the increment of the state $\delta x(k) = x(k) - x(k-1)$ and of the other variables: $\delta d(k) = d(k) - d(k-1)$, $\delta y(k+1) = C\delta x(k+1)$, $y(k+1) = C\delta x(k+1) + y(k) = CA\delta x(k) + CB\delta u(k) + CM\delta d(k) + y(k)$. We introduce the augmented state

$$x_a(k) = \begin{bmatrix} \delta x(k) \\ y(k) \end{bmatrix} \tag{2.17}$$

that leads to the augmented model

$$\begin{aligned} \begin{bmatrix} \delta x(k+1) \\ y(k+1) \end{bmatrix} &= \begin{bmatrix} A & 0 \\ CA & I \end{bmatrix} \begin{bmatrix} \delta x(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} B \\ CB \end{bmatrix} \delta u(k) + \begin{bmatrix} M \\ CM \end{bmatrix} \delta d(k) \\ y(k) &= \begin{bmatrix} 0 & I \end{bmatrix} \begin{bmatrix} \delta x(k) \\ y(k) \end{bmatrix} \end{aligned} \tag{2.18}$$

This strategy, known as the full-increment velocity form (FIVF), enables zero tracking error in steady state. In Figure 2.7, we can observe its control scheme.

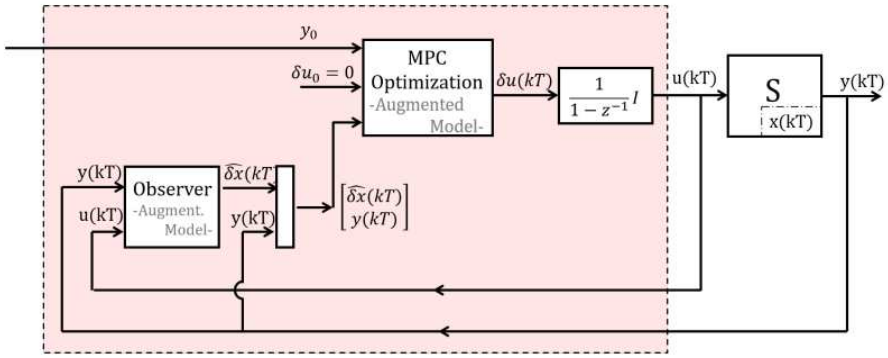


Figure 2.7: Full increment velocity form control scheme, taken from [32].

In our case, the augmented system becomes

$$\begin{bmatrix}\delta\Delta Q_{plasma}(k+1)\\ \Delta BG(k+1)\end{bmatrix} = \begin{bmatrix}A & 0\\ CA & 1\end{bmatrix}\begin{bmatrix}\delta\Delta Q_{plasma}(k)\\ \Delta BG(k)\end{bmatrix} + \begin{bmatrix}B\\ CB\end{bmatrix}\delta g(k)$$
$$\Delta BG(k) = \begin{bmatrix}0 & 1\end{bmatrix}\begin{bmatrix}\delta\Delta Q_{plasma}(k)\\ \Delta BG(k)\end{bmatrix} \tag{2.19}$$

where the matrices A, B and C are the discrete version of the matrices of the continuous-time system.

2.2.1.2 Unmeasured Disturbance Estimator

An alternative approach to the velocity form is the Unmeasured Disturbance Estimator (UDE). The idea behind this strategy is that if the MPC computation does not provide a zero-error tracking in steady-state, a constant unmeasured disturbance $d_u(k)$ might be acting on the system. The new model becomes:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) + Md(k) + M_u d_u(k)\\ d_u(k+1) = d_u(k)\\ y(k) = Cx(k) + C_u d_u(k)\end{cases} \tag{2.20}$$

After introducing the augmemted state $x_a(k) = \begin{bmatrix}x(k)\\ d_u(k)\end{bmatrix}$, the augmented model is

$$\begin{bmatrix}x(k+1)\\ d_u(k+1)\end{bmatrix} = \begin{bmatrix}A & M_u\\ 0 & I\end{bmatrix}\begin{bmatrix}x(k)\\ d_u(k)\end{bmatrix} + \begin{bmatrix}B\\ 0\end{bmatrix}u(k) + \begin{bmatrix}M\\ 0\end{bmatrix}d(k)$$
$$y(k) = \begin{bmatrix}C & C_u\end{bmatrix}\begin{bmatrix}x(k)\\ d_u(k)\end{bmatrix} \tag{2.21}$$

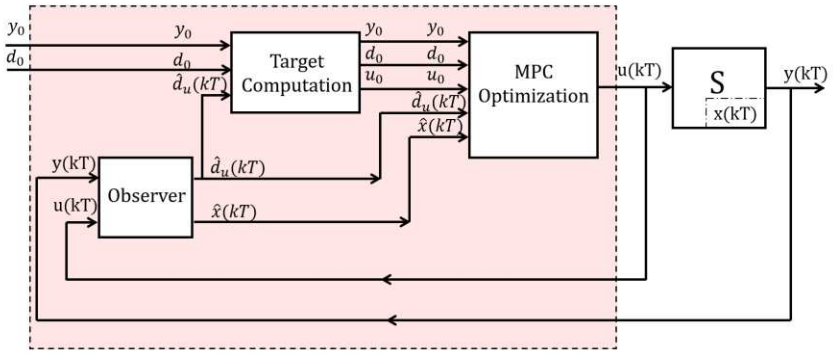


Figure 2.8: Unmeasured Disturbance Estimator control scheme, taken from [32].

The matrices M_u and C_u are design parameters to be chosen by the algorithm designer. It has been shown that estimating the unmeasured disturbance (by estimating the augmented state) leads to zero-error tracking in steady-state. In Figure 2.8, we can observe its control scheme. In our problem the augmented model becomes

$$\begin{aligned} \begin{bmatrix} \Delta Q_{plasma}(k+1) \\ d_u(k+1) \end{bmatrix} &= \begin{bmatrix} A & M_u \\ 0 & I \end{bmatrix} \begin{bmatrix} \Delta Q_{plasma}(k) \\ d_u(k) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} g(k) \\ \Delta BG(k) &= \begin{bmatrix} C & C_u \end{bmatrix} \begin{bmatrix} \Delta Q_{plasma}(k) \\ d_u(k) \end{bmatrix} \end{aligned} \tag{2.22}$$

again, the matrices A, B and C are the discrete version of the matrices of the continuous-time system.

2.2.1.3 State Estimation: the Kalman Filter

As we have seen in the previous section, to estimate the unmeasured disturbance acting on our system we need to estimate the augmented state $x_a(k) = \begin{bmatrix} x(k) \\ d_u(k) \end{bmatrix}$. we will do this using the Kalman filter.

The Kalman filter is a recursive algorithm that estimates the state of a dynamic system from a series of noisy measurements. It operates on a series of measurements over time, providing an optimal estimate of the state of the system by taking into account both the system's dynamics and the uncertainty in the measurements.

State equation with noise:

$$x(k+1) = Ax(k) + Bu(k) + w(k)$$

Where $x(k)$ is the true state at time k and $w(k)$ is the process noise, assumed to be Gaussian with zero mean and covariance Q_k.

Measurement equation with noise:

$$y(k) = Cx(k) + v(k)$$

Where $y(k)$ is the measurement at time k and $v(k)$ is the measurement noise, assumed to be Gaussian with zero mean and covariance R_k.

The filter consists of two main steps: prediction and update. In the prediction step, the filter uses the system's dynamic model to predict the state at the next time step, incorporating the process noise. Mathematically, the prediction step is represented by the following equations:

$$\hat{x}(k+1|k) = A\hat{x}(k|k) + Bu(k)$$
$$P(k+1|k) = AP(k|k)A^T + Q$$

Where $\hat{x}(k+1|k)$ is the predicted state estimate at time $k+1$ given measurements up to time k, $\hat{x}(k|k)$ is the previous state estimate, P is the covariance matrix of the estimation error $\tilde{x(k)} = x(k) - \hat{x}(k)$.

In the update step, the filter incorporates a new measurement, adjusting the predicted state estimate based on the difference between the predicted measurement and the actual measurement. The update equations are as follows:

$$M(k+1) = P(k+1|k)C^T(CP(k+1|k)C^T + R)^{-1}$$
$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + M(k+1)(y(k+1) - C\hat{x}(k+1|k))$$
$$P(k+1|k+1) = P(k+1|k) - P(k+1|k)C^T(CP(k+1|k)C^T + R)^{-1}CP(k+1|k)$$

Where M is called Kalman gain.

These equations encapsulate the core operations of the Kalman filter, enabling it to provide an optimal estimate of the system's state even in the presence of noise and uncertainty.

Regarding the Full Increment Velocity form, luckily the augmented state is defined as $x_a(k) = \begin{bmatrix} \delta x(k) \\ y(k) \end{bmatrix}$, so the Kalman filter is not needed to estimate it since both $\delta x(k)$ and $y(k)$ are variables we have access to. However, in the Unmeasured Disturbance Estimator approach, we do not have this luxury because we need to estimate the disturbance, which is inherently unknown.

Chapter 3

MATLAB and Simulink

To tune and test the control algorithms, we implemented them using MATLAB and Simulink. MATLAB (MATrix LABoratory) and Simulink are powerful computational tools widely used in various fields, including engineering, science, mathematics, and finance. Developed by MathWorks, MATLAB provides an interactive environment for numerical computation, data analysis, visualization, and algorithm development. Simulink, on the other hand, is a graphical simulation and model-based design platform that allows engineers and researchers to simulate, model, and analyze dynamic systems using block diagrams.

MATLAB is renowned for its versatility and ease of use, offering a rich set of built-in functions and toolboxes for solving a wide range of mathematical and engineering problems. Its intuitive syntax and extensive library of functions make it an ideal choice for prototyping algorithms, conducting numerical simulations, and performing data analysis tasks. MATLAB supports various data types, including scalars, vectors, matrices, and multidimensional arrays, making it suitable for handling complex mathematical operations and large datasets efficiently.

In addition to its core functionality, MATLAB provides specialized toolboxes for specific applications, such as signal processing, image processing, control systems, optimization, and machine learning. These toolboxes extend MATLAB's capabilities, enabling users to tackle domain-specific challenges and accelerate their research and development efforts.

Simulink complements MATLAB by offering a graphical environment for designing, simulating, and analyzing dynamic systems. Using a block diagram approach, users can model complex systems by connecting predefined blocks representing various components and subsystems. Simulink supports both continuous-time and discrete-time simulations, allowing engineers to simulate the behavior of systems over time and analyze their dynamic responses.

One of the key advantages of Simulink is its seamless integration with MATLAB, enabling users to incorporate MATLAB code directly into Simulink models and vice versa. This integration facilitates the development of hybrid models combining numerical computation, algorithmic logic, and system dynamics,

making it easier to prototype and verify complex control algorithms and dynamic systems.

3.1 Type 1 Diabetic Simulator

To fine-tune the control algorithm and conduct comprehensive testing before running a trial in humans, we utilized a realistic simulator to perform glucose clamp experiments in silico. The simulator comprises an accurate model of glucose-insulin physiology.

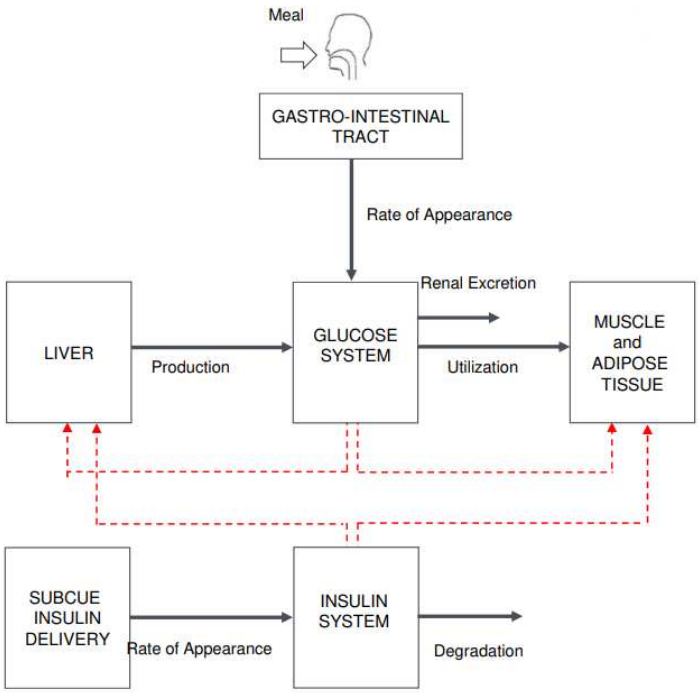


Figure 3.1: Scheme of T1D model proposed by Dalla Man et al. [3].

To model the pharmacodynamics and pharmacokinetics of glucose and insulin in type 1 diabetic patients (see Fig. 3.1), we utilized a simplified version of the simulator proposed by Dalla Man et al. [3], [14], [15]. This simplified version of the simulator includes 100 virtual patients to mimic the metabolic variability observed in real subjects. The UVA/Padova Type 1 Diabetes Simulator is accepted by the Food and Drug Administration as a substitute to animal trials for studies on Type 1 Diabetes.

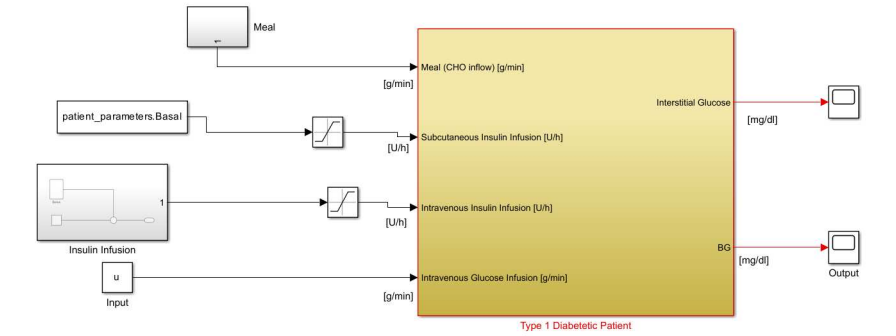


Figure 3.2: Simulink scheme of the simplified version of the T1D model proposed by Dalla Man et al. [3].

In Fig.3.2 the Simulink scheme of the simplified version of the T1D simulator is displayed. We have 4 main inputs:

- Meal intake
- Subcutaneous Insulin Infusion
- Intravenous Insulin Infusion
- Intravenous Glucose Infusion

and the main output is the blood glucose level (BG) of the patient. In our study, the control variable that both the PID and MPC algorithms can manipulate is the intravenous glucose infusion rate. The other inputs of the simulator remain unchanged by the controllers.

3.2 PID controller

To design and tune the PID algorithm we developed the Simulink scheme shown in Fig.3.3 and Fig.3.4.

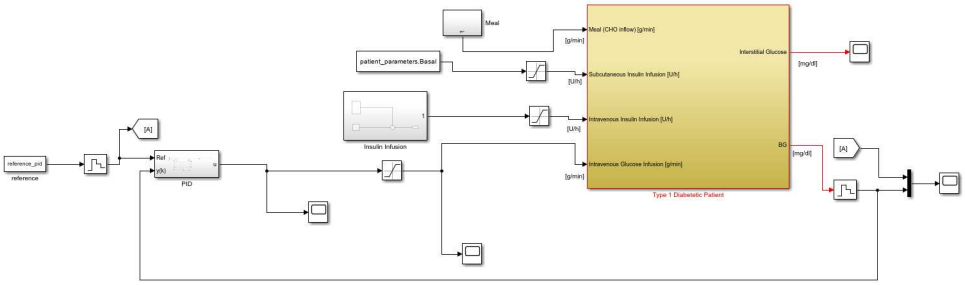


Figure 3.3: Simulink scheme of the closed loop PID controller.

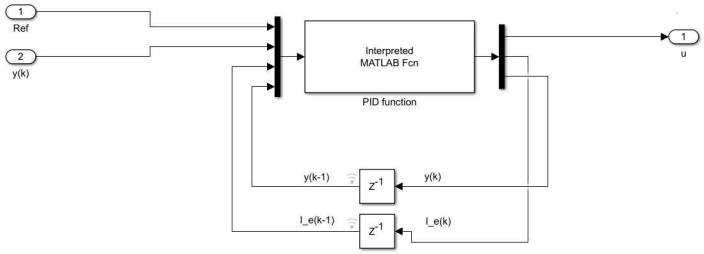


Figure 3.4: Zoom inside the PID block

```
function output=compute_PID_control(ref,y,Ie_old,y_old,controller_paramters)


%% Proportional Action
    P=controller_paramters.Kp*(ref-y);          %[g/min]

%% Integral Action

    % Integral of the error
        Ie=Ie_old+(ref-y)*controller_paramters.Ts; %[(mg/dl)*min]

    % AntiWind-up
        Ie=min(Ie,controller_paramters.Ie_max);
        Ie=max(Ie,controller_paramters.Ie_min);

      I=controller_paramters.Ki*Ie;             %[g/min]

%% Derivative Action
     dy=(y-y_old)/controller_paramters.Ts;    %[mg/dl/min]
     D= -controller_paramters.Kd*dy;            %[g/min]


%% Total Control Action
    u= P+I+D; %[g/min]

%% Building the Output Vector
    output=[u,Ie,y,P,I,D]';
```

Figure 3.5: MATLAB Function to calculate the GIR with PID controller.

The "Interpreted MATLAB function" we used is the one displayed in Fig.3.5. The function takes the following inputs:

1. The reference BG signal (ref).
2. The BG $y(k)$ resulting from the suggested GIR passed through feedback (y).
3. The last system output value $y(k-1)$, necessary for computing the discrete approximation of the derivative when the current system output becomes available (y_old).
4. The past integration of the error, defined as the sum of all previous errors to calculate the integral action (Ie_old).
5. The chosen controller parameters (Kp, Ki, Kd, Ie_max, Ie_min).

3.3 MPC controller

Similarly to what has been accomplished for the PID controller, we have devised a Simulink scheme for the MPC controller. In the upcoming sections, we will present the MATLAB and Simulink components for both the Full Increment Velocity form and the Unmeasured Disturbance Estimator approaches, with and without the look ahead feature.

3.3.1 Full Increment Velocity Form

In Fig.3.6 and Fig.3.7 the Simulink scheme for the MPC controller with Full Increment Velocity form without look ahead is displayed.

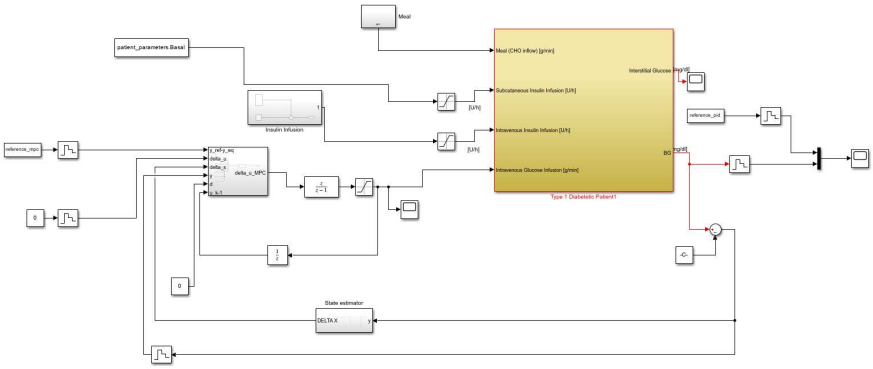


Figure 3.6: Simulink scheme of the closed loop MPC with FIVF controller.

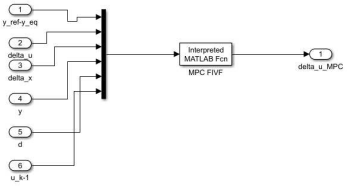


Figure 3.7: Zoom inside the MPC with FIVF block.

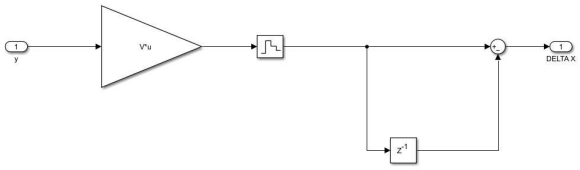


Figure 3.8: Zoom inside the block used to access to the state of the system.

```
function delta_u_MPC=compute_MPC_control_fivf(y0_current,delta_u0_current,delta_x,y_current,d_current,u_k_meno_1,condensedMatrices,num_of_input,N)
y0=kron(ones(N,1),y0_current);
u0=kron(ones(N,1),delta_u0_current);
d=kron(ones(N,1),d_current);

x_augmented=[delta_x;y_current];


Q_qp= condensedMatrices.call_BC'*condensedMatrices.call_Q*condensedMatrices.call_BC+condensedMatrices.call_R;

c_qp=condensedMatrices.call_BC'*condensedMatrices.call_Q*(condensedMatrices.call_AC*x_augmented+condensedMatrices.call_MC*d-y0)-condensedMatrices.call_R*u0;


A_qp=-condensedMatrices.call_F;
b_qp=u_k_meno_1*ones(N,1);



options = optimset('Display','off','Algorithm','interior-point-convex');
[delta_u_sequence,~,exit_flag]=quadprog((Q_qp+Q_qp')/2,c_qp,A_qp,b_qp,[],[],[],[],[],options); % To Avoid warning


delta_u_MPC_sequence=reshape(delta_u_sequence,num_of_input,size(delta_u_sequence,1)/num_of_input);

delta_u_MPC=delta_u_MPC_sequence(:,1);
```

Figure 3.9: MATLAB Function to calculate the GIR with MPC with FIVF controller.

The "Interpreted MATLAB function" we used is the one displayed in Fig.3.9. The function takes the following inputs:

1. The reference BG signal to which is subtracted BG_{eq}, to have a step reference that starts from 0 ($y0\_current$).
2. the discrete version of the derivative of the reference control variable $delta\_u0\_current$, we set it to 0.
3. The discrete version of the derivative of the state of the system ($delta\_x$).
4. The BG $y(k)$ resulting from the suggested GIR passed through feedback ($y\_current$).
5. The value of the disturbance acting on the system $d\_current$ (since it is unannounced we set it to 0).
6. The value of the previous control action $u\_k\_meno\_1$
7. The condensed matrices needed for the MPC algorithm
8. the number of inputs m (in our case is 1)
9. the prediction horizon N

In Fig.3.10 the Simulink scheme for the Full Increment Velocity form with look ahead is shown. It is quite similar to the one without this feature, but we can see that now the scheme has the "clock" block.

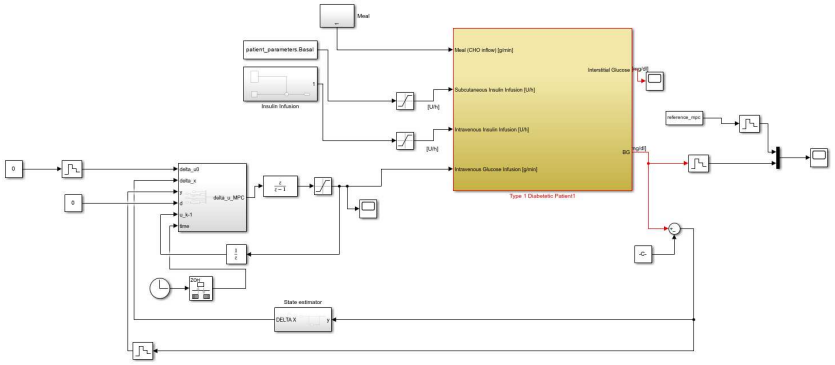


Figure 3.10: Simulink scheme of the closed loop MPC with FIVF controller with look ahead.

```
function delta_u_MPC=compute_MPC_control_fivf_look_ahead(delta_u0_current,delta_x,y_current,d_current,u_k_meno_1,current_time,reference,condensedMatrices,num_of_input,N)
PH=N;
u0=kron(ones(N,1),delta_u0_current);
d=kron(ones(N,1),d_current);
t0=reference.time;
y0_total=reference.signals.values;
current_index = find(t0==current_time);

if current_index+PH>length(y0_total)
    index_remaining=current_index+PH-length(y0_total);
    y0_call=y0_total(current_index+1:length(y0_total));
    for i=1:index_remaining
        y0_call=[y0_call;y0_total(end)];
    end
else
    y0_call = y0_total(current_index+1:current_index+PH);
end
x_augmented=[delta_x;y_current];

Q_qp= condensedMatrices.call_BC'*condensedMatrices.call_Q*condensedMatrices.call_BC+condensedMatrices.call_R;
c_qp=condensedMatrices.call_BC'*condensedMatrices.call_Q*(condensedMatrices.call_AC*x_augmented+condensedMatrices.call_MC*d-y0_call)-condensedMatrices.call_R*u0;

A_qp=-condensedMatrices.call_F;
b_qp=u_k_meno_1*ones(N,1);

options = optimset('Display','off','Algorithm','interior-point-convex');
[delta_u_sequence,~,exit_flag]=quadprog((Q_qp+Q_qp')/2,c_qp,A_qp,b_qp,[],[],[],[],[],options); % To Avoid warning
delta_u_MPC_sequence=reshape(delta_u_sequence,num_of_input,size(delta_u_sequence,1)/num_of_input);
delta_u_MPC=delta_u_MPC_sequence(:,1);
```

Figure 3.11: MATLAB Function to calculate the GIR with MPC with FIVF controller with look ahead.

By analyzing the function that computes MPC FIVF control with look ahead (Fig. 3.11), we can observe that the reference value is no longer constant over the prediction horizon. Through the utilization of the clock, we determine the current time instance in the simulation. Consequently, we can compute the vector of reference values over the prediction horizon ($y0\_call$), taking into account the impact of the future N control actions when making decisions in the present.

To understand the importance of the look ahead feature we propose an example on a simple "toy" problem: the tank level control (Fig.3.12). The input of the system is the inflow of water ϕ and the output is the height of the water column.

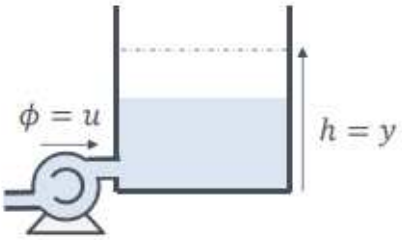


Figure 3.12: Tank level problem.

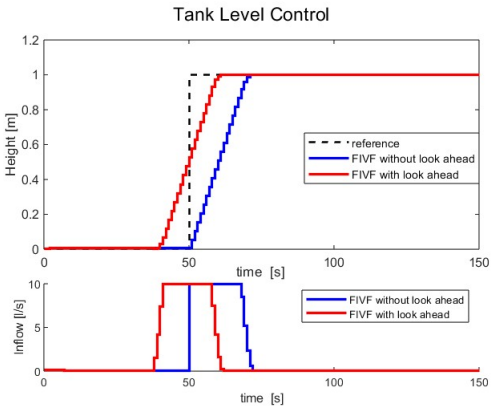


Figure 3.13: Tank level control: MPC FIVF with and without look ahead.

In Fig. 3.13, we can observe the performance of two MPC controllers with full increment velocity form: one with look ahead and one without. It is apparent that the former possesses the capability to anticipate the change of the reference and consider its transition over the prediction horizon (in this case, $N = 20$). This feature is essential to all MPC controllers.

3.3.2 Unmeasured Disturbance Estimator

As we have seen for the Full Increment Velocity form, also for the Unmeasured Disturbance Estimator technique we implemented the controllers with and without look ahead. In this section we will introduce only the Simulink schemes and MATLAB functions used to derive a MPC UDE controller with look ahead. The considerations outlined in section 3.3.1 regarding the difference between look ahead and non-look ahead controllers hold true for the Unmeasured Disturbance Estimator approach as well.

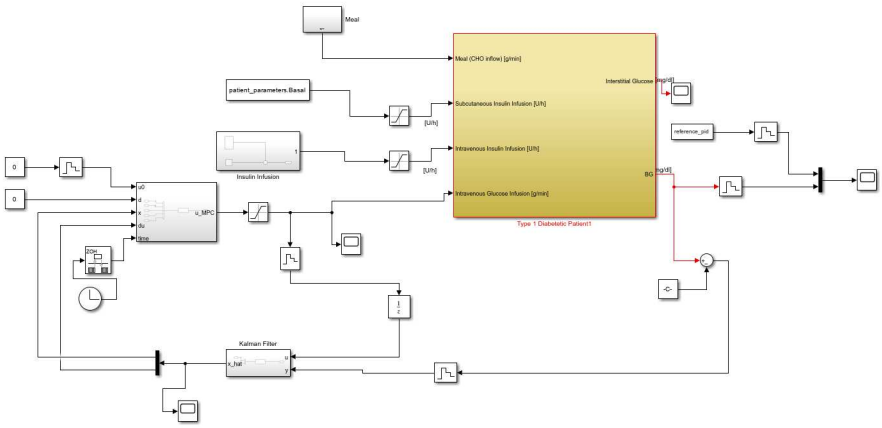


Figure 3.14: Simulink scheme of the closed loop MPC with UDE controller with look ahead.

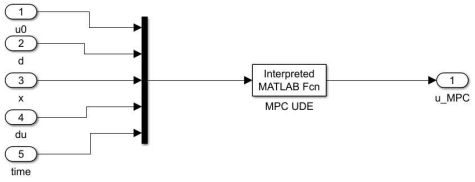


Figure 3.15: Zoom inside the MPC with UDE block.

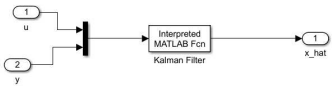


Figure 3.16: Zoom inside the Kalman filter block used to estimate the state.

```
function u_MPC=compute_MPC_control_UDE_look_ahead(u0_current,d_current,x_hat_current,du_hat_current,current_time,reference,condensedMatrices,num_of_input,N)
PH=N;
u0=kron(ones(N,1),u0_current);
d=kron(ones(N,1),d_current);

x_augmented=[x_hat_current;du_hat_current];

t0=reference.time;
y0_total=reference.signals.values;
current_index = find(t0==current_time);
if current_index+PH>length(y0_total)
    index_remaining=current_index+PH-length(y0_total);
    y0_call=y0_total(current_index+1:length(y0_total));
    for i=1:index_remaining
        y0_call=[y0_call;y0_total(end)];
    end
else
    y0_call = y0_total(current_index+1:current_index+PH);
end

Q_qp= condensedMatrices.call_BC'*condensedMatrices.call_Q*condensedMatrices.call_BC+condensedMatrices.call_R;
c_qp=condensedMatrices.call_BC'*condensedMatrices.call_Q*(condensedMatrices.call_AC*x_augmented+condensedMatrices.call_MC*d-y0_call)-condensedMatrices.call_R*u0;

A_qp=condensedMatrices.call_F;
b_qp=condensedMatrices.call_f;

options = optimset('Display','off','Algorithm','interior-point-convex');
[u_sequence,~,exit_flag]=quadprog((Q_qp+Q_qp')/2,c_qp,A_qp,b_qp,[],[],[],[],options); % To Avoid warning
u_MPC_sequence=reshape(u_sequence,num_of_input,size(u_sequence,1)/num_of_input);
u_MPC=u_MPC_sequence(:,1);
```

Figure 3.17: MATLAB Function to calculate the GIR with MPC with UDE controller.

