

M.Sc.  
Master of Science in Engineering

# Optimization based Control Algorithms for the Artificial Pancreas and Insulin Dosing Systems

Haotian Gao(s192232)

Kongens Lyngby 2021



**DTU Compute**  
**Department of Applied Mathematics and Computer Science**  
**Technical University of Denmark**

Matematiktorvet  
Building 303B  
2800 Kongens Lyngby, Denmark  
Phone +45 4525 3031  
[compute@compute.dtu.dk](mailto:compute@compute.dtu.dk)  
[www.compute.dtu.dk](http://www.compute.dtu.dk)

# Summary

---

Since  $\beta$  cells cannot produce insulin, patients with type 1 diabetes mellitus(T1DM) have to infuse exogenous insulin to keep their blood glucose levels within a normal range. As a closed-loop control system, the artificial pancreas(AP) monitors the blood glucose concentration in real time, and infuses the calculated exogenous insulin through the insulin pump. In this thesis, we focus on the design and implementation of linear model predictive control(LMPC) and nonlinear model predictive control(NMPC) algorithm for AP problems. In addition to a single-hormone system that controls blood glucose by infusion of insulin, a multiple-hormone system that simultaneously infuses insulin and glucagon is also considered. In the design of the objective function of NMPC, in order to obtain a better control effect, the penalty form of the manipulated variables is discussed in detail. Both LMPC and NMPC algorithms are implemented using the framework of CasADI[And+18]. The Matlab and C++ versions of NMPC are implemented, and the Matlab version of LMPC is implemented. The extended Hovorka model is used as the simulation model, and the extended MVP model is used as the control model. The method of parameter identification and experimental design for the control model are introduced. The closed-loop simulation test is designed to verify the effectiveness and robustness of the implemented algorithm. Through the test results, the designed and implemented LMPC and NMPC algorithms of the single-hormone system and the multi-hormone system can well control blood glucose within the normal range, and the calculated infusion dose also meets the desired control strategy. In addition, stable and robust control effects can be maintained for different patients.



# Preface

---

This master thesis was prepared at the department of Applied Mathematics and Computer Science at the Technical University of Denmark in fulfillment of the requirements for acquiring a Master degree in Electrical Engineering.

, June 22, 2021

Haojun Guo



# Acknowledgements

---

First of all, I would like to thank my supervisor John Bagterp Jørgensen for his support, help and encouragement throughout the project.

I would like to thank Asbjørn Thode Reenberg and Morten Ryberg Wahlgreen. They always help me when I encounter difficulties.

Finally, I would like to thank my family and girlfriend for their support.



# Abbreviations

---

AP	Artificial Pancreas
API	Application Programming Interface
BG	Blood Glucose
CDEKF	Continuous Discrete Extended Kalman Filter
CGM	Continuous Glucose Monitoring
CSII	Continuous Subcutaneous Insulin Infusion
DM	Diabetes Mellitus
ERK	Explicit Runge Kutta
IDF	International Diabetes Federation
IPOPT	Interior Point OPTimizer
LS	Least Square
LMPC	Linear Model Predictive Control
MDI	Multiple daily injections
MLE	Maximum Likelihood Estimation
MPC	Model Predictive Control
MVP	Medtronic Virtual Patient
NLP	Nonlinear Programming
NMPC	Nonlinear Model Predictive Control
OCP	Optimal Control Problem
ODE	Ordinary Differential Equations
PID	Proportional Integral Derivative
QP	Quadratic Programming
RMSE	Root Mean Square Error
SDE	Stochastic Differential Equations
T1DM	Type 1 Diabetes Mellitus
T2DM	Type 2 Diabetes Mellitus



# Contents

---

<b>Summary</b>	i
<b>Preface</b>	iii
<b>Acknowledgements</b>	v
<b>Abbreviations</b>	vii
<b>Contents</b>	ix
<b>1 Introduction</b>	1
1.1 Structure of thesis . . . . .	2
<b>2 Mathematical model</b>	5
2.1 The Hovorka Model . . . . .	6
2.1.1 Glucose-insulin system(Single-hormone) . . . . .	6
2.1.2 Glucose-insulin system(Multiple-hormone) . . . . .	8
2.1.3 Meal absorption . . . . .	9
2.1.4 Subcutaneous insulin absorption . . . . .	9
2.1.5 Parameters . . . . .	10
2.1.6 Steady state . . . . .	11
2.2 The MVP model . . . . .	12
2.2.1 Glucose-insulin system(Single-hormone) . . . . .	12
2.2.2 Glucose-insulin system(Multiple-hormone) . . . . .	12
2.2.3 Meal absorption . . . . .	13
2.2.4 Subcutaneous insulin absorption . . . . .	13
2.2.5 Steady state . . . . .	14
2.3 CGM model . . . . .	15
2.4 Exercise model . . . . .	15
2.5 Glucagon model . . . . .	16
2.6 The extended Hovorka model . . . . .	17
2.7 The extended MVP model . . . . .	18
<b>3 Parameter Estimation</b>	19
3.1 Model . . . . .	19

3.2	Numerical integration methods . . . . .	21
3.2.1	Explicit Runge-Kutta 4 methods for ODE . . . . .	21
3.2.2	Euler-Maruyama method for SDE . . . . .	22
3.3	Experimental design of data set for parameter estimation . . . . .	23
3.4	Least squares estimation . . . . .	25
3.5	Maximum likelihood estimation . . . . .	26
3.6	Parameter estimation for closed-loop simulation . . . . .	29
<b>4</b>	<b>Nonlinear Model Predictive Control for artificial pancreas</b>	<b>31</b>
4.1	Mathematical model for NMPC . . . . .	32
4.2	Continuous-discrete extended Kalman filter . . . . .	32
4.2.1	Filtering . . . . .	33
4.2.2	Prediction . . . . .	33
4.3	Optimal control problem formulation . . . . .	34
4.3.1	Objective function . . . . .	34
4.3.2	Discretization of optimal control problem . . . . .	36
4.3.3	Smoothing function of penalty term . . . . .	40
4.4	Implementation of NLP problem in CasADi . . . . .	42
4.4.1	Symbolic framework and function objects . . . . .	42
4.4.2	Mathematical Model . . . . .	45
4.4.3	Self-defined functions . . . . .	49
4.4.4	Declaration of decision variables . . . . .	50
4.4.5	Objective function . . . . .	51
4.4.6	Constraints . . . . .	52
4.4.7	NLP solver . . . . .	53
4.5	Closed-loop simulation test . . . . .	57
4.5.1	Comparison of closed-loop test of different forms for objective function . . . . .	58
4.5.2	Closed-loop simulation test of virtual patient . . . . .	67
<b>5</b>	<b>Linear Model Predictive Control for artificial pancreas</b>	<b>73</b>
5.1	Mathematical model for LMPC . . . . .	74
5.1.1	Linearization . . . . .	74
5.1.2	Discretization . . . . .	75
5.2	Discrete dynamic Kalman filters . . . . .	76
5.2.1	Filtering . . . . .	76
5.2.2	Prediction . . . . .	77
5.3	Optimal control problem formulation . . . . .	78
5.3.1	Objective function . . . . .	78
5.3.2	Discretization of optimal control problem . . . . .	80
5.4	Matrix for constructing the QP problem . . . . .	84
5.5	Closed-loop simulation test . . . . .	89
<b>6</b>	<b>Conclusion</b>	<b>95</b>

<b>A Parameters of 10 virtual patients</b>	<b>99</b>
<b>B Source Code</b>	<b>103</b>
<b>Bibliography</b>	<b>105</b>



# CHAPTER 1

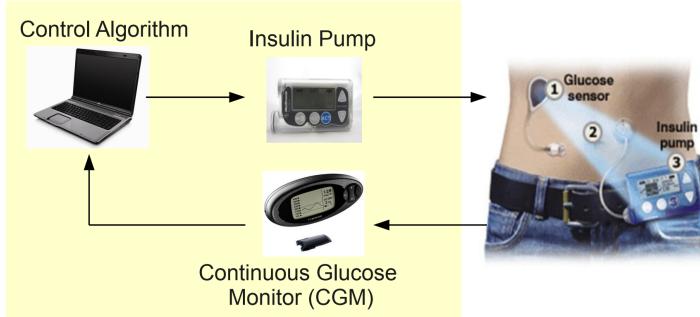
# Introduction

---

Diabetes mellitus(DM) is a group of metabolic diseases characterized by chronic hyperglycemia resulting from defects in insulin secretion, insulin action, or both[KD15]. The International Diabetes Federation (IDF) estimated the 451 million (age 18-99 years) people with diabetes worldwide and the global prevalence were expected to increase to 693 million by 2045[Cho+18]. Type 1 diabetes mellitus(T1DM) and type 2 diabetes mellitus(T2DM) are the two most common types of diabetes. The  $\beta$  cells of patients with T1DM cannot produce insulin due to the destruction of autoimmune response. Therefore, in order to maintain life, the injection of exogenous insulin is necessary for patients with T1DM. T2DM is caused by abnormal  $\beta$  cell function and insulin resistance. Therefore, patients with T2DM generally use drugs that reduce insulin resistance and increase insulin secretion. In order to maintain the normal absorption and utilization of glucose, patients with T1DM maintain normal blood glucose concentration by injecting exogenous insulin. However, in the control process of intensive insulin treatment, incorrect or accidental insulin injection can cause hypoglycemia and weight gain. Injection of low-doses of glucagon can be applied as a method to reduce the risk of hypoglycemia and weight gain[Rei+14]. In addition, only using exogenous insulin as the input, the risk of hypoglycemia caused by exercise cannot be prevented. Therefore, in this thesis, the treatment of the single-hormone system and the multiple-hormone system for patients with T1DM will be considered.

Multiple daily injections (MDI) or continuous subcutaneous insulin infusion (CSII) is applied for infusion of exogenous insulin. As a closed-loop control system, artificial pancreas(AP) includes continuous glucose monitoring (CGM), controller, and insulin pump. CGM can continuously monitor blood glucose in real time and the required insulin is injected through the insulin pump. There are many control algorithms such as model-free algorithms including PID [Wat+11], model-based algorithms including optimal control[SMM09], model predictive control(MPC)[Mag+09],[Boi+11], and so on. In this thesis, the linear model predictive control(LMPC) algorithm and the non-linear model predictive control(NMPC) algorithm of the single-hormone system and the multiple-hormone system will be designed, implemented by using the framework of CasADi and tested by closed-loop simulation. The Matlab version of LMPC is implemented, and the Matlab and C++ versions of NMPC are implemented.

## Artificial Pancreas



**Figure 1.1:** Diagram of an artificial pancreas(AP)[Boi12].

### 1.1 Structure of thesis

The thesis is structured as following

- **Chapter 2** presents the physiological mathematical model used. The physiological model(extended Hovorka model)[Hov+04][Boi12] used for closed-loop simulation and the identifiable physiological model(extended Medtronic Virtual Patient(MVP) model)[Boi+19] used for MPC control are introduced. Both models are divided into a single-hormone system version and a multiple-hormone system version. The parameter distribution of Hovorka model is used to generate virtual patient, which is shown in Table 2.2.
- **Chapter 3** discusses methods and specific implementations of parameter identification for artificial pancreas(AP) problems. The stochastic differential equations(SDE) version of the control model and the parameters to be identified are introduced. Whether in the maximum likelihood estimation method, or in the prediction of estimator and controller of NMPC, the nonlinear model needs to be numerically integrated. We introduced the SDE-based numerical integration method, explicit Euler-Maruyama method with a fixed step size[Mar55] and the ordinary differential equations(ODE)-based numerical integration method, explicit Runge–Kutta 4 method(ERK4) method with a fixed step size. The generation of the data set used for parameter identification is described. Two parameter identification methods, least square(LS) method and maximum likelihood estimation(MLE) are discussed. The parameters of the 10 virtual patients in the closed-loop simulation test are generated by the simulation model and his parameter distribution. The specific parameter values are shown in Table A.1

and A.2 in Appendix A. Maximum likelihood estimation(MLE) is used to verify the effectiveness of parameter identification. The parameters of the control model corresponding to 10 virtual patients can be found in Table 3.5.

- **Chapter 4** proposes a nonlinear model predictive control(NMPC) algorithm applied for artificial pancreas(AP) problems. The SDE version of the nonlinear MVP model mentioned in Chapter 3 is used as the control model, and the continuous-discrete extended Kalman filter(CDEKF) is designed to observe the state of the MVP model through the blood glucose of CGM. In addition, We use the multiple shooting method to convert the optimal control problem into a nonlinear programming(NLP) problem. The objective function and constraints of the optimal control problem will be introduced, in particular, the penalty form of the manipulated variable  $u$  in the objective function is discussed, classified and compared through closed-loop simulation tests to find a better combination of penalty forms. The interior point method(IPOPT[WB06] called by CasADi) is used to solve the nonlinear programming problem. As a numerical integration method, Explicit Runge-Kutta 4(ERK4) with fixed step is used in the prediction of CDEKF and the controller to calculate the objective function in the optimal control problem. The Euler-Maruyama method with a fixed step size is used to simulate the extended Hovorka model. The NMPC algorithm is implemented based on the framework of CasADi. The implemented Matlab version and C++ version of the NMPC algorithm are compared through closed-loop simulation tests. Finally, the effectiveness and robustness of the implemented NMPC algorithm is verified by closed-loop simulation tests on 10 virtual patients.
- **Chapter 5** proposes a linear model predictive control(LMPC) algorithm applied for artificial pancreas problems. LMPC is designed and the Matlab version is implemented. First, we describe the linearization and discretization methods of the nonlinear control model. Discrete dynamic Kalman filters are designed to observe the state of the linear model through the blood glucose of CGM. In addition, the single shooting method is used to convert the optimal control problem into a quadratic programming(QP) problem. How to construct matrix to represent the objective function and constraints of the optimal control problem will be introduced. We use the active set method(qpOASES[Fer+14] called by CasADi) to solve the QP problem. The matrix construction and calculation in LMPC are completed by Matlab and converted to the data type required by the API of CasADi. Finally, the effectiveness and robustness of the implemented LMPC algorithm is verified by closed-loop simulation tests on 10 virtual patients.



# CHAPTER 2

## Mathematical model

There are several physiological models which have been developed to simulate virtual patients with type 1 diabetes. Such as the model developed by Hovorka et al[Hov+04], the Minimal Model developed by Bergman et al[BPC81] and the Medtronic Virtual Patient (MVP) model[Kan+09]. In this thesis, the Hovorka model is used to simulate patients with type 1 diabetes, and an identified model[Boi+19] derived from the MVP model is used as the control model of the model predictive control(MPC). In addition, exercise model, glucagon model and CGM model[BK08] are also introduced and added to Hovorka model and MVP model. The extended Hovorka model is mainly divided into single-hormone model and multiple-hormone model. Since the MPC algorithm implemented in this thesis is a model-based control algorithm, the corresponding extended MVP model is also divided into single-hormone model and multiple-hormone model.