In Fig.3.14 the Simulink scheme for this technique to achieve zero-error reference tracking in steady state is displayed, with a zoom inside the "MPC" and "Kalman Filter" block (Fig.3.15 and Fig.3.16 respectively). The "Interpreted MATLAB function" we used to compute MPC control is the one displayed in Fig.3.17. The function takes the following inputs:

1. The reference control variable $u0\_current$, we set it to 0.
2. The value of the disturbance acting on the system $d\_current$ (since it is unannounced we set it to 0).
3. The value of the estimate of the state of the system ($x\_hat\_current$).
4. The value of the estimate of the unmeasured disturbance acting on the system ($du\_hat\_current$).
5. The current time of the simulation to activate the feature of look ahead ($current\_time$)
6. the reference value to track
7. The condensed matrices needed for the MPC algorithm
8. the number of inputs m (in our case is 1)
9. the prediction horizon N

On the other hand, the function responsible for providing an estimate of the augmented state using the Kalman Filter is depicted in Fig. 3.18. It takes as inputs the current blood glucose measurement and the previous glucose infusion rate, in addition to the system model matrices and a structure named "$tuning\_parameters$" which encompasses all parameters necessary for tuning the controller to achieve the desired performance. As we have already explained

in Section 2.2.1.3, the Kalman filter provides an estimate of the augmented state in this configuration, which is then used to compute MPC control and achieve zero-error tracking. In all our configurations, we set the initial estimation of the state to 0, while the initial variance of the estimation error to the identity matrix.

```
function x_hat = KalmanFilter(y, u, SystemModel,tuning_parameters)

    persistent P_hat x_hat_prev;

    A=SystemModel.A;
    B=SystemModel.B;
    C=SystemModel.C;
    D=SystemModel.D;
    Q=tuning_parameters.Qk;
    R=tuning_parameters.Rk;
    P0=tuning_parameters.P0;
    x0=tuning_parameters.x0;

    if isempty(P_hat)

        P_hat = P0;
        x_hat_prev = x0;
    end


    x_hat_minus = A * x_hat_prev + B * u;
    P_minus = A * P_hat * A' + Q;


    K = P_minus * C' / (C * P_minus * C' + R);
    x_hat = x_hat_minus + K * (y - C * x_hat_minus - D * u);

    P_hat = (eye(size(A)) - K * C) * P_minus;
    x_hat_prev = x_hat;
end
```

Figure 3.18: MATLAB Function to calculate the estimate of the augmented state by using the Kalman filter.

3.4 Reference Profiles

As mentioned in section 1, we have three main types of clamps: hyperglycemic, huglycemic and hypoglycemic.

At the onset of each simulated trial, the patient's blood glucose level is at its basal range (approximately 130-160 mg/dl). During the initial 90 minutes, blood glucose measurements are collected, and no intravenous glucose is administered. The experiment will last for approximately 250 minutes for an hyperglycemic clamp, and around 400 minutes for euglycemic and hypoglycemic clamps.

For the hyperglycemic clamp, the target reference is 230 mg/dl, as we can see in Fig.3.19 (the BG level before the step is the basal one of patient #001). The two grey lines represent ±10% of the BG target. No intravenous insulin will be administered in these glucose clamps , only basal subcutaneous infusion simulating pancreas-generated basal insulin.

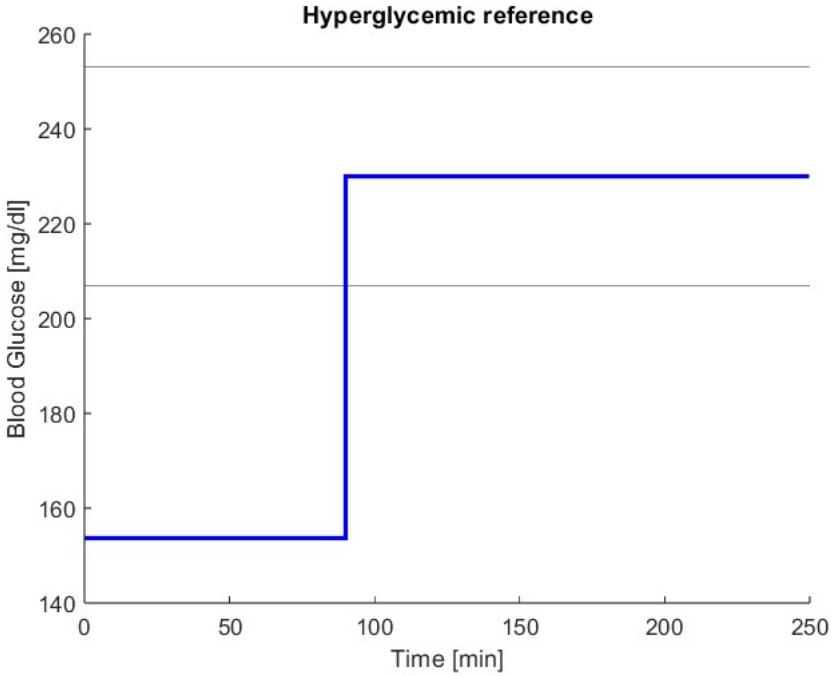


Figure 3.19: Reference profile for an hyperglycemic clamp (patient #001).

For the euglycemic clamp, the target is 100 mg/dl (Fig.3.20). Here, beyond the basal subcutaneous insulin, intravenous insulin infusion is administered corrisponding to half the basal one, which is of course patient-dependent.

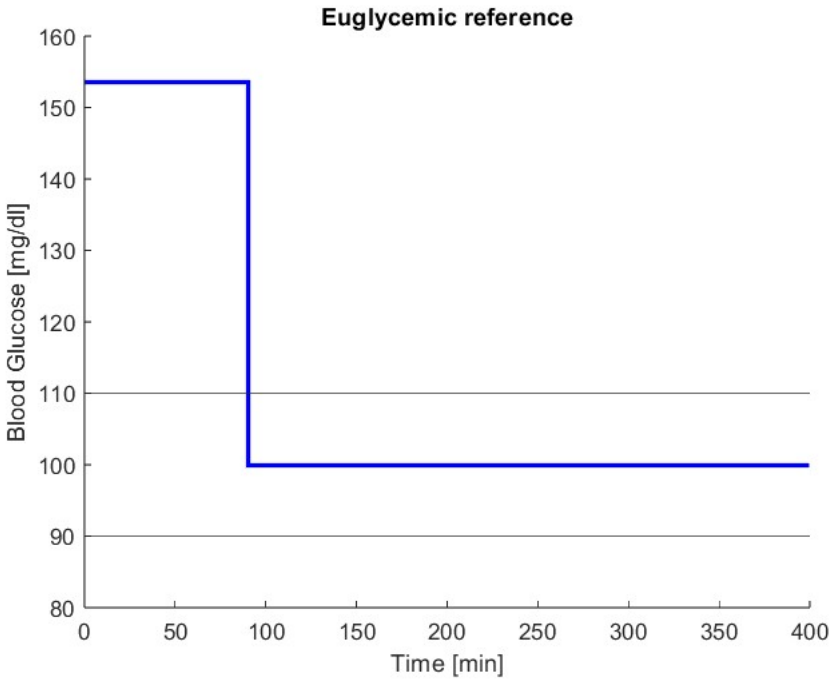


Figure 3.20: Reference profile for an euglycemic clamp (patient #001).

Lastly, for hypoglycemic clamps the target BG level is 50 mg/dl (Fig.3.21. Similarly to euglycemic clamps, here we administer both subcutaneous and intravenous insulin, the latter being 1.5 times the former. Here, the two grey lines are the values of ±20% of the BG target. This is considered to be the safety range of BG values. In all three types of glucose clamps we define T_{ref} $=[T_{step}, T_{end}]$ as the period of time that goes from the change in the reference to the end of the experiment.

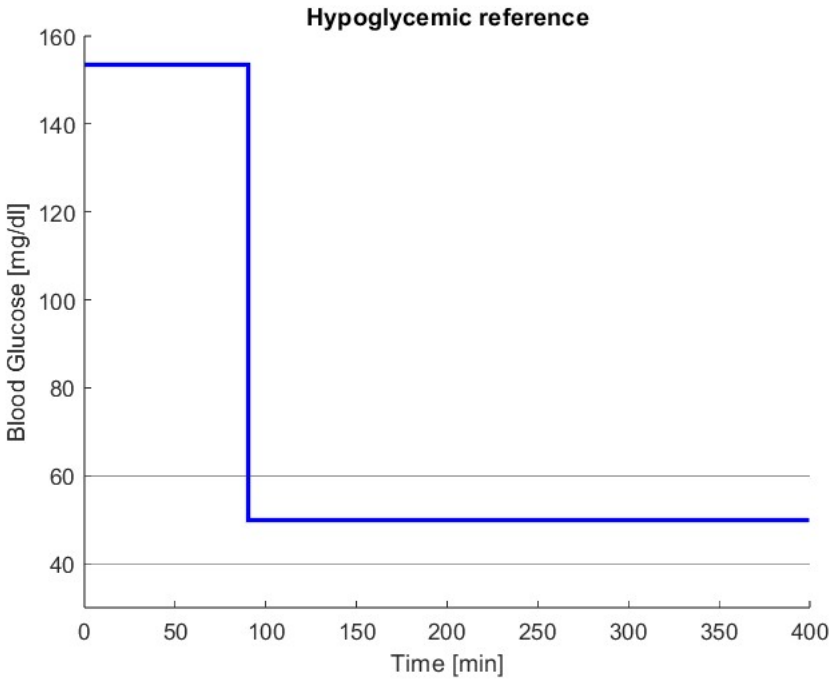


Figure 3.21: Reference profile for an Hypoglycemic clamp (patient #001).

As previously introduced, we wanted to explore the "ISO-IV" clamp. In this type of clamp we gather information about the Oral Glucose Tolerance Test (OGTT) response of the patient on the first day and on the second we try to replicate it with intravenous glucose infusion instead of carbohydrate ingestion. This helps to quantify the incretin effect and provide valuable insights into diabetes and other metabolic disorders related to mealtime glucose control. To do this we need the OGTT profile of the patient, which is obviously patient-specific. The UVa/Padova Type 1 Diabetes Simulator, as we have seen in Fig.3.2, has an input corresponding to the meal. Here we can simulate the ingestion of glucose, both 40g and 75g as previously explained. In Fig.3.22 and Fig.3.23 the response to 40g and 75g respectively are shown (notice that for the latter the BG values are higher). Here we do nothave administered any bolus of subcutaneous insulin, so the patient returns to basal range quite slowly (this simulation lasts 1200 minutes).

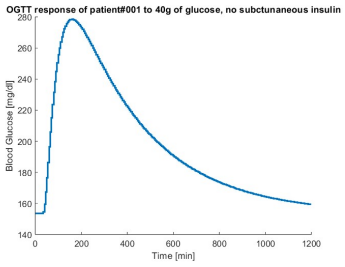


Figure 3.22: OGTT response of patient#001 at 40g of glucose, no bolus of subcutaneous insulin administered.

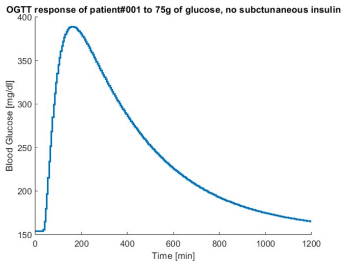


Figure 3.23: OGTT response of patient#001 at 75g of glucose, no bolus of subcutaneous insulin administered.

In Fig.3.24 and Fig.3.25 the response of a 40g OGTT is shown where a subcutaneous bolus of insulin is administered. The rate of the bolus is calculated like this, measured in U/h: $r_{bolus} = \frac{bolus}{D_{bolus}} = \frac{gt_{carbo}}{CR*D_{bolus}} = \frac{r_{carbo}*D_{carbo}}{CR*D_{bolus}}$. Since the duration of the bolus and of the assumption of carbohydrates are both set to 1 minute, we have: $r_{bolus} = \frac{r_{carbo}*60}{CR}$ [U/h]. In Fig.3.26 and Fig.3.27 we see the response to the 75g OGTT with the bolus of insulin.

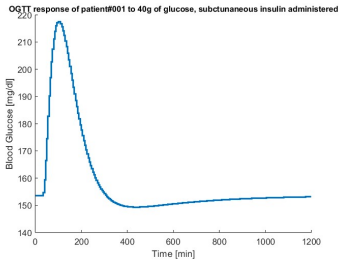


Figure 3.24: OGTT response of patient#001 at 40g of glucose, bolus of subcutaneous insulin administered.

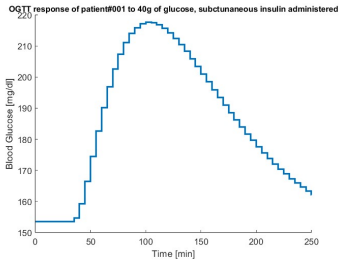


Figure 3.25: OGTT response of patient#001 at 40g of glucose, bolus of subcutaneous insulin administered, zoom.

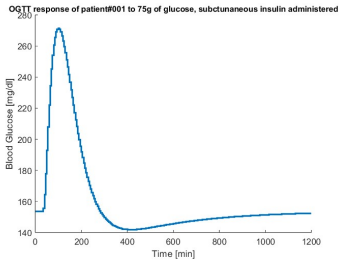


Figure 3.26: OGTT response of patient#001 at 75g of glucose, bolus of subcutaneous insulin administered.

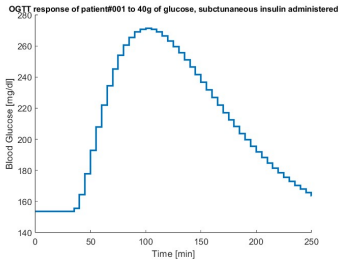


Figure 3.27: OGTT response of patient#001 at 75g of glucose, bolus of subcutaneous insulin administered, zoom.

The UVa/Padova Type 1 Diabetes Simulator is equipped with 100 virtual patients, each exhibiting a distinct response to an OGTT. These responses are illustrated in Fig.3.28 and Fig.3.29, with the mean values highlighted in bold and purple. For the ISO-IV clamp, each patient had its response from the OGTT of the day before as reference. This type of experiment lasts around 250 minutes.

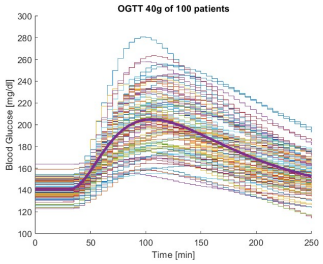


Figure 3.28: OGTT response of the 100 patients at 40g of glucose.

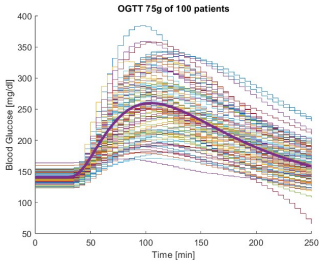


Figure 3.29: OGTT response of the 100 patients at 75g of glucose.

3.5 Performance metrics

In our analysis of the results obtained from the application of control algorithms to glucose clamp experiments, it is essential to employ appropriate performance metrics to quantitatively assess the efficacy and accuracy of the control strategies. These metrics serve as objective measures to evaluate how well the control algorithms regulate blood glucose levels and estimate physiological parameters.

One commonly used performance metric in the field of control systems and data analysis is the Mean Absolute Percentage Error (MAPE). MAPE is a measure of the accuracy of a forecasting or estimation technique and is particularly suitable for assessing the performance of control algorithms in maintaining glycemic control within target ranges.

MAPE is defined as:

$$\text{MAPE} = 100 \cdot \frac{1}{N} \sum_{t_k \in T_{\text{ref}}} \left(\left| \frac{y(t_k) - y_0(t_k)}{y_0(t_k)} \right| \right) \tag{3.1}$$

where $T_{ref} = [T_{step}, T_{end}]$, as defined in Section 3.4 (for the ISO-IV clamps the MAPE will be calculated on the entire length of the experiment). The closer the Mean Absolute Percentage Error (MAPE) is to 0%, the better the control performance. This performance metric will be calculated only for the results achieved across the entire population.

It is important to note that for certain types of clamps, the MAPE may be relatively small, while for others, such as the euglycemic and hypoglycemic clamps, it may be considerably larger. This is a consequence of the action of insulin, which takes time to lower BG levels, as we will see in Section 4. The controller's performance will be evaluated relative to each other. Comparing the MAPE between the hyperglycemic and hypoglycemic clamps would not be meaningful; instead, we will compare the MAPE of the PID in hypoglycemia with that of the MPC in hypoglycemia, for instance.

Chapter 4

Results

In this section, we present the outcomes of our simulations and analyses conducted using MATLAB and Simulink. The results are divided into two main categories: the Noise-free Scenario, where BG measurements are assumed to be without measurement error, and the Realistic Scenario, where measurement errors are considered. The selected parameters for the PID and MPC controllers are those that yielded the optimal performance for each type of glucose clamp. These parameters were chosen following automated and manual tuning processes.

4.1 Noise-free Scenario

In the Noise-free Scenario, we assume ideal conditions where BG measurements are free from errors. This scenario serves as a baseline for evaluating the performance of our control algorithms under optimal conditions.

4.1.1 PID controller

In this section we will provide results using PID controller on the different types of glucose clamps analyzed.

4.1.1.1 Hyperglycemic Clamp

As previously indicated in Section 3.4, it was established that the target blood glucose level during the hyperglycemic clamp procedure is set at 230 mg/dl. Specifically, in Fig. 4.1, the trajectory of patient#001's blood glucose levels over time is depicted under the control of the PID controller (first half of the image). The suggested GIR provided by the controller is shown in the second half of the image.

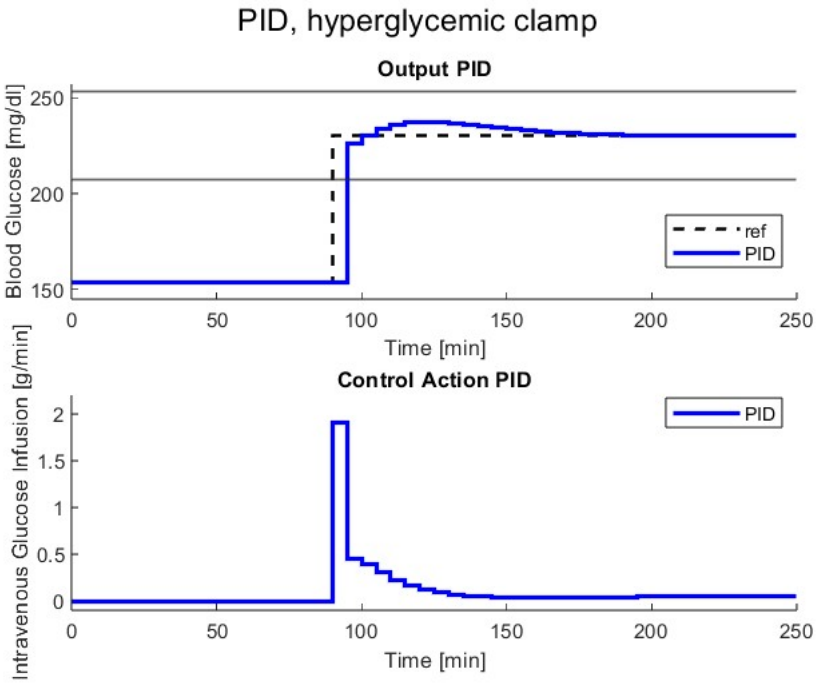


Figure 4.1: Evolution of BG level in a hyperglycemic clamp using the PID controller, patient#001.

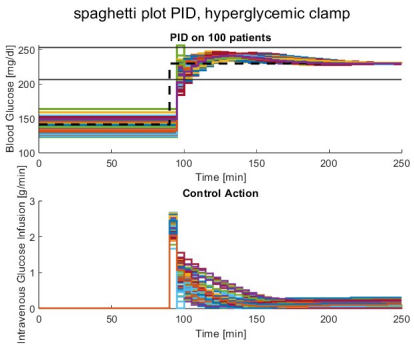


Figure 4.2: Spaghetti plot of 100 patients in a hyperglycemic clamp using PID controller.

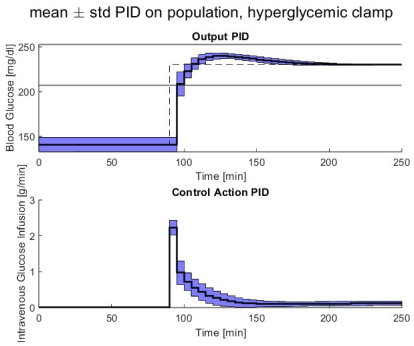


Figure 4.3: Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using PID controller.

In Fig. 4.2 we can see the "spaghetti" plot of the entire population, while in Fig.4.3 the mean output (in black) ± 1 standard deviation confidence interval

(the blue shaded area) is displayed.

PID parameters are shown in Tab.4.1, for both patient#001 and the entire population. In the case of the entire population subjected to hyperglycemic clamp and controlled using the PID controller, we obtained a MAPE value of 2.87%.

It is apparent that five minutes after the reference changes from the basal blood glucose level to the target level of 230 mg/dl, the PID controller initiates the glucose infusion rate and promptly delivers a substantial control action. This action rapidly brings the BG levels close to the target, resulting in a minor overshoot, followed by a gradual settling to 230 mg/dl.

	Patient#001	Population
K_p $[\frac{g \cdot dl}{mg \cdot min}]$	0.02	0.02
K_i $[\frac{g \cdot dl}{mg \cdot min^2}]$	0.001	0.001
K_d $[\frac{g \cdot dl}{mg}]$	0.002	0.001

Table 4.1: PID controller parameters for hyperglycemic clamp on patient#001 and on population.

4.1.1.2 Euglycemic Clamp

For the euglycemic clamp, we set the target of BG to 100 mg/dl. In Fig.4.4 the trajectory of patient#001's blood glucose levels over time is depicted under the control of the PID controller, together with the suggested glucose infusion rate. In Fig.4.5 and Fig.4.6 the spaghetti and mean ± std plot are shown. The PID controller parameters are in Table 4.2. In the case of the entire population subjected to euglycemic clamp and controlled using the PID controller, we obtained a MAPE value of 14.62%. As previously mentioned, a MAPE of 14.62% may appear high, but it is crucial to consider that during euglycemic and hypoglycemic clamps, the blood glucose levels remain above the target for a considerable duration, accumulating a significant error. This is due to the time required for insulin to lower BG levels, during which the controller does not intervene with any control action.

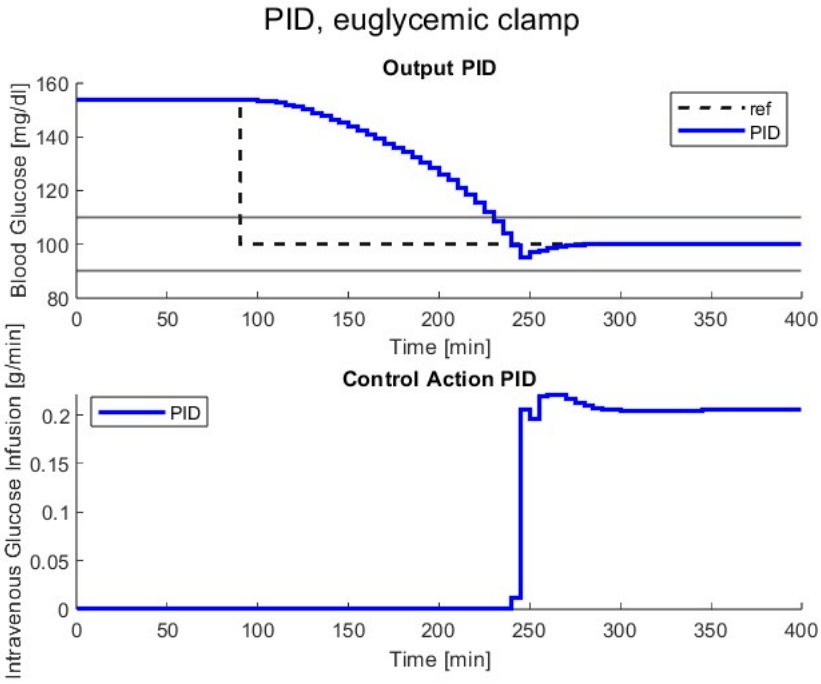


Figure 4.4: Evolution of BG level in a euglycemic clamp using the PID controller, patient#001.

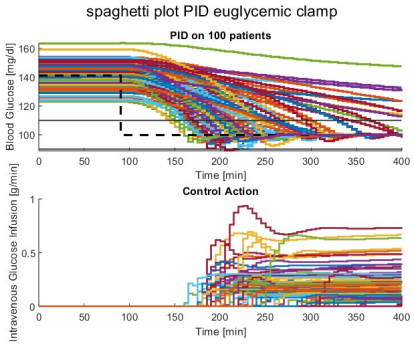


Figure 4.5: Spaghetti plot of 100 patients in a euglycemic clamp using PID controller.

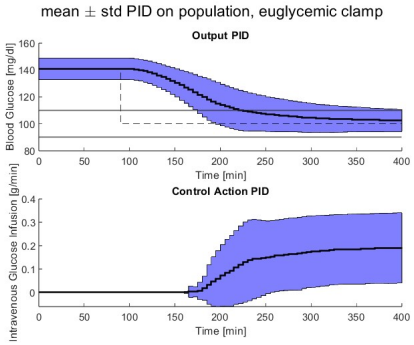


Figure 4.6: Mean ± std confidence interval of 100 patients in a euglycemic clamp using PID controller.

	Patient#001	Population
K_p $[\frac{g \cdot dl}{mg \cdot min}]$	0.025	0.025
K_i $[\frac{g \cdot dl}{mg \cdot min^2}]$	0.003	0.003
K_d $[\frac{g \cdot dl}{mg}]$	0.001	0.001

Table 4.2: PID controller parameters for euglycemic clamp on patient#001 and on population.

We can see from Fig 4.4 that the PID controller remains inactive until around minute 240/245, during which time the injected insulin causes a reduction in the blood glucose level. After reaching values close to the target (100 mg/dl), the controller starts to provide GIR and, after a small undershoot, reaches the desired BG level. It is apparent from the plots of the entire population that in this clamp, we observe higher variability both in the output of the controller (blood glucose values) and in the control action (glucose infusion rate). This variability is a consequence of the individual insulin sensitivity among different patients.

4.1.1.3 Hypoglycemic Clamp

In Fig. 4.7, we observe the evolution of the blood glucose (BG) level in patient#001 under the control of the PID controller, with the target set to 50 mg/dl. Additionally, Fig. 4.5 and Fig. 4.6 depict the spaghetti plot and the mean ± standard deviation plot, respectively. The PID controller parameters are listed in Table 4.2. For the entire population subjected to euglycemic clamp and controlled using the PID controller, we obtained a MAPE value of 45.72%.

Similarly to the euglycemic clamp, in the hypoglycemic clamp, the PID controller remains inactive until the insulin has sufficiently lowered the blood glucose levels close to the target (50 mg/dl). Subsequently, following a minor undershoot, the controller initiates glucose infusion rate to reach the desired target value. Also in this scenario, the variability is higher due to individual differences in insulin sensitivity among the patients.

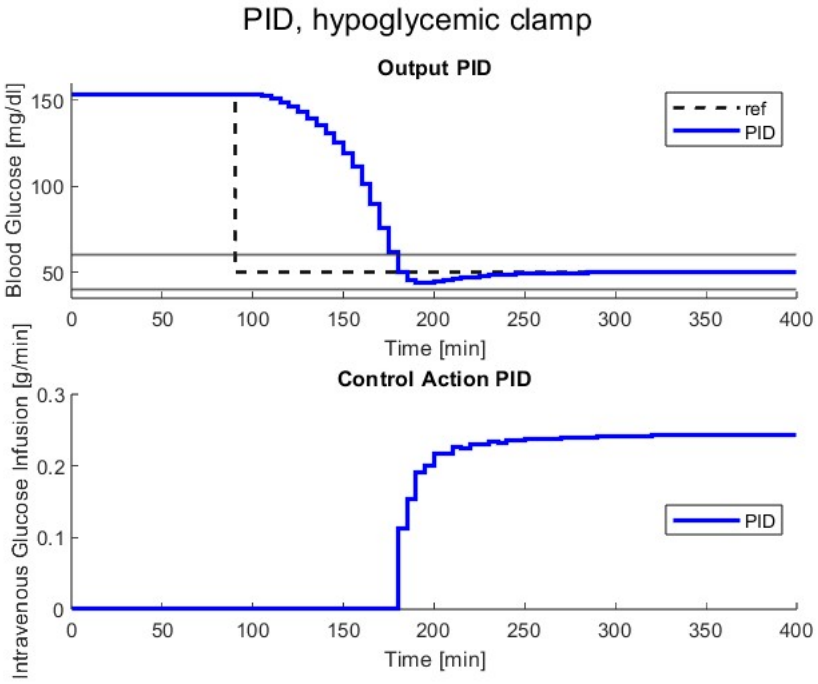


Figure 4.7: Evolution of BG level in a hypoglycemic clamp using the PID controller, patient#001.

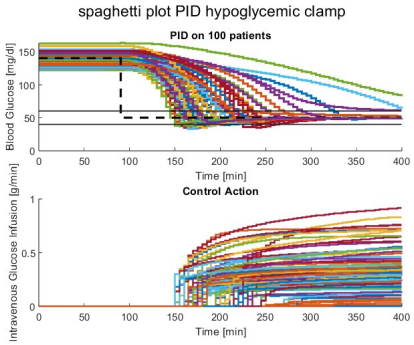


Figure 4.8: Spaghetti plot of 100 patients in a hypoglycemic clamp using PID controller.

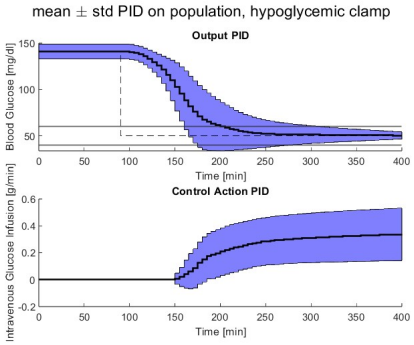


Figure 4.9: Mean ± std confidence interval of 100 patients in a hypoglycemic clamp using PID controller.

	Patient#001	Population
K_p $[\frac{g \cdot dl}{mg \cdot min}]$	0.02	0.02
K_i $[\frac{g \cdot dl}{mg \cdot min^2}]$	0.001	0.001
K_d $[\frac{g \cdot dl}{mg}]$	0.05	0.001

Table 4.3: PID controller parameters for hypoglycemic clamp on patient#001 and on population.

4.1.1.4 ISO-IV Clamp

In this paragraph we will analyze the performance of the PID controller on the ISO-IV clamps, both on OGTTs with 40g and 75g of glucose. As already stated, each patient will follow its OGTT response. In Fig.4.10, the evolution of blood glucose (BG) levels for patient#001, patient#002, and patient#003 is depicted. Since displaying the responses of all 100 patients and their variability would not convey meaningful insights due to variations in the reference throughout the entire population (as apparent in Fig.4.10), we have chosen to visualize the mean relative tracking error, as shown in Fig.4.11. This vector of values is defined as the mean of the relative tracking error between all subjects. The same rationale applies to the 75g OGTT, where the responses of the first 3 patients and the tracking error are presented in Fig.4.12 and Fig.4.13, respectively. PID parameters (for the population) can be found in Table 4.4, while the Mean Absolute Percentage Error (MAPE) in Table 4.5.

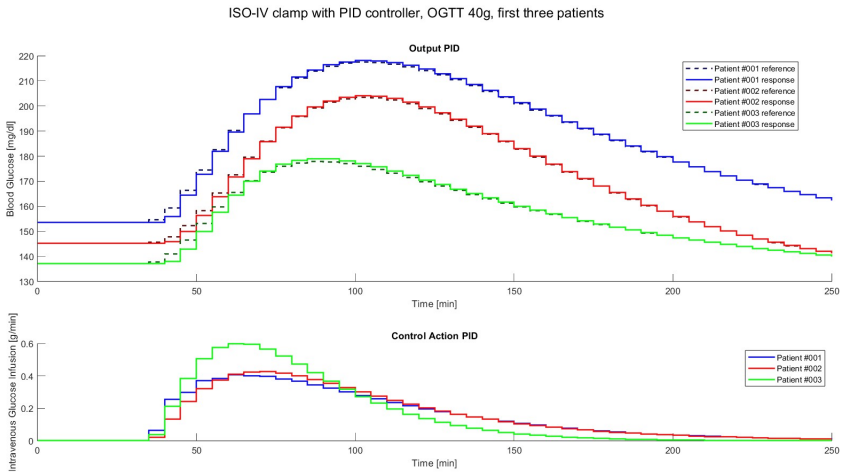


Figure 4.10: Evolution of BG level in a ISO-IV clamp (OGTT 40g) using the PID controller, first three patients.

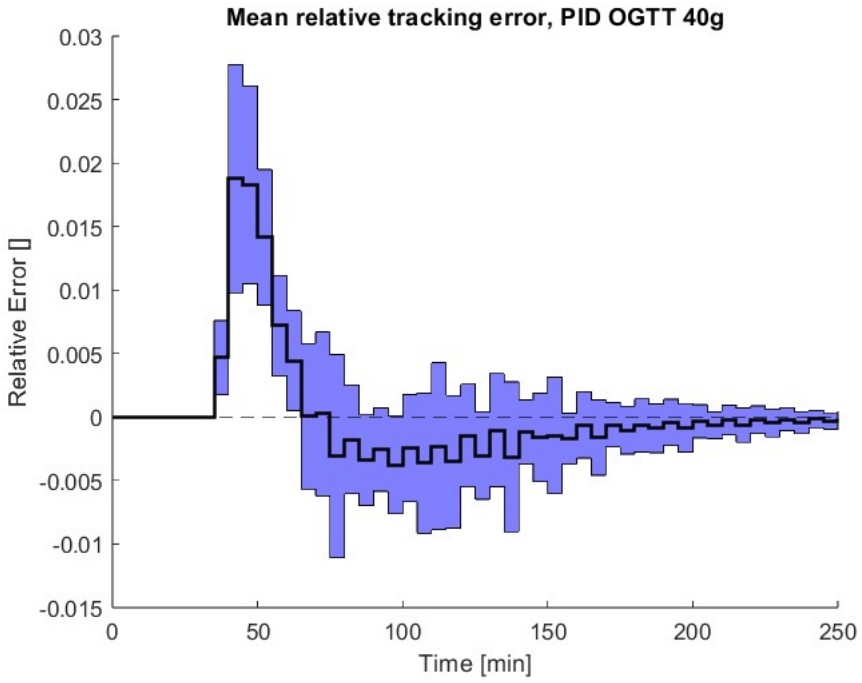


Figure 4.11: Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the PID controller on population of 100 patients.

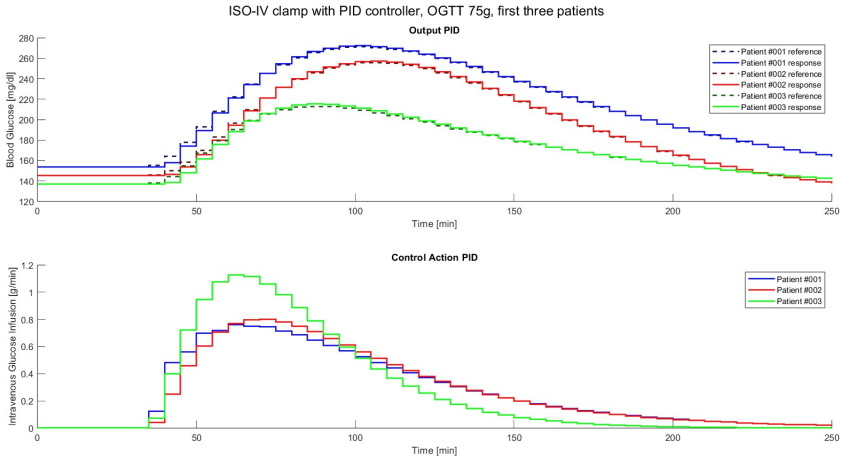


Figure 4.12: Evolution of BG level in a ISO-IV clamp (OGTT 75g) using the PID controller, first three patients.

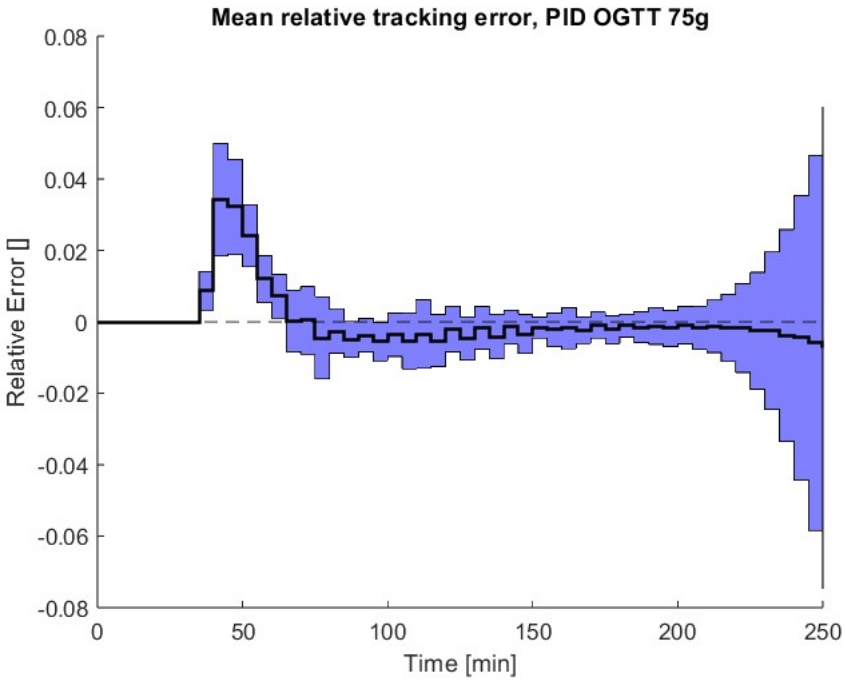


Figure 4.13: Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the PID controller on population of 100 patients.

	OGTT 40g	OGTT 75g
K_p $[\frac{g \cdot dl}{mg \cdot min}]$	0.02	0.02
K_i $[\frac{g \cdot dl}{mg \cdot min^2}]$	0.0085	0.0085
K_d $[\frac{g \cdot dl}{mg}]$	0.005	0.005

Table 4.4: PID controller parameters for ISO-IV clamp on population.

	OGTT 40g	OGTT 75g
MAPE (%)	0.29	0.51

Table 4.5: Mean Absolute Percentage Error (MAPE) of PID controller in a ISO-IV clamp, response to OGTT 40g and OGTT 75g.

We can observe that the controller effectively tracks the reference very well in both the 40g and 75g scenarios. Additionally, we notice that the higher relative tracking error is typically observed in the initial phase of the experiment, where the PID controller exhibits slight delay in tracking the reference due to its lack

of ability to look ahead into the future. We can observe that in Fig.4.13 the variability of the mean relative tracking error is a little bit high at the end of the experiment, this is probably due to the fact that with the 75g OGTT we reach higher values of BG. Nevertheless, despite the observed variability, the tracking error remains consistently small, indicating excellent performance achieved using this simple controller.

4.1.2 MPC Controller with Full Increment Velocity Form

In this section we will provide results using the MPC controller with Full Increment Velocity Form on the different types of glucose clamps analyzed. Since the procedures for our glucose clamps remain consistent with those described in Sections 3.4 and 4.1.1, only images and results will be presented in the upcoming sections concerning MPC controllers.

4.1.2.1 Hyperglycemic Clamp

In Fig.4.14, the performance of the MPC controller with Full Increment Velocity Form is depicted, both with and without the look ahead feature. It is apparent that the former, with look ahead activated, can anticipate future changes in the reference, allowing for earlier control action and achieving the target value before the latter. This is confirmed in Fig.4.15 to Fig.4.18 representing the MPC controller results on the population in a hyperglycemic clamp. In Table 4.6 the MPC controller parameters are shown, while in Table 4.7 the Mean Absolute Percentage Error (MAPE) is displayed.

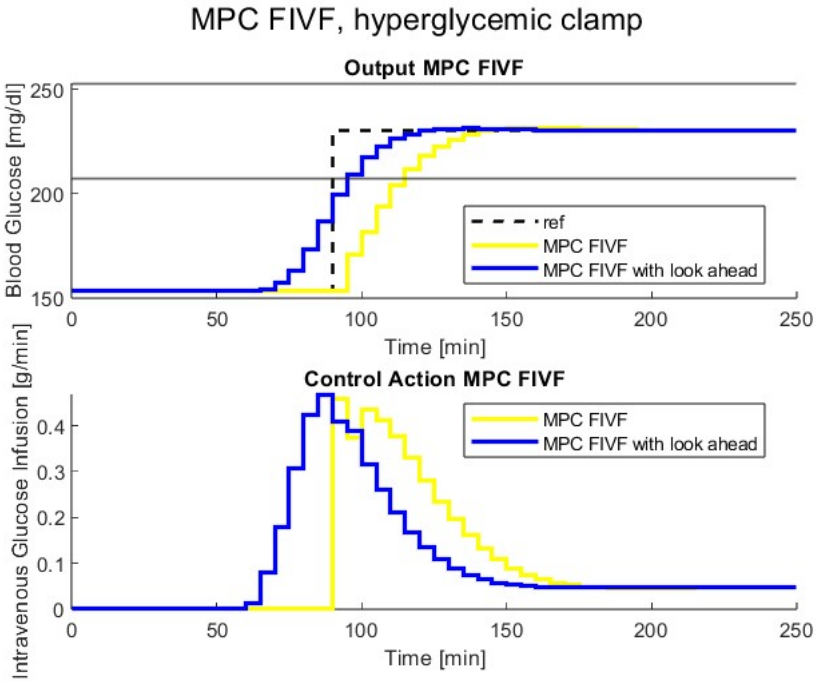


Figure 4.14: Evolution of BG level in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form, patient#001.

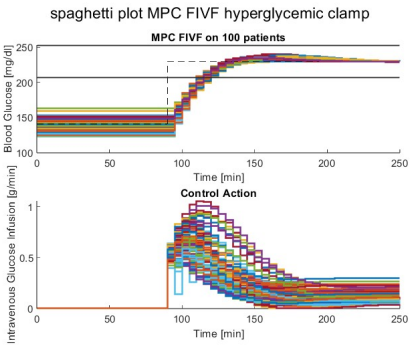


Figure 4.15: Spaghetti plot of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form (no look ahead implemented).

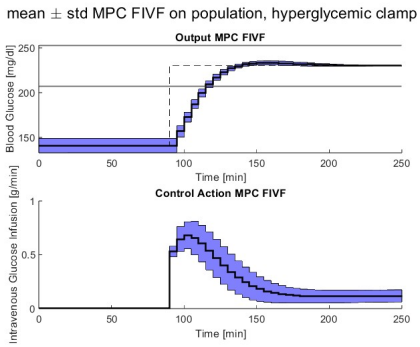


Figure 4.16: Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form (no look ahead implemented).

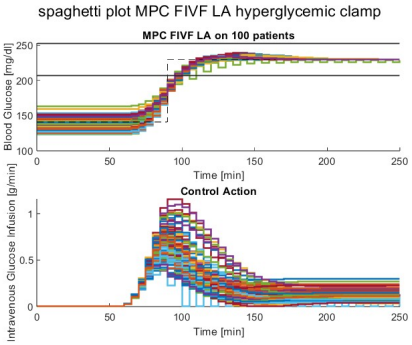


Figure 4.17: Spaghetti plot of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated).

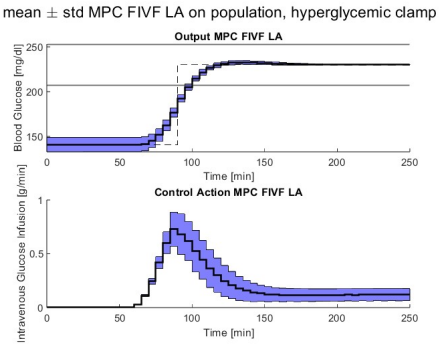


Figure 4.18: Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated).

	Patient#001, FIVF	Patient#001, FIVF LA	Population, FIVF	Population, FIVF LA
PH	15	15	10	15
Q	0.01	0.01	0.01	0.01
R	150	50	150	60

Table 4.6: MPC with Full Increment Velocity Form parameters in a hyperglycemic clamp. With and without look ahead feature, on patient#001 and on population. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness.

	FIVF	FIVF LA
MAPE (%)	4.71	2.21

Table 4.7: Mean Absolute Percentage Error (MAPE) of MPC controller with Full Increment Velocity Form in a hyperglycemic clamp, with and without look ahead feature.

4.1.2.2 Euglycemic Clamp

From Fig.4.19, it is apparent that in the euglycemic clamp, the action of the look ahead feature does not offer any improvements. In fact, the outputs of the controller with and without look ahead are practically identical. This is a consequence of the administration of insulin, which lowers blood glucose levels. As previously mentioned, the controller does not administer any glucose infusion rate until the BG levels almost reach the target. At this point, the look ahead feature does not provide any improvements because both controllers already know that the reference is 100 mg/dl, so their actions are essentially the same. The same considerations can be made on the results on the population of 100 patients (Fig.4.20 to Fig.4.23). In Table 4.8 the MPC controller parameters are shown, while in Table 4.9 the Mean Absolute Percentage Error (MAPE) is displayed. As one could imagine, they are identical.

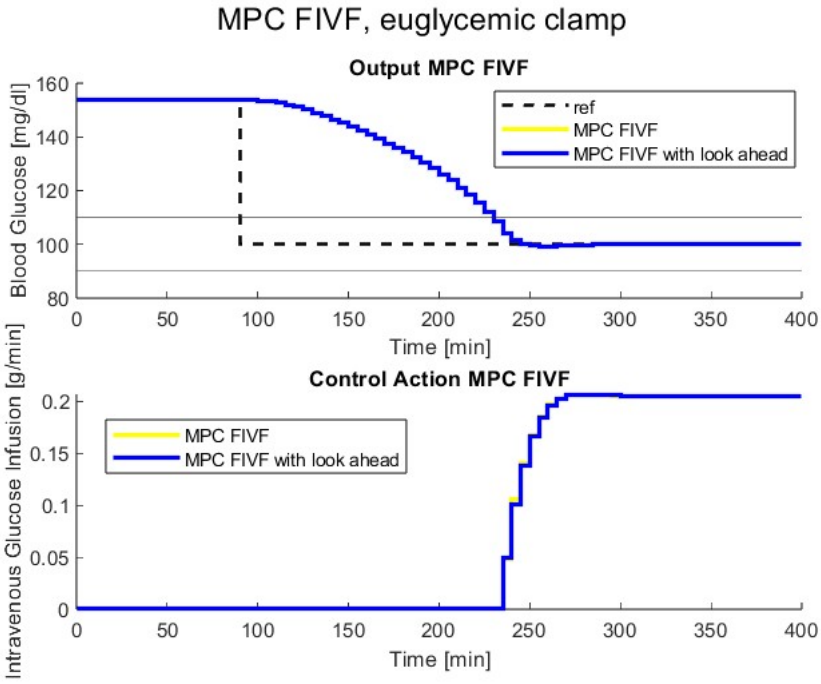


Figure 4.19: Evolution of BG level in a euglycemic clamp using the MPC controller with Full Increment Velocity Form, patient#001.

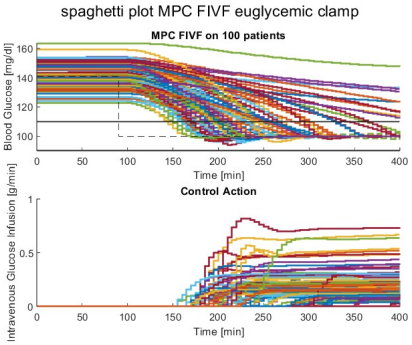


Figure 4.20: Spaghetti plot of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form (no look ahead implemented).

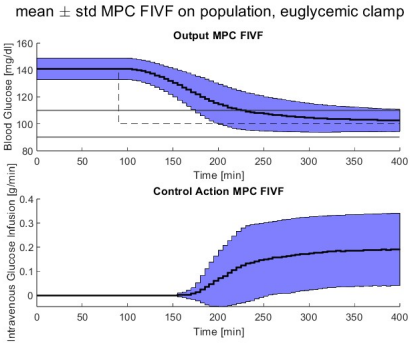


Figure 4.21: Mean ± std confidence interval of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form (no look ahead implemented).

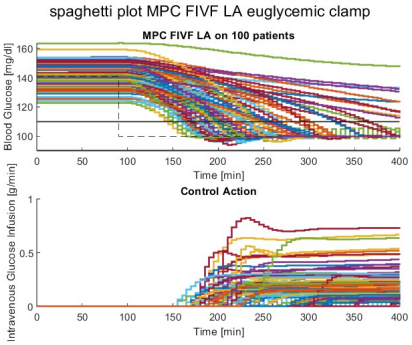


Figure 4.22: Spaghetti plot of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated).

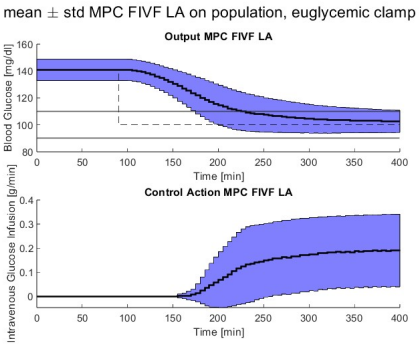


Figure 4.23: Mean ± std confidence interval of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated).

	Patient#001, FIVF	Patient#001, FIVF LA	Population, FIVF	Population, FIVF LA
PH	15	15	15	15
Q	0.01	0.01	0.01	0.01
R	10	10	10	10

Table 4.8: MPC with Full Increment Velocity Form parameters in a euglycemic clamp. With and without look ahead feature, on patient#001 and on population. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness.

	FIVF	FIVF LA
MAPE (%)	14.51	14.51

Table 4.9: Mean Absolute Percentage Error (MAPE) of MPC controller with Full Increment Velocity Form in a euglycemic clamp, with and without look ahead feature.

4.1.2.3 Hypoglycemic Clamp

The considerations made for the euglycemic clamp can be made also for the hypoglycemic clamp. As we can see in Fig.4.24 the results with and without look ahead are very similar (if not almost identical). Because of this we will show only the results on the population given by the MPC controller with look ahead (Fig.4.25 and Fig.4.26). MPC parameters are shown in Table4.10. The MAPE found is 44.33%.

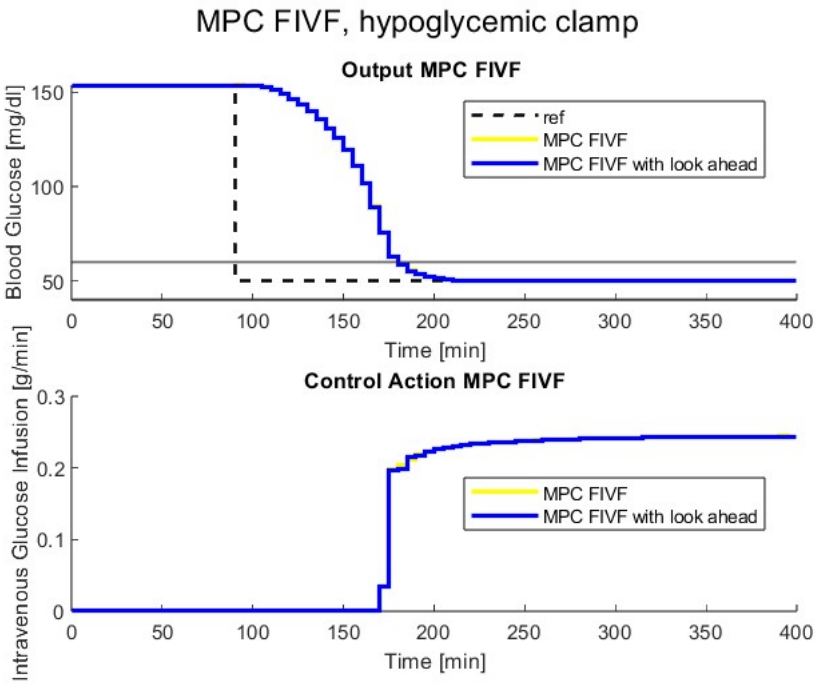


Figure 4.24: Evolution of BG level in a hypoglycemic clamp using the MPC controller with Full Increment Velocity Form, patient#001.

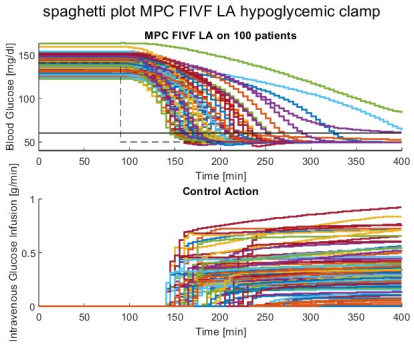


Figure 4.25: Spaghetti plot of 100 patients in a hypoglycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated).

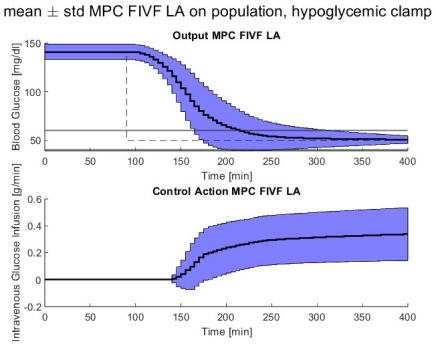


Figure 4.26: Mean ± std confidence interval of 100 patients in a hypolycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated).

	patient#001	Population
PH	15	15
Q	0.01	0.01
R	20	50

Table 4.10: MPC with Full Increment Velocity Form parameters in a hypoglycemic clamp. With look ahead feature, on patient#001 and on population. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness.

4.1.2.4 ISO-IV Clamp

As done for the PID controller, we will also analyze the performance of the MPC controller on the ISO-IV clamp. Given the effectiveness of the look ahead feature of the MPC controller, we will exclusively present the results achieved with it.

The evolution of blood glucose (BG) levels for the first three patients using an MPC controller with Full Increment Velocity Form for 40g and 75g OGTTs are depicted in Fig.4.27 and Fig.4.29 respectively. The mean relative tracking error for the two experiments is displayed in Fig.4.28 and Fig.4.30 .

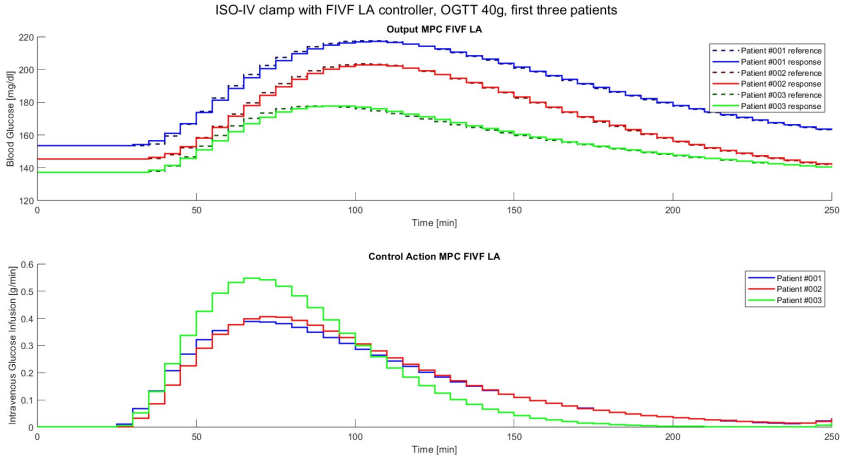


Figure 4.27: Evolution of BG levels in a ISO-IV clamp (OGTT 40g) using the MPC controller with Full Increment Velocity Form, first three patients.

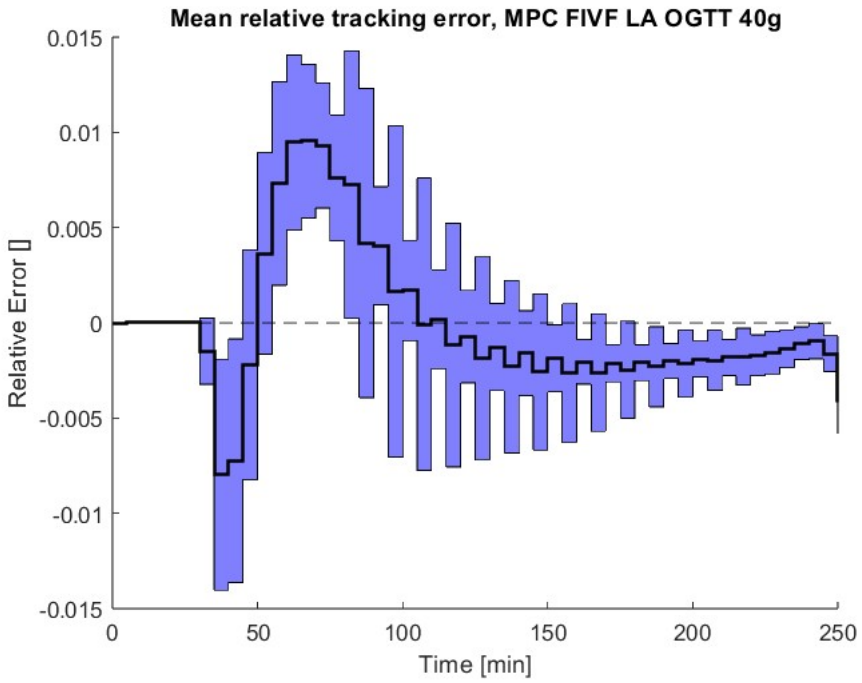


Figure 4.28: Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the MPC controller with Full Increment Velocity Form on population of 100 patients.

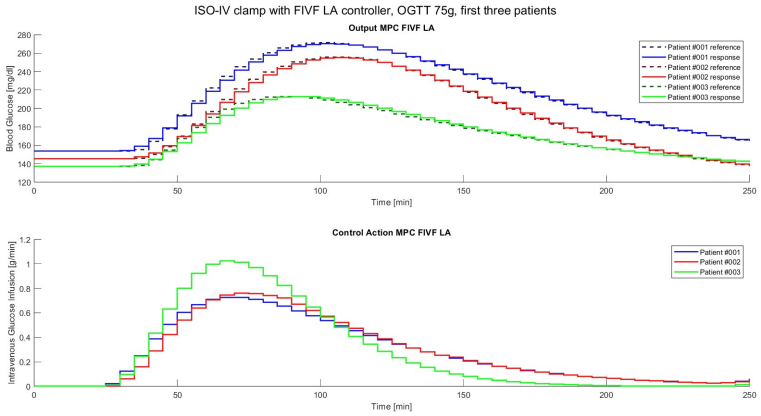


Figure 4.29: Evolution of BG levels in a ISO-IV clamp (OGTT 75g) using the MPC controller with Full Increment Velocity Form, first three patients.

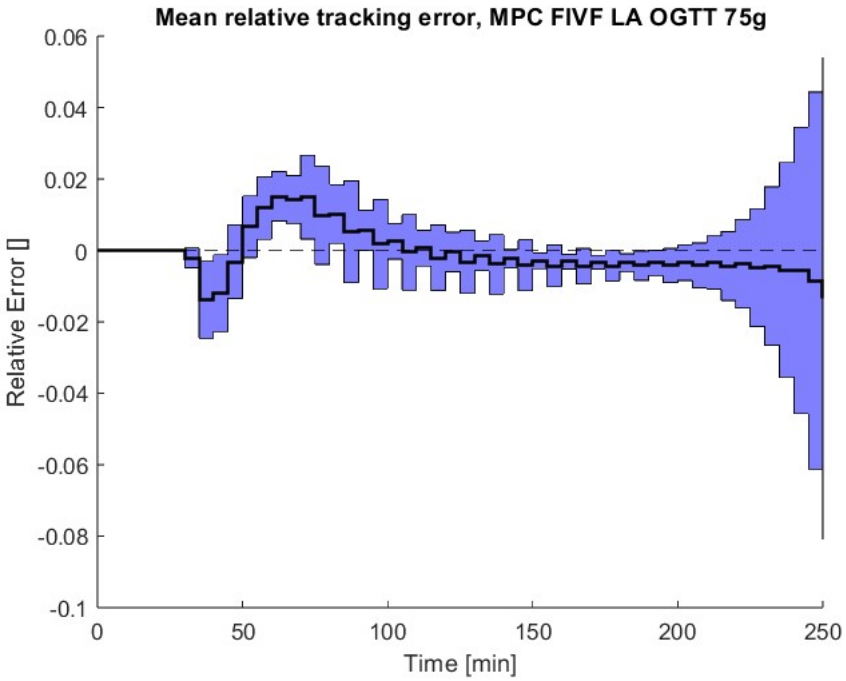


Figure 4.30: Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the MPC controller with Full Increment Velocity Form on population of 100 patients.

	OGTT 40g	OGTT 75g
PH	15	15
Q	0.01	0.01
R	5	5

Table 4.11: MPC with Full Increment Velocity Form parameters in a ISO-IV clamp, on OGTT 40g and OGTT 75g. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness.

	OGTT 40g	OGTT 75g
MAPE (%)	0.31	0.55

Table 4.12: Mean Absolute Percentage Error (MAPE) of MPC controller with Full Increment Velocity in a ISO-IV clamp.

It is apparent from the spaghetti plot of the three patients (and confirmed by the plot of the error) that, both in the 40g and 75g cases, the MPC tends to anticipate the reference during the ascending phase and near the peak tends to stay below it. Nevertheless, we have achieved very good performance.

4.1.3 MPC Controller with the Unmeasured Disturbance Estimator Approach

In this section we will provide results using the MPC controller with the Unmeasured Disturbance Estimator approach on the different types of glucose clamps analyzed. Since the procedures for our glucose clamps remain consistent with those described in Sections 3.4 and 4.1.1, only images and results will be presented in the upcoming sections concerning MPC controllers. Moreover, we will show only MPC controllers with the look ahead feature.