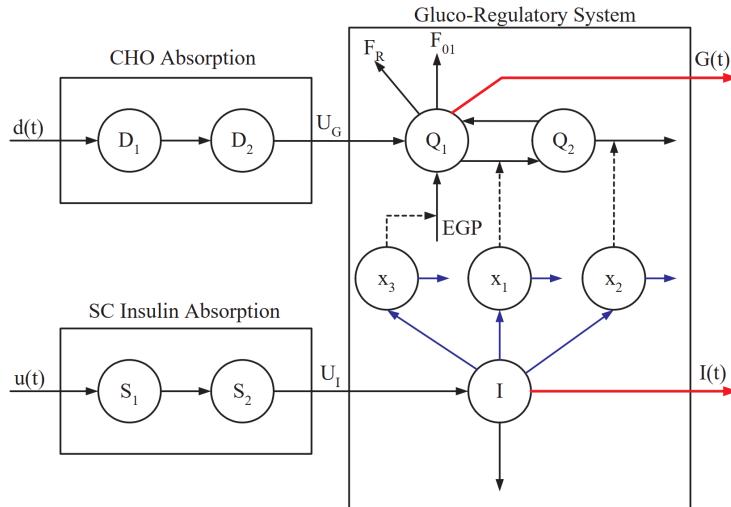
**Table 2.1:** Single-hormone and multiple-hormone model.

	Simulation model
Single-hormone	Hovorka model(single-homone)+CGM
Multiple-hormone	Hovorka(multiple-homone)+Exercise+Glucagon+CGM

	Control model
Single-hormone	MVP(single-homone)+CGM
Multiple-hormone	MVP(multiple-homone)+Glucagon+CGM

## 2.1 The Hovorka Model

There are 10 states in the Hovorka Model, including 6 states for modeling the glucose-insulin system, 2 states for modeling the meal absorption, and 2 other states for modeling the subcutaneous insulin absorption. A diagram of the Hovorka model is showed in Figure 2.1.



**Figure 2.1:** A diagram of the Hovorka model[Boi12].

### 2.1.1 Glucose-insulin system(Single-hormone)

A two-compartment representation is used to model the blood glucose kinetics. There are two states,  $Q_1(t)$  [mmol] and  $Q_2(t)$  [mmol], which indicates the glucose in the blood stream and peripheral tissue respectively.

$$\frac{dQ_1(t)}{dt} = U_G - (F_{01}^c(t) + F_R(t) + x_1(t)Q_1(t)) + k_{12}Q_2(t) + EGP_0(1 - x_3(t)) \quad (2.1a)$$

$$\frac{dQ_2(t)}{dt} = x_1(t)Q_1(t) - (k_{12} + x_2(t))Q_2(t) \quad (2.1b)$$

The output which is the blood glucose concentration  $G(t)$  [mmol/L] is given by

$$y(t) = G(t) = \frac{Q_1(t)}{V_G} \quad (2.2)$$

Where the glucose distribution volume  $V_G$  [L] is dependent on the body weight  $BW$  [kg].

The change of first compartment  $Q_1(t)$  [mmol] depends on:

- $U_G(t)$  [mmol/min] is the gut absorption rate from meals intake.
- $F_{01}^c(t)$  [mmol/min] is the non-insulin-dependent glucose consumed in the central nervous system and is defined as

$$F_{01,c}(t) = \begin{cases} F_{01} & G(t) \geq 4.5\text{mmol/L} \\ \frac{F_{01}G(t)}{4.5} & \text{otherwise} \end{cases} \quad (2.3)$$

- $F_R(t)$  [mmol/min] states the renal glucose excretion in the kidneys and is defined as

$$F_R(t) = \begin{cases} 0.003(G(t) - 9)V_G & G(t) \geq 9\text{mmol/L} \\ 0 & \text{otherwise} \end{cases} \quad (2.4)$$

- $x_1(t)Q_1(t)$  [mmol/min] is the insulin-dependent glucose consumed in the peripheral tissue, and  $x_1(t)$  is the action of insulin on glucose distribution.
- $k_{12}Q_2(t)$  [mmol/min] is glucose transferred from the peripheral tissue to the blood stream, and  $k_{12}$  is the transfer rate constant.
- $EGP_0(1 - x_3(t))$  [mmol/min] is the endogenous glucose produced by liver, and  $x_3(t)$  is the action of insulin on endogenous glucose production.

The change of second compartment  $Q_2(t)$  [mmol] depends on:

- $x_1(t)Q_1(t)$  [mmol/min] is the insulin-dependent glucose consumed in the peripheral tissue, and  $x_1(t)$  is the action of insulin on glucose distribution.
- $k_{12}Q_2(t)$  [mmol/min] is glucose transferred from the peripheral tissue to the blood stream, and  $k_{12}$  is the transfer rate constant.
- $x_2(t)Q_2(t)$  [mmol/min] is the insulin-dependent disposal of glucose, and  $x_2(t)$  is the action of insulin on glucose disposal.

Four states are used to describe insulin action. The plasma insulin concentration  $I(t)$ [mU/L] is described as

$$\frac{dI(t)}{dt} = \frac{U_I(t)}{V_I} - k_e I(t) \quad (2.5)$$

Where

- $U_I(t)$  [mU/min] is the absorption rate of insulin in the blood stream.
- $V_I$  [L] is the insulin distribution volume.
- $k_e$  [1/min] is the insulin elimination rate.

Three states representing the actions of insulin on glucose kinetics ( $x_1(t)Q_1(t)$ ,  $x_2(t)Q_2(t)$  and  $EGP_0(1 - x_3(t))$ ) are glucose distribution  $x_1(t)$ , glucose disposal  $x_2(t)$  and endogenous glucose production  $x_3(t)$  respectively.

$$\frac{dx_1(t)}{dt} = -k_{a1}x_1(t) + k_{b1}I(t) \quad (2.6a)$$

$$\frac{dx_2(t)}{dt} = -k_{a2}x_2(t) + k_{b2}I(t) \quad (2.6b)$$

$$\frac{dx_3(t)}{dt} = -k_{a3}x_3(t) + k_{b3}I(t) \quad (2.6c)$$

Where

- $k_{a1}$ ,  $k_{a2}$ ,  $k_{a3}$  [1/min] are the deactivation rate constants.
- $k_{b1}$ ,  $k_{b2}$ ,  $k_{b3}$  [L/mU/min] are the activation rate constants.

$$k_{b1} = S_{IT}k_{a1} \quad (2.7a)$$

$$k_{b2} = S_{ID}k_{a2} \quad (2.7b)$$

$$k_{b3} = S_{IE}k_{a3} \quad (2.7c)$$

### 2.1.2 Glucose-insulin system(Multiple-hormone)

Compared with the single-hormone model, the exercise model, glucagon model and CGM model included in the multiple-hormone model affect the system differential equation of the glucose-insulin system.  $Q_1(t)$  [mmol] and  $Q_2(t)$  [mmol] are rewritten as

$$\begin{aligned} \frac{dQ_1(t)}{dt} &= U_G(t) - (F_{01}^c(t) + F_R(t) + x_1(t)Q_1(t)) + k_{12}Q_2(t) + EGP_0(1 - x_3(t)) \\ &\quad + \frac{K_g V_G Z_2(t)}{18.01577} - \alpha E_2(t)^2 x_1(t) Q_1(t) \end{aligned} \quad (2.8a)$$

$$\begin{aligned} \frac{dQ_2(t)}{dt} &= x_1(t)Q_1(t) - (k_{12} + x_2(t)) Q_2(t) + \alpha E_2(t)^2 x_1(t) Q_1(t) - \alpha E_2(t)^2 x_2(t) Q_2(t) \\ &\quad - \frac{\beta E_1(t)}{HR_{base}} \end{aligned} \quad (2.8b)$$

Compared with the single-hormone model, the change of first compartment  $Q_1(t)$  [mmol] is also dependent on:

- $\frac{K_g V_G Z_2(t)}{18.01577}$  [mmol/min] is the effect of glucagon on blood glucose, and  $Z_2(t)$  is the state describing the glucagon.  $K_g = 0.075$
- $\alpha E_2(t)^2 x_1(t) Q_1(t)$  [mmol/min] is the exercise-induced insulin-dependent blood glucose consumption in the peripheral tissue, and  $E_2(t)$  is the prolonged response to physical activity.

The change of second compartment  $Q_2(t)$  [mmol] is also dependent on:

- $\alpha E_2(t)^2 x_1(t) Q_1(t)$  [mmol/min] is the exercise-induced insulin-dependent blood glucose consumption in the peripheral tissue.
- $\alpha E_2(t)^2 x_2(t) Q_2(t)$  [mmol/min] is the exercise-induced insulin-dependent disposal of glucose.
- $\frac{\beta E_1(t)}{HR_{base}}$  [mmol/min] is the exercise-induced insulin-independent blood glucose consumption.

### 2.1.3 Meal absorption

The meal absorption is simulated using a two-compartment model, which describes the effects of oral intake of carbohydrate on the glucose in the blood stream.  $D_1(t)$  [mmol] and  $D_2(t)$  [mmol] are the states describing the glucose in the two compartments.

$$\frac{dD_1(t)}{dt} = A_G D(t) - \frac{1}{\tau_D} D_1(t) \quad (2.9a)$$

$$\frac{dD_2(t)}{dt} = \frac{1}{\tau_D} D_1(t) - \frac{1}{\tau_D} D_2(t) \quad (2.9b)$$

Where

- $A_G$  [1/min] is a utilization factor of carbohydrates to glucose.
- $\tau_D$  [min] is a time constant.
- $D(t)$  [mmol/min] is the oral intake of carbohydrate in glucose equivalents.  $d(t)$  [g/min] is the oral intake rate of carbohydrate and  $M_{wG}$  [g/mol] is the molecular weight of glucose.

$$D(t) = \frac{1000}{M_{wG}} d(t) \quad (2.10)$$

The gut absorption rate from meals intake  $U_G(t)$  [mmol/min] can be calculated

$$U_G(t) = \frac{1}{\tau_D} D_2(t) \quad (2.11)$$

### 2.1.4 Subcutaneous insulin absorption

In this thesis, the insulin is injected subcutaneously using a CSII pump. A two-compartment model is used to describe the Subcutaneous insulin absorption.  $S_1(t)$

[mmol] and  $S_2(t)$  [mmol] are the states describing the insulin in the two compartments.

$$\frac{dS_1(t)}{dt} = u(t) - \frac{S_1(t)}{\tau_S} \quad (2.12a)$$

$$\frac{dS_2(t)}{dt} = \frac{S_1(t)}{\tau_S} - \frac{S_2(t)}{\tau_S} \quad (2.12b)$$

Where

- $u(t)$  [mU/min] is the injected insulin.
- $\tau_S$  [min] is a time constant.

The absorption rate of insulin in the blood stream  $U_I(t)$  [mU/min] can be calculated

$$U_I(t) = \frac{1}{\tau_S} S_2(t) \quad (2.13)$$

### 2.1.5 Parameters

The parameters of Hovorka model mentioned above are listed in the table [Boi12]. There are four parameters,  $V_I$ ,  $V_G$ ,  $EGP_0$  and  $F_{01}$  which are dependent on the body weight  $BW$  [kg], so they will be listed in the form of per kg.

**Table 2.2:** Parameters for the Hovorka model.

Parameter	Unit	Distribution
$EGP_0/BW$	mmol/kg/min	$EGP_0 \sim N(0.0161, 0.0039^2)$
$F_{01}/BW$	mmol/kg/min	$F_{01} \sim N(0.0097, 0.0022^2)$
$k_{12}$	1/min	$k_{12} \sim N(0.0649, 0.0282^2)$
$k_{a,1}$	1/min	$k_{a,1} \sim N(0.0055, 0.0056^2)$
$k_{a,2}$	1/min	$k_{a,2} \sim N(0.0683, 0.0507^2)$
$k_{a,3}$	1/min	$k_{a,3} \sim N(0.0304, 0.0235^2)$
$S_{I,1}$	L/mU	$S_{I,1} \sim N(51.2e^{-4}, (32.09e^{-4})^2)$
$S_{I,2}$	L/mU	$S_{I,2} \sim N(8.2e^{-4}, (7.84e^{-4})^2)$
$S_{I,3}$	L/mU	$S_{I,3} \sim N(520e^{-4}, (306.2e^{-4})^2)$
$k_e$	1/min	$k_e \sim N(0.14, 0.035^2)$
$V_I/BW$	L/kg	$V_I/BW \sim N(0.12, 0.012^2)$
$V_G/BW$	L/kg	$\exp(V_G/BW) \sim N(1.16, 0.23^2)$
$\tau_D$	min	$\ln\left(\frac{1}{\tau_D}\right) \sim N(-3.689, 0.025^2)$
$\tau_S$	min	$\frac{1}{\tau_I} \sim N(0.018, 0.0045^2)$
$A_G$	Unitless	$A_G \sim U(0.7, 1.2)$
$BW$	kg	$BW \sim U(65, 95)$

### 2.1.6 Steady state

In order to obtain the steady-state of the Hovorka model, the following equations needs to be solved

$$\dot{x}(t) = \begin{bmatrix} \frac{D_1(t)}{\frac{dt}{dt}} \\ \frac{D_2(t)}{\frac{dt}{dt}} \\ \frac{S_1(t)}{\frac{dt}{dt}} \\ \frac{S_2(t)}{\frac{dt}{dt}} \\ \frac{Q_1(t)}{\frac{dt}{dt}} \\ \frac{Q_2(t)}{\frac{dt}{dt}} \\ \frac{I(t)}{\frac{dt}{dt}} \\ \frac{x_1(t)}{\frac{dt}{dt}} \\ \frac{x_2(t)}{\frac{dt}{dt}} \\ \frac{x_3(t)}{\frac{dt}{dt}} \end{bmatrix} = f(x(t), u(t), d(t), 0) = \mathbf{0} \quad (2.14)$$

In addition, the equation  $f(x(t), u(t), d(t), 0) = \mathbf{0}$  needs to be satisfied near some desired blood glucose concentration value, such as the normoglycemia at  $z_{ss} = 108$  [mg/dL].

$$G(t) = \frac{Q_1(t)}{V_G} = z_{ss} \implies h(x(t)) - z_{ss} = 0 \quad (2.15)$$

Since the blood glucose concentration  $G(t)$  is only related to the first glucose compartment state  $Q_1(t)$ ,  $Q_1(t)$  can be calculated and expressed using only  $z(t)$ . In addition, the oral intake of carbohydrate  $D(t)$  is zero under the steady-state. Therefore, all states can be represented by the controlled variables  $z(t)$  and the manipulated variables  $u(t)$ , and  $\begin{bmatrix} x_{ss} \\ u_{ss} \end{bmatrix}$  can be calculated by the following equations.

$$\begin{bmatrix} f(x_{ss}, u_{ss}, d_{ss}, 0) \\ h(x_{ss}) - z_{ss} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 0 \end{bmatrix} \quad (2.16)$$

## 2.2 The MVP model

In this thesis, an identified model derived from Medtronic Virtual Patient(MVP) model is used as the control model. There are 6 states in the control model, inculding 2 states for modeling glucose-insulin system, 2 states for modeling the meal absorption, and 2 other states for modeling the subcutaneous insulin absorption.

### 2.2.1 Glucose-insulin system(Single-hormone)

There are 2 states, the effect of insulin  $I_{EFF}(t)$  [ $\text{min}^{-1}$ ] and the blood glucose  $G(t)$  [mg/dL]. The action of insulin on blood glucose is described by

$$\frac{dI_{EFF}(t)}{dt} = -k_1 I_{EFF}(t) + k_1 S_I I_P(t) \quad (2.17a)$$

$$\frac{dG(t)}{dt} = -I_{EFF}(t)G(t) + EGP + R_A(t) \quad (2.17b)$$

Where

- $S_I$  [mL/mU] is the insulin sensitivity.
- $EGP$  [mg/dL/min] is the endogenous glucose production.
- $R_A(t)$  [mg/dL/min] is the absorption rate from meals.

### 2.2.2 Glucose-insulin system(Multiple-hormone)

Compared with the single-hormone model, the glucagon model included in the multiple-hormone model affect the system differential equation of the glucose-insulin system.  $I_{EFF}(t)$  [ $\text{min}^{-1}$ ] and  $G(t)$  [mg/dL] are rewritten as

$$\frac{dI_{EFF}(t)}{dt} = -k_1 I_{EFF}(t) + k_1 S_I I_P(t) \quad (2.18a)$$

$$\frac{dG(t)}{dt} = -I_{EFF}(t)G(t) + EGP + R_A(t) + K_g Z_2(t) \quad (2.18b)$$

Where

- $K_g$  indicates the effect of glucagon on blood glucose.  $K_g = 0.075$ .
- $Z_2(t)$  [pg] is the state describing the glucagon.

### 2.2.3 Meal absorption

The same two-compartment model as in the Hovorka model is used to describe the meal absorption.  $D_1(t)$  [mmol] and  $D_2(t)$  [mmol] are the states describing the glucose in the two compartments.

$$\frac{dD_1(t)}{dt} = d(t) - k_m D_1(t) \quad (2.19a)$$

$$\frac{dD_2(t)}{dt} = k_m (D_1(t) - D_2(t)) \quad (2.19b)$$

Where

- $d(t)$  [mg/min] is the meals intake.
- $k_m$  [ $\text{min}^{-1}$ ] the inverse of the meal absorption time constant production.
- $R_A(t)$  is the absorption rate from meals.

The absorption rate from meals  $R_A(t)$  [mg/dL/min] can be calculated as

$$R_A(t) = D_2(t) \frac{k_m}{V_G} \quad (2.20)$$

Where  $V_G$  [dL] is the glucose distribution volume.

### 2.2.4 Subcutaneous insulin absorption

The subcutaneous insulin absorption is described by a two-compartment model.  $I_{SC}(t)$  [mU/L/min] and  $I_P(t)$  [mU/L] are the states describing the insulin concentration in the two compartments.

$$\frac{dI_{SC}(t)}{dt} = k_1 \left( \frac{u(t)}{C_I} - I_{SC}(t) \right) \quad (2.21a)$$

$$\frac{dI_P(t)}{dt} = k_1 (I_{SC}(t) - I_P(t)) \quad (2.21b)$$

Where

- $k_1$  [ $\text{min}^{-1}$ ] is the inverse of the insulin action time constant.
- $u(t)$  [mU/min] is the injected insulin.
- $C_I$  [L/min] is the clearance rate.

### 2.2.5 Steady state

In order to obtain the steady-state of the MVP model, the following equations needs to be solved

$$\dot{x}(t) = \begin{bmatrix} \frac{I_{SC}(t)}{\frac{dt}{dt}} \\ \frac{I_P(t)}{\frac{dt}{dt}} \\ \frac{I_{EFF}(t)}{\frac{dt}{dt}} \\ \frac{G(t)}{\frac{dt}{dt}} \\ \frac{D_1(t)}{\frac{dt}{dt}} \\ \frac{D_2(t)}{\frac{dt}{dt}} \end{bmatrix} = f(x(t), u(t), d(t), 0) = \mathbf{0} \quad (2.22)$$

Same as the Hovorka model, the equation  $f(x(t), u(t), d(t), 0) = \mathbf{0}$  needs to be satisfied near some desired blood glucose concentration value, such as the normoglycemia at  $z_{ss} = 108$  [mg/dL].

$$G(t) = G(t) = z_{ss} \implies h(x(t)) - z_{ss} = 0 \quad (2.23)$$

The blood glucose concentration  $G(t)$  is also a state and the oral intake of carbohydrate  $D(t)$  is zero under the steady-state. Therefore, all states can be represented by the controlled variables  $z_{ss}$  and the manipulated variables  $u_{ss}$ , and  $\begin{bmatrix} x_{ss} \\ u_{ss} \end{bmatrix}$  can be calculated by the following equations.

$$\begin{bmatrix} f(x_{ss}, u_{ss}, d_{ss}, 0) \\ h(x_{ss}) - z_{ss} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 0 \end{bmatrix} \quad (2.24)$$

## 2.3 CGM model

In this thesis, the CGM model from [Fac+14] is used for Hovorka model and MVP model. This model consists of two parts, the glucose transport from plasma to interstitial tissues and the sensor noise. The glucose transport is described as

$$\frac{dG_I(t)}{dt} = \frac{1}{\tau_{GI}} (G(t) - G_I(t)) \quad (2.25)$$

Where the time constant  $\tau_{GI}$  is 6.7 min. The sensor noise is described by the sum of the two auto-regressive processes

$$cc_k = 1.23cc_{k-1} - 0.3995cc_{k-2} + w_{cc,k} \quad (2.26a)$$

$$\hat{v}_k = 1.013\hat{v}_{k-1} - 0.2135\hat{v}_{k-2} + w_k \quad (2.26b)$$

Where

- $w_{cc,k} \sim N(0, 11.3 \text{ mg}^2/\text{dL}^2)$ .
- $w_k \sim N(0, 14.45 \text{ mg}^2/\text{dL}^2)$ .

Therefore, the discrete glucose value returned by the CGM at time  $t_k$  is

$$y_k = G_I(t_k) + cc_k + \hat{v}_k \quad (2.27)$$

## 2.4 Exercise model

There are three compartments which describe the increase in heart rate caused by exercise and the effect on blood glucose. Effect of increased heart rate  $E_1(t)$  [BPM], effect under the influence of exercise under different conditions  $T_E(t)$  [min] and the prolonged response to physical activity  $E_2(t)$  is described as

$$\frac{dE_1(t)}{dt} = \frac{1}{t_{HR}} (\Delta HR(t) - E_1(t)) \quad (2.28a)$$

$$\frac{dT_E(t)}{dt} = \frac{1}{t_{ex}} \cdot (c_1 f(E_1(t)) + c_2 - T_E(t)) \quad (2.28b)$$

$$\frac{dE_2(t)}{dt} = - \left( \frac{f(E_1(t))}{t_{in}} + \frac{1}{T_E(t)} \right) E_2(t) + \frac{f(E_1(t)) T_E(t)}{c_1 + c_2} \quad (2.28c)$$

Where

- $\Delta HR(t)$  [BPM] is the value higher than the basal heart rate.
- $t_{HR}$  [min] is the time constant for the effect of exercise when the heart rate is higher than the base heart rate.

- $t_{ex}$  [min] is the time constant for  $T_E(t)$  to return to steady state.
- $c_1$  and  $c_2$  [min] are parameters related to the steady-state value of  $T_E(t)$ .
- $f(E_1(t)) = \frac{\left(\frac{E_1(t)}{a \cdot HR_{base}}\right)^n}{1 + \left(\frac{E_1(t)}{a \cdot HR_{base}}\right)^n}$

The parameters of exercise model mentioned above are listed in the Table 2.3

**Table 2.3:** Parameters for the exercise model.

Parameter	Unit	Value
$a$	unitless	0.77
$t_{HR}$	min	5
$t_i n$	min	1
$n$	unitless	3
$t_{ex}$	min	200
$c_1$	min	500
$c_2$	min	100
$\beta$	mmol/min	0.78
$\alpha$	unitless	1.79
$HR_{base}$	bpm	70

## 2.5 Glucagon model

The glucagon model uses the same two-compartment model as the subcutaneous insulin absorption model,  $Z_1(t)$  [pg] and  $Z_2(t)$  [pg] describe the glucagon concentration in the two compartments

$$\frac{dZ_1(t)}{dt} = u_g(t) - k_g Z_1(t) \quad (2.29a)$$

$$\frac{dZ_2(t)}{dt} = k_g (Z_1(t) - Z_2(t)) \quad (2.29b)$$

Where

- $u_g(t)$  [pg/min] is the glucagon infusion rate.
- $k_g$  [ $\text{min}^{-1}$ ] is the inverse of time constant for the absorption rate of glucagon and  $k_g = 0.0526$ .

## 2.6 The extended Hovorka model

The extended Hovorka model can be modeled by using systems of ordinary differential equations. Without process noise and measurement noise, it can be expressed as

$$\frac{dx(t)}{dt} = f(x(t), u(t), d(t), p) \quad x(t_0) = x_0 \quad (2.30a)$$

$$y(t) = g(x(t)) \quad (2.30b)$$

$$z(t) = h(x(t)) \quad (2.30c)$$

For the single-hormone system, in addition to 1 state of the CGM model, the extended Hovorka model used in this thesis which is revised by Boiroux[Boi12] has a total of 11 state variables. There is 1 manipulated variable, subcutaneous insulin injections(including basal insulin and bolus insulin), and there is 1 disturbance, the oral intake of carbohydrate. There is 1 controlled variable which is the blood glucose concentration.

$$x(t) = [D_1(t), D_2(t), S_1(t), S_2(t), Q_1(t), Q_2(t), I(t), x_1(t), x_2(t), x_3(t), G_I(t)]' \quad (2.31a)$$

$$u(t) = [u_i(t)(u_{ba}(t) + u_{bo}(t))] \quad (2.31b)$$

$$d(t) = [d_{meal}(t)] \quad (2.31c)$$

$$y(t) = [G_I(t)] \quad (2.31d)$$

$$z(t) = [G(t)] \quad (2.31e)$$

For the multiple-hormone system, in addition to the CGM model, exercise model and glucagon model, the extended Hovorka model has a total of 16 state variables. There are 2 manipulated variables, subcutaneous insulin injections(including basal insulin and bolus insulin) and glucagon injections, and there are 2 disturbances, the oral intake of carbohydrate and exercise. There is 1 controlled variable same as the single-hormone system.

$$x(t) = [D_1(t), D_2(t), S_1(t), S_2(t), Q_1(t), Q_2(t), I(t), x_1(t), x_2(t), x_3(t), G_I(t), Z_1(t), Z_1(t), E_1(t), T_E(t), E_2(t)]' \quad (2.32a)$$

$$u(t) = [u_i(t)(u_{ba}(t) + u_{bo}(t)), u_g(t)]' \quad (2.32b)$$

$$d(t) = [d_{meal}(t), \Delta HR(t)]' \quad (2.32c)$$

$$y(t) = [G_I(t)] \quad (2.32d)$$

$$z(t) = [G(t)] \quad (2.32e)$$

## 2.7 The extended MVP model

The extended MVP model can be modeled by using systems of ordinary differential equations. Without process noise and measurement noise, it can be expressed as

$$\frac{dx(t)}{dt} = f(x(t), u(t), d(t), p) \quad x(t_0) = x_0 \quad (2.33a)$$

$$y(t) = g(x(t)) \quad (2.33b)$$

$$z(t) = h(x(t)) \quad (2.33c)$$

For the single-hormone system, in addition to 1 state of the CGM model, the extended MVP model has a total of 7 state variables. Same as the Hovorka Model, There is 1 manipulated variable, subcutaneous insulin injections(including basal insulin and bolus insulin), and there is 1 disturbance, the oral intake of carbohydrate. There is 1 controlled variable which is the blood glucose concentration.

$$x(t) = [I_{SC}(t), I_P(t), I_{EFF}(t), G(t), D_1(t), D_2(t), G_I(t)]' \quad (2.34a)$$

$$u(t) = [u_i(t)(u_{ba}(t) + u_{bo}(t))] \quad (2.34b)$$

$$d(t) = [d_{meal}(t)] \quad (2.34c)$$

$$y(t) = [G_I(t)] \quad (2.34d)$$

$$z(t) = [G(t)] \quad (2.34e)$$

For the multiple-hormone system, in addition to the CGM model and glucagon model, the extended MVP model has a total of 9 state variables. There are 2 manipulated variable, subcutaneous insulin injections(including basal insulin and bolus insulin) and glucagon injections, but there is only 1 disturbance, the oral intake of carbohydrate, and 1 controlled variable, blood glucose concentration.

$$x(t) = [I_{SC}(t), I_P(t), I_{EFF}(t), G(t), D_1(t), D_2(t), G_I(t), Z_1(t), Z_2(t)]' \quad (2.35a)$$

$$u(t) = [u_i(t)(u_{ba}(t) + u_{bo}(t)), u_g(t)]' \quad (2.35b)$$

$$d(t) = [d_{meal}(t)] \quad (2.35c)$$

$$y(t) = [G_I(t)] \quad (2.35d)$$

$$z(t) = [G(t)] \quad (2.35e)$$

# CHAPTER 3

# Parameter Estimation

---

In this thesis, the extended Hovorka model is used to simulate patients with type 1 diabetes, and an identified model[Boi+19] derived from the MVP model is used as the control model of the model predictive controller.

The accuracy of the model is very important for the design and tuning of model-based control algorithms, such as model predictive control(MPC). The prediction model derived from the MVP model is a minimal model, which is simpler and identifiable than the Hovorka model. With only 6 differential equations and fewer parameters, although the MVP model can simulate the dynamics of the glucose/insulin system, there is still a certain mismatch between the MVP model and the patient's physiological model. For individual insulin therapy, the MVP model is required to be adjusted to match the individual's physiological model. Therefore, parameter identification is used to make the model more accurately simulate the dynamics of the glucose/insulin system of the individual by finding a set of parameters.