4.1.3.1 Hyperglycemic Clamp

In Fig.4.31 we can see how the MPC controller with the Unmeasured Disturbance Estimator technique performs on the hyperglycemic clamp, while in Fig.4.32 and 4.33 it is behaviour on the entire population is shown. We clearly see a greater promptness in the controller output compared with the Full Increment Velocity Form one. This is confirmed by the MAPE, which for this controller is of 2.00%. The controller parameters can be found in Table 4.13. As already stated, the initial estimation of the augmented state and of the covariance matrix of the estimation error were set to 0 and to the identity matrix, respectively.

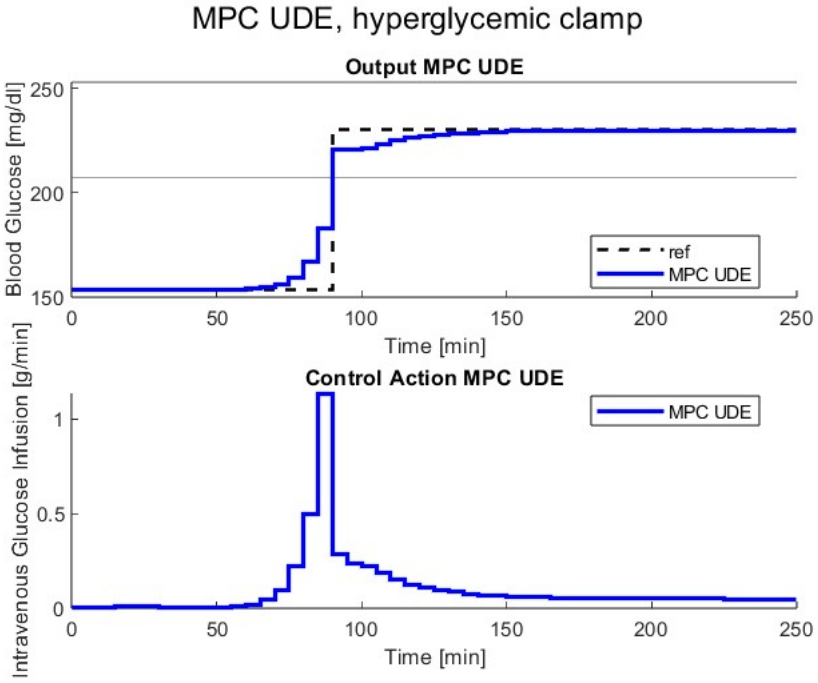


Figure 4.31: Evolution of BG level in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, patient#001.

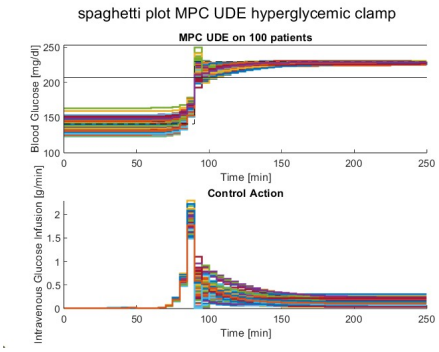


Figure 4.32: Spaghetti plot of 100 patients in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method.

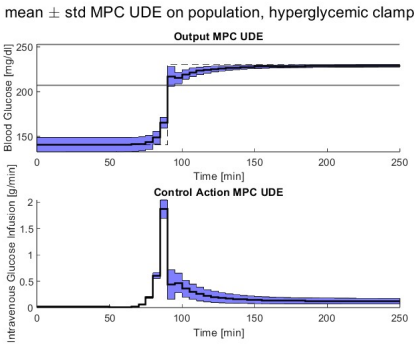


Figure 4.33: Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method.

	patient#001	Population
PH	15	15
Q	0.01	0.01
R	8	4
M_u	10	40
C_u	5	5
Q_k	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$
R_k	1	1

Table 4.13: Parameters of the MPC controller with the Unmeasured Disturbance Estimator method, hyperglycemic clamp. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively.

4.1.3.2 Euglycemic Clamp

In Fig.4.34 the response of patient#001 in a euglycemic clamp is displayed, while in Figures 4.35 and 4.36 the results on the population. The controller parameters are in Table 4.14. The MAPE for this experiment is of 14.59%.

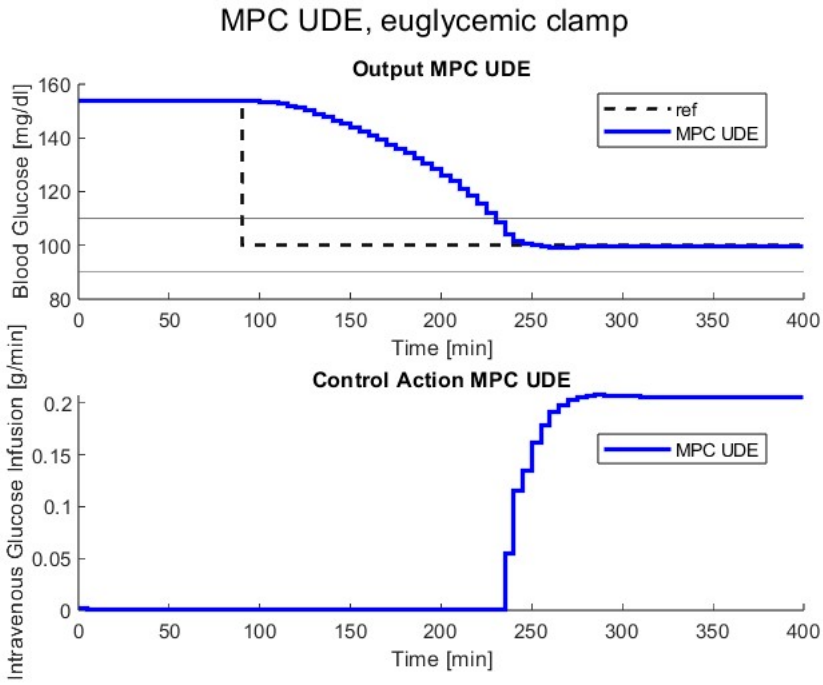


Figure 4.34: Evolution of BG level in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, patient#001.

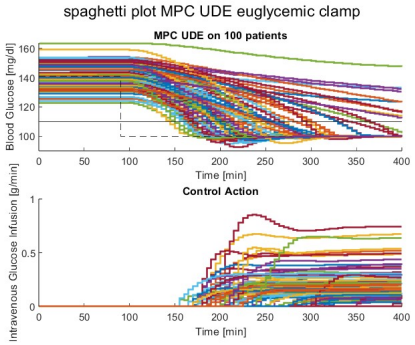


Figure 4.35: Spaghetti plot of 100 patients in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method.

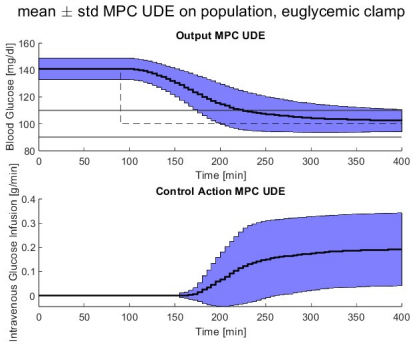


Figure 4.36: Mean ± std confidence interval of 100 patients in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method.

	patient#001	Population
PH	15	15
Q	0.01	0.01
R	10	10
M_u	10000	10000
C_u	100	100
Q_k	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$
R_k	1	1

Table 4.14: Parameters of the MPC controller with the Unmeasured Disturbance Estimator method, euglycemic clamp. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively.

4.1.3.3 HypoGlycemic clamp

In Fig.4.37 the response of patient#001 in a hypoglycemic clamp is displayed, while in Figures 4.38 and 4.39 the results on the population. The controller parameters are in Table 4.15. The MAPE for this experiment is of 44.59%.

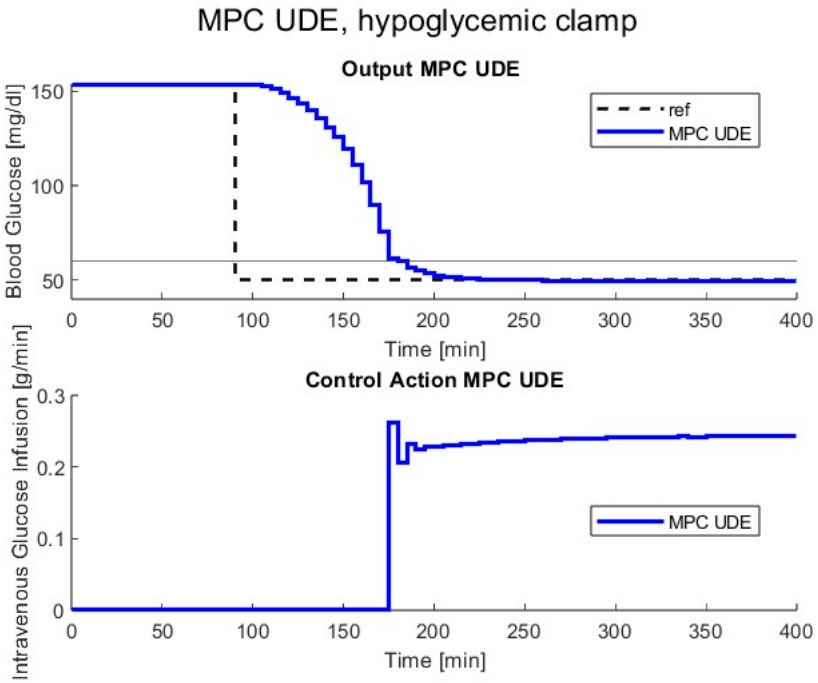


Figure 4.37: Evolution of BG level in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, patient#001.

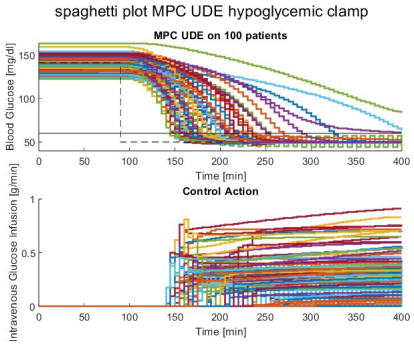


Figure 4.38: Spaghetti plot of 100 patients in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method.

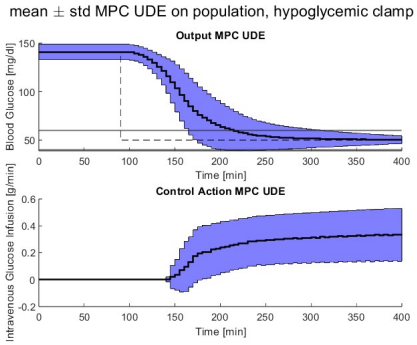


Figure 4.39: Mean ± std confidence interval of 100 patients in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method.

	patient#001	Population
PH	15	15
Q	0.01	0.01
R	10	10
M_u	10000	10000
C_u	100	100
Q_k	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$
R_k	1	1

Table 4.15: Parameters of the MPC controller with the Unmeasured Disturbance Estimator method, hypoglycemic clamp. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively.

4.1.3.4 ISO-IV clamp

As done for the other controllers, we will analyze the performance of the Unmeasured Disturbance Estimator approach on the ISO-IV clamp. The evolution of blood glucose (BG) levels for the first three patients using an MPC controller with Full Increment Velocity Form for 40g and 75g OGTTs are depicted in Fig.4.40 and Fig.4.42 respectively. The mean relative tracking error for the two experiments is displayed in Fig.4.41 and Fig.4.43

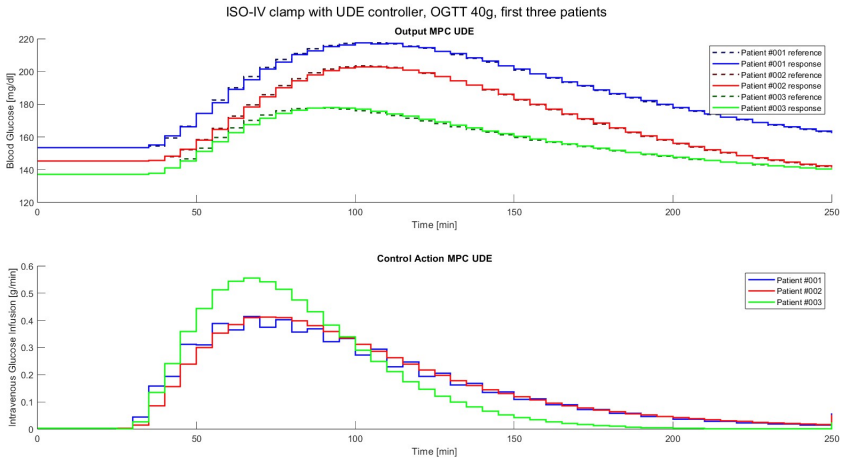


Figure 4.40: Evolution of BG level in a ISO-IV clamp (OGTT 40g) using the MPC controller with Unmeasured Disturbance Estimator approach, first three patients.

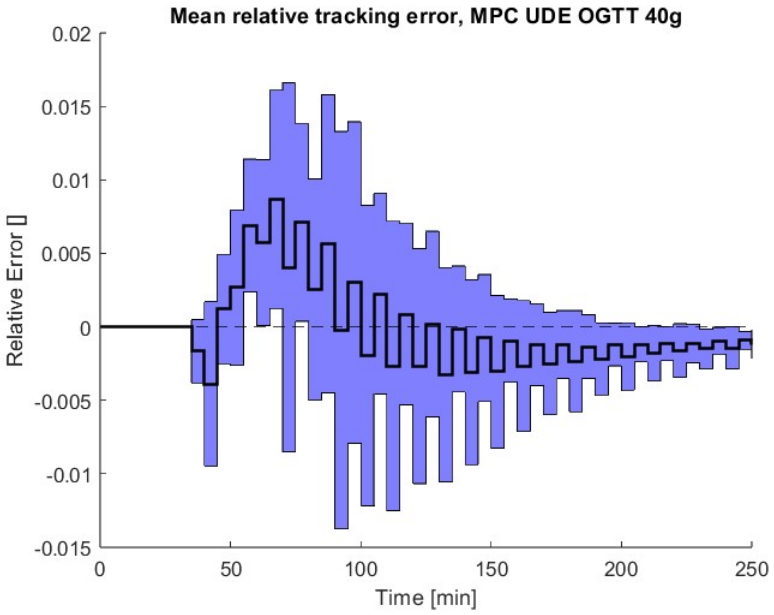


Figure 4.41: Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the MPC controller with Unmeasured Disturbance Estimator approach on population of 100 patients.

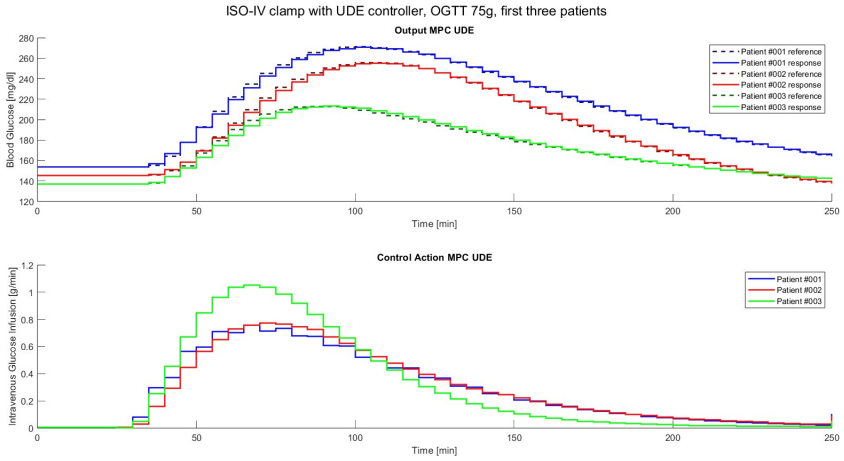


Figure 4.42: Evolution of BG level in a ISO-IV clamp (OGTT 75g) using the MPC controller with Unmeasured Disturbance Estimator approach, first three patients.

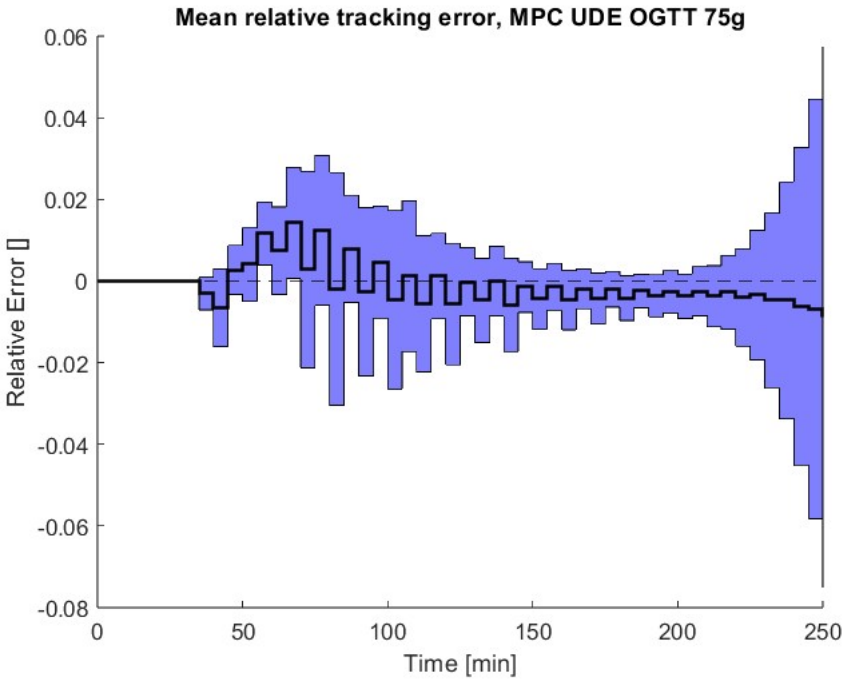


Figure 4.43: Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the MPC controller with Unmeasured Disturbance Estimator approach on population of 100 patients.

	OGTT 40g	OGTT 75g
PH	15	10
Q	1000	1000
R	1	1
M_u	3000	3000
C_u	-5	-5
Q_k	$\begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$
R_k	0.5	1

Table 4.16: Parameters of the MPC controller with the Unmeasured Disturbance Estimator method, ISO-IV clamp. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively.

	OGTT 40g	OGTT 75g
MAPE (%)	0.26	0.51

Table 4.17: Mean Absolute Percentage Error (MAPE) of MPC controller with Unmeasured Disturbance Estimator method in a ISO-IV clamp.

4.2 Realistic Scenario

In the Realistic Scenario, we introduce measurement errors to simulate conditions more akin to real-world scenarios. These measurement errors, which can arise from sensor inaccuracies, noise, and other sources of variability inherent in clinical settings, have a significant impact on the performance of control algorithms in regulating glucose levels.

By incorporating measurement errors into our simulations, we aim to evaluate the robustness and resilience of the control algorithms under conditions that more closely resemble clinical practice. This allows us to assess how well the control strategies perform in the presence of uncertainties and disturbances, providing insights into their real-world applicability and effectiveness.

The measurement noise is modeled as zero-mean, Gaussian white noise with a constant coefficient of variation (CV). The measured output of the system is given by:

$$y(k) = \mathrm{BG}(k) + w(k) \tag{4.1}$$

where $w(k) \sim \mathcal{N}(0, \sigma^2(k))$, and $\sigma(k) = \mathrm{CV} \cdot \mathrm{BG}(k)$. For our simulation, we set the CV equal to 2%, representing the accuracy of a YSI Glucose/Lactate Analyzer, a commonly used system for blood glucose measurements in glucose clamp experiments.

4.2.1 PID controller

4.2.1.1 Hyperglycemic Clamp

The Mean Absolute Percentage Error (MAPE)for the hyperglycemic clamp using the PID controller in presence of measurement noise is of 3.46%.

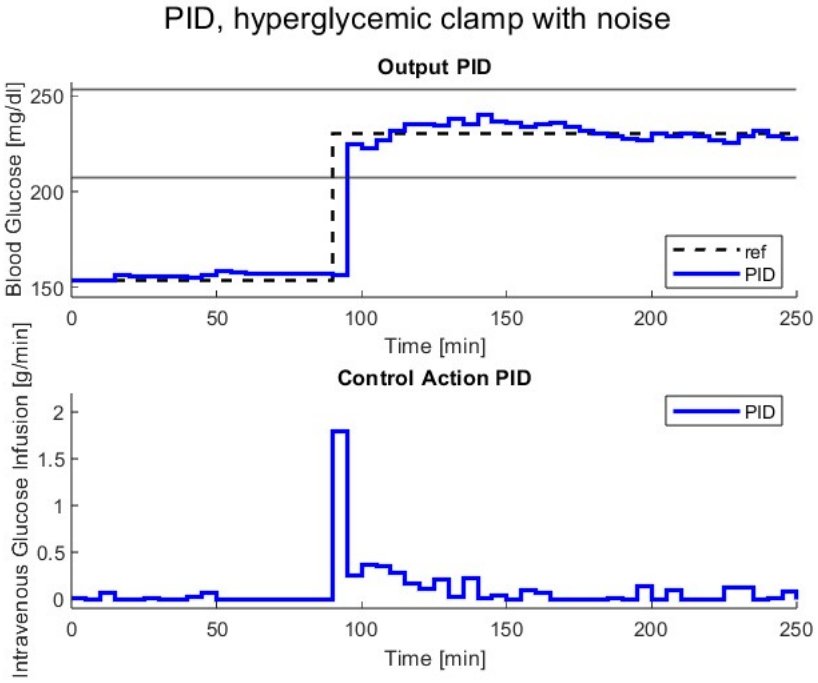


Figure 4.44: Evolution of BG level in a hyperglycemic clamp using the PID controller on patient#001, measurement noise present.

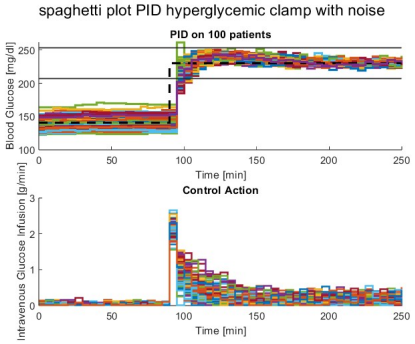


Figure 4.45: Spaghetti plot of 100 patients in a hyperglycemic clamp using PID controller, measurement noise present.

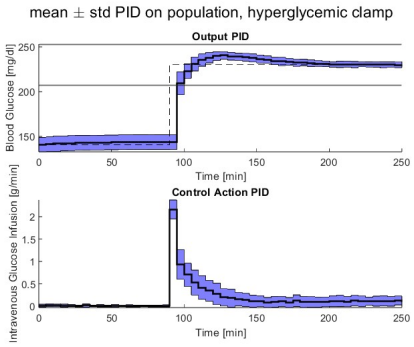


Figure 4.46: Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the PID controller, measurement noise present.

	Patient#001	Population
K_p $[\frac{g \cdot dl}{mg \cdot min}]$	0.02	0.02
K_i $[\frac{g \cdot dl}{mg \cdot min^2}]$	0.001	0.001
K_d $[\frac{g \cdot dl}{mg}]$	0.002	0.001

Table 4.18: PID controller parameters for hyperglycemic clamp on patient#001 and on population, measurement noise present.

4.2.1.2 Euglycemic Clamp

The Mean Absolute Percentage Error (MAPE) for the euglycemic clamp using the PID controller in presence of measurement noise is of 16.85%.

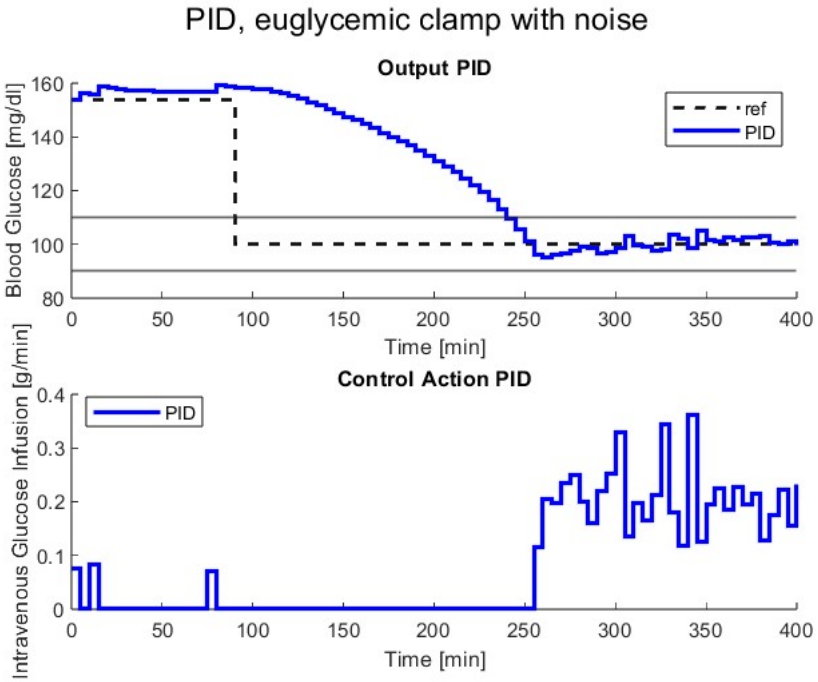


Figure 4.47: Evolution of BG level in a euglycemic clamp using the PID controller on patient#001, measurement noise present.

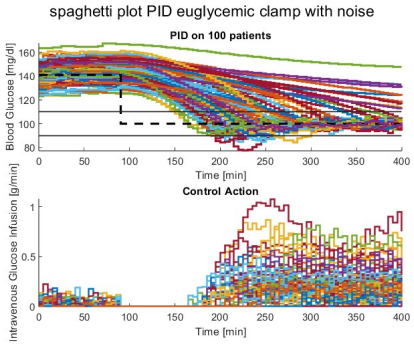


Figure 4.48: Spaghetti plot of 100 patients in a euglycemic clamp using PID controller, measurement noise present.

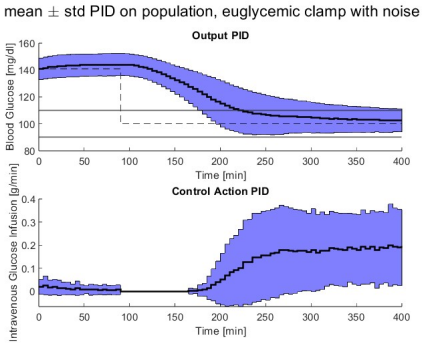


Figure 4.49: Mean ± std confidence interval of 100 patients in a euglycemic clamp using PID controller, measurement noise present.

	Patient#001	Population
K_p $[\frac{g \cdot dl}{mg \cdot min}]$	0.02	0.025
K_i $[\frac{g \cdot dl}{mg \cdot min^2}]$	0.001	0.003
K_d $[\frac{g \cdot dl}{mg}]$	0.001	0.001

Table 4.19: PID controller parameters for euglycemic clamp on patient#001 and on population, measurement noise present.

4.2.1.3 Hypoglycemic Clamp

The Mean Absolute Percentage Error (MAPE) for the hypoglycemic clamp using the PID controller in presence of measurement noise is of 48.22%.

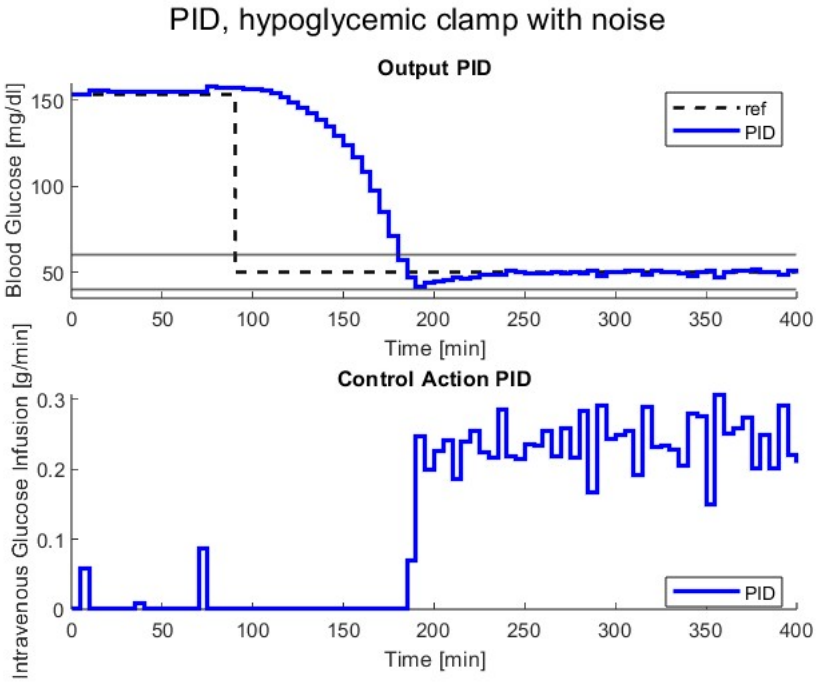


Figure 4.50: Evolution of BG level in a hypoglycemic clamp using the PID controller on patient#001, measurement noise present.

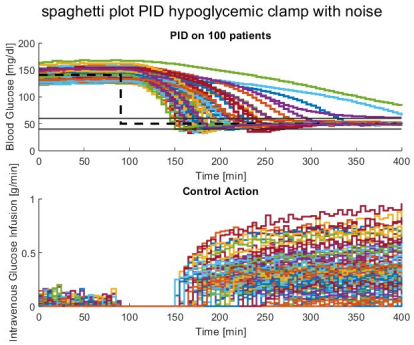


Figure 4.51: Spaghetti plot of 100 patients in a hypoglycemic clamp using PID controller, measurement noise present.

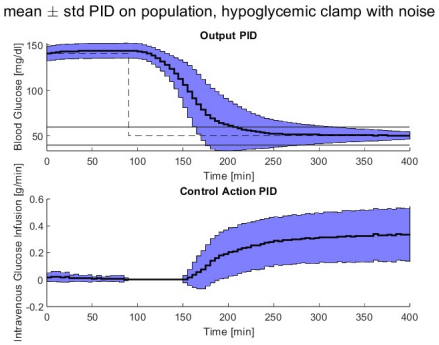


Figure 4.52: Mean ± std confidence interval of 100 patients in a hypoglycemic clamp using PID controller, measurement noise present.

	Patient#001	Population
K_p $[\frac{g \cdot dl}{mg \cdot min}]$	0.02	0.02
K_i $[\frac{g \cdot dl}{mg \cdot min^2}]$	0.001	0.001
K_d $[\frac{g \cdot dl}{mg}]$	0.001	0.001

Table 4.20: PID controller parameters for hypoglycemic clamp on patient#001 and on population, measurement noise present.

4.2.1.4 ISO-IV Clamp

The MAPE for the ISO-IV clamp can be found in Table 4.22.