## 3.1 Model

In this thesis, in order to verify the performance of parameter identification and control algorithms, the Hovorka model and its parameter distribution mentioned in Chapter 2 will be used to generate 10 virtual patients, and the parameters of 10 patients are shown in Table A.1 and A.2 in Appendix A. Clinical experiments will be designed to generate data set used for parameter identification. Since the parameters of the CGM model, the exercise model and the glucagon model are all fixed values, the parameters that need to be identified for the single-hormone system or the multiple-hormone system are only the parameters of the MVP model. The control model derived from the MVP model mentioned in section 2.2 is considered for parameter identification. The model can be formulated as a continuous-discrete stochastic differential equation model

$$dx(t) = f(x(t), u(t), d(t), p)dt + \sigma d\omega(t) \quad (3.1a)$$

$$y(t_k) = h(x(t_k)) + v_k \quad (3.1b)$$

The SDE model includes subcutaneously infused insulin,  $u(t)$ , the CHO absorption rate,  $d(t)$ , the model states,  $x(t)$ , and blood glucose  $y(t_k)$  which taken at discrete times. The process noise  $\omega(t)$  is assumed as a standard Wiener process with  $d\omega(t) \sim N_{iid}(0, Idt)$  and the sensor noise  $v_k$  is described in CGM model of section 2.3. In order to specifically describe the parameters and diffusion terms to be identified, the differential equation of the SDE model is expressed as

$$dI_{SC}(t) = k_1 \left( \frac{u(t)}{C_I} - I_{SC}(t) \right) dt \quad (3.2)$$

$$dI_P(t) = k_1 (I_{SC}(t) - I_P(t)) dt \quad (3.3)$$

$$dI_{EFF}(t) = (-k_1 I_{EFF}(t) + k_1 S_I I_P(t)) dt \quad (3.4)$$

$$dG(t) = (-I_{EFF}(t)G(t) + EGP + R_A(t)) dt + \sigma_4 dw_G \quad (3.5)$$

$$dD_1(t) = k_m (d(t) - D_1(t)) dt \quad (3.6)$$

$$dD_2(t) = k_m (D_1(t) - D_2(t)) dt \quad (3.7)$$

$$dG_I(t) = \frac{1}{\tau_{GI}} (G(t) - G_I(t)) dt \quad (3.8)$$

Where

- The absorption rate from meals  $R_A(t)$  [ $mg/dL/min$ ] is

$$R_A(t) = D_2(t) \frac{k_m}{V_G} \quad (3.9)$$

- The insulin clearance rate  $C_I$  [ $L/min$ ] is fixed as 1.5, because it cannot be distinguished from the insulin sensitivity when only CGM measurements are available[BJ18].
- The time constant of CGM lag  $\tau_{GI}$  [min] is set as 6.7[Fac+14].
- Only the diffusion term  $\sigma_4$  related to blood glucose  $G(t)$  is identified. Other diffusion terms are set as 0.

Therefore, the parameters to be identified are

$$\theta = [ k_1 \quad S_I \quad EGP \quad k_m \quad V_G \quad \sigma_4 ]'$$

## 3.2 Numerical integration methods

In this thesis, numerical integration methods are needed to solve differential equations. In the nonlinear model predictive control(NMPC) algorithm described in the following chapters, whether for a single-hormone system or a multiple-hormone system, the numerical integration of the ordinary differential equations(ODE) version of the extended MVP model is required. For example, the nonlinear system differential equation is numerically integrated to predict the state and its covariance in the continuous-discrete extended Kalman filter(CDEKF). It is also used to construct the objective function of the optimal control problem in the NMPC controller.

$$dx(t) = f(x(t))dt \quad (3.10)$$

In addition, the stochastic differential equations(SDE) version of extended MVP model is used to calculate error and its covariance in the maximum likelihood estimation(MLE) method for parameter identification. The SDE version of the extended Hovorka model is used for simulation in closed-loop testing.

$$dx(t) = f(x(t))dt + \sigma(x(t))d\omega(t) \quad (3.11)$$

Where  $\omega(t)$  is a standard Wiener process. In this thesis, explicit Runge-Kutta 4(ERK4) with fixed step is introduced to numerically integrate ODE, and the Euler-Maruyama method with a fixed step size is used to numerically integrate the SDE.

### 3.2.1 Explicit Runge-Kutta 4 methods for ODE

In this section, the numerical integration method for ODE is introduced. The ODE can be formulated as

$$dx(t) = f(x(t))dt \quad x(t_0) = x_0 \quad (3.12)$$

Both explicit and implicit methods can perform numerical integration on ODE. In this thesis, explicit Runge-Kutta 4(ERK4) method is used. The calculation steps of explicit Runge-Kutta method can be expressed as

$$T_i = t_k + c_i h, \quad i = 1, 2, \dots, s \quad (3.13a)$$

$$X_i = x_k + h \sum_{j=1}^s a_{ij} f(T_j, X_j), \quad i = 1, 2, \dots, s \quad (3.13b)$$

$$x_{k+1} = x_k + h \sum_{i=1}^s b_i f(T_i, X_i) \quad (3.13c)$$

Where  $h$  is the step size. In this thesis, a fixed step size is used to perform numerical integration from start time  $t_0$  to the end time  $t_f$ . For example, the sampling time of the simulation is 5 minutes and the step size is 1.25, and the calculation steps

mentioned above will be calculated four times for an interval. The parameters  $a_{ij}$ ,  $b_i$ ,  $c_i$  can be found in the Butcher tableau of an explicit Runge Kutta scheme

**Table 3.1:** Butcher tableau of an explicit Runge Kutta scheme.

0					
$c_2$	$a_{21}$				
$c_3$	$a_{31}$	$a_{32}$			
$\vdots$	$\vdots$		$\ddots$		
$c_s$	$a_{s1}$	$a_{s2}$	$\cdots$	$a_{s,s-1}$	
$x$	$b_1$	$b_2$	$\cdots$	$b_{s-1}$	$b_s$

The Butcher tableau of explicit Runge-Kutta 4(ERK4) methods used in this thesis is

**Table 3.2:** Butcher tableau of an explicit Runge Kutta 4 method.

0				
$1/2$	$1/2$			
$1/2$	0	$1/2$		
1	0	0	0	
$x$	$1/6$	$1/3$	$1/3$	$1/6$

### 3.2.2 Euler-Maruyama method for SDE

In this section, the numerical integration method for SDE is introduced. The SDE can be formulated as

$$dx(t) = f(x(t))dt + \sigma(x(t))d\omega(t) \quad x(t_0) = x_0 \quad (3.14)$$

Where  $\omega(t)$  is a standard Wiener process and  $d\omega(t) \sim N_{iid}(0, Idt)$ . Further, SDE is discretized and expressed as

$$x(t_{k+1}) = x(t_k) + \int_{t_k}^{t_{k+1}} f(t, x(t))dt + \int_{t_k}^{t_{k+1}} \sigma(x(t))d\omega(t) \quad (3.15)$$

In this thesis, the explicit Euler-Maruyama method is used to numerically integrate the SDE. In this method, the state function  $f(x(t))$  and the  $\sigma(x(t))$  method are kept a constant value between the interval of  $t_k$  and  $t_{k+1}$ , and the calculation steps can be expressed as

$$x_{k+1} = x_k + f(t_k, x_k) \Delta t_k + \sigma(x_k) \Delta \omega_k \quad (3.16)$$

Where  $\Delta t_k$  is the fixed step size and  $\Delta \omega(k) \sim N_{iid}(0, \Delta t_k)$

### 3.3 Experimental design of data set for parameter estimation

Generally, frequently sampled intravenous glucose Tolerance tests (IVGTTs) are used to identify the minimal models, such as the Bergman minimal model and the MVP model[Gal+09]. In standard IVGTTs, glucose is intravenously injected and blood glucose concentrations are measured following the injection. However, compared with standard IVGTTs, infusing glucose and insulin in different time patterns can make parameter identification faster and more accurate. For example, in modified IVGTTs, insulin is infused after a certain period of glucose intake. In addition, although virtual patients rather than real patients are used for testing, the designed experimental test must be safe, that is, Blood glucose and insulin concentration must always be kept within the normal physiological range.

A 30-hour data set is used, and the output is taken by CGM measurement with a sampling time of 5 minutes. The data set is obtained by simulation based on the individual parameters generated by the Hovorka model and its parameter distribution. The SDE version of Hovorka Model is used to simulate the scenario.

$$dx(t) = f(x(t), u(t), d(t), p)dt + \sigma d\omega(t) \quad (3.17a)$$

$$y(t_k) = h(x(t_k)) + v_k \quad (3.17b)$$

For each state, the diffusion term

$$\sigma_i = 0.01x_{ss,i} \quad i = 1, 2, \dots, 11 \quad (3.18)$$

Where  $x_{ss,i}$  is the steady state for each state.

The design of the meal size and time of the virtual patient is taken from[Kan+09]:

**Table 3.3:** Size and time of meals.

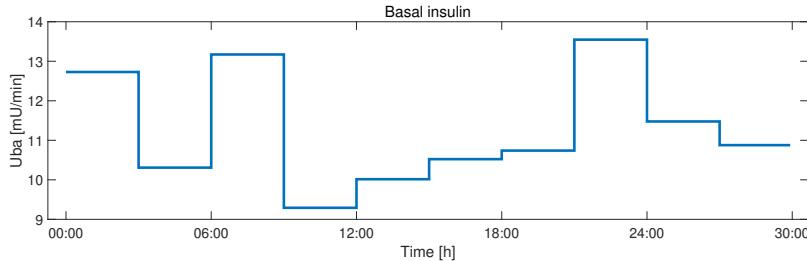
	Breakfast	Snack	Lunch	Dinner	Snack
Time	8:00	11:30	13:15	18:00	22:00
Carbohydrates(g)	72	36	131	51	70

At the same time, a certain amount of bolus insulin will be injected at the time of food intake.

**Table 3.4:** Size and time of Bolus insulin.

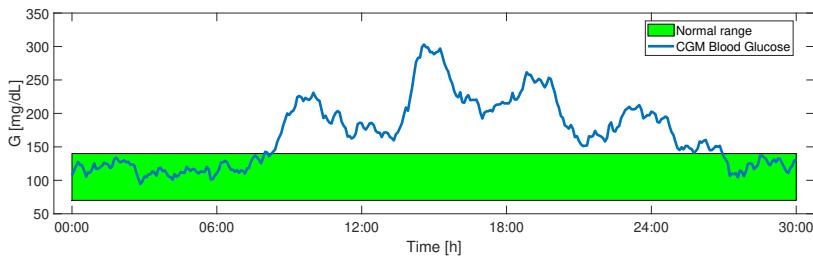
Time	8:00	11:30	13:15	18:00	22:00
Bolus insulin(mU/min)	800	800	800	800	800

In order to better identify the time constant, basal insulin will maintain a constant injection in a 3-hour interval and the value is calculated as  $u_{ba,k} = u_{ss} + \eta_k$ , where  $u_{ss}$  is the steady-state basal insulin injection rate and  $\eta_k \sim N(0, 0.3u_{ss})$ . Below is a set of example data for randomly generated basal insulin.



**Figure 3.1:** A set of sample data for randomly generated basal insulin.

Two parameter estimation methods, least squares(LS) estimation and maximum likelihood estimation(MLE), will be introduced and implemented. A set of parameters of the Hovorka Model is used as an example to explain the experimental design of the data set and to verify the effectiveness of two parameter identification methods. In this thesis, the maximum likelihood estimation(MLE) method is selected as a parameter identification method for closed-loop testing of the model-based control algorithm. The inputs including basal insulin infusion rate, bolus insulin infusion rate and meal size are assumed to be known. An example of the generated 30-hour data set is



**Figure 3.2:** An example of the generated 30-hour data set.

The relative mean square error (RMSE) is used to quantify the performance of parameter estimation. The RMSE is obtained by

$$RMSE = \sqrt{\frac{\sum_{k=1}^N (\bar{y}_i - \hat{y}_i)^2}{N}} \quad (3.19)$$

Where  $\bar{y}_i$  is the measured output and  $\hat{y}_i$  is the estimated output based on the parameter identified.

### 3.4 Least squares estimation

The least squares(LS) method is used to find a set of parameters to minimize the objective function related to the residual of the whole process. Regarding the form of the objective function expressed by the residual, it can be divided into unweighted square and weighted square. A general representation is

$$\hat{\theta} = \operatorname{argmin}_{\theta} \sum_{i=1}^N \|y_i(\theta) - \bar{y}_i\|_{W_i}^2 \quad (3.20)$$

Where  $\hat{\theta}$  is the optimal parameter,  $y_i(\theta)$  is the estimated output based on the parameter found and  $\bar{y}_i$  is the measured output. The weights in the unweighted square form are  $W_i = \operatorname{diag}(\mathbf{1})$ . In this thesis, the identification effect of the whole process is equally valued, so the unweighted square form is described and implemented.

The NLP problem form of the least squares problem is

$$\min_{\theta} \quad \Phi(\theta) = \sum_{k=1}^N \|y_k - \bar{y}_k\|^2 \quad (3.21a)$$

$$\text{s.t.} \quad x_0 = \bar{x}_0 \quad (3.21b)$$

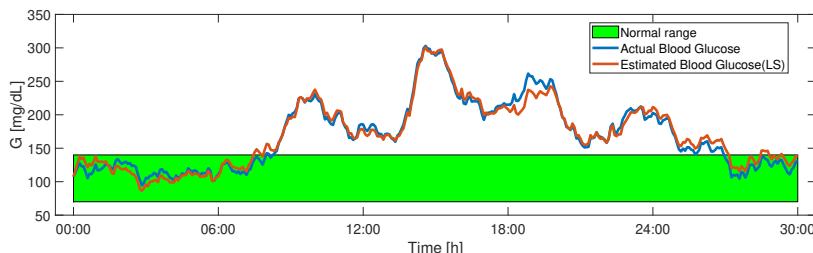
$$F(x_k, u_k, d_k, \theta_k) - x_{k+1} = 0 \quad \forall k \in [0, N-1] \quad (3.21c)$$

$$G(x_k) - y_k = 0 \quad \forall k \in [0, N-1] \quad (3.21d)$$

$$\theta_{min} \leq \theta \leq \theta_{max} \quad (3.21e)$$

Where  $y_k$  is the measured output and  $x_0$  is the initial states. The single shooting method is used as the discrete method. In the calculation of the objective function, where  $\bar{y}_k$ ,  $\bar{x}_0$ ,  $u_k$ ,  $d_k$  are known,  $x_{k+1}$  is calculated by numerical integration using the Euler-Maruyama method with a fixed step size[Mar55].

The result of parameter identification using the least square(LS) estimation method on the example data is



**Figure 3.3:** The result of parameter identification using the least square(LS) estimation.

The RMSE of Least squares(LS) estimation's result is

$$RMSE_{LS} = 10.7820$$

### 3.5 Maximum likelihood estimation

The problem of Maximum likelihood estimation(MLE) for nonlinear continuous-discrete systems is formulated

$$dx(t) = f(x(t), u(t), d(t), p)dt + \sigma d\omega(t) \quad (3.22a)$$

$$y(t_k) = h(x(t_k)) + v_k \quad (3.22b)$$

Assumed  $\mathcal{Y}_N$  is a set of data sampled from the continuous-discrete stochastic model above

$$\mathcal{Y}_N = \{y_0, y_1, \dots, y_N\} \quad (3.23)$$

The joint probability density is maximized

$$p(\mathcal{Y}_N | \theta) = p(y_N, y_{N-1}, \dots, y_0 | \theta) \quad (3.24)$$

Make  $V(\theta) = -\log(p(\mathcal{Y}|\theta))$  the negative log-likelihood function, which is equivalent to minimize

$$V(\theta) = \frac{(N+1)n_y}{2} \ln(2\pi) + \frac{1}{2} \sum_{k=0}^N \left[ \ln [\det(R_{e,k})] + e'_k R_{e,k}^{-1} e_k \right] \quad (3.25)$$

Where

$$e_k = y_k - \hat{y}_{k|k-1} \quad (3.26)$$

The optimal parameters to be found can be expressed as

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}}(V(\theta)) \quad (3.27)$$

The continuous-discrete extended Kalman filter(CDEKF) is used to compute the error  $e_k$  and its covariance  $R_{e,k}$ . In the chapters 4 of the nonlinear model predictive control algorithm, CDEKF is also used as an estimator. The calculation steps are as follows

Filtering: Given predicted state and its covariance  $\hat{x}_{k|k-1}$ ,  $\hat{P}_{k|k-1}$  and discrete measurement  $y_k$

$$C_k = \frac{\partial h}{\partial x} (\hat{x}_{k|k-1}) \quad (3.28a)$$

$$R_{k|k-1} = C_k \hat{P}_{k|k-1} C'_k + R_k \quad (3.28b)$$

$$K_k = \hat{P}_{k|k-1} C'_k R_{k|k-1}^{-1} \quad (3.28c)$$

The innovation is calculated by

$$e_k = y_k - h(\hat{x}_{k|k-1}) \quad (3.29)$$

The estimated state and its covariance are obtained by

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k e_k \quad (3.30a)$$

$$\hat{P}_{k|k} = \hat{P}_{k|k-1} - K_k R_{k|k-1} K'_k \quad (3.30b)$$

Prediction: Given estimated state and its covariance  $\hat{x}_{k|k}$ ,  $\hat{P}_{k|k}$ , Compute  $\hat{x}_{k+1|k}$ ,  $\hat{P}_{k+1|k}$  by solving

$$\frac{d\hat{x}_k(t)}{dt} = f(\hat{x}_k(t), u_k, d_k, p) \quad (3.31a)$$

$$\frac{dP_k(t)}{dt} = A_k(t)P_k(t) + P_k(t)A_k(t)' + \sigma\sigma' \quad (3.31b)$$

Where

$$A_k(t) = \frac{\partial f}{\partial x}(\hat{x}_k(t), u_k, d_k, p) \quad (3.32)$$

With the initial conditions

$$\hat{x}_k(t_k) = \hat{x}_{k|k} \quad (3.33a)$$

$$\hat{P}_k(t_k) = \hat{P}_{k|k} \quad (3.33b)$$

The NLP problem form of the Maximum likelihood estimation problem is

$$\min_{\theta} V(\theta) \quad (3.34a)$$

$$\text{s.t. } \hat{x}_{0|0} = \bar{x}_0 \quad (3.34b)$$

$$\hat{P}_{0|0} = \bar{P}_0 \quad (3.34c)$$

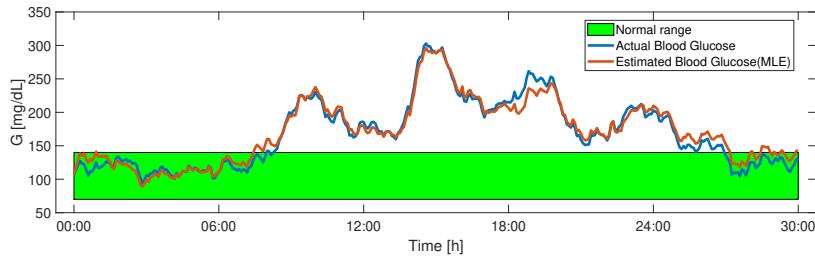
$$F\left(\hat{x}_{k|k}, \hat{P}_{k|k}, u_k, d_k, \theta_k\right) - \hat{x}_{k+1|k+1} = 0 \quad \forall k \in [0, N-1] \quad (3.34c)$$

$$G\left(\hat{P}_{k|k}\right) - \hat{P}_{k+1|k+1} = 0 \quad \forall k \in [0, N-1] \quad (3.34d)$$

$$\theta_{min} \leq \theta \leq \theta_{max} \quad (3.34e)$$

The single shooting method is used as the discrete method. The function  $F$  and  $G$  correspond to the prediction and filtering steps of CDEKF. the Euler-Maruyama method with a fixed step size[Mar55] is used for numerical integration.

The result of parameter identification using the maximum likelihood estimation(MLE) method on the example data is



**Figure 3.4:** The result of parameter identification using the Maximum likelihood estimation(MLE).

The RMSE of maximum likelihood estimation's result is

$$RMSE_{MLE} = 9.5128$$

### 3.6 Parameter estimation for closed-loop simulation

In this thesis, we conduct a closed-loop test on the implemented control algorithm. The control performance of the model-based control algorithm, such as model predictive control(MPC), depends on the accuracy of the control model. For different patients, the parameters of the control model need to be identified. In the tests of the LMPC and NMPC algorithms, the maximum likelihood estimation(MLE) will be used to estimate the parameters of 10 virtual patients based on the Hovorka Model. The parameters of 10 virtual patients generated using the Hovorka model and his parameter distribution is shown in Table A.1 and A.2 in Appendix A. The following are the parameter identification results of 10 virtual patients

**Table 3.5:** Estimated Parameters of MVP model of 10 virtual patients.

Patient	$k_1$ (1/min)	$S_I$ (mL/mU/min)	$EGP_0$ (mg/dL/min)
1	0.0114	0.00199	2.159
2	0.0089	0.00185	2.008
3	0.00598	0.00160	1.884
4	0.0077	0.00191	2.034
5	0.0123	0.00215	2.296
6	0.0109	0.00210	2.172
7	0.00736	0.00197	2.057
8	0.0101	0.00201	2.188
9	0.0098	0.00212	2.151
10	0.00807	0.00204	2.112

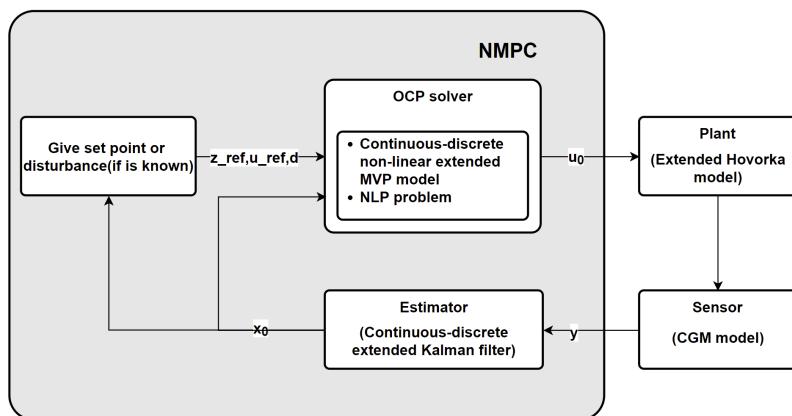
Patient	$k_m$ (1/min)	$V_G$ (dL)	$\sigma_4$ (mg/dL/min)
1	0.0135	119.143	0.345
2	0.0214	108.901	0.350
3	0.0172	114.880	0.359
4	0.0144	93.867	0.384
5	0.0136	91.953	0.356
6	0.0198	171.698	0.383
7	0.0188	189.526	0.362
8	0.0171	127.453	0.359
9	0.0310	288.052	0.440
10	0.0172	116.018	0.368



# CHAPTER 4

# Nonlinear Model Predictive Control for artificial pancreas

In this chapter, the nonlinear model predictive control(NMPC) is applied for the artificial pancreas(AP) problem. Figure 4.1 shows the block-diagram of closed-loop simulation with NMPC.



**Figure 4.1:** Block-diagram of close loop simulation with NMPC.

In NMPC, in every control calculation, according to the form of the objective function, the construction of the optimal control problem(OCP) needs to be provided with the set point of the controlled variable  $z$ , the target value of the manipulated variable  $u$  and the value of the disturbance term observed by the estimator or directly given from the outside. In addition, the initial value of the state of the control model observed by the estimator is necessary. Since NMPC uses a nonlinear control model, we use continuous-discrete extended Kalman filter(CDEKF) as an estimator. In the solution of the OCP, the OCP is discretized as a nonlinear programming(NLP) problem and solved. The OCP solution in NMPC is implemented using the framework of CasADI. The solver of the NLP optimization problem is IPOPT, which is also called by CasADI. In section 4.4, how to use the CasADI framework to implement the NMPC algorithm of the single-hormone system is introduced, and the code of the version of Matlab and C++ are also shown.

## 4.1 Mathematical model for NMPC

In the section 3.1 of parameter estimation, the MVP model for identification is introduced. When the parameters of the MVP model are properly identified, the control model for NMPC can be formed.

$$dx(t) = f(x(t), u(t), d(t), p)dt + \sigma d\omega(t) \quad (4.1a)$$

$$y(t_k) = h(x(t_k)) + v_k \quad (4.1b)$$

The process noise  $\omega(t)$  is assumed as a standard Wiener process with  $d\omega(t) \sim N_{iid}(0, Idt)$  and the sensor noise  $v_k$  is described in CGM model of section 2.3. The control model of the NMPC algorithm based on the extended MVP model of single-hormone system contains 7 states (6 states in the MVP model + 1 state in the CGM model), The control model of the NMPC algorithm based on the the extended MVP model of multiple-hormone system contains 9 states (6 states in the MVP model + 1 state in the CGM model + 2 states in the glucagon model). The SDE version of the extended MVP model is used as the control model, and the method of solving the steady-state value has been introduced in section 2.2.5.

## 4.2 Continuous-discrete extended Kalman filter

For the nonlinear continuous-discrete systems used in NMPC, the continuous-discrete extended Kalman filter(CDEKF) is introduced. The function of CDEKF is to predict and estimate the state of the system and its covariance, given a continuous state equation and discrete measurement.

### 4.2.1 Filtering

Given predicted state and its covariance  $\hat{x}_{k|k-1}$ ,  $\hat{P}_{k|k-1}$  and discrete measurement  $y_k$

$$C_k = \frac{\partial h}{\partial x} (\hat{x}_{k|k-1}) \quad (4.2a)$$

$$R_{k|k-1} = C_k \hat{P}_{k|k-1} C'_k + R_k \quad (4.2b)$$

$$K_k = \hat{P}_{k|k-1} C'_k R_{k|k-1}^{-1} \quad (4.2c)$$

The innovation is calculated by

$$e_k = y_k - h(\hat{x}_{k|k-1}) \quad (4.3)$$

The estimated state and its covariance are obtained by

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k e_k \quad (4.4a)$$

$$\hat{P}_{k|k} = \hat{P}_{k|k-1} - K_k R_{k|k-1} K'_k \quad (4.4b)$$

### 4.2.2 Prediction

Given estimated state and its covariance  $\hat{x}_{k|k}$ ,  $\hat{P}_{k|k}$ , Compute  $\hat{x}_{k+1|k}$ ,  $\hat{P}_{k+1|k}$  by solving

$$\frac{d\hat{x}_k(t)}{dt} = f(\hat{x}_k(t), u_k, d_k, p) \quad (4.5a)$$

$$\frac{dP_k(t)}{dt} = A_k(t)P_k(t) + P_k(t)A_k(t)' + \sigma\sigma' \quad (4.5b)$$

Where

$$A_k(t) = \frac{\partial f}{\partial x} (\hat{x}_k(t), u_k, d_k, p) \quad (4.6)$$

With the initial conditions

$$\hat{x}_k(t_k) = \hat{x}_{k|k} \quad (4.7a)$$

$$\hat{P}_k(t_k) = \hat{P}_{k|k} \quad (4.7b)$$

$u_k$  indicates the insulin infusion rate calculated by the NMPC.  $d_k$  is the announced oral meal intake and  $p$  is the estimated parameters of MVP model. The numerical integration is calculated by ERK4.

## 4.3 Optimal control problem formulation

An optimal control problem includes a objective function that is a function of state and control variables. NMPC solves an optimal control problem (OCP) to obtain the optimal trajectory for input  $u$ , which describes the paths of the control variables that minimize the cost function subject to input bounds and dynamics describing the states  $x$  and outputs  $y$ . A general OCP formulation for NMPC is

$$\min_u \quad \Phi = \int_{t_0}^{t_f} g(z(t), u(t), p) dt + \lambda(z(t_f), p) \quad (4.8a)$$

$$\text{s.t.} \quad x(t_0) = x_0 \quad (4.8b)$$

$$\dot{x}(t) = f(x(t), u(t), d(t), p) \quad (4.8c)$$

$$z(t) = h(x(t), p) \quad (4.8d)$$

$$u_{\min} \leq u(t) \leq u_{\max} \quad (4.8e)$$

The objective function consists of two terms: a stage cost term  $g(z(t), u(t), p)$  and a stage-to-go term  $\lambda(z(t_f), p)$ .  $\dot{x}(t) = f(x(t), u(t), d(t), p)$  and  $z(t) = h(x(t), p)$  represents the model equations.  $t_0$ ,  $t_f$ ,  $x_0$  are known parameters.

In the artificial pancreas(AP) problem, for the single-hormone system,  $u(t)$  represents the basal insulin infusion rate and bolus insulin infusion rate at time  $t$  and  $d(t)$  represents the carbohydrates (CHO) intake rate at time  $t$ .

For the multiple-hormone system,  $u(t)$  represents the basal insulin infusion rate, bolus insulin infusion rate and glucagon infusion rate at time  $t$  and  $d(t)$  represents the carbohydrates (CHO) intake rate and increased heart rate caused by exercise at time  $t$ . Model equations  $\dot{x}(t) = f(x(t), u(t), d(t), p)$  and  $z(t) = h(x(t), p)$  is derived from the extended MVP model.

### 4.3.1 Objective function

For the single-hormone system, blood glucose fluctuates due to the influence of meals intake and endogenous glucose production, and the injection rate of insulin(including basal insulin and bolus insulin) is controlled to keep blood glucose in a normal range. For the multiple-hormonal system, in addition to meals intake, exercise will also affect blood glucose fluctuations, but at the same time glucagon will be controlled to prevent blood glucose from falling below the normal range. A large number of constraints make the solution of nonlinear programming problems time-consuming and difficult to obtain a global optimal solution. In order to improve computational efficiency, the upper and lower bound constraints of the controlled variable  $z$  are penalized in the objective function in the form of soft constraints rather than directly set as the form of inequality constraints.