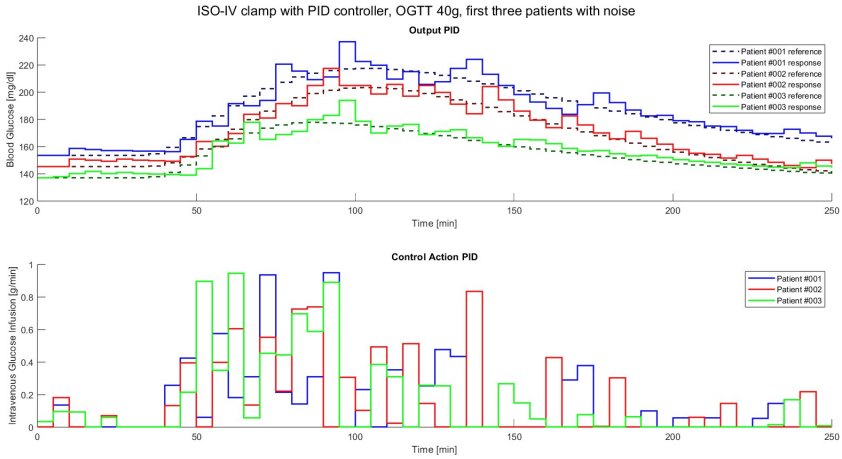


Figure 4.53: Evolution of BG level in a ISO-IV clamp (OGTT 40g) using the PID controller on first three patients, measurement noise present.

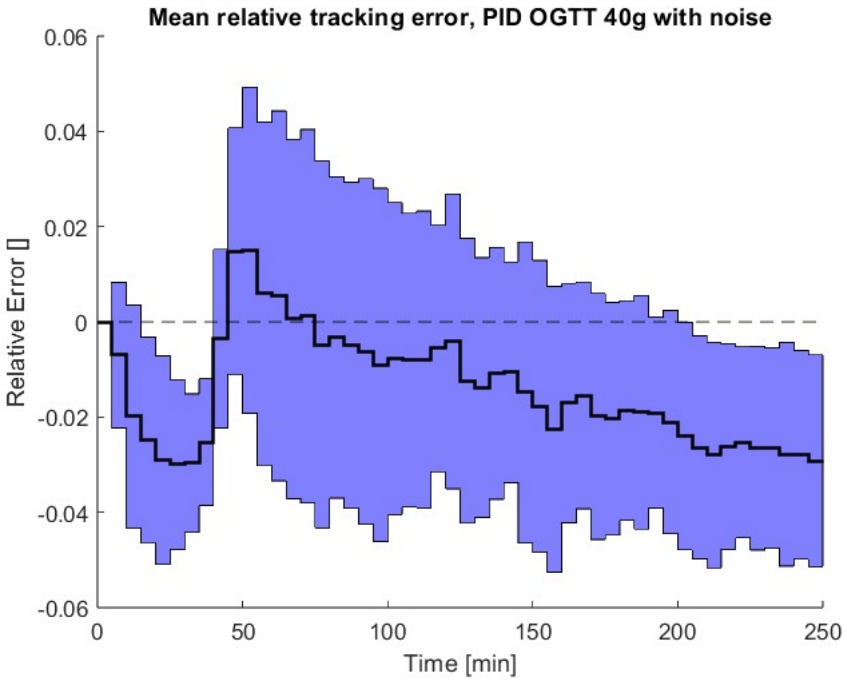


Figure 4.54: Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the PID controller on population of 100 patients, measurement noise present.

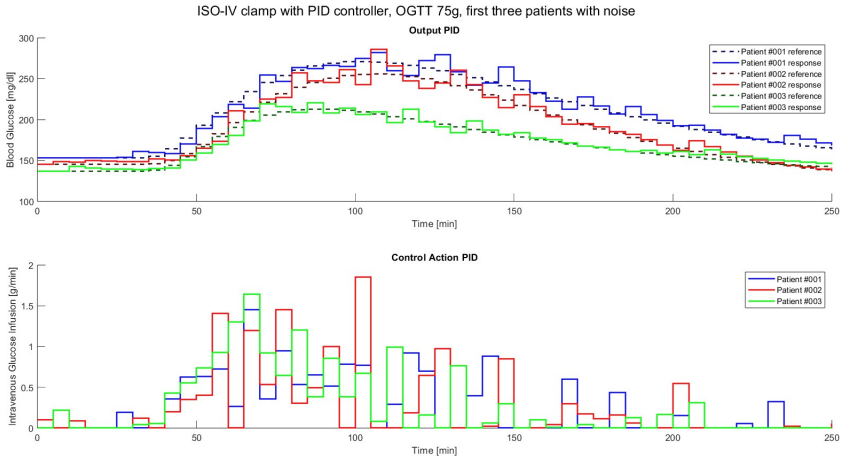


Figure 4.55: Evolution of BG level in a ISO-IV clamp (OGTT 75g) using the PID controller on first three patients, measurement noise present.

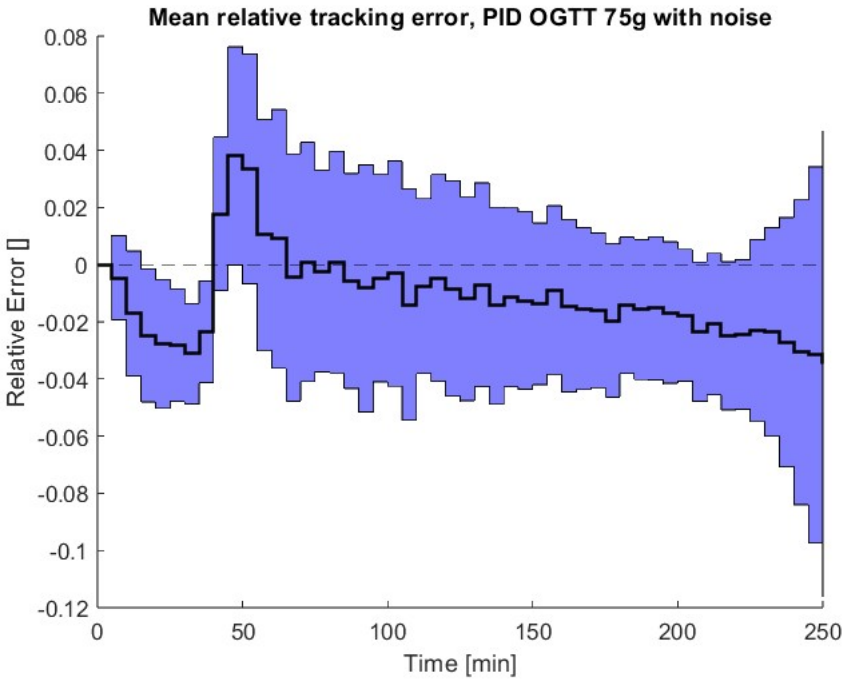


Figure 4.56: Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the PID controller on population of 100 patients, measurement noise present.

	OGTT 40g	OGTT 75g
K_p $[\frac{g \cdot dl}{mg \cdot min}]$	0.02	0.02
K_i $[\frac{g \cdot dl}{mg \cdot min^2}]$	0.0085	0.0085
K_d $[\frac{g \cdot dl}{mg}]$	0.005	0.005

Table 4.21: PID controller parameters for ISO-IV clamp on population, measurement noise present.

	OGTT 40g	OGTT 75g
MAPE (%)	2.52	2.71

Table 4.22: Mean Absolute Percentage Error (MAPE) of PID controller in a ISO-IV clamp, response to OGTT 40g and OGTT 75g, measurement noise present.

4.2.2 MPC controller with Full Increment Velocity Form

4.2.2.1 Hyperglycemic Clamp

The Mean Absolute Percentage Error (MAPE) for the hyperglycemic clamp using the MPC controller with Full Increment Velocity Form in presence of measurement noise is of 3.99%.

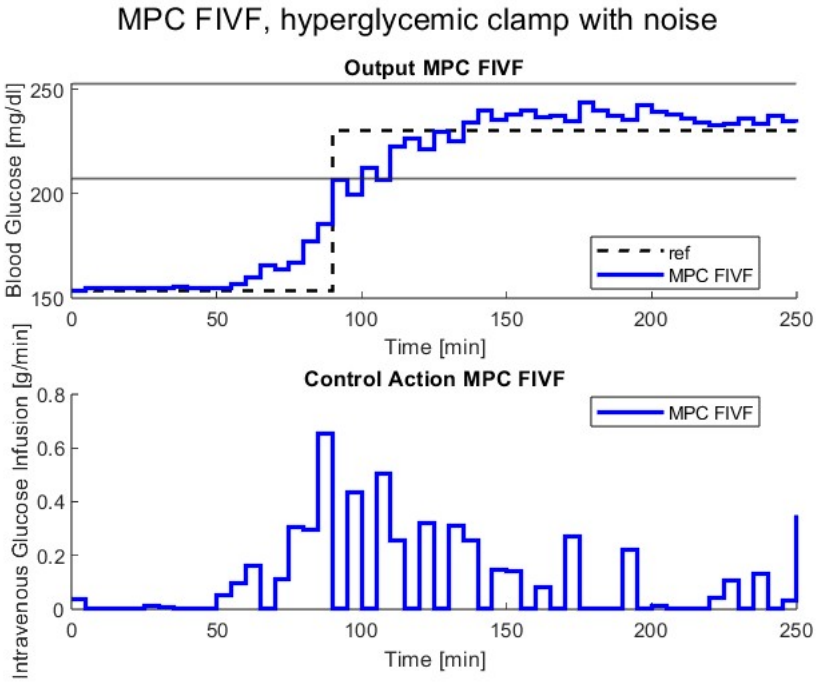


Figure 4.57: Evolution of BG level in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form on patient#001, measurement noise present.

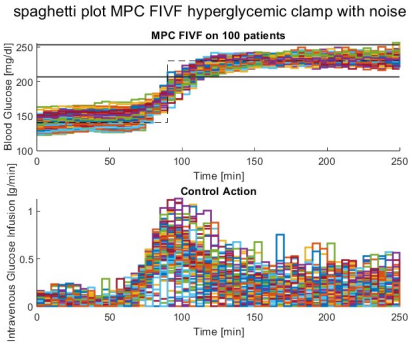


Figure 4.58: Spaghetti plot of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present.

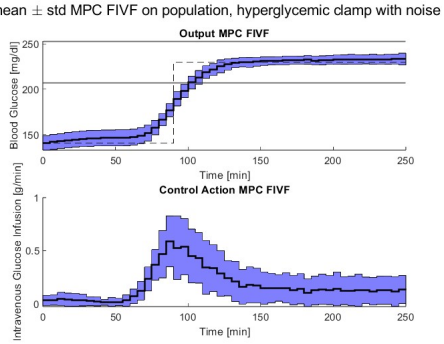


Figure 4.59: Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present.

	Patient#001, FIVF	Population, FIVF
PH	15	15
Q	0.01	0.01
R	60	100

Table 4.23: MPC with Full Increment Velocity Form parameters in a hyperglycemic clamp on patient#001 and on population, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness.

4.2.2.2 Euglycemic Clamp

The Mean Absolute Percentage Error (MAPE) for the euglycemic clamp using the MPC controller with Full Increment Velocity Form in presence of measurement noise is of 17.94%.

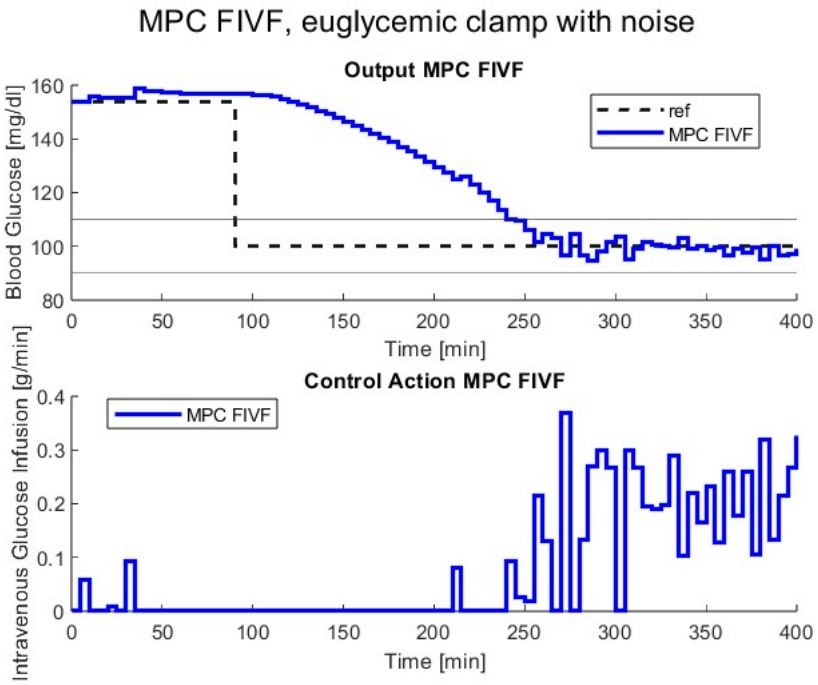


Figure 4.60: Evolution of BG level in a euglycemic clamp using the MPC controller with Full Increment Velocity Form on patient#001, measurement noise present.

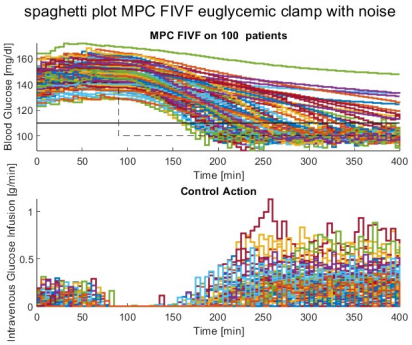


Figure 4.61: Spaghetti plot of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present.

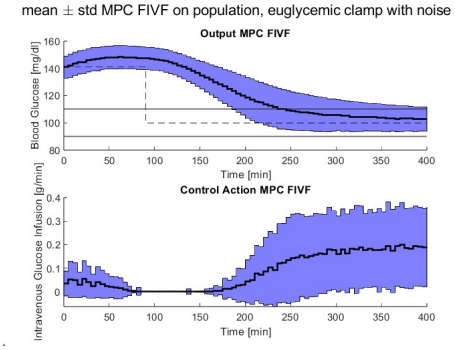


Figure 4.62: Mean ± std confidence interval of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present.

	Patient#001, FIVF	Population, FIVF
PH	15	15
Q	0.01	0.01
R	50	50

Table 4.24: MPC with Full Increment Velocity Form parameters in a euglycemic clamp on patient#001 and on population, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness.

4.2.2.3 Hypoglycemic Clamp

The Mean Absolute Percentage Error (MAPE) for the hypoglycemic clamp using the MPC controller with Full Increment Velocity Form in presence of measurement noise is of 48.28%.

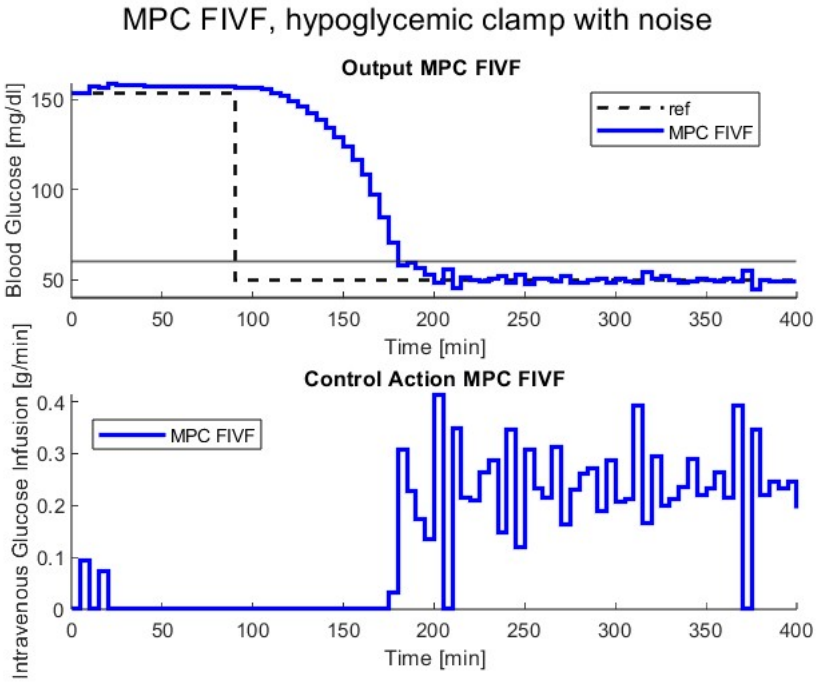


Figure 4.63: Evolution of BG level in a hypoglycemic clamp using the MPC controller with Full Increment Velocity Form on patient#001, measurement noise present.

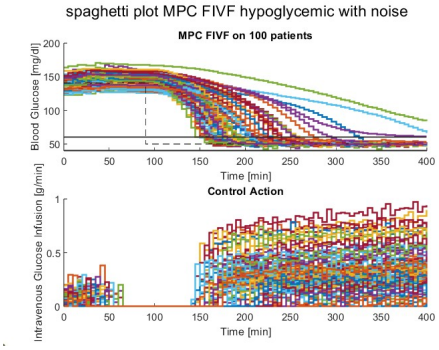


Figure 4.64: Spaghetti plot of 100 patients in a hypoglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present.

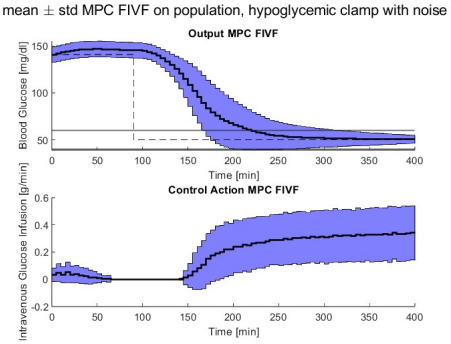


Figure 4.65: Mean ± std confidence interval of 100 patients in a hypoglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present.

	Patient#001, FIVF	Population, FIVF
PH	15	15
Q	0.01	0.01
R	20	50

Table 4.25: MPC with Full Increment Velocity Form parameters in a hypoglycemic clamp on patient#001 and on population, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness.

4.2.2.4 ISO-IV clamp

The MAPE for the ISO-IV clamp can be found in Table 4.27.

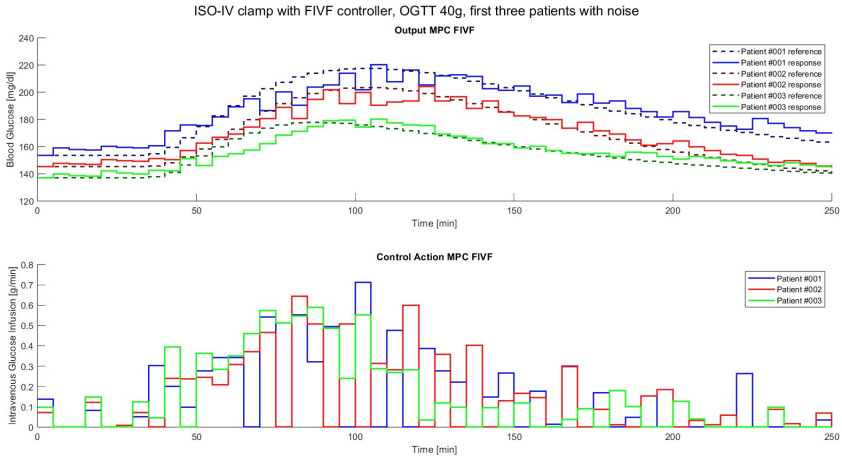


Figure 4.66: Evolution of BG level in a ISO-IV clamp (OGTT 40g) using the MPC controller with Full Increment Velocity Form on first three patients, measurement noise present.

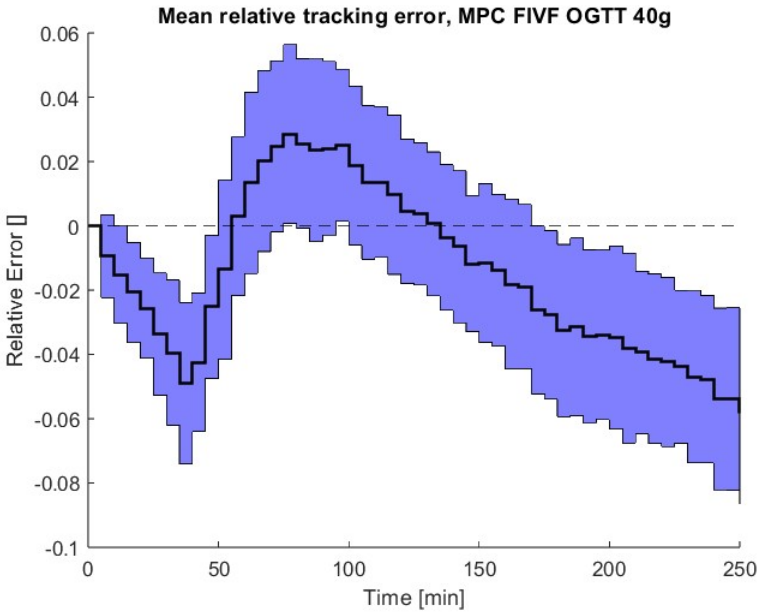


Figure 4.67: Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the MPC controller with Full Increment Velocity Form on population of 100 patients, measurement noise present.

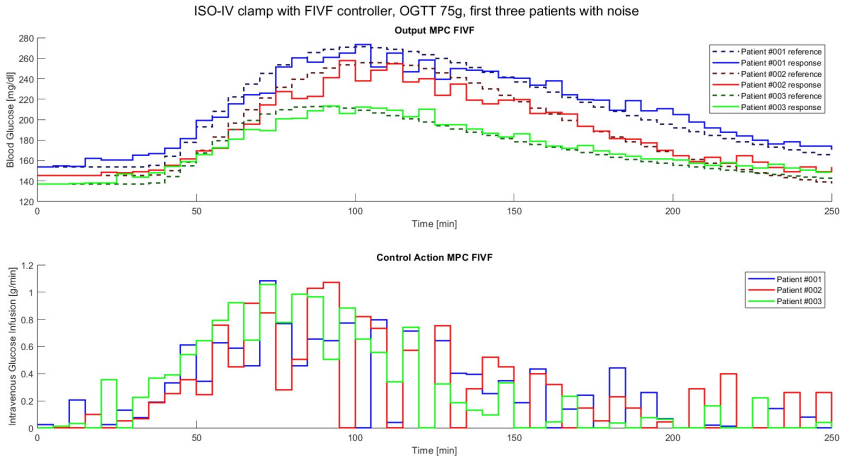


Figure 4.68: Evolution of BG level in a ISO-IV clamp (OGTT 75g) using the MPC controller with Full Increment Velocity Form on first three patients, measurement noise present.

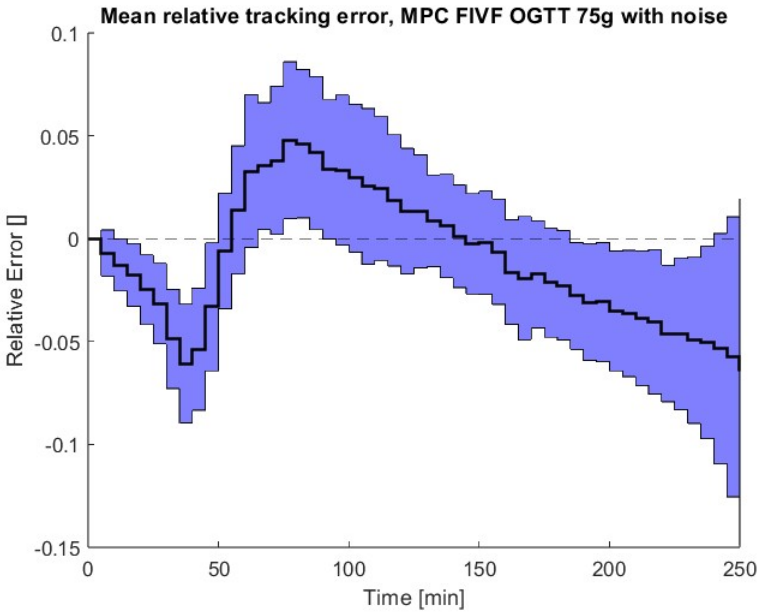


Figure 4.69: Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the MPC controller with Full Increment Velocity Form on population of 100 patients, measurement noise present.

	OGTT 40g	OGTT 75g
PH	15	15
Q	0.01	0.01
R	50	50

Table 4.26: MPC with Full Increment Velocity Form parameters in a ISO-IV clamp, on OGTT 40g and OGTT 75g, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness.

	OGTT 40g	OGTT 75g
MAPE (%)	2.99	3.38

Table 4.27: Mean Absolute Percentage Error (MAPE) of MPC controller with Full Increment Velocity Form in a ISO-IV clamp, response to OGTT 40g and OGTT 75g, measurement noise present.

4.2.3 MPC controller with the Unmeasured Disturbance Estimator approach

4.2.3.1 Hyperglycemic Clamp

The Mean Absolute Percentage Error (MAPE) for the hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, in presence of measurement noise is of 2.55%.

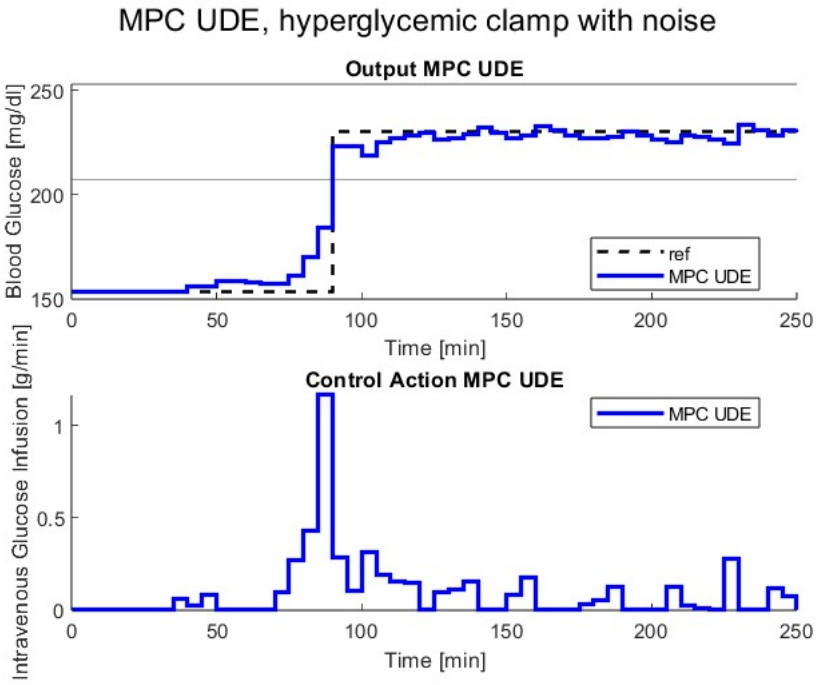


Figure 4.70: Evolution of BG level in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method on patient#001, measurement noise present.

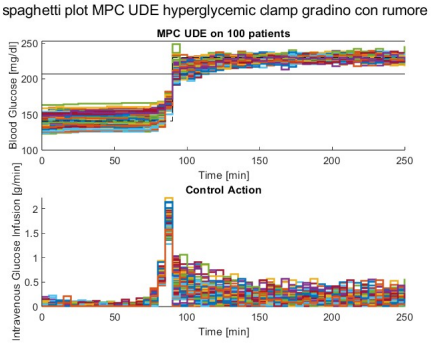


Figure 4.71: Spaghetti plot of 100 patients in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present.

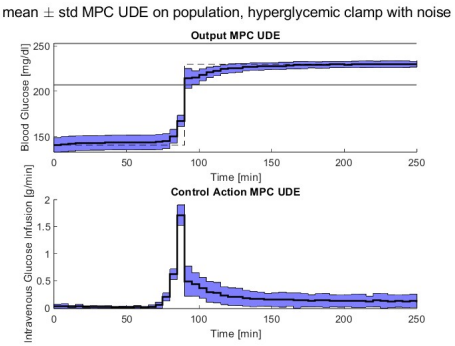


Figure 4.72: Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present.

	patient#001	Population
PH	15	15
Q	0.01	0.01
R	8	5
M_u	10	15
C_u	5	15
Q_k	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$
R_k	1	1

Table 4.28: Parameters of the MPC controller with the Unmeasured Disturbance Estimator method in a hyperglycemic clamp, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter , respectively.

[H]

4.2.3.2 Euglycemic Clamp

The Mean Absolute Percentage Error (MAPE) for the euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, in presence of measurement noise is of 16.76%.

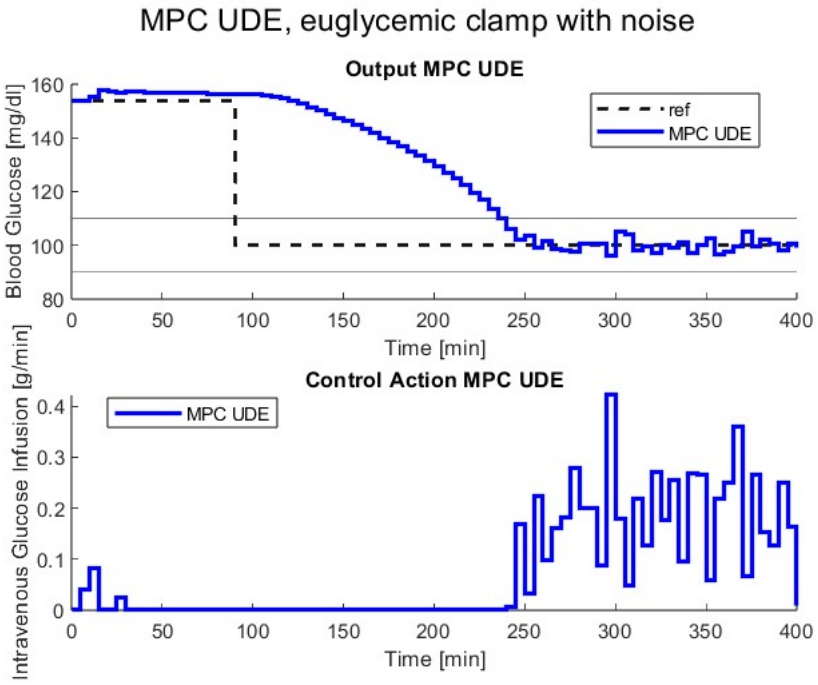


Figure 4.73: Evolution of BG level in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method on patient#001, measurement noise present.

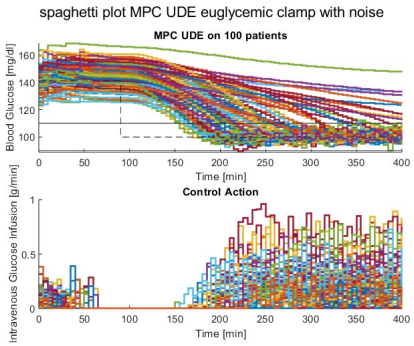


Figure 4.74: Spaghetti plot of 100 patients in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present.