For the single-hormone system, the objective function consists of the penalty term of the controlled variable  $z$  related to the blood glucose and the penalty terms of the manipulated variables  $u_i$  related to the insulin including  $u_{ba}$  and  $u_{bo}$  and their change rate  $\Delta u_i$  including  $\Delta u_{ba}$  and  $\Delta u_{bo}$ .

$$\begin{aligned}\Phi &= \Phi_z + \Phi_u(\Phi_{ui}) \\ &= \underbrace{\int_{t_0}^{t_f} \rho_z(z(t)) dt}_{\text{BG penalty terms}} + \underbrace{\int_{t_0}^{t_f} \rho_{ui}(u_{i,k}) dt + \sum_{k=0}^{N-1} \rho_{\Delta ui}(\Delta u_{i,k})}_{\text{Insulin penalty terms}} \quad (4.9)\end{aligned}$$

For the multiple-hormone system, in addition to the penalty terms of blood glucose and insulin infusion rate, there is also a penalty term of the manipulated variable  $u_g$  and its change rate related to the glucagon.

$$\begin{aligned}\Phi &= \Phi_z + \Phi_u(\Phi_{ui} + \Phi_{ug}) \\ &= \underbrace{\int_{t_0}^{t_f} \rho_z(z(t)) dt}_{\text{BG penalty terms}} + \underbrace{\int_{t_0}^{t_f} \rho_{ui}(u_{i,k}) dt + \sum_{k=0}^{N-1} \rho_{\Delta ui}(\Delta u_{i,k})}_{\text{Insulin penalty terms}} \\ &\quad + \underbrace{\int_{t_0}^{t_f} \rho_{ug}(u_{g,k}) dt + \sum_{k=0}^{N-1} \rho_{\Delta ug}(\Delta u_{g,k})}_{\text{Glucagon penalty terms}} \quad (4.10)\end{aligned}$$

The penalty term of the controlled variable  $z$  includes the penalty term of deviation from the target value, and the soft constraint penalty of the upper and lower bounds of  $z_k$ .

$$\rho_z(z(t)) = \alpha_{\bar{z}}\rho_{\bar{z}}(z(t)) + \alpha_{z_{\min}}\rho_{z_{\min}}(z(t)) + \alpha_{z_{\max}}\rho_{z_{\max}}(z(t)) \quad (4.11)$$

Where  $\alpha_{\bar{z}}$ ,  $\alpha_{z_{\min}}$  and  $\alpha_{z_{\max}}$  are the weight values of the penalty items. Through tuning, the weight value is set to  $\alpha_{\bar{z}} = 0.01$ ,  $\alpha_{z_{\min}} = 300000$  and  $\alpha_{z_{\max}} = 1$ . The respective penalties can be expressed as

$$\rho_{\bar{z}}(z(t)) = \frac{1}{2}\|z(t) - \bar{z}\|_2^2 \quad (4.12a)$$

$$\rho_{z_{\min}}(z(t)) = \frac{1}{2}\|\min\{z(t) - z_{\min}, 0\}\|_2^2 \quad (4.12b)$$

$$\rho_{z_{\max}}(z(t)) = \frac{1}{2}\|\max\{z(t) - z_{\max}, 0\}\|_2^2 \quad (4.12c)$$

$\bar{z} = 108$  mg/dL is the blood glucose set point.  $z_{\min}$  and  $z_{\max}$  are thresholds for hypoglycemia ( $BG < 70$  mg/dL) and hyperglycemia ( $BG > 180$  mg/dL). In addition, since hypoglycemia is very dangerous for patients, a high penalty value will be given

when the blood glucose concentration is lower than 70 mg/dL.

Since the penalty term of  $\Delta u_i$  is a discrete form, and the discrete method will be described in section 4.3.2, only the penalty term of blood glucose is described in this section, and the penalty term of  $u_i$ ,  $\Delta u_i$ ,  $u_g$  and  $\Delta u_g$  in discrete form will be introduced in detail in 4.3.2.

### 4.3.2 Discretization of optimal control problem

In order to solve the OCP problem with numerical methods, the continuous OCP problem needs to be discretized, also known as the direct method. Direct method includes direct single shooting method, direct multiple shooting method, and direct collocation method.

For the NMPC of artificial pancreas(AP), direct multiple shooting is used to discretize the continuous OCP problem. We divide the control horizon,  $[t_0; t_f]$ , into  $N$  intervals each of length  $T_s$ . Let  $\mathcal{N} = \{0, 1, \dots, N-1\}$  and  $t_k = t_0 + kT_s$  with  $k \in \mathcal{N}$ . Manipulated variables  $u$  are discretized by applying zero-order-hold parameterization.

$$u(t) = u_k \quad t_k \leq t < t_{k+1} \quad k \in \mathcal{N} \quad (4.13)$$

In contrast to the single shooting method, direct multiple shooting method also discretizes the integration of model equations and state variables  $x$  into  $N$  intervals. The discretized state variables  $x$  is added to the decision variable and form equality constraint with the corresponding integration of dynamic Problem decision variables

$$W = [u_0, x_0, \dots, u_{N-1}, x_{N-1}, x_N] \quad (4.14)$$

$$F(x_k, u_k, d_k, p) - x_{k+1} = 0 \quad k \in \mathcal{N} \quad (4.15)$$

The OCP problem in discrete form can be expressed as

$$\min_{\{x_0, x_{k+1}, u_k\}_{k=0}^{N-1}} \Phi = \sum_{k=0}^{N-1} G_k(z_k, u_k, d_k, p) + \lambda(z_N, p) \quad (4.16a)$$

$$\text{s.t.} \quad x_0 = \bar{x}_0 \quad (4.16b)$$

$$F_k(x_k, u_k, d_k, p) - x_{k+1} = 0 \quad k \in \mathcal{N} \quad (4.16c)$$

$$H_k(x_k, p) - z_k = 0 \quad k \in \mathcal{N} \quad (4.16d)$$

$$u_{\min} \leq u_k \leq u_{\max} \quad k \in \mathcal{N} \quad (4.16e)$$

The discrete stage cost term is

$$G_k(z_k, u_k, d_k, p) = \left\{ \int_{t_k}^{t_{k+1}} g(z(t), u_k) dt : z_k = H(x_k, p), \dot{x}(t) = f(x(t), u_k, d_k, p), x(t_k) = x_k \right\} \quad (4.17)$$

The discrete-time state transition function is

$$F_k(x_k, u_k, d_k, p) = \{x(t_{k+1}) : \dot{x}(t) = f(x(t), u_k, d_k, p), x(t_k) = x_k\} \quad (4.18)$$

The discrete OCP problem can be expressed in NLP form and solved using numerical nonlinear optimization methods such as SQP algorithm and interior-point algorithm. The general formulation of NLP problem is

$$\min_w \quad f(w, p) \quad (4.19a)$$

$$\text{s.t.} \quad g_{eq}(w) = 0 \quad (4.19b)$$

$$g_{ieq}(w) \geq 0 \quad (4.19c)$$

$$w_l \leq w \leq w_u \quad (4.19d)$$

For the problem of artificial pancreas, the formulation of NLP problem for single-hormone system can be further expressed as

$$\begin{aligned} \min_{\{x_0, x_{k+1}, u_k\}_{k=0}^{N-1}} \quad & \Phi = \Phi_z + \Phi_u(\Phi_{ui}) \\ = \quad & \underbrace{\sum_{k=0}^{N-1} \rho_z(z_k)}_{\text{BG penalty terms}} + \underbrace{\sum_{k=0}^{N-1} \rho_{ui}(u_{i,k}) + \sum_{k=0}^{N-1} \rho_{\Delta ui}(\Delta u_{i,k})}_{\text{Insulin penalty terms}} \end{aligned} \quad (4.20a)$$

$$\text{s.t.} \quad g_{eq}(w) = \begin{bmatrix} \bar{x}_0 - x_0 \\ F(x_0, u_0, d_0, p) - x_1 \\ \vdots \\ F(x_{N-1}, u_{N-1}, d_{N-1}, p) - x_N \end{bmatrix} = 0 \quad (4.20b)$$

$$u_{min} \leq u_k \leq u_{max} \quad (4.20c)$$

Where

$$u_k = u_{i,k}$$

The formulation of NLP problem for multiple-hormone system can be further expressed as

$$\begin{aligned} \min_{\{x_0, x_{k+1}, u_k\}_{k=0}^{N-1}} \quad & \Phi = \Phi_z + \Phi_u (\Phi_{ui} + \Phi_{ug}) \\ = \quad & \underbrace{\sum_{k=0}^{N-1} \rho_z(z_k)}_{\text{BG penalty terms}} + \underbrace{\sum_{k=0}^{N-1} \rho_{ui}(u_{i,k})}_{\text{Insulin penalty terms}} + \sum_{k=0}^{N-1} \rho_{\Delta ui}(\Delta u_{i,k}) \\ & + \underbrace{\sum_{k=0}^{N-1} \rho_{ug}(u_{g,k})}_{\text{Glucagon penalty terms}} + \sum_{k=0}^{N-1} \rho_{\Delta ug}(\Delta u_{g,k}) \end{aligned} \quad (4.21a)$$

$$\text{s.t.} \quad g_{eq}(w) = \begin{bmatrix} \bar{x}_0 - x_0 \\ F(x_0, u_0, d_0, p) - x_1 \\ \vdots \\ F(x_{N-1}, u_{N-1}, d_{N-1}, p) - x_N \end{bmatrix} = 0 \quad (4.21b)$$

$$u_{min} \leq u_k \leq u_{max} \quad (4.21c)$$

Where

$$u_k = \begin{bmatrix} u_{i,k} \\ u_{g,k} \end{bmatrix}$$

The penalty term of the manipulated variable  $u_i$ (including  $u_{ba}$  and  $u_{bo}$ ) and  $u_g$  and their change rate  $\Delta u_i$ (including  $\Delta u_{ba}$  and  $\Delta u_{bo}$ ) and  $\Delta u_g$  include the  $\ell_1$  norm and  $\ell_2$  norm penalty form, which can be generally expressed as

$$\rho_{ui}(u_{i,k}) = \frac{1}{2} \|u_{i,k} - \bar{u}_{i,k}\|_{2,Q_{ui,2}}^2 + \|u_{i,k} - \bar{u}_{i,k}\|_{1,Q_{ui,1}} \quad (4.22)$$

$$\rho_{\Delta ui}(\Delta u_{i,k}) = \frac{1}{2} \|\Delta u_{i,k}\|_{2,Q_{\Delta ui,2}}^2 + \|\Delta u_{i,k}\|_{1,Q_{\Delta ui,1}} \quad (4.23)$$

$$\rho_{ug}(u_{g,k}) = \frac{1}{2} \|u_{g,k} - \bar{u}_{g,k}\|_{2,Q_{ug,2}}^2 + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}} \quad (4.24)$$

$$\rho_{\Delta ug}(\Delta u_{g,k}) = \frac{1}{2} \|\Delta u_{g,k}\|_{2,Q_{\Delta ug,2}}^2 + \|\Delta u_{g,k}\|_{1,Q_{\Delta ug,1}} \quad (4.25)$$

Where

$$u_{i,k} = \begin{bmatrix} u_{ba,k} \\ u_{bo,k} \end{bmatrix} \quad \bar{u}_{i,k} = \begin{bmatrix} \bar{u}_{ba,k} \\ \bar{u}_{bo,k} \end{bmatrix} \quad \Delta u_{i,k} = \begin{bmatrix} \Delta u_{ba,k} \\ \Delta u_{bo,k} \end{bmatrix}$$

Where  $\bar{u}_{ba,k}$  and  $\bar{u}_{bo,k}$  is the target input of basal insulin and bolus insulin.

In the insulin treatment of artificial pancreas(AP), basal insulin and bolus insulin are used together to maintain the blood glucose concentration in the normal range. The

role of basal insulin is to keep blood glucose constant during fasting, so the input of basal insulin should not change too drastically. Especially under the influence of sensor noise and process noise, the calculated input amount of basal insulin may change suddenly. The following two forms of penalty can achieve the restraint effect on basal insulin:

- $\frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_2^2$  : As the target value of basal insulin,  $\bar{u}_{ba,k}$  can be the doctor's recommended dose of basal insulin for a specific patient, or it can be the dose of basal insulin calculated by the patient under the steady-state of normal blood glucose. The  $\ell_2$  norm penalty makes the  $u_{ba,k}$  calculated by the controller as close to the target  $\bar{u}_{ba,k}$  as possible, so as to achieve the purpose of preventing  $u_{ba,k}$  from changing drastically.
- $\frac{1}{2} \|\Delta u_{ba,k}\|_2^2$  : The  $\ell_2$  norm penalty is directly applied to  $\Delta u_{ba,k}$  to keep  $u_{ba,k}$  a relatively smooth change.

Insulin bolus is mainly injected at time of meals intake to prevent high blood glucose after meals. So the bolus insulin should be kept at 0 during the fasting period and injected in large amounts at the time of meals. The following form of penalty can achieve the constraint effect on bolus insulin:

- $\|u_{bo,k} - \bar{u}_{bo,k}\|_1$  : The  $\ell_1$  norm penalty is directly applied to  $u_{bo,k}$  and the target bolus insulin  $\bar{u}_{bo,k}$  is set to 0. During the fasting period, the blood glucose rises less and bolus insulin is punished, so basal insulin injection is given priority. On the contrary, when eating, blood glucose rises greatly, and the change of basal insulin is constrained, so a large dose of bolus insulin is used to infuse.

Glucagon is mainly injected in large doses when the blood glucose is lower than the normal range, such as hypoglycemia caused by exercise or excessive insulin injection. The following form of penalty can achieve the constraint effect on glucagon:

- $\|u_{g,k} - \bar{u}_{g,k}\|_1$  : The  $\ell_1$  norm penalty is directly applied to  $u_{g,k}$  and the target glucagon  $\bar{u}_{g,k}$  is set to 0. The form of penalty for glucagon is the same as that of bolus insulin, so that it can be injected in large dose only when needed.

So far, all penalty forms have been introduced. Among them, the penalty term of controlled variable  $z$  are deterministic, but for the penalty term of manipulated variable  $u$ , multiple forms of penalty can be combined to achieve the purpose of control. Therefore, in the closed-loop simulation of section 4.5, we test the NMPC algorithm of the objective function of different penalty forms(different combinations of the penalty terms of manipulated variable  $u$  including  $u_{ba}$ ,  $u_{bo}$  and  $u_g$ ) and the results is compared and discussed.