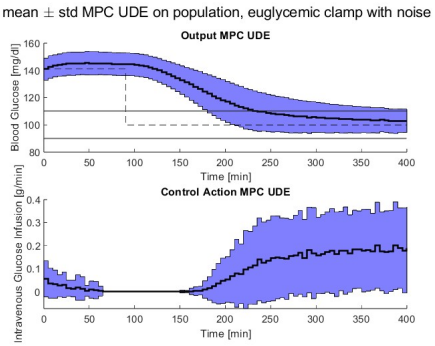


Figure 4.75: Mean ± std confidence interval of 100 patients in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present.

	patient#001	Population
PH	15	15
Q	0.01	0.01
R	10	10
M_u	10000	10000
C_u	100	100
Q_k	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$
R_k	1	1

Table 4.29: Parameters of the MPC controller with the Unmeasured Disturbance Estimator method in a euglycemic clamp, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively.

4.2.3.3 Hypoglycemic Clamp

The Mean Absolute Percentage Error (MAPE) for the hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, in presence of measurement noise is of 47.58%.

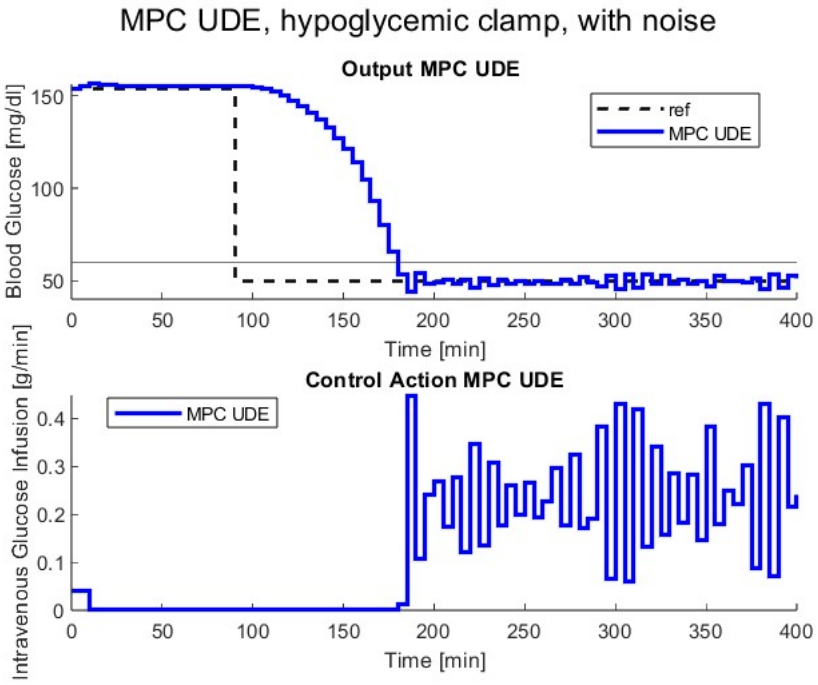


Figure 4.76: Evolution of BG level in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method on patient#001, measurement noise present.

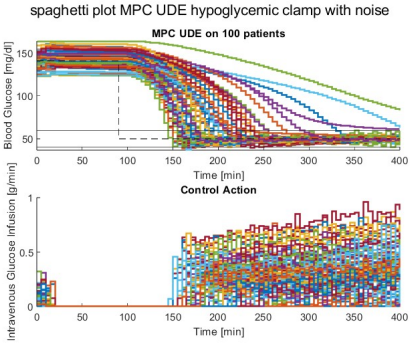


Figure 4.77: Spaghetti plot of 100 patients in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present.

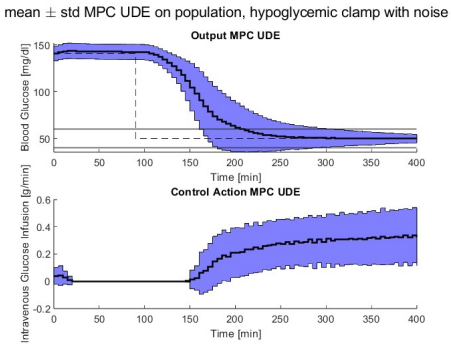


Figure 4.78: Mean ± std confidence interval of 100 patients in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present.

	patient#001	Population
PH	15	15
Q	0.01	0.01
R	0.5	0.5
M_u	1000	1000
C_u	100	100
Q_k	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$
R_k	1	1

Table 4.30: Parameters of the MPC controller with the Unmeasured Disturbance Estimator method in a hypoglycemic clamp, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively.

	Hyperglycemic	Euglycemic	Hypoglycemic
MAPE (%)	2.55	16.76	47.58

Table 4.31: MAPE for hyper, eu and hypoglycemic clamps using the MPC controller with Unmeasured Disturbance Estimator approach in presence of measurement noise

4.2.3.4 ISO-IV clamp

The MAPE for the ISO-IV clamp can be found in Table 4.33.

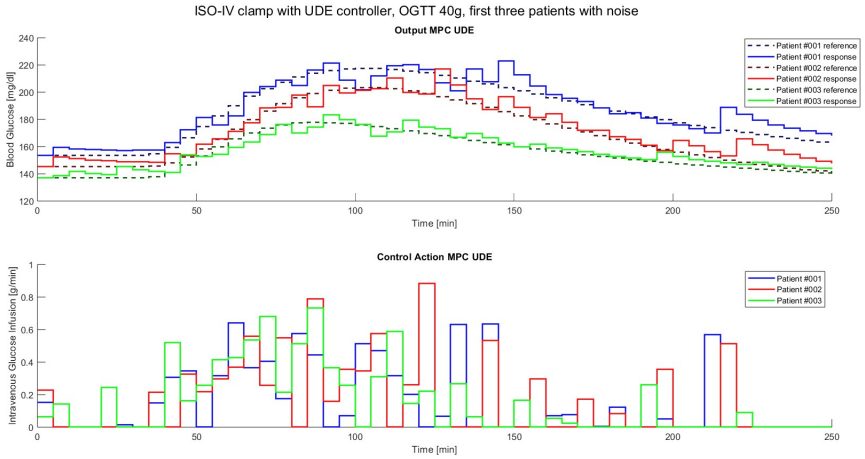


Figure 4.79: Evolution of BG level in a ISO-IV clamp (OGTT 40g) using the MPC controller with Unmeasured Disturbance Estimator approach on first three patients, measurement noise present.

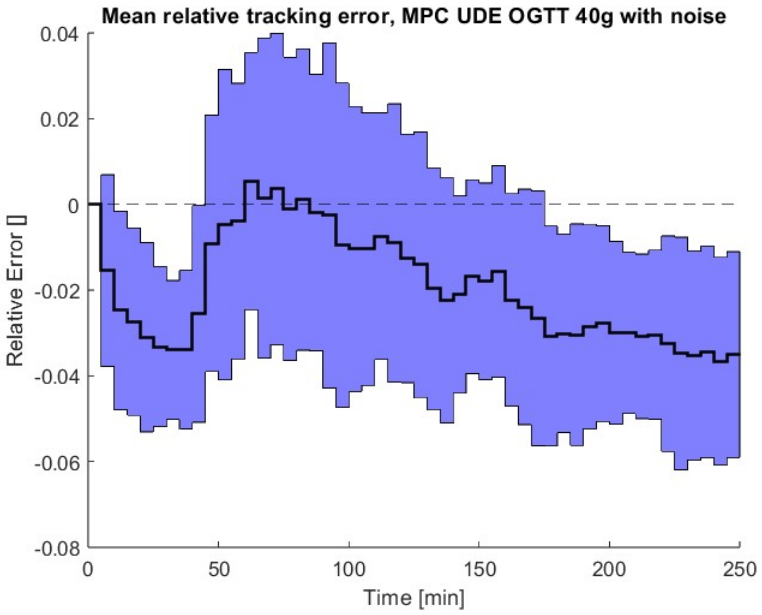


Figure 4.80: Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the MPC controller with Unmeasured Disturbance Estimator approach on population of 100 patients, measurement noise present.

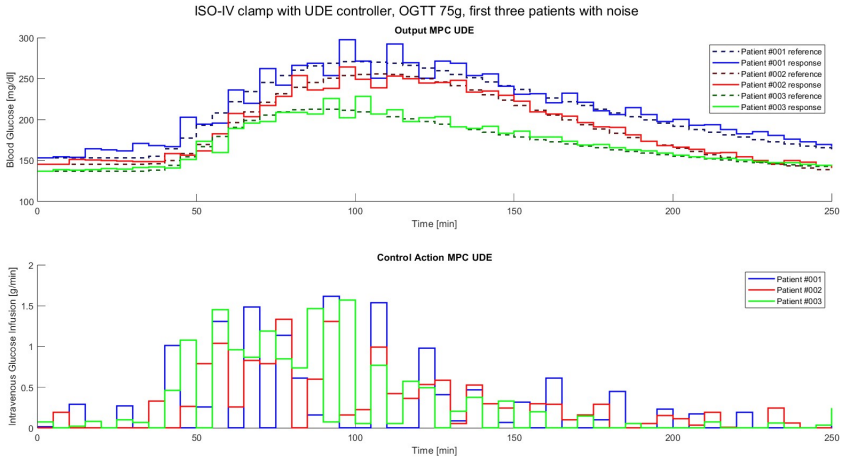


Figure 4.81: Evolution of BG level in a ISO-IV clamp (OGTT 75g) using the MPC controller with Unmeasured Disturbance Estimator approach on first three patients, measurement noise present.

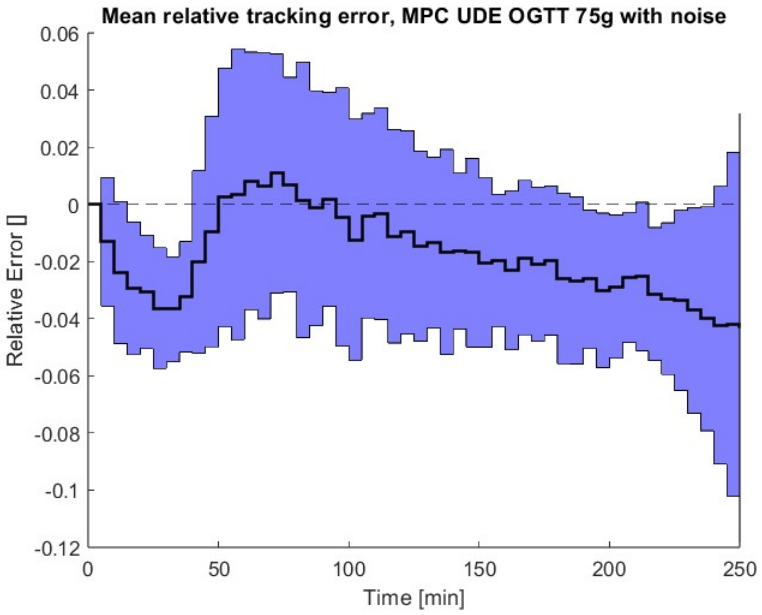


Figure 4.82: Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the MPC controller with Unmeasured Disturbance Estimator approach on population of 100 patients, measurement noise present.

	OGTT 40g	OGTT 75g
PH	15	10
Q	1000	2000
R	1	1
M_u	2500	2000
C_u	-4	-4
Q_k	$\begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$
R_k	0.5	1

Table 4.32: Parameters of the MPC controller with the Unmeasured Disturbance Estimator method in a ISO-IV clamp, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively.

	OGTT 40g	OGTT 75g
MAPE (%)	2.75	2.93

Table 4.33: Mean Absolute Percentage Error (MAPE) of MPC controller with Unmeasured Disturbance Estimator method in a ISO-IV clamp, measurement noise present.

4.2.4 Comments on the results

In this more realistic scenario, we introduced measurement errors. This real-world configuration aims to assess the performance of control algorithms under conditions that more closely resemble clinical practice, taking into account the inherent variability and challenges present in realistic settings.

As apparent from the displayed results, the performance of the controllers has not significantly changed. Introducing fluctuations to the measured blood glucose naturally leads to less efficient tracking, as observed from the plots of the controller outputs and the Mean Absolute Percentage Error (MAPE) calculated for each experiment. Nevertheless, both PID and MPC controllers demonstrated high-level performance in tracking the references across all types of glucose clamps explored.

Chapter 5

Comparing PID and MPC Performance

After presenting the results obtained by each controller—PID, MPC with Full Increment Velocity Form, and MPC with the Unmeasured Disturbance Estimator method—it is time to compare their performance across the four types of glucose clamp examined. As done for the Section 4, where we presented the results accomplished first in a noise-free scenario and then in a more realistic scenario, also in this Section we will see how the controllers performed first in an ideal configuration and then with the presence of measurement noise.

5.1 Noise-free Scenario

5.1.1 Hyperglycemic clamp

In Fig.5.1, we observe the performance of the three different controllers implemented compared in a hyperglycemic clamp on the entire population. The effectiveness of the MPC controllers is apparent: with the lookahead feature, the controller can reach the target reference faster by acting earlier and administering a glucose infusion rate before the reference changes. As depicted in the figure and further corroborated by Table 5.1, the MPC with the Unmeasured Disturbance Estimator approach performed the best, achieving a MAPE of 2.00%. The MPC with Full Increment Velocity Form is slightly slower but still performs admirably with a MAPE of 2.21%. This discrepancy is apparent when observing the glucose infusion rate administered by the controllers: the former provides a higher value of GIR just before the change in the reference, while the latter initiates with smaller values beforehand and remains under 1 g/min. The PID controller, lacking the ability to anticipate changes in the reference, starts later. However, by administering a relatively high glucose infusion rate, it eventually reaches the target blood glucose level, after a small overshoot that the two MPC controller managed to avoid, achieving a MAPE of 2.87%.

	PID	MPC FIVF	MPC UDE
MAPE (%)	2.87	2.21	2.00

Table 5.1: Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers, hyperglycemic clamp.

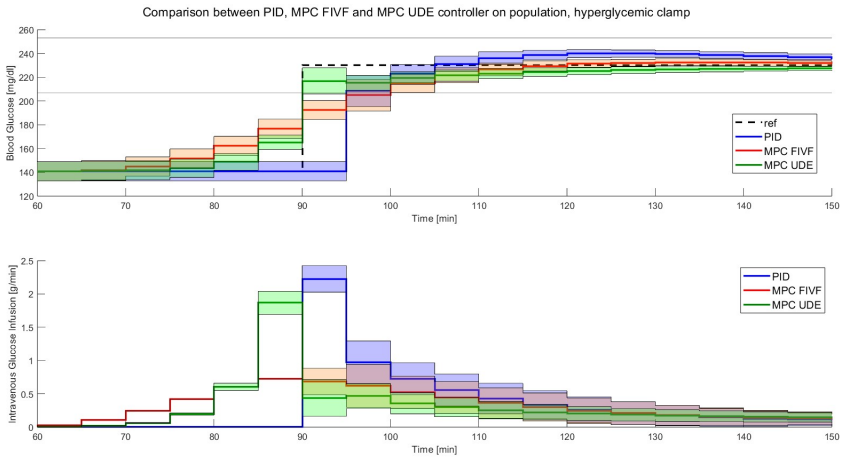


Figure 5.1: Comparison between PID, MPC FIVF and MPC UDE controllers on population, hyperglycemic clamp. Zoom between minute 60 and 150.

5.1.2 Euglycemic clamp

For the euglycemic clamp, the comparison between the results of the three controllers on the population is depicted in Fig.5.2. As previously mentioned, in this type of clamp, insulin plays a crucial role as it lowers blood glucose concentration. However, since our controllers can only manipulate the input of the system, which is glucose infusion, they cannot expedite the process of lowering BG levels. Therefore, until the blood glucose remains above a certain level, they remain inactive. From the image, we can observe that around minute 180, all three controllers switch on and provide a glucose infusion rate to avoid undershoots and reach the target level. Due to this limitation, all three controllers perform almost identically, as they cannot significantly influence the process until the blood glucose concentration is sufficiently lowered. Their performance, measured with the MAPE, is summarized in Table 5.2.

	PID	MPC FIVF	MPC UDE
MAPE (%)	14.62	14.51	14.59

Table 5.2: Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers, euglycemic clamp.

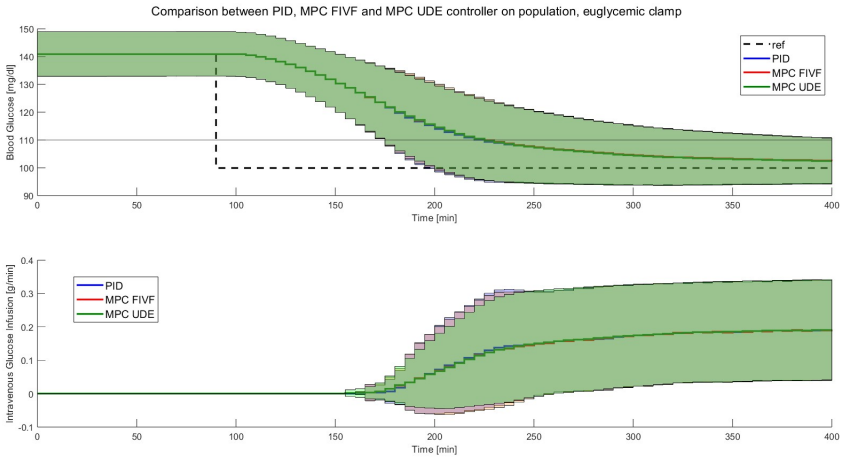


Figure 5.2: Comparison between PID, MPC FIVF and MPC UDE controllers on population, euglycemic clamp.

5.1.3 Hypoglycemic clamp

For the hypoglycemic clamp experiment, the comparison between the results of the three controllers on the population is shown in Fig.5.3. Similar considerations made for the results obtained in the euglycemic clamp also apply to this type of glucose clamp. In Table 5.3, the MAPE for this type of experiment is displayed, demonstrating that the performance of the controllers is very similar. However, the PID controller appears to perform slightly poorer by activating a few minutes later than the two MPCs and by causing a small undershoot. A MAPE of 45% may seem high, but as previously mentioned, we must consider that the BG levels before the reference changes are around 130-150 mg/dl. Therefore, in the first 50-75 minutes after the step, the error accumulates for quite some time. Some potential future developments could involve varying the intravenous insulin administered to observe its effects on control performance..

	PID	MPC FIVF	MPC UDE
MAPE (%)	45.72	44.33	44.59

Table 5.3: Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers, hypoglycemic clamp.

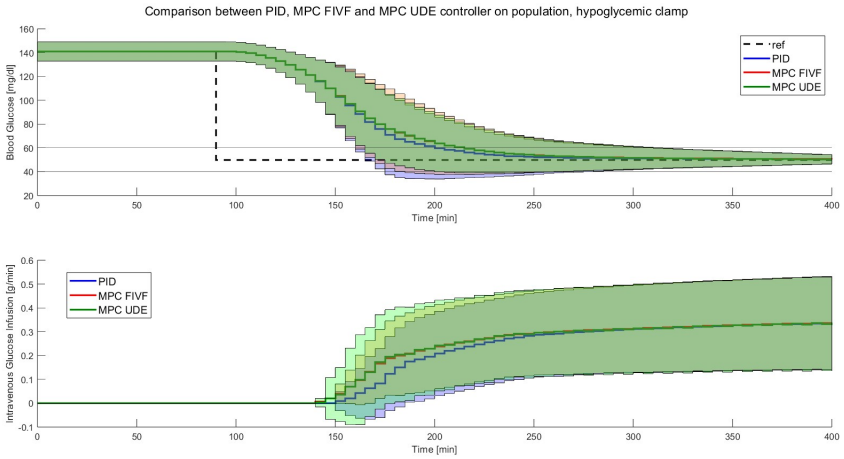


Figure 5.3: Comparison between PID, MPC FIVF and MPC UDE controllers on population, hypoglycemic clamp.

5.1.4 ISO-IV clamp

The comparison of the results obtained in our last type of glucose clamp, the ISO-IV clamp, is illustrated in Fig.5.4. Specifically, the mean relative tracking error achieved by the three different controllers is displayed. As mentioned previously, a plot of mean $\pm$ std for these types of clamps would not be too meaningful since each patient uses its OGTT response from the previous day as a reference. Much more informative is the plot of the tracking error, which provides insights into the performance of the three controllers. (Reminder: The relative tracking error is defined as $\frac{y_0-y}{y_0}$, where y_0 is the reference and y is the output.)

As observed from the image, the PID controller generates a positive error in the ascending phase of the OGTT, attributable to the absence of the look ahead feature, resulting in the reference being greater than the output. Subsequently, it produces a small negative error, indicating that the output remains slightly above the reference. On the other hand, the two MPC controllers initially anticipate the ascending phase of the OGTT, as evidenced by the negative tracking error. They then transition to staying slightly under the reference curve (characterized by the positive error around minute 70 where the OGTT reaches its peak), and finally follow the PID controller by remaining slightly above the OGTT curve. The MAPE for the three controllers is shown in Table 5.4 (first row). All three controllers provided excellent tracking, we have in fact MAPE very close to 0%.

	PID	MPC FIVF	MPC UDE
MAPE (%), OGTT 40g	0.29	0.31	0.26
MAPE (%), OGTT 75g	0.51	0.55	0.51

Table 5.4: Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers, ISO-IV clamp.

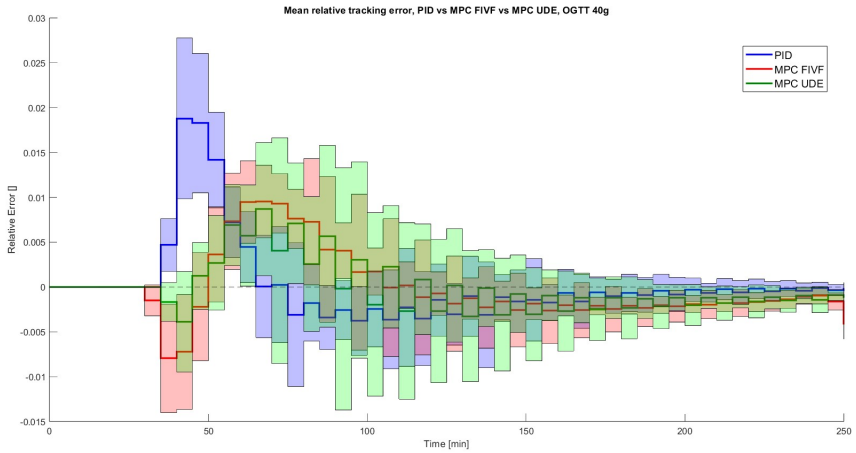


Figure 5.4: Comparison between PID, MPC FIVF and MPC UDE controllers on population, relative tracking error, ISO-IV clamp (OGTT 40g).

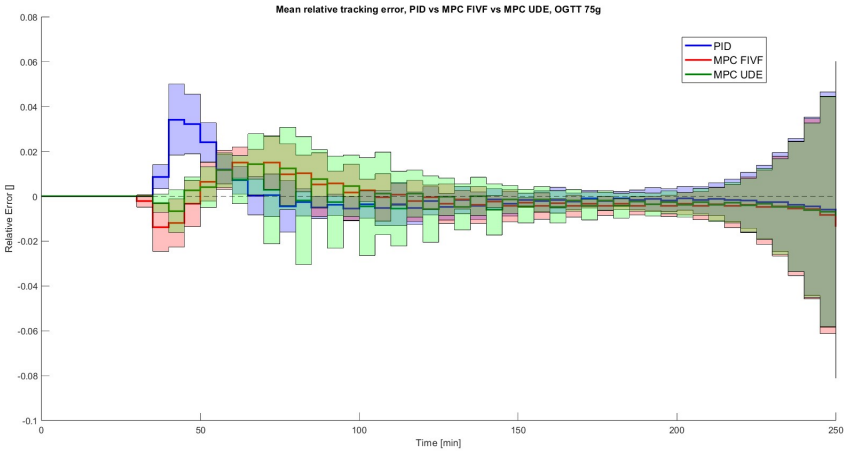


Figure 5.5: Comparison between PID, MPC FIVF and MPC UDE controllers on population, relative tracking error, ISO-IV clamp (OGTT 75g).

For the ISO-IV clamp on the 75g OGTT the same considerations done for the 40g OGTT apply. The relative tracking error (Fig.5.5) is a bit higher due to higher values of blood glucose, but the behaviour of the 3 controllers stay the same. The performance of the controller can be found in Table 5.4 (second row). We can see that, even if slightly greater, the MAPE stays very close to 0% for all three controllers, signifying excellent tracking.

5.2 Realistic Scenario

5.2.1 Hyperglycemic Clamp

In Fig.5.6, we observe the comparison between the three controllers after incorporating measurement noise to simulate a more realistic scenario. It is apparent that the noise does not significantly affect the controllers' performance; there is only a slight impact observed at the beginning of the experiment and during steady state after the step. We can also see this in the Mean Absolute Percentage Error (Table 5.5). The MPC controller with the Unmeasured Disturbance Estimator method confirms to be the more efficient in tracking the reference in a hyperglycemic clamp experiment.

	PID	MPC FIVF	MPC UDE
MAPE (%)	3.46	3.99	2.55

Table 5.5: Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers in a hyperglycemic clamp, measurement noise present.

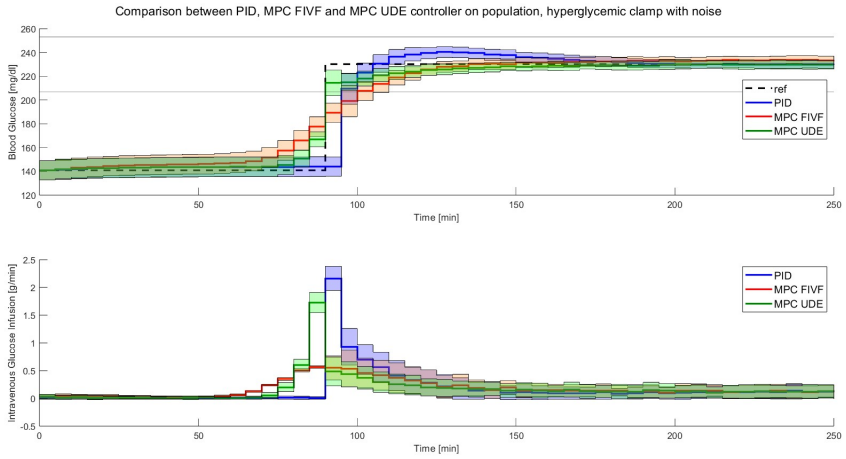


Figure 5.6: Comparison between PID, MPC FIVF and MPC UDE controllers on population in a hyperglycemic clamp, measurement noise present.

5.2.2 Euglycemic Clamp

With regard to the euglycemic clamp, the comparison between the three controllers with the presence of measurement noise is shown in Fig.5.7. As anticipated, the existence of noise does not alter the behavior of the controllers

significantly. The MAPE, displayed in Table 5.6, has increased slightly, but we still achieve very reasonable performance. The MPC controller with Full Increment Velocity Form appears to be slightly slower than the other two controllers, as evidenced by the higher MAPE, but the difference is minimal.

	PID	MPC FIVF	MPC UDE
MAPE (%)	16.85	17.94	16.76

Table 5.6: Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers in a euglycemic clamp, measurement noise present.

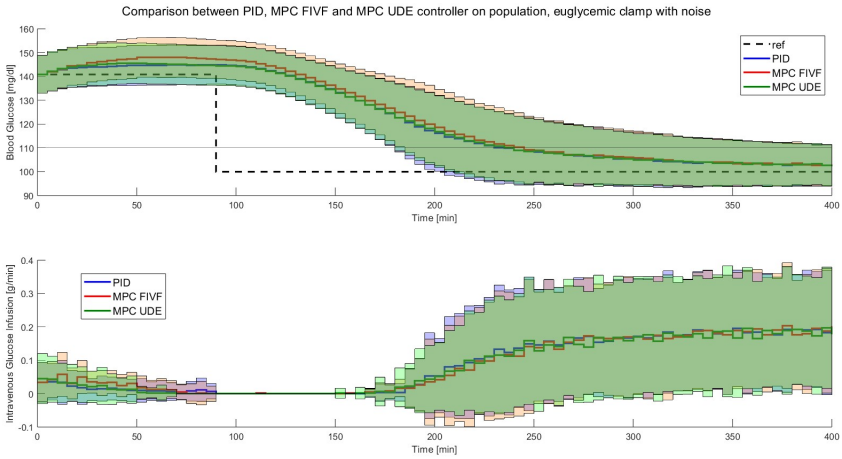


Figure 5.7: Comparison between PID, MPC FIVF and MPC UDE controllers on population in a euglycemic clamp, measurement noise present.

5.2.3 Hypoglycemic Clamp

As usual, the same considerations made for the euglycemic clamp can be made for the hypoglycemic clamp (Fig.5.8). Here as well, obviously, the performance worsen a little bit with the presence of measurement noise (Table 5.7). As seen before the MPC with Full Increment Velocity Form stays marginally above the others, while the PID controller execute a slightly undershoot, but the differences are really minimal.

	PID	MPC FIVF	MPC UDE
MAPE (%)	48.22	48.28	47.58

Table 5.7: Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers in a hypoglycemic clamp, measurement noise present.

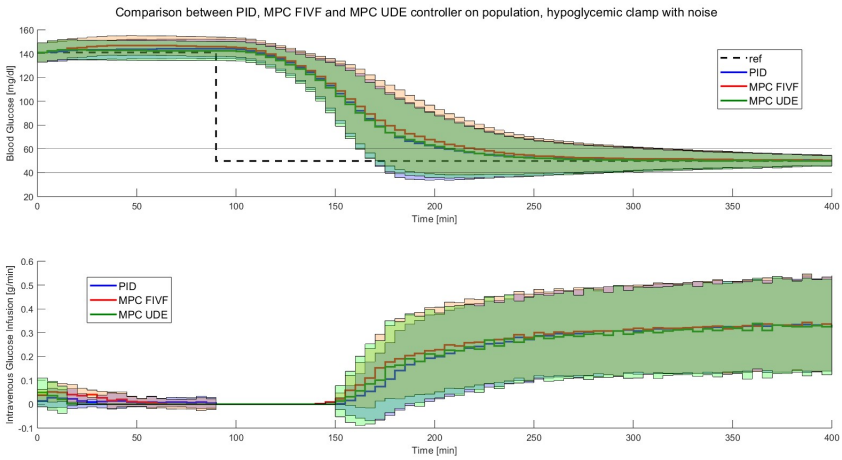


Figure 5.8: Comparison between PID, MPC FIVF and MPC UDE controllers on population in a hypoglycemic clamp, measurement noise present.

5.2.4 ISO-IV Clamp

Finally, the mean relative tracking error achieved by the three controllers with the presence of measurement noise is displayed in Fig.5.9 and Fig.5.10 for the 40g and 75g OGTTs, respectively. In both the 40g and 75g experiments, although the relative error has clearly increased, the behavior of the controllers has not changed significantly, as apparent from Table 5.8. If anything, it appears that the addition of noise has led the PID controller to produce the smallest Mean Absolute Percentage Error, both in the 40g and 75g ISO-IV clamps.

	PID	MPC FIVF	MPC UDE
MAPE (%), OGTT 40g	2.52	2.99	2.75
MAPE (%), OGTT 75g	2.71	3.38	2.93

Table 5.8: Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers in a ISO-IV clamp, measurement noise present.