### 4.3.3 Smoothing function of penalty term

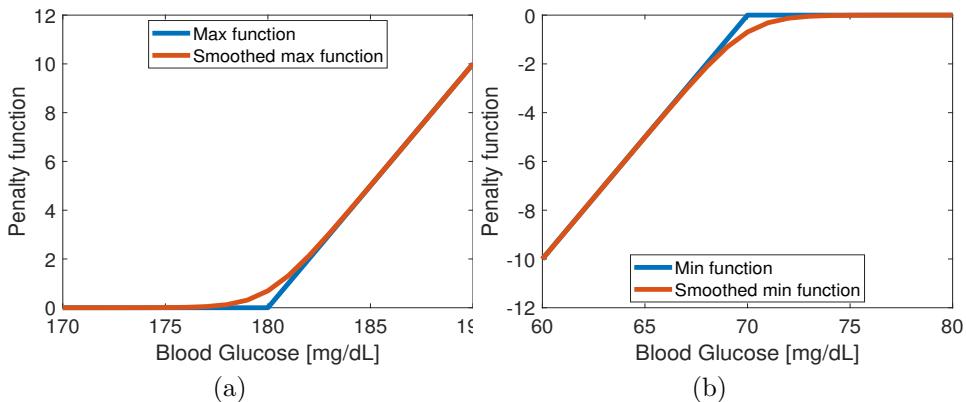
In the description of the objective function, the maximum value function ("max" in Matlab) and the minimum value function ("min" in Matlab) are used in the penalty term of the controlled variable  $z$ , and the absolute value function ("abs" in Matlab) is used in the penalty term of manipulated variable  $u$ . When the NLP solver works, the gradient information of the objective function is needed. Therefore, a suitable smooth function needs to be found to approximate the above non-smooth function.

Logarithm function and exponential function are used to form smooth functions of maximum value function and minimum value function

$$\max(x, y) \approx \log(\exp(x) + \exp(y)) \quad (4.26)$$

$$\min(x, y) \approx -\log(\exp(-x) + \exp(-y)) \quad (4.27)$$

Taking  $\max\{z(t) - z_{\max}, 0\}$  ( $z_{\max} = 180$ ) and  $\min\{z(t) - z_{\min}, 0\}$  ( $z_{\min} = 70$ ) used in the objective function as an example, (a) and (b) of Figure 4.2 show the approximate effect of the smoothing function

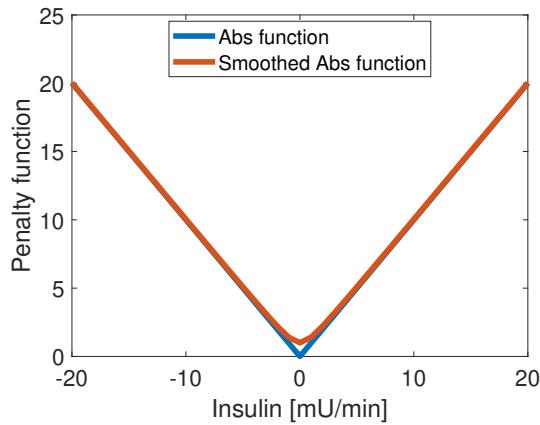


**Figure 4.2:** Smoothing function of max and min.

Square function and square root function are used to form smooth functions of absolute value function

$$\text{abs}(x) \approx \sqrt{x^2 + \beta^2} \quad (4.28)$$

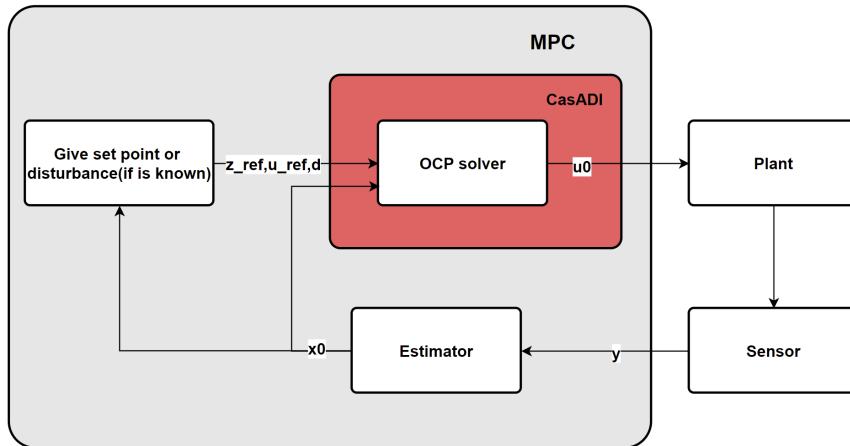
The smoothing function is related to the size of parameter  $\beta$ . The smaller the beta, the closer the smoothing function is to the "abs" function. In this thesis,  $\beta = 0.1$ . Taking  $\|u_{bo,k} - \bar{u}_{bo,k}\|_1$  ( $\bar{u}_{bo,k} = 0$ ) used in the objective function as an example, Figure 4.3 show the approximate effect of the smoothing function.



**Figure 4.3:** Smoothing function of abs.

## 4.4 Implementation of NLP problem in CasADI

CasADI is an open-source software tool for numerical optimization in general and optimal control in particular[And13]. It is worth mentioning that CasADI is not an optimal control solver, but provides users with a set of powerful high-level building blocks to help users build their own general-purpose or specific-purpose OCP solvers more simply and effectively. In this section, the nonlinear model predictive control(NMPC) algorithm of single-hormone system for artificial pancreas(AP) and insulin dosing systems is taken as an example to introduce how to use CasADI to implement the NMPC algorithm.



**Figure 4.4:** Block-diagram of close loop simulation with MPC using CasADI.

The "Building blocks" provided by CasADI is to help users establish the NLP problem form of the OCP problem that needs to be solved, and call the numerical nonlinear programming interface for calculation. In this section, basic syntax of CasADI is first introduced, and the implementation of the NLP problem Transcribed from OCP problem of artificial pancreas by CasADI is discussed.

### 4.4.1 Symbolic framework and function objects

Similar to Matlab's symbolic calculation, CasADI has a self-contained symbolic framework that allows the user to establish symbolic expressions using a syntax of everything-is-a-matrix. All matrices are sparse and are stored by a general sparse format com-

pressed column storage (CCS). In addition, CasADI allows the user to create function objects which defined by a symbolic expression. A very important and basic function of CasADI is automatic differentiation(AD), which plays an important role in the implementation of NLP problems, such as when NLP solver is used, the CasADI's interface will automatically generate the information that it needs to solve the NLP. But the AD tool of CasADI can only be used under its own symbolic framework, which means that all equations, models and functions used in NLP problems need to be written in CasADI's syntax.

Several basic symbolic data types and function definition will be introduced.

#### 4.4.1.1 The SX symbolics

The SX data type is used to represent matrices whose elements consist of symbolic expressions made up by a sequence of unary and binary operations.

```

1 %Matlab
2 x = SX.sym('x');
3 y = SX.sym('y', 5);
4 z = SX.sym('z', 4, 2);
```

```

1 //C++
2 SX x = SX::sym("x");
3 SX y = SX::sym("y", 5);
4 SX z = SX::sym("z", 4, 2);
```

#### 4.4.1.2 The MX symbolics

The MX type is a more general matrix expression type, and allows, like SX, to build up expressions consisting of a sequence of elementary operations. But unlike SX, these elementary operations are not restricted to be scalar unary or binary operations.

```

1 %Matlab
2 x = MX.sym('x');
3 y = MX.sym('y', 5);
4 z = MX.sym('z', 4, 2);
```

```

1 //C++
2 MX x = MX::sym("x");
3 MX y = MX::sym("y", 5);
4 MX z = MX::sym("z", 4, 2);
```

#### 4.4.1.3 Combination of MX and SX

Functions defined by SX expressions have a much lower cost for each operation than ones by MX. The SX expressions are more suitable to be used for low level operations. MX can be more economical when working with operations that are naturally vector or matrix valued with many elements.

#### 4.4.1.4 Function objects

Function objects are generally defined with the syntax

```
1 f = functionname(name, arguments,...,[options])
```

```
1 %Matlab
2 f = Function('f,{x,y},{x+y}')
```

```
1 //C++
2 Function f = Function("f",{x,y},{x+y})
```

The options structure is a dictionary type in Python, a struct in MATLAB or CasADi's Dict type in C++.

#### 4.4.2 Mathematical Model

MPC is a model based control method. In the NLP problem transcribed from the OCP problem, both the implementation of objective function and constraints need to be model-based, so the function of the model equation and its numerical integration need to be written in CasADI syntax. In CasADI, the model parameters are initialized, and the variables of the MVP model are declared, including 7 states, 2 manipulated variables and 1 disturbance.

```

1 %Matlab
2 k1 = 0.0182; CI = 1.5; SI = 0.00092; EGP = 1.25;
3 km = 0.027; VG = 200; tau_GI = 6.7; rou4 = 3.05;
4 p = [k1; CI; SI; EGP; km; VG; tau_GI; rou4];
5
6 x1 = SX.sym('x1'); x2 = SX.sym('x2'); x3 = SX.sym('x3');
7 x4 = SX.sym('x4'); x5 = SX.sym('x5'); x6 = SX.sym('x6');
8 x7 = SX.sym('x7');
9 states = [x1;x2;x3;x4;x5;x6;x7]; n_states = length(states);
10 u1 = SX.sym('u1'); u2 = SX.sym('u2');
11 controls = [u1;u2]; n_controls = length(controls);
12 d = SX.sym('d');
13 disturbance=d;
```

```

1 //C++
2 MVP_parameters MVP_param = default_parameters();
3
4 SX x1 = SX::sym("x1"); SX x2 = SX::sym("x2"); SX x3 = SX::sym("x3");
5 SX x4 = SX::sym("x4"); SX x5 = SX::sym("x5"); SX x6 = SX::sym("x6"); SX x7 = SX::sym("x7");
6 vector<SX> states = {x1,x2,x3,x4,x5,x6,x7};
7 int n_states = states.size();
8 SX states_arr = vertcat(states);
9
10 SX u1 = SX::sym("u1"); SX u2 = SX::sym("u2");// control
11 vector<SX> controls = {u1,u2};
12 int n_controls = controls.size();
13 SX controls_arr = vertcat(controls);
14
15 SX d = SX::sym("d");
16 vector<SX> disturbance = {d};
17 SX disturbance_arr = vertcat(disturbance);
```

The function of MVP model is calculated directly based on the declared symbol variable and defined as

```

1 %Matlab
2 xdot = [p(1)*((u1+u2)/p(2)-x1);
3 p(1)*(x1-x2);
4 -p(1)*x3+p(1)*p(3)*x2;
5 -x3*x4+p(4)+(x6*p(5)/p(6));
6 d-x5*p(5);
7 (x5-x6)*p(5);
8 (x4-x7)/p(7)];
9 f=Function('f',{states,controls,disturbance},{xdot});
```

```

1 //C++
2 vector<SX> xdot(n_states);
3 xdot[0] = MVP_param.k1*((u1+u2)/MVP_param.CI-x1);
4 xdot[1] = MVP_param.k1*(x1-x2);
5 xdot[2] = -MVP_param.k1*x3+MVP_param.k1*MVP_param.SI*x2;
6 xdot[3] = -x3*x4+MVP_param.EGP+(x6*MVP_param.km/MVP_param.VG);
7 xdot[4] = d-x5*MVP_param.km;
8 xdot[5] = (x5-x6)*MVP_param.km;
9 xdot[6] = (x4-x7)/MVP_param.tau_GI;
10
11 vector<SX> XUD(1,states_arr);
12 XUD.push_back(controls_arr);
13 XUD.push_back(disturbance_arr);
14 Function f = Function("f", XUD, xdot);
```

Fixed step Explicit Runge-Kutta 4(ERK4) integrator with 4 intervals is used for numerical integration.

$$k_1 = f(x_k) \quad (4.29a)$$

$$k_2 = f\left(x_k + h \frac{k_1}{2}\right) \quad (4.29b)$$

$$k_3 = f\left(x_k + h \frac{k_2}{2}\right) \quad (4.29c)$$

$$k_4 = f(x_k + hk_3) \quad (4.29d)$$

$$x_{k+1} = x_k + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4) \quad (4.29e)$$

Based on the state function  $\dot{x}(t) = f(x(t), u(t), d(t), p)$ , the numerical integral function of the state of the model can be defined.

```

1 %Matlab
2 T = 5; M = 4; DT = T/M;
3 states_RK4 = states;
4 for j=1:M
5     RK1 = f(states_RK4, controls,disturbance);
6     RK2 = f(states_RK4 + DT/2 * RK1, controls,disturbance);
7     RK3 = f(states_RK4 + DT/2 * RK2, controls,disturbance);
8     RK4 = f(states_RK4 + DT * RK3, controls,disturbance);
9     states_RK4 = states_RK4+DT/6*(RK1 +2*RK2 +2*RK3 +RK4);
10 end
11 fun_RK4 = Function('F', {states, controls, disturbance}, {states_RK4}, {'xk','u','d'}, {'xk1'});

```

```

1 //C++
2 const double T = 5;//[min]
3 const int M = 4;
4 double DT = T/M;
5 SX states_arr_RK4 = states_arr, states_arr_tmp = states_arr;
6 vector<SX> XUD_RK4, RK1(1),RK2(1),RK3(1),RK4(1);
7 for(int i=0;i<M;++i){
8     XUD_RK4.push_back(states_arr_RK4), XUD_RK4.push_back(controls_arr);
9     XUD_RK4.push_back(disturbance_arr);
10    RK1 = f(XUD_RK4);
11    XUD_RK4.clear();
12
13    states_arr_tmp = states_arr_RK4+DT/2*vertcat(RK1);
14    XUD_RK4.push_back(states_arr_tmp), XUD_RK4.push_back(controls_arr);
15    XUD_RK4.push_back(disturbance_arr);
16    RK2 = f(XUD_RK4);
17    XUD_RK4.clear();
18
19    states_arr_tmp = states_arr_RK4+DT/2*vertcat(RK2);
20    XUD_RK4.push_back(states_arr_tmp), XUD_RK4.push_back(controls_arr);
21    XUD_RK4.push_back(disturbance_arr);
22    RK3 = f(XUD_RK4);
23    XUD_RK4.clear();
24
25    states_arr_tmp = states_arr_RK4+DT*vertcat(RK3);
26    XUD_RK4.push_back(states_arr_tmp), XUD_RK4.push_back(controls_arr);
27    XUD_RK4.push_back(disturbance_arr);
28    RK4 = f(XUD_RK4);
29    XUD_RK4.clear();
30
31    states_arr_RK4 = states_arr_RK4 + DT/6*(vertcat(RK1)+2*vertcat(RK2)+2*vertcat(RK3)+vertcat(RK4));
32 }
33 Function fun_RK4("F", XUD,vector<SX>{states_arr_RK4});

```

### 4.4.3 Self-defined functions

Before constructing the NLP problem, in addition to the two self-defined functions in the previous section (the state function  $\dot{x} = f(x, u, d)$  and its numerical integration function  $x_{k+1} = \text{fun\_RK4}(x_k, u_k, d_k)$ ), other self-defined functions to be used in the calculation of the objective function and constraint conditions also need to be written in CasADI syntax.

In the problem of artificial pancreas, in order to implement soft constraints on the controlled variable  $z$ , it is necessary to smooth and define the min function and the max function. In addition, in order to implement soft constraints on the manipulated variable  $u_{bo}$  of bolus insulin, the absolute value function needs to be smoothed and defined.