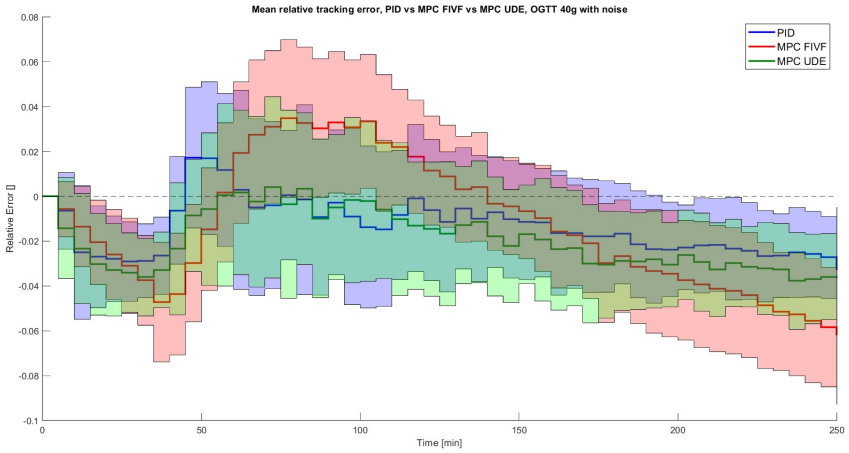


Figure 5.9: Comparison between PID, MPC FIVF and MPC UDE controllers on population, relative tracking error, ISO-IV clamp (OGTT 40g), measurement noise present.

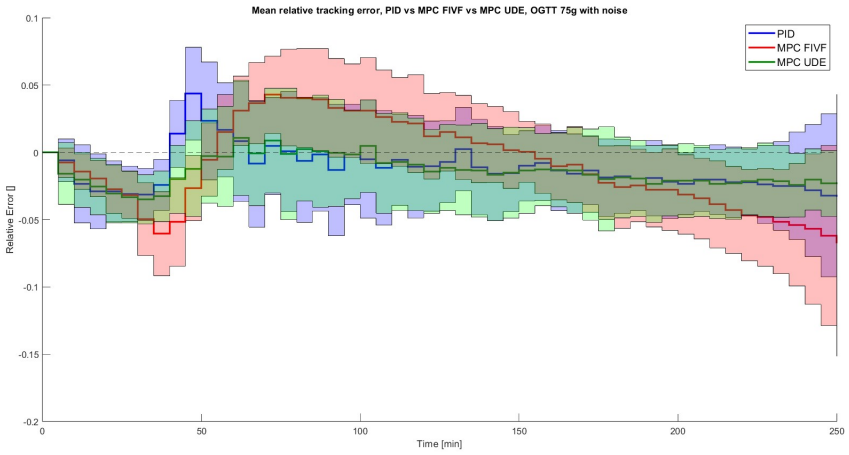


Figure 5.10: Comparison between PID, MPC FIVF and MPC UDE controllers on population, relative tracking error, ISO-IV clamp (OGTT 75g), measurement noise present.

5.3 Final Comparison

By analyzing the results presented in the previous sections, determining a clear winner among the three controllers is not straightforward. In the hyperglycemic clamp, the Model Predictive Control (MPC) with the Unmeasured Disturbance Estimator approach appears to be the most effective in tracking the reference, attributed to its swift response in administering a significant glucose infusion rate prior to the change in the target curve. This superiority is confirmed by the Mean Absolute Percentage Error, observed in both ideal and real-world scenarios.

In contrast, for the euglycemic and hypoglycemic clamp experiments, no obvious winner emerges. All three controllers perform similarly, largely due to the influence of insulin. In such cases, developing a more complex MPC controller may not be necessary, as the simpler Proportional-Integral-Derivative controller achieves practically identical performance.

Regarding the ISO-IV clamps, excellent tracking is achieved with all three controllers, with the Mean Absolute Percentage Error (MAPE) nearly approaching 0%. Similar to the previous two clamp types, identifying a clear winner is challenging. Under ideal conditions, where measurement noise is absent, the MPC with the Unmeasured Disturbance Estimator Approach appears to be slightly more accurate. However, in realistic scenarios, the PID controller demonstrates marginally better performance. Once again, it appears that the PID controller may be the preferable choice due to its simplicity compared to the more complex MPC controller, especially considering the comparable performance observed across all clamp types.

Chapter 6

Mobile Application

6.1 Introduction

In the previous sections we developed both PID and MPC algorithms to suggest Glucose Infusion in glucose clamp experiments (Hyperglycemic, Euglycemic, Hypoglycemic and ISO-IV). Given the good performance achieved, we chose to design a mobile application as a decision support system for clinical researchers conducting glucose clamp experiments (the glucose clamp supported are Hyperglycemic, Euglyucemic and Hypoglycemic). The concept behind the app is quite simple: after inserting a blood glucose (BG) measurement, a glucose infusion rate (GIR) is suggested.

Due to its simplicity compared to an MPC, we integrated a PID controller into the application. The parameters chosen are the ones that permitted the best performance on the three type of clamps in the realistic scenario (Tab.6.1).

Glucas (GLUcose CLamp ASsistant) has already been successfully deployed as computer software [1]. Recognizing the increasing prevalence of mobile platforms and the benefits they offer in terms of accessibility and convenience, we have made the strategic decision to extend its functionality by developing a dedicated mobile application. This expansion to mobile platforms aims to further enhance the usability and accessibility of Glucas, ensuring that researchers can easily access its features and capabilities directly from their smartphone. We implemented Glucas 2.0 in the Flutter environment, in the next section a brief overview of the Flutter framework is provided.

	Hyperglycemic	Euglycemic	Hypoglycemic
K_p $[\frac{g \cdot dl}{mg \cdot min}]$	0.02	0.025	0.02
K_i $[\frac{g \cdot dl}{mg \cdot min^2}]$	0.001	0.003	0.001
K_d $[\frac{g \cdot dl}{mg}]$	0.002	0.001	0.001

Table 6.1: PID controller parameters used for Glucas 2.0.

6.2 Dart and Flutter

Mobile applications have become essential in our daily lives, with a significant increase in network traffic on mobile devices since November 2016, surpassing desktops or laptops. To meet the diverse user base's needs, developers face the challenge of adapting apps to both Android and iOS platforms, requiring expertise in different programming languages such as Java/Kotlin and Objective-C/Swift.

Cross-platform development presents hurdles due to the lack of a unified codebase across mobile platforms. Frameworks like Apache Cordova and Ionic aim to streamline development by combining native and web app approaches. However, they may not fully utilize platform-specific features, leading to potential limitations in performance and resource access.

Google introduced Flutter in 2018 to address these challenges, offering a seamless cross-platform development experience for iOS and Android. Flutter's features, including hot reload for quick code changes and modular widgets for flexible UIs, have gained popularity among major companies like Alibaba and Google.

Developers leverage Flutter with development environments like Android Studio and Visual Studio Code. Dart, the programming language behind Flutter, provides essential features like clear syntax and support for reactive programming, contributing to Flutter's efficiency in app development.

The synergy between Dart and Flutter enables the creation of high-performance apps, with Dart's flexibility in compilation playing a crucial role. Developers can utilize tools like Visual Studio or Android Studio to build Flutter apps effectively.

6.3 Project Structure

The project was crafted within the Visual Studio Code environment, complemented by the utilization of Android Studio to simulate an Android smartphone. In the following sections, we will provide a brief overview of the key components comprising a mobile application developed using Flutter.

6.3.1 UI

In Flutter-based applications, the main building blocks are called Widgets, which can represent different elements such as containers, columns, rows, bottons and many others. Conceptually, everything in a flutter app is a widget. They have a hierarchical structure, so the "child" widget nests within the "parent" one and so on. There are two main types of widgets, the Stateless ones and the Stateful ones. Stateless widget remain unchanged once created and do not alter their behavior based on user event, while Stateful widgets are dynamic, allowing for changes in content over time based on user-generated actions. They

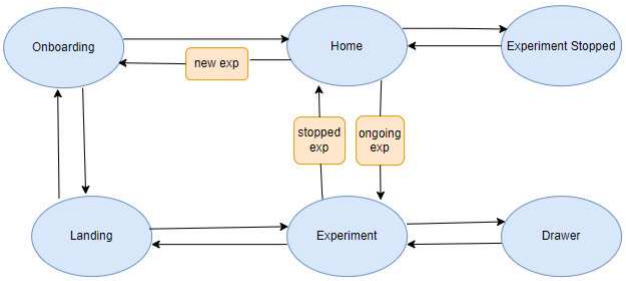


Figure 6.1: Glucas 2.0 navigation flow chart.

are particularly useful for developing interfaces with dynamic components, offering options for managing dynamism and immutability.

During the app development process, numerous widgets are used to create various elements and functionalities. In our application, we employed a wide range of widgets including containers, rows, columns, texts, buttons, pop-up dialogs, timers, and charts, among others. These widgets collectively contribute to crafting a rich and interactive user experience within the application.

6.3.2 Navigation

Gluclas is designed as a user-friendly mobile application with a streamlined structure organized into several screens:

1. Home
2. Onboarding
3. Landing
4. Experiment
5. Experiment Stopped

Each screen is carefully crafted to provide intuitive navigation and seamless user experience, ensuring that users can efficiently access and utilize Glucas 2.0's features and functionalities. In Figure 6.1, we present the simple navigation flowchart outlining the structure of the app's screens.

6.3.3 State Management and Data Persistence

To efficiently manage state within our application, we made use of a Provider, a powerful tool provided by Flutter. The Provider allows seamless passing of information among all widgets within the application, ensuring effective state management throughout the user interface.

For data persistence within the application, we adopted two distinct approaches, each chosen based on the nature of the data to be saved. We utilized the SharedPreferences approach, a tool that enables the storage of key-value (KV) pairs comprising primitive values such as strings, integers, floating-point numbers, or boolean values. SharedPreferences are commonly employed for saving information that does not require the complexity of a database. In our application, we utilized SharedPreferences for storing simple data such as the current experiment number (IdExp) and a boolean value (thereIsExperiment) indicating whether there is an ongoing experiment. This approach ensures that when the app is reopened, the user is directed straight to the experiment page if an experiment is in progress.

To save more complex information, almost every app implements a local database. In mobile app development with Flutter, the sqflite plugin is essential for SQLite database management. It supports iOS and Android, offering transaction control and batch operations for efficient database modifications. The initial phase of developing the database involves creating an Entity-Relationship (E-R) diagram. A database follows a designated data model, particularly a model designed for organizing data. Typically, a relational database is utilized, consisting of interconnected relations depicted as tables. Each table has its schema and stores data instances. Designing a relational database involves two main stages: conceptual design and logical design.

Conceptual design results in a formal depiction of database requirements known as the Entity-Relationship (E-R) model. This model includes three key elements: Entities, representing classes of objects with common characteristics and independent existence; Relationships, indicating conceptual connections between two or more entities; and Attributes, describing basic properties of entities or relationships. The schema should also specify cardinalities for each entity involved in a relationship and identifiers for each entity. Identifiers are defined as the set of attributes in the schema uniquely identifying instances of the entity. The conceptual schema used for Glucas 2.0 is represented if Fig.6.2.

We have two entities: "Experiment" and "Data". An experiment can generate from 0 to N data points, and each data point belongs to one and only one experiment. The "Experiment" entity is defined with "ID" as its primary key, serving as the attribute uniquely identifying each row in the database. Additional attributes include "tStart," "TicoGlobal," and "stopped," which hold information regarding the experiment's status. Conversely, attributes like "pzId," "pzGroup," "pzAge," "pzWeight," "pzHeight," and "typeOfClamp" are designated to store patient and clamp-related information. On the other hand, the "Data" entity has "ID" as its primary key and "IdExp" as a foreign key. This signifies that two data points with the same ID may exist, but they cannot belong to the same experiment. The "Data" entity includes attributes related to data such as "BG", "BG_time", "suggestedDose", "actualDose", "doseTime" , "sumError" , and "time_tic". Additionally, there are boolean values like "selectedIsSuggested" and "isDoseAlreadyInserted" which aid in the insertion of dose values into the database. The logical schema precisely and effectively encapsulates all the information outlined by an E-R schema generated during the

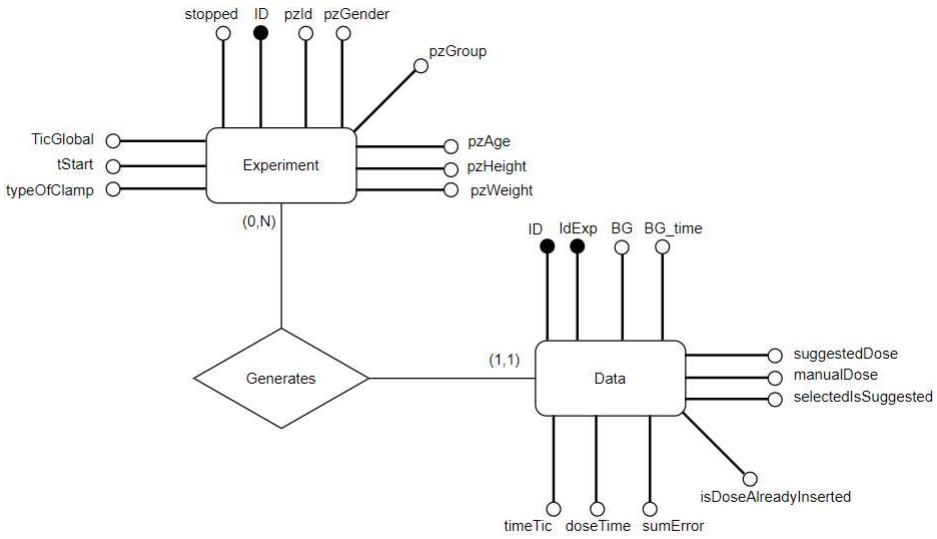


Figure 6.2: E-R model on which the App database is based.

conceptual design phase. The elements of the E-R schema are translated into tables following specific rules determined by the cardinalities of the entities. In summary, our app relies on two tables:

- **EXPERIMENT**(ID, stopped, ticGlobal, tStart, pzId, pzGroup, pzGender, pzAge, pzHeight, pzWeight, typeOfClamp)
- **DATA**(ID, IdExp, BG, BG_time, suggestedDose, manualDose, timeTic, doseTime, sumError, selectedIsSuggested, isDoseAlreadyInserted)

6.4 App Workflow

In the following pages, we will outline the main features of Glucas 2.0 during a typical run of the mobile application. Upon opening the app, the user is directed to the "home" page, depicted in Fig.6.3, where they can initiate a new glucose clamp experiment. Upon pressing the "Create New Experiment" button, the user is navigated to the "onboarding" page (Fig.6.4). Here, patient data is collected, and the user can select from three types of clamps. It is worth noting that providing these details is optional, allowing users to navigate without mandatory input. The only required information is the type of glucose clamp and the ID of the patient. Although not strictly required for the control algorithm itself, additional information such as age, height, and weight can be helpful for physicians in assessing the most appropriate control actions

to take. The GDPR (General Data Protection Regulation) is a comprehensive data protection regulation implemented by the EU in 2016, representing one of the most extensive and progressive data protection regulations globally. Each EU member state has adopted a national regulation on data protection compliant with the GDPR. For example, in Italy, it is the DECRETO LEGISLATIVO 10 agosto 2018, n.101. The GDPR includes seven main principles, with the "Data Minimisation" principle emphasizing that personal data should be adequate, relevant, and limited to what is necessary for processing purposes. To adhere to GDPR standards, the app avoids requesting unnecessary data and does not mandate their entry. Additionally, it refrains from soliciting sensitive patient information, such as names or surnames, to prevent recognition and identification of individuals.

Subsequently, the user is directed to the "landing" page (Fig.6.5), which serves as a transition screen before commencing the experiment. Once ready, the user can initiate the actual experiment by pressing the noticeable button.

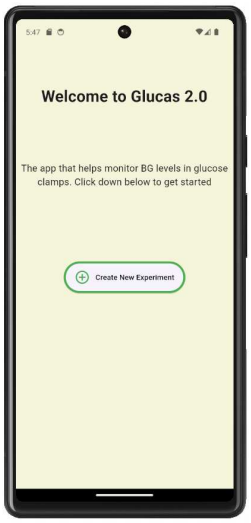


Figure 6.3: Home.

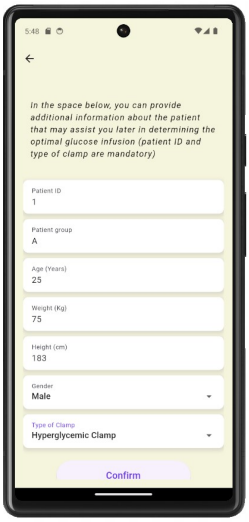


Figure 6.4: Onboarding.

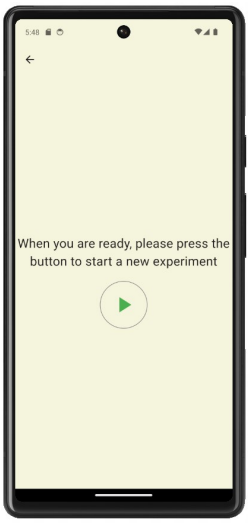
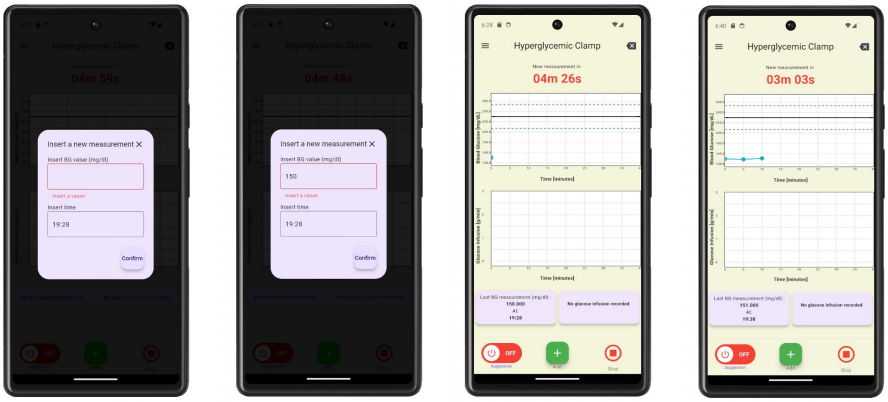


Figure 6.5: Landing.

When the experiment initiates, a pop-up alert dialog prompts the user to input the blood glucose (BG) of the patient (Fig.6.6a). Although the default insertion time is set to the current time, we have integrated the option to adjust the insertion time, providing users with enhanced flexibility. Upon entering the BG measurement (Fig.6.6b) and pressing the "confirm" button, the dialog will close. As depicted in Figure6.6c, the experiment page displays two charts: one for BG measurements and another for glucose infusion. Notably, a value of 150 mg/dl has been incorporated into the first chart. Furthermore, Fig.6.6d illustrates three additional measurements that have been included.

For these glucose clamp experiments, we have configured the sampling time



(a) (b) (c) (d)

Figure 6.6: a) Measurement request. b) Measurement insertion. c) Plot of the first measurement. d) Plot of three measurements.

(a) (b)

Figure 6.7: a) Add measurement and doses button. b) Activate suggestion of PID controller.

T_s to be 5 minutes, consistent with the control algorithms discussed in Section 2. Positioned above the charts is a timer set to 5 minutes. Once the timer expires, it will reset, the pop-up dialog will reappear and the procedure repeats. If the user does not close the dialog or does not insert the measurement within one minute the dialog will close automatically. It can be reopened using the "add" button. (Fig.6.7a). Below the two charts we find two areas where the last BG and glucose infusion value are displayed, togheter with the time of insertion.

Glucas 2.0 has two operating modalities for glucose infusion: suggestion OFF and suggestion ON. The user can switch between modes clicking the "suggestion" button at the bottom-left corner of the screen. In the first modality, called "reading mode", the PID controller will not be active (Fig.6.8a). This mode simulates a scenario where a physician has taken some blood glucose (BG) measurements but has not yet initiated the experiment. In this case, which is the one we have seen until now, after inserting a BG value the user has the possibility to manually administer intravenous glucose. Conversely, when suggestions are activated, the dosage dialog will effectively propose the output of the PID controller (Fig.6.8b), which aims to bring the patient's blood glucose as close as possible to the reference value (depicted by the black line in the first chart). Upon selecting "confirm," the user accepts the suggested dosage and administers it accordingly.

In Fig.6.9 we can see other 3 data points added to the charts. If the clinical researcher wishes to administer a different quantity of glucose for any reason, they can do so by manually entering the dose in the text field (Fig.6.10a). The glucose infusion rate is then added in the second chart, in red the suggested GIR and in blue the actual dose (Fig.6.10b).

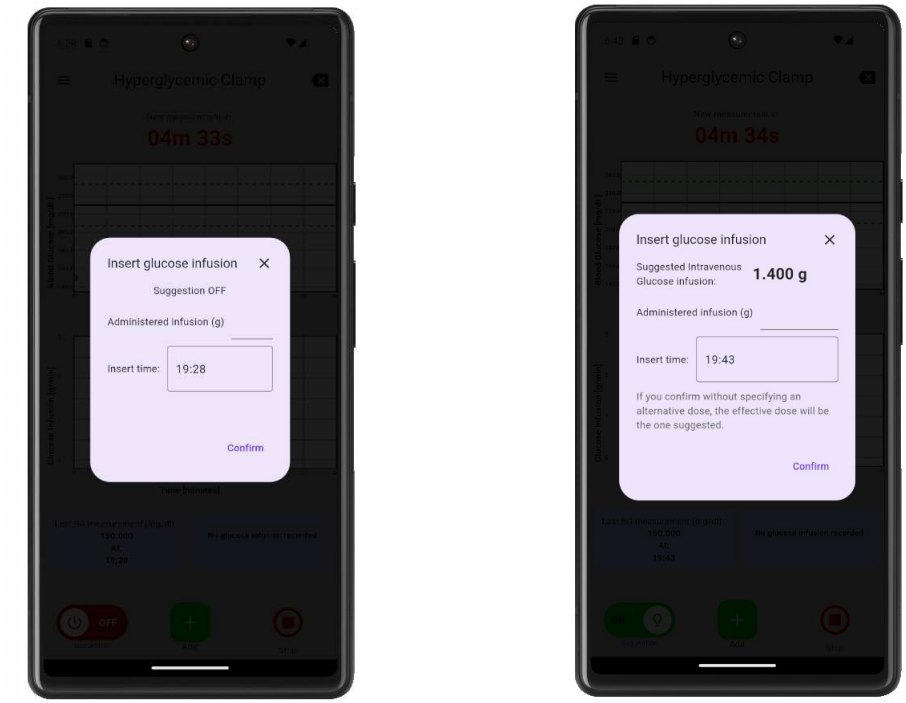
The experiment continues for its designated duration, typically lasting between 180 to 240 minutes. In Fig. 6.11, we can observe a simulation of a hyperglycemic clamp conducted using Glucas 2.0 (for convienence, the simulation is abbreviated).

In Figures 6.12 and 6.13, we can observe brief simulations of a euglycemic and hypoglycemic clamp, respectively.

By selecting the hamburger icon located in the top-left corner of the screen, the drawer will slide out, revealing the patient's data collected during the onboarding page (fig.6.14).

A strong feature of Glucas 2.0 is its ability to allow clinical researchers to conduct more than one experiment simultaneously. By utilizing the "back" arrow located in the top-right corner of the screen (Fig.6.15a), the user can return to the home screen (Fig.6.15b). From there, the user can select any ongoing experiment from the list (6.15c), effectively allowing experiments to run concurrently in the background. This ensures that timers, data points, and all other features remain accessible and operational. The mobile application is designed to be robust against crashes or closures. In the event of such occurrences, when the application is reopened, the experiment will seamlessly continue from where it left off. This ensures uninterrupted progress and data integrity throughout the experiment.

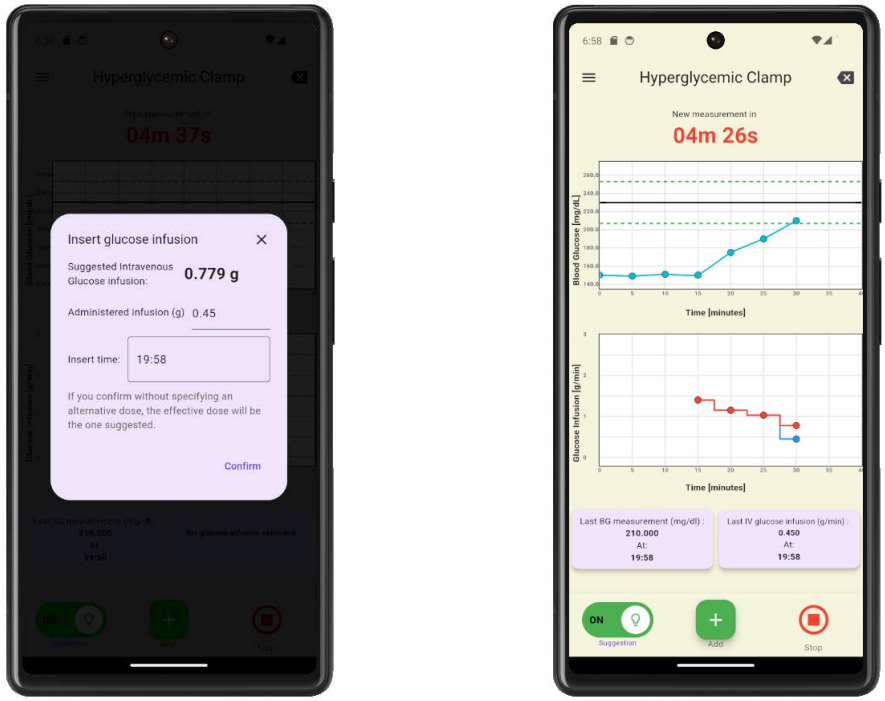
Once the experiment ends, the user can stop it with the "stop" red button



(a) (b)

Figure 6.8: a) Alert dialog for entering the infusion in case of suggestions turned off. b) Alert dialog for entering the infusion in case of suggestions turned on.

Figure 6.9: Experiment (hyperglycemic clamp).



(a) (b)

Figure 6.10: a) Dose dialog with administered infusion different from the suggested one. b) Plot of administered and suggested glucose infusion.

Figure 6.11: Experiment (hyperglycemic clamp).

Figure 6.12: Experiment (euglycemic clamp).

Figure 6.13: Experiment (hypoglycemic clamp).

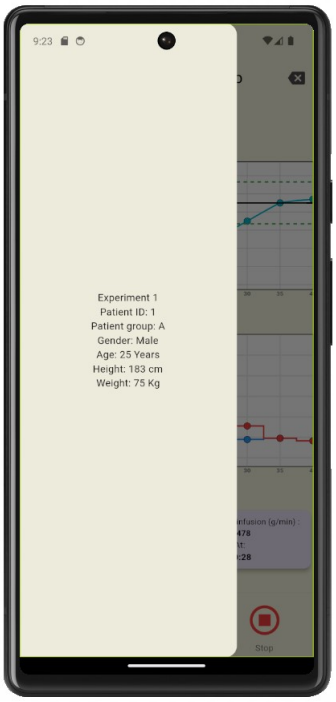
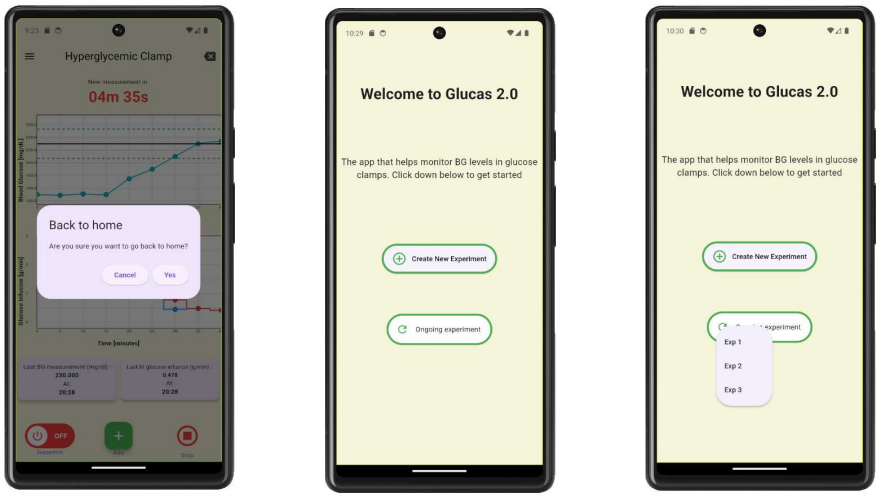


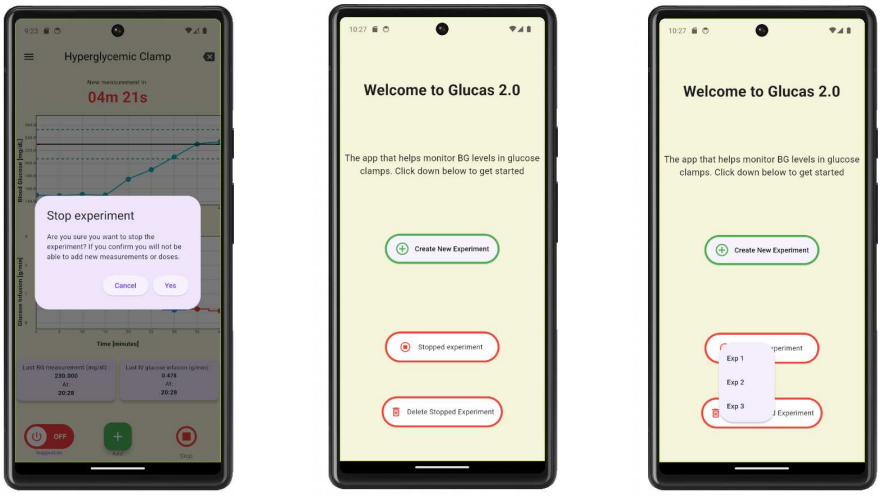
Figure 6.14: Drawer.



(a) (b) (c)

Figure 6.15: a) Back to home confirmation dialog. b) Homepage with "ongoing experiments" button. c) List of ongoing experiments.

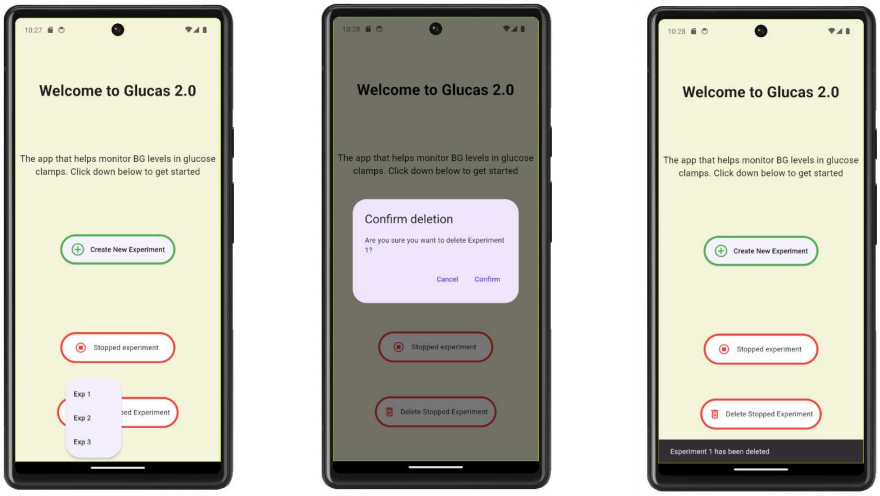
in the bottom-right of the page (Fig.6.16a). Once an experiment is terminated, it will not be possible to add new data. Researchers can still access to the stopped trial by choosing from the list of "stopped experiments" in the home page (Fig.6.16c).



(a) (b) (c)

Figure 6.16: a) Stop experiment confirmation dialog. b) Homepage with "stopped experiments" button. c) List of stopped experiments.

Finally, users can also delete stopped trials with the "Delete Stopped Experiment" button (Fig.6.17). This will erase the entire experiment from the database.



(a) (b) (c)

Figure 6.17: a) List of stopped experiments that can be deleted. b) Confirmation dialog for deletion. c) Confirmation message of deletion.

Chapter 7

Conclusions and Future Developments

The aim of this work was to design closed-loop strategies to assist physicians in modulating glucose infusion rates during glucose clamp experiments. These experiments are crucial for quantifying insulin resistance and β-cell functioning, providing essential insights into metabolic health and potential treatment strategies for conditions such as diabetes.

The first contribution of this thesis involved implementing PID and MPC controllers in the four types of glucose clamps analyzed. We explored both the noise-free scenario, under ideal conditions, and the realistic scenario, with the presence of measurement noise. From the results obtained, we did not find a controller that outperformed the others, except for the MPC with Unmeasured Disturbance Estimator in the hyperglycemic clamp, as discussed in Section 5.3, even if the other two controllers still achieved effective control. For the other types of clamps - euglycemic, hypoglycemic and ISO-IV - all three controller performed excellent and accomplished very similar results. For this reason the choice of a PID controller may be more suitable due to its simplicity compared to a MPC controller.

The second contribution involved the design of a Glucas 2.0, a mobile application developed in the Flutter environment. Glucas 2.0 aims to support clinical researchers in hyperglycemic, euglycemic and hypoglycemic clamp experiments. By providing a blood glucose measurement, the PID controller implemented in the application gives a suggested glucose infusion rate.

As possible future developments, with regard to the control algorithms design, several considerations can be made. First of all, the full UVA/Padova Type 1 Diabetes Simulator could be used to implement the three controllers. In this project the time was limited so we utilized a simplified version of it. Moreover, testing the algorithms on healthy patients instead of diabetic ones could give meaningful insights in the analysis of insulin resistance and β-cell functioning. Another possible improvement could be to modify insulinization in euglycemic

and hypoglycemic clamps, to speed up the process of lowering BG levels, leading to a more in-depth evaluation of the performance of PID and MPC controllers in these glucose clamps. Furthermore, conducting randomized control trials to validate the proposed control strategies on real patients is essential. This step would help validate the findings obtained through in-silico simulations and provide a clearer understanding of the effectiveness of the controllers in clinical settings.

The primary future development for Glucas 2.0 would involve integrating the MPC controller into the application. Despite achieving similar performance overall, the MPC controller in the hyperglycemic clamp demonstrated superior performance. In terms of app functionality, several additions could enhance user experience. For instance, enabling physicians to modify data points inserted into the application would provide greater flexibility and customization. Additionally, incorporating a feature for adding notes could be valuable for annotating and contextualizing data, enhancing the utility of the application for healthcare professionals. Finally, improving the user interface design to enhance aesthetics and usability could be considered.

Bibliography

[1] J. Pavan, C. Dalla Man, D. Herzig, L Bally, S. Del Favero *Gluclas: A software for computer-aided modulation of glucose infusion in glucose clamp experiments* https://www.sciencedirect.com/science/article/pii/S0169260722004850

[2] J. Pavan, D. Herzig, A.Tripyla, C. Dalla Man, L Bally, S. Del Favero *Clinical evaluation of a decision support system for glucose infusion in hypoglycaemic clamp experiments* https://dom-pubs.onlinelibrary.wiley.com/doi/10.1111/dom.15265?af=R

[3] C. Dalla Man, R.A. Rizza, C. Cobelli *Meal simulation model of the glucose-insulin system* https://pubmed.ncbi.nlm.nih.gov/17926672/

[4] R.A. DeFronzo, J.D. Tobin, R. Andres *Glucose clamp technique: a method for quantifying insulin secretion and resistance* https://pubmed.ncbi.nlm.nih.gov/382871/

[5] D. Elahi *In praise of the hyperglycemic clamp. A method for assessment of beta-cell sensitivity and insulin resistance* https://pubmed.ncbi.nlm.nih.gov/8742583/

[6] L. Sjaarda, S. Lee, H. Tfayli, F. Bacha, M. Bertolet, S. Arslanian *Measuring β-cell function relative to insulin sensitivity in youth: does the hyperglycemic clamp suffice?* https://pubmed.ncbi.nlm.nih.gov/23275361/

[7] E. Bonora, P. Moghetti, C. Zancanaro, M.Cigolini, M. Querena, V. Cacciatori, A. Corgnati, M. Muggeo *Estimates of in vivo insulin action in man: comparison of insulin tolerance tests with euglycemic and hyperglycemic glucose clamp studies* https://pubmed.ncbi.nlm.nih.gov/2645308/

[8] C. S. Tam, W. Xie, W. D. Johnson, W. T. Cefalu, L. M. Redman, E. Ravussin *Defining insulin resistance from hyperinsulinemic-euglycemic clamps* https://pubmed.ncbi.nlm.nih.gov/22511259/

[9] T. W Fabricius et. al. *Hyperinsulinaemic–hypoglycaemic glucose clamps in human research: a systematic review of the literature* https://pubmed.ncbi.nlm.nih.gov/33566134/

[10] L. Leelarathna, S. A. Little, E. Walkinshaw, H. Kai Tan, A. Lubina-Solomon, K. Kumareswaran, A. P. Lane, T. Chadwick, S. M. Marshall, J. Speight, D. Flanagan, S. R. Heller, J. A.M. Shaw, M. L. Evans *Restoration of Self-Awareness of Hypoglycemia in Adults With Long-Standing Type 1 Diabetes: Hyperinsulinemic-hypoglycemic clamp substudy results from the HypoCOMPaSS trial* https://diabetesjournals.org/care/article/36/12/4063/33202/Restoration-of-Self-Awareness-of-Hypoglycemia-in

[11] C. Benesch et.al. *How to Assess the Quality of Glucose Clamps? Evaluation of Clamps Performed With ClampArt, a Novel Automated Clamp Device* https://journals.sagepub.com/doi/full/10.1177/1932296815576957

[12] T. Heise, T. R. Pieber *Towards peakless, reproducible and long-acting insulins. An assessment of the basal analogues based on isoglycaemic clamp studies* https://dom-pubs.onlinelibrary.wiley.com/doi/full/10.1111/j.1463-1326.2007.00756.x?casa_token=59ZlRixUAsgAAAAA%3A4lbPPgxKLC6jGnhYXLadtucAxISuNlZW13yuW84Fruz6t7McYpoaB4lgVN3uGwAvbp3cRv5L0DhPRiPt

[13] T. Heise, E. Zijlstra, L. Nosek, S. Heckermann, L. Plum-Mörschel, T. Forst *Euglycaemic glucose clamp: what it can and cannot do, and how to do it* https://dom-pubs.onlinelibrary.wiley.com/doi/full/10.1111/dom.12703?casa_token=Dq6mLeyXEogAAAAA%3A-iAVPabN8K86NcRB45hfYxnQwK4YcrMSBWLiRTxS35R4A0EaX5yNAGSi0ow5WxMCJniGY_mQjxXbLL_6

[14] C. Dalla Man, F. Micheletto, D. Lv, M. Breton, B. Kovatchev, C. Cobelli *The UVA/PADOVA Type 1 Diabetes Simulator: New Features* https://pubmed.ncbi.nlm.nih.gov/29451021/

[15] R. Visentin, E. Campos-Náñez, M. Schiavon, D. Lv, M. Vettoretti, M. Breton, B. P Kovatchev, C. Dalla Man, C. Cobelli *The UVA/Padova Type 1 Diabetes Simulator Goes From Single Meal to Single Day* https://pubmed.ncbi.nlm.nih.gov/29451021/

[16] L. Leelarathna, S. A. Little, E. Walkinshaw, H. Kai Tan, A. Lubina-Solomon, K. Kumareswaran, A. P. Lane, T. Chadwick, S. M. Marshall, J. Speight, D. Flanagan, S. R. Heller, J. A.M. Shaw, M. L. Evans *The Padova Type 2 Diabetes Simulator from Triple-Tracer Single-Meal Studies: In Silico Trials Also Possible in Rare but Not-So-Rare Individuals* https://pubmed.ncbi.nlm.nih.gov/32324063/

[17] R. Visentin, C. Cobelli, C. Dalla Man *The Padova Type 2 Diabetes Simulator from Triple-Tracer Single-Meal Studies: In Silico Trials Also Possible in Rare but Not-So-Rare Individuals* https://diabetesjournals.org/care/article/36/12/4063/33202/Restoration-of-Self-Awareness-of-Hypoglycemia-in

[18] J. J. Burmeister, M. A. Arnold *Accuracy of the Ysi Stat Plus Analyzer for Glucose and Lactate* https://www.tandfonline.com/doi/abs/10.1080/00032719508001119

[19] B. A. Aulinger, D. A. D'alessio *Assessment of the incretin effect in healthy subjects: concordance between clamp and OGTT methods* https://journals.physiology.org/doi/abs/10.1152/ajpendo.00104.2022#:~:text=Different%20estimates%20of%20the%20incretin,the%20OGTT%2Fisoglycemic%20infusion%20method.

[20] M. Hompesch, K. Rave *An Analysis of How to Measure Glucose during Glucose Clamps: Are Glucose Meters Ready for Research?* https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2769796/

[21] E. Bartoli, G.P. Fra, G.P. Carnevale Schianca *The oral glucose tolerance test (OGTT) revisited* https://www.sciencedirect.com/science/article/pii/S0953620510001482

[22] A. De Gaetano, S. Panunzi, A. Matone, A. Samson, J. Vrbikova, B. Bendlova, G. Pacini *Routine OGTT: A Robust Model Including Incretin Effect for Precise Identification of Insulin Sensitivity and Secretion in a Single Individual* https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0070875

[23] W. M. Bayliss, E. H. Starling *The mechanism of pancreatic secretion* https://pubmed.ncbi.nlm.nih.gov/16992627/

[24] J. F. Rehfeld *The origin and understanding of the incretin concept* https://www.researchgate.net/publication/326418263_The_Origin_and_Understanding_of_the_Incretin_Concept

[25] J. B. Lee, E. Dassau, R. Gondhalekar, D. E. Seborg, J. E. Pinsker, F. J. Doyle *Enhanced Model Predictive Control (eMPC) Strategy for Automated Glucose Control* https://pubs.acs.org/doi/10.1021/acs.iecr.6b02718

[26] E. Matamoros-Alcivar et.al. *Implementation of MPC and PID Control Algorithms to the Artificial Pancreas for Diabetes Mellitus Type 1* https://ieeexplore.ieee.org/document/9690529

[27] L. M. Huyett, . Dassau, H. C. Zisser, F. J. Doyle *Design and Evaluation of a Robust PID Controller for a Fully Implantable Artificial Pancreas* https://www.researchgate.net/publication/326418263_The_Origin_and_Understanding_of_the_Incretin_Concept

[28] MathWorks *MathWorks. "Simulink by mathworks."* https://it.mathworks.com/products/simulink.html

[29] M. L. Darby, M. Nikolaou *Mpc: Current practice and challenges* https://www.sciencedirect.com/science/article/pii/S0967066111002528

[30] T. Antonio *"Guida flutter: Creare app mobile multipiattaforma* https://www.html.it/guide/flutter-creare-app-mobile-multipiattaforma/

[31] P. Mridul Paul. *The widget-based architecture: Building blocks of a flutter app (part 1)* https://medium.com/@mridulpaul2000/the-widget-based-architecture-building-blocks-of-a-flutter-app-part-1-1e9635e5cf1d

[32] S. Del Favero *Control of biological systems 2022-2023* https://stem.elearning.unipd.it/course/view.php?id=4939

List of Figures

1.1 Common protocol of glycemic clamp experiments including possible outcomes. 4
1.2 Glucose response for OGTT vs ISO-IV, taken from [19]. 5
1.3 Insulin response for OGTT vs ISO-IV , taken from [19]. 5

2.1 General closed-loop controller architecture, taken from [32]. . . . 7
2.2 General PID controller architecture, taken from [32]. 9
2.3 MPC for reference tracking controller scheme, taken from [32]. . 11
2.4 Discrete-time system, taken from [32]. 11
2.5 Monocompartimental model for MPC reasoning. 12
2.6 Monocompartimental model for MPC reasoning in euglycemic and hypoglycemic clamps. 12
2.7 Full increment velocity form control scheme, taken from [32]. . . 15
2.8 Unmeasured Disturbance Estimator control scheme, taken from [32]. 16

3.1 Scheme of T1D model proposed by Dalla Man et al. [3]. 19
3.2 Simulink scheme of the simplified version of the T1D model proposed by Dalla Man et al. [3]. 20
3.3 Simulink scheme of the closed loop PID controller. 21
3.4 Zoom inside the PID block 21
3.5 MATLAB Function to calculate the GIR with PID controller. . . 22
3.6 Simulink scheme of the closed loop MPC with FIVF controller. . 23
3.7 Zoom inside the MPC with FIVF block. 23
3.8 Zoom inside the block used to access to the state of the system. . 23
3.9 MATLAB Function to calculate the GIR with MPC with FIVF controller. 24
3.10 Simulink scheme of the closed loop MPC with FIVF controller with look ahead. 25
3.11 MATLAB Function to calculate the GIR with MPC with FIVF controller with look ahead. 25
3.12 Tank level problem. 26
3.13 Tank level control: MPC FIVF with and without look ahead. . . 26

3.14 Simulink scheme of the closed loop MPC with UDE controller with look ahead. 27
3.15 Zoom inside the MPC with UDE block. 27
3.16 Zoom inside the Kalman filter block used to estimate the state. . 27
3.17 MATLAB Function to calculate the GIR with MPC with UDE controller. 28
3.18 MATLAB Function to calculate the estimate of the augmented state by using the Kalman filter. 29
3.19 Reference profile for an hyperglycemic clamp (patient #001). . . 30
3.20 Reference profile for an euglycemic clamp (patient #001). 31
3.21 Reference profile for an Hypoglycemic clamp (patient #001). . . 32
3.22 OGTT response of patient#001 at 40g of glucose, no bolus of subcutaneous insulin administered. 33
3.23 OGTT response of patient#001 at 75g of glucose, no bolus of subcutaneous insulin administered. 33
3.24 OGTT response of patient#001 at 40g of glucose, bolus of subcutaneous insulin administered. 33
3.25 OGTT response of patient#001 at 40g of glucose, bolus of subcutaneous insulin administered, zoom. 33
3.26 OGTT response of patient#001 at 75g of glucose, bolus of subcutaneous insulin administered. 34
3.27 OGTT response of patient#001 at 75g of glucose, bolus of subcutaneous insulin administered, zoom. 34
3.28 OGTT response of the 100 patients at 40g of glucose. 34
3.29 OGTT response of the 100 patients at 75g of glucose. 34

4.1 Evolution of BG level in a hyperglycemic clamp using the PID controller, patient#001. 37
4.2 Spaghetti plot of 100 patients in a hyperglycemic clamp using PID controller. 37
4.3 Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using PID controller. 37
4.4 Evolution of BG level in a euglycemic clamp using the PID controller, patient#001. 39
4.5 Spaghetti plot of 100 patients in a euglycemic clamp using PID controller. 40
4.6 Mean ± std confidence interval of 100 patients in a euglycemic clamp using PID controller. 40
4.7 Evolution of BG level in a hypoglycemic clamp using the PID controller, patient#001. 41
4.8 Spaghetti plot of 100 patients in a hypoglycemic clamp using PID controller. 42
4.9 Mean ± std confidence interval of 100 patients in a hypoglycemic clamp using PID controller. 42
4.10 Evolution of BG level in a ISO-IV clamp (OGTT 40g) using the PID controller, first three patients. 43

4.11 Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the PID controller on population of 100 patients. 44
4.12 Evolution of BG level in a ISO-IV clamp (OGTT 75g) using the PID controller, first three patients. 44
4.13 Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the PID controller on population of 100 patients. 45
4.14 Evolution of BG level in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form, patient#001. . . . 47
4.15 Spaghetti plot of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form (no look ahead implemented). 48
4.16 Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form (no look ahead implemented). 48
4.17 Spaghetti plot of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated). 48
4.18 Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated). 48
4.19 Evolution of BG level in a euglycemic clamp using the MPC controller with Full Increment Velocity Form, patient#001. . . . 50
4.20 Spaghetti plot of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form (no look ahead implemented). 51
4.21 Mean ± std confidence interval of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form (no look ahead implemented). 51
4.22 Spaghetti plot of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated). 51
4.23 Mean ± std confidence interval of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated). 51
4.24 Evolution of BG level in a hypoglycemic clamp using the MPC controller with Full Increment Velocity Form, patient#001. . . . 53
4.25 Spaghetti plot of 100 patients in a hypoglycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated). 54
4.26 Mean ± std confidence interval of 100 patients in a hypolycemic clamp using the MPC controller with Full Increment Velocity Form (look ahead activated). 54
4.27 Evolution of BG levels in a ISO-IV clamp (OGTT 40g) using the MPC controller with Full Increment Velocity Form, first three patients. 55

4.28 Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the MPC controller with Full Increment Velocity Form on population of 100 patients. 56
4.29 Evolution of BG levels in a ISO-IV clamp (OGTT 75g) using the MPC controller with Full Increment Velocity Form, first three patients. 57
4.30 Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the MPC controller with Full Increment Velocity Form on population of 100 patients. 57
4.31 Evolution of BG level in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, patient#001. 60
4.32 Spaghetti plot of 100 patients in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method. 60
4.33 Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method. 60
4.34 Evolution of BG level in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, patient#001. 62
4.35 Spaghetti plot of 100 patients in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method. 63
4.36 Mean ± std confidence interval of 100 patients in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method. 63
4.37 Evolution of BG level in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, patient#001. 64
4.38 Spaghetti plot of 100 patients in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method. 65
4.39 Mean ± std confidence interval of 100 patients in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method. 65
4.40 Evolution of BG level in a ISO-IV clamp (OGTT 40g) using the MPC controller with Unmeasured Disturbance Estimator approach, first three patients. 66
4.41 Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the MPC controller with Unmeasured Disturbance Estimator approach on population of 100 patients. 67
4.42 Evolution of BG level in a ISO-IV clamp (OGTT 75g) using the MPC controller with Unmeasured Disturbance Estimator approach, first three patients. 67
4.43 Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the MPC controller with Unmeasured Disturbance Estimator approach on population of 100 patients. 68

4.44 Evolution of BG level in a hyperglycemic clamp using the PID controller on patient#001, measurement noise present. 71
4.45 Spaghetti plot of 100 patients in a hyperglycemic clamp using PID controller, measurement noise present. 71
4.46 Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the PID controller, measurement noise present. . . . 71
4.47 Evolution of BG level in a euglycemic clamp using the PID controller on patient#001, measurement noise present. 73
4.48 Spaghetti plot of 100 patients in a euglycemic clamp using PID controller, measurement noise present. 74
4.49 Mean ± std confidence interval of 100 patients in a euglycemic clamp using PID controller, measurement noise present. 74
4.50 Evolution of BG level in a hypoglycemic clamp using the PID controller on patient#001, measurement noise present. 75
4.51 Spaghetti plot of 100 patients in a hypoglycemic clamp using PID controller, measurement noise present. 76
4.52 Mean ± std confidence interval of 100 patients in a hypoglycemic clamp using PID controller, measurement noise present. 76
4.53 Evolution of BG level in a ISO-IV clamp (OGTT 40g) using the PID controller on first three patients, measurement noise present. 77
4.54 Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the PID controller on population of 100 patients, measurement noise present. 78
4.55 Evolution of BG level in a ISO-IV clamp (OGTT 75g) using the PID controller on first three patients, measurement noise present. 78
4.56 Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the PID controller on population of 100 patients, measurement noise present. 79
4.57 Evolution of BG level in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form on patient#001, measurement noise present. 80
4.58 Spaghetti plot of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present. 81
4.59 Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present. 81
4.60 Evolution of BG level in a euglycemic clamp using the MPC controller with Full Increment Velocity Form on patient#001, measurement noise present. 82
4.61 Spaghetti plot of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present. 83
4.62 Mean ± std confidence interval of 100 patients in a euglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present. 83

4.63 Evolution of BG level in a hypoglycemic clamp using the MPC controller with Full Increment Velocity Form on patient#001, measurement noise present. 84
4.64 Spaghetti plot of 100 patients in a hypoglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present. 85
4.65 Mean ± std confidence interval of 100 patients in a hypoglycemic clamp using the MPC controller with Full Increment Velocity Form, measurement noise present. 85
4.66 Evolution of BG level in a ISO-IV clamp (OGTT 40g) using the MPC controller with Full Increment Velocity Form on first three patients, measurement noise present. 86
4.67 Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the MPC controller with Full Increment Velocity Form on population of 100 patients, measurement noise present. 87
4.68 Evolution of BG level in a ISO-IV clamp (OGTT 75g) using the MPC controller with Full Increment Velocity Form on first three patients, measurement noise present. 87
4.69 Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the MPC controller with Full Increment Velocity Form on population of 100 patients, measurement noise present. 88
4.70 Evolution of BG level in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method on patient#001, measurement noise present. 89
4.71 Spaghetti plot of 100 patients in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present. 90
4.72 Mean ± std confidence interval of 100 patients in a hyperglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present. 90
4.73 Evolution of BG level in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method on patient#001, measurement noise present. 91
4.74 Spaghetti plot of 100 patients in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present. 92
4.75 Mean ± std confidence interval of 100 patients in a euglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present. 92
4.76 Evolution of BG level in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method on patient#001, measurement noise present. 93
4.77 Spaghetti plot of 100 patients in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present. 94

4.78 Mean ± std confidence interval of 100 patients in a hypoglycemic clamp using the MPC controller with Unmeasured Disturbance Estimator method, measurement noise present. 94
4.79 Evolution of BG level in a ISO-IV clamp (OGTT 40g) using the MPC controller with Unmeasured Disturbance Estimator approach on first three patients, measurement noise present. 95
4.80 Mean relative tracking error for ISO-IV clamp (OGTT 40g) using the MPC controller with Unmeasured Disturbance Estimator approach on population of 100 patients, measurement noise present. 96
4.81 Evolution of BG level in a ISO-IV clamp (OGTT 75g) using the MPC controller with Unmeasured Disturbance Estimator approach on first three patients, measurement noise present. 96
4.82 Mean relative tracking error for ISO-IV clamp (OGTT 75g) using the MPC controller with Unmeasured Disturbance Estimator approach on population of 100 patients, measurement noise present. 97

5.1 Comparison between PID, MPC FIVF and MPC UDE controllers on population, hyperglycemic clamp. Zoom between minute 60 and 150. 100
5.2 Comparison between PID, MPC FIVF and MPC UDE controllers on population, euglycemic clamp. 101
5.3 Comparison between PID, MPC FIVF and MPC UDE controllers on population, hypoglycemic clamp. 102
5.4 Comparison between PID, MPC FIVF and MPC UDE controllers on population, relative tracking error, ISO-IV clamp (OGTT 40g).103
5.5 Comparison between PID, MPC FIVF and MPC UDE controllers on population, relative tracking error, ISO-IV clamp (OGTT 75g).104
5.6 Comparison between PID, MPC FIVF and MPC UDE controllers on population in a hyperglycemic clamp, measurement noise present.105
5.7 Comparison between PID, MPC FIVF and MPC UDE controllers on population in a euglycemic clamp, measurement noise present. 106
5.8 Comparison between PID, MPC FIVF and MPC UDE controllers on population in a hypoglycemic clamp, measurement noise present.107
5.9 Comparison between PID, MPC FIVF and MPC UDE controllers on population, relative tracking error, ISO-IV clamp (OGTT 40g), measurement noise present. 108
5.10 Comparison between PID, MPC FIVF and MPC UDE controllers on population, relative tracking error, ISO-IV clamp (OGTT 75g), measurement noise present. 108

6.1 Glucas 2.0 navigation flow chart. 112
6.2 E-R model on which the App database is based. 114
6.3 Home. 115
6.4 Onboarding. 115
6.5 Landing. 115

6.6 a) Measurement request. b) Measurement insertion. c) Plot of the first measurement. d) Plot of three measurements. 116
6.7 a) Add measurement and doses button. b) Activate suggestion of PID controller. 116
6.8 a) Alert dialog for entering the infusion in case of suggestions turned off. b) Alert dialog for entering the infusion in case of suggestions turned on. 118
6.9 Experiment (hyperglycemic clamp). 118
6.10 a) Dose dialog with administered infusion different from the suggested one. b) Plot of administered and suggested glucose infusion. 119
6.11 Experiment (hyperglycemic clamp). 119
6.12 Experiment (euglycemic clamp). 120
6.13 Experiment (hypoglycemic clamp). 120
6.14 Drawer. 121
6.15 a) Back to home confirmation dialog. b) Homepage with "ongoing experiments" button. c) List of ongoing experiments. 121
6.16 a) Stop experiment confirmation dialog. b) Homepage with "stopped experiments" button. c) List of stopped experiments. 122
6.17 a) List of stopped experiments that can be deleted. b) Confirmation dialog for deletion. c) Confirmation message of deletion. . . 123

List of Tables

2.1 Model parameters. 13

4.1 PID controller parameters for hyperglycemic clamp on patient#001 and on population. 38
4.2 PID controller parameters for euglycemic clamp on patient#001 and on population. 40
4.3 PID controller parameters for hypoglycemic clamp on patient#001 and on population. 42
4.4 PID controller parameters for ISO-IV clamp on population. . . . 45
4.5 Mean Absolute Percentage Error (MAPE) of PID controller in a ISO-IV clamp, response to OGTT 40g and OGTT 75g. 45
4.6 MPC with Full Increment Velocity Form parameters in a hyperglycemic clamp. With and without look ahead feature, on patient#001 and on population. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness. 49
4.7 Mean Absolute Percentage Error (MAPE) of MPC controller with Full Increment Velocity Form in a hyperglycemic clamp, with and without look ahead feature. 49
4.8 MPC with Full Increment Velocity Form parameters in a euglycemic clamp. With and without look ahead feature, on patient#001 and on population. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness. 52
4.9 Mean Absolute Percentage Error (MAPE) of MPC controller with Full Increment Velocity Form in a euglycemic clamp, with and without look ahead feature. 52
4.10 MPC with Full Increment Velocity Form parameters in a hypoglycemic clamp. With look ahead feature, on patient#001 and on population. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness. 54

4.11 MPC with Full Increment Velocity Form parameters in a ISO-IV clamp, on OGTT 40g and OGTT 75g. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness. 58
4.12 Mean Absolute Percentage Error (MAPE) of MPC controller with Full Increment Velocity in a ISO-IV clamp. 58
4.13 Parameters of the MPC controller with the Unmeasured Disturbance Estimator method, hyperglycemic clamp. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively. 61
4.14 Parameters of the MPC controller with the Unmeasured Disturbance Estimator method, euglycemic clamp. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively. 63
4.15 Parameters of the MPC controller with the Unmeasured Disturbance Estimator method, hypoglycemic clamp. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively. 65
4.16 Parameters of the MPC controller with the Unmeasured Disturbance Estimator method, ISO-IV clamp. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively. 68
4.17 Mean Absolute Percentage Error (MAPE) of MPC controller with Unmeasured Disturbance Estimator method in a ISO-IV clamp. 69
4.18 PID controller parameters for hyperglycemic clamp on patient#001 and on population, measurement noise present. 72
4.19 PID controller parameters for euglycemic clamp on patient#001 and on population, measurement noise present. 74
4.20 PID controller parameters for hypoglycemic clamp on patient#001 and on population, measurement noise present. 76
4.21 PID controller parameters for ISO-IV clamp on population, measurement noise present. 79
4.22 Mean Absolute Percentage Error (MAPE) of PID controller in a ISO-IV clamp, response to OGTT 40g and OGTT 75g, measurement noise present. 79

4.23 MPC with Full Increment Velocity Form parameters in a hyperglycemic clamp on patient#001 and on population, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness. 81
4.24 MPC with Full Increment Velocity Form parameters in a euglycemic clamp on patient#001 and on population, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness. 83
4.25 MPC with Full Increment Velocity Form parameters in a hypoglycemic clamp on patient#001 and on population, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness. 85
4.26 MPC with Full Increment Velocity Form parameters in a ISO-IV clamp, on OGTT 40g and OGTT 75g, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R the parameters regulating control aggressiveness. 88
4.27 Mean Absolute Percentage Error (MAPE) of MPC controller with Full Increment Velocity Form in a ISO-IV clamp, response to OGTT 40g and OGTT 75g, measurement noise present. . . . 88
4.28 Parameters of the MPC controller with the Unmeasured Disturbance Estimator method in a hyperglycemic clamp, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter , respectively. 90
4.29 Parameters of the MPC controller with the Unmeasured Disturbance Estimator method in a euglycemic clamp, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively. 92
4.30 Parameters of the MPC controller with the Unmeasured Disturbance Estimator method in a hypoglycemic clamp, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively. 94
4.31 MAPE for hyper, eu and hypoglycemic clamps using the MPC controller with Unmeasured Disturbance Estimator approach in presence of measurement noise 94

4.32 Parameters of the MPC controller with the Unmeasured Disturbance Estimator method in a ISO-IV clamp, measurement noise present. PH is the Prediction Horizon of the MPC controller, Q and R regulate control aggressiveness, M_u and C_u are the parameters used to augment the model, Q_k and R_k are the covariances of the process and of the measurement noise of the Kalman filter, respectively. 97
4.33 Mean Absolute Percentage Error (MAPE) of MPC controller with Unmeasured Disturbance Estimator method in a ISO-IV clamp, measurement noise present. 98

5.1 Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers, hyperglycemic clamp. 100
5.2 Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers, euglycemic clamp. . 100
5.3 Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers, hypoglycemic clamp. 101
5.4 Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers, ISO-IV clamp. . . . 103
5.5 Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers in a hyperglycemic clamp, measurement noise present. 105
5.6 Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers in a euglycemic clamp, measurement noise present. 106
5.7 Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers in a hypoglycemic clamp, measurement noise present. 107
5.8 Mean Absolute Percentage Error (MAPE) comparison between PID, MPC FIVF and MPC UDE controllers in a ISO-IV clamp, measurement noise present. 107

6.1 PID controller parameters used for Glucas 2.0. 110