In CasADI, the functions is defined as

```

1 %Matlab
2 x_max = log(exp(x1)+exp(x2));
3 x_min = -log(exp(-x1)+exp(-x2));
4 x_abs = sqrt(x1*x1+1);
5 myMax = Function('myMax',{x1,x2},{x_max});
6 myMin = Function('myMin',{x1,x2},{x_min});
7 myAbs = Function('myAbs',{x1},{x_abs});
```

```

1 //C++
2 SX val_max = log(exp(x1)+exp(x2));
3 SX val_min = -log(exp(-x1)+exp(-x2));
4 SX val_abs = sqrt(x1*x1+1);
5 Function myMax("myMax",{x1,x2},{val_max});
6 Function myMin("myMin",{x1,x2},{val_min});
7 Function myAbs("myAbs",{x1},{val_abs});
```

#### 4.4.4 Declaration of decision variables

Starting from this chapter, the construction of NLP problems in CasADi will be introduced. The NLP solver interface in CasADi allows NLP problems of the following form

$$\min_W f(W, P) \quad (4.30a)$$

$$\text{s.t.} \quad W_{lb} \leq W \leq W_{ub} \quad (4.30b)$$

$$g_{lb} \leq g(W, P) \leq g_{ub} \quad (4.30c)$$

In CasADi, since the objective function and constraint conditions of the NLP problem are all represented by symbolic variables in CasADi, decision variables  $W$  and parameter variables  $P$  must first be declared as symbolic variables of CasADi.

$$W = [u_0, x_0, \dots, u_{N-1}, x_{N-1}, x_N] \quad (4.31)$$

$$P = [\bar{x}_0, \bar{z}, d] \quad (4.32)$$

It is worth mentioning that among the parameter variables,  $P$ , there are only the initial value of the model,  $\bar{x}_0$ , the set point of the controlled variable,  $\bar{z}$  and the disturbance,  $d$ . The model parameters or the weight value of the objective function and other parameters are not declared as CasADi symbolic variables. This is because the NMPC algorithm solves an OCP problem in each iteration, and each time the parameter declared as a symbolic variable will change, so every time the NLP solver of CasADi is called, the  $P$  will be reassigned. In contrast, parameters such as model parameters are invariant values, so they do not need to be declared as CasADi symbolic variables

In the artificial pancreas(AP) problem of, the control horizon is set as  $[0; 360]$  [min] and is divided into  $N = 72$  intervals each of length  $T_s = 5$  [min].

```

1 %Matlab
2 T = 5;
3 N = 72;
4 U = SX.sym('U',n_controls,N);
5 X = SX.sym('X',n_states,N+1);
6 P = SX.sym('P',n_states +1 +1);
7 OPT_variables = [reshape(X,n_states*(N+1),1);reshape(U,n_controls*N,1)];
```

```

1 //C++
2 const double T = 5;
3 const int N =72;
4 MX X = MX::sym("X",n_states*(N+1));
5 MX U = MX::sym("U",n_controls*N);
6 MX P = MX::sym("P",n_states+1+1);
7 MX OPT_variables = vertcat(X,U);
```

#### 4.4.5 Objective function

In this section, the implementation of the objective function of the NLP problem in CasADi is discussed. In the artificial pancreas(AP) problem, the objective function consists of the penalty term of the controlled variable  $z$  and the penalty terms of the manipulated variables  $u_{ba}$  and  $u_{bo}$ .

$$\Phi = \underbrace{\sum_{k=0}^{N-1} \rho_z(z(t))}_{\text{BG penalty terms}} + \underbrace{\sum_{k=0}^{N-1} \rho_{ui}(u_{i,k}) + \sum_{k=0}^{N-1} \rho_{\Delta ui}(\Delta u_{i,k})}_{\text{Insulin penalty terms}} \quad (4.33)$$

The implementation of the objective function in CasADi is

```

1 %Matlab
2 Qz = 0.01; Qz_max = 1; Qz_min = 300000; Qu_bo = 100; Qu_delta_ba = 150;
3 z_max = 180; z_min = 70;
4 obj = 0; % Objective function
5 for k = 1:N
6     st = X(:,k); zk=st(4); con=U(:,k);
7     obj = obj+(zk-P(n_states+1))*Qz*(zk-P(n_states+1)) + myMax(zk-z_max,0)*Qz_max*myMax(zk-z_max,0) +
        myMin(zk-z_min,0)*Qz_min*myMin(zk-z_min,0)+Qu_bo*myAbs(con(2));
8     if k ~= N
9         con_next = U(:,k+1);
10        obj = obj+(con_next(1)-con(1))*Qu_delta_ba*(con_next(1)-con(1));
11    end
12 end

```

```

1 //C++
2 const double Qz = 0.01,Qz_max = 1,Qz_min = 300000,Qu_bo = 100,Qu_delta_ba = 150;
3 const double z_max = 180, z_min = 70;
4 vector<MX> X_vec(1,X(Slice(0,n_states),Slice()));
5 vector<MX> U_vec(1,U(Slice(0,n_controls),Slice()));
6 for(int i=1;i<N+1;++i){
7     X_vec.push_back(X(Slice(i*n_states,(i+1)*n_states),Slice()));
8     if(i!=N)
9         U_vec.push_back(U(Slice(i*n_controls,(i+1)*n_controls),Slice()));
10 }
11 MX obj = 0;
12 for(int i=0;i<N;++i){
13     MX st = X_vec[i], zk = st(3), con = U_vec[i];
14     MX max_k = vertcat(myMax({zk - z_max, 0}));
15     MX min_k = vertcat(myMin({zk - z_min, 0}));
16     MX abs_k = vertcat(myAbs(con(1)));
17     obj = obj + (zk-P(n_states))*Qz*(zk-P(n_states)) + max_k*Qz_max*max_k + min_k*Qz_min*min_k + Qu_bo
        *abs_k;
18     if(i!=N-1){

```

```

19     MX con_next = U_vec[i+1];
20     obj = obj + (con_next(0)-con(0))*Qu_delta_ba*(con_next(0)-con(0));
21 }
22 }
```

#### 4.4.6 Constraints

In this section, implementation of constraints of NLP problem in CasADi is introduced. In the artificial pancreas(AP) problem, the multiple shooting method is used and the state variables,  $x$  becomes a decision variable, and the corresponding equation constraint is a nonlinear equation constraint

$$g(W, P) = \begin{bmatrix} \bar{x}_0 - x_0 \\ F(x_0, u_0, d_0, p) - x_1 \\ \vdots \\ F(x_{N-1}, u_{N-1}, d_{N-1}, p) - x_N \end{bmatrix} = 0 \quad (4.34)$$

The implementation of the constraints in CasADi is

```

1 %Matlab
2 g = [];  
3 st = X(:,1); zk = st(4);
4 g = [g;st-P(1:n_states)]; % initial condition constraints
5 for k=1:N
6     st = X(:,k); zk = st(4); con = U(:,k);
7     st_next = X(:,k+1);
8     if k == 1
9         st_next_RK4 = fun_RK4(st,con,P(n_states+2));
10    else
11        st_next_RK4 = fun_RK4(st,con,0);
12    end
13    g = [g;st_next-st_next_RK4];
14 end
```

```

1 //C++
2 vector<MX> g;
3 MX st = X_vec[0], zk = st(3);
4 g.push_back(st-P(Slice(0,n_states),Slice()));
5 for(int i=0;i<N;++i){
6     st = X_vec[i], zk = st(3);
7     MX st_next = X_vec[i+1], con = U_vec[i], st_next_RK4;
```

```

8   if(i==0){
9     vector<MX> XUD_tmp(1,st);
10    XUD_tmp.push_back(con), XUD_tmp.push_back(P(n_states+1));
11    st_next_RK4 = vertcat(fun_RK4(XUD_tmp));
12  }
13  else{
14    vector<MX> XUD_tmp(1,st);
15    XUD_tmp.push_back(con), XUD_tmp.push_back(0);
16    st_next_RK4 = vertcat(fun_RK4(XUD_tmp));
17  }
18  g.push_back(st_next-st_next_RK4);
19}

```

#### 4.4.7 NLP solver

After decision variables, objective functions and nonlinear constraints are all represented by symbolic variables, CasADI's "nlpssol" function is used to create NLP solver objects. CasADI has several NLP solvers interfaced, such as IPOPT, SNOPT, WORHP and KNITRO. IPOPT, an open-source primal-dual interior point method which is included in CasADI installations, is used here.

```

1 %Matlab
2 nlp_prob = struct('f', obj, 'x', OPT_variables, 'g', g, 'p', P);
3 opts = struct;
4 opts.ipopt.max_iter = 500;
5 opts.ipopt.print_level = 3;
6 opts.print_time = 0;
7 opts.ipopt.acceptable_tol = 1e-8;
8 opts.ipopt.acceptable_obj_change_tol = 1e-8;
9 solver = nlpssol('solver', 'ipopt', nlp_prob,opts);

```

```

1 //C++
2 MXDict nlp_prob = {{"x", OPT_variables}, {"f", obj}, {"g", vertcat(g)}, {"p", P}};
3 Dict opts;
4 opts["ipopt.tol"] = 1e-8;
5 opts["ipopt.max_iter"] = 500;
6 opts["ipopt.print_level"] = 5;
7 Function solver = nlpssol("nlpssol", "ipopt", nlp_prob, opts);

```

After the NLP solver object "solver" is created, when the NMPC algorithm solves the OCP problem in each iteration, an initial guess of the decision variables needs to be given firstly. Except for the first iteration, the initial guesses required for all iterations are given by the optimal decision variables calculated in the previous iteration. In the problem of artificial pancreas, examples of implementation are

```

1 %Matlab
2 args = struct;
3 x0 = [12.5805; 12.5805; 0.0116; 108.0000; 0; 0; 108.0000];
4 u0 = zeros(N,n_controls);
5 X0 = repmat(x0,N+1,1);
6 args.x0 = [reshape(X0', n_states*(N+1), 1) ; reshape(u0',n_controls*N,1)];

```

```

1 //C++
2 map<string, vector<double>> args;
3 vector<double> x0 = {12.5805,12.5805, 0.0116, 108.0000, 0, 0, 108.0000};
4 vector<double> OPT_varX0,OPT_varU0(n_controls*N,0);
5 for(int i=0;i<N+1;++i)
6     OPT_varX0.insert(OPT_varX0.end(),x0.begin(),x0.end());
7 OPT_varX0.insert(OPT_varX0.end(),OPT_varU0.begin(),OPT_varU0.end());
8 args["x0"] = OPT_varX0;

```

The upper and lower bounds of nonlinear constraints and decision variable constraints also need to be given. It should be noted that the nonlinear constraints of CasADi's NLP solver interface only allow constraints to be inequality constraints. But as long as the upper and lower bounds are set to 0, it is equivalent to the equality constraint.

```

1 %Matlab
2 u_min = 0; u_max = inf;
3 x_min = -inf; x_max = inf;
4 args.lbg(1:n_states*(N+1),1) = 0;
5 args.ubg(1:n_states*(N+1),1) = 0;
6 args.lbx(1:n_states*(N+1),1) = x_min;
7 args.ubx(1:n_states*(N+1),1) = x_max;
8 args.lbx(n_states*(N+1)+1:n_states*(N+1)+n_controls*N,1) = u_min;
9 args.ubx(n_states*(N+1)+1:n_states*(N+1)+n_controls*N,1) = u_max;
```

```

1 //C++
2 const double u_min = 0, u_max = DBL_MAX, x_min = -DBL_MAX, x_max = DBL_MAX;
3 vector<double> OPT_varXmin(n_states*(N+1),x_min);
4 vector<double> OPT_varXmax(n_states*(N+1),x_max);
5 vector<double> OPT_varUmin(n_controls*N,u_min);
6 vector<double> OPT_varUmax(n_controls*N,u_max);
7 OPT_varXmin.insert(OPT_varXmin.end(),OPT_varUmin.begin(),OPT_varUmin.end());
8 OPT_varXmax.insert(OPT_varXmax.end(),OPT_varUmax.begin(),OPT_varUmax.end());
9 args["lbx"] = OPT_varXmin;
10 args["ubx"] = OPT_varXmax;
11 args["lbg"] = vector<double>(n_states*(N+1),0);
12 args["ubg"] = vector<double>(n_states*(N+1),0);
```

Parameters,  $P = [\bar{x}_0, \bar{z}, d]$  declared as symbolic variables also need to be given. The initial value of the model state,  $\bar{x}_0$  is calculated by continuous-discrete extended Kalman filter(CDEKF), and the disturbance(meal intake [g]) is given before the meal.

```
1 %Matlab
2 x_bar = x0; z_ref = 108; d_meal = 50;
3 args.p = [x_bar;z_ref;d_meal*1000];
```

```
1 //C++
2 vector<double> x_bar = x0;
3 double z_ref = 108, d_meal = 50;
4 vector<double> OPT_p = x_bar;
5 OPT_p.push_back(z_ref);
6 OPT_p.push_back(d_meal);
7 args["p"] = OPT_p;
```

The NLP solver object "solver" is called to solve the NLP problem. The NMPC algorithm only executes the calculated input,  $u_k$  at the current time point, and the optimal decision variable for the entire control horizon,  $u_{hor}$  will be used as the initial guess for the next iterative calculation.

```
1 %Matlab
2 sol = solver('x0', args.x0, 'lbx', args.lbx, 'ubx', args.ubx,'lbg', args.lbg, 'ubg', args.ubg,'p',args
    .p);
3 u_hor = reshape(full(sol.x(n_states*(N+1)+1:end)'),n_controls,N);
4 uk = u_hor(:,1);
5 u0 = [u_hor(:,2:size(u_hor,2)),u_hor(:,size(u_hor,2))]';
6 X0 = reshape(full(sol.x(1:n_states*(N+1))),n_states,N+1)';
7 X0 = [X0(2:end,:);X0(end,:)];
```

```
1 //C++
2 vector<double> sol,xsol,usol,uk;
3 solver({{"lbx", args["lbx"]}, {"ubx", args["ubx"]}, {"x0", args["x0"]}, {"lbg", args["lbg"]}, {"ubg",
    args["ubg"]}, {"p",args["p"]}}, {{"x", &sol}});
4 xsol.insert(xsol.begin(),sol.begin(),sol.begin()+n_states*(N+1));
5 usol.insert(usol.begin(),sol.begin()+n_states*(N+1),sol.end());
6 uk.insert(uk.begin(),usol.begin(),usol.begin()+2);
7 OPT_varX0.clear();
8 OPT_varX0.insert(OPT_varX0.begin(),xsol.begin()+n_states,xsol.end());
9 OPT_varX0.insert(OPT_varX0.end(),xsol.begin()+n_states*N, xsol.end());
10 OPT_varX0.insert(OPT_varX0.end(),usol.begin()+n_controls,usol.end());
11 OPT_varX0.insert(OPT_varX0.end(),usol.begin()+n_controls*(N-1),usol.end());
```

## 4.5 Closed-loop simulation test

In the previous part, the NMPC algorithm for the artificial pancreas(AP) problem is introduced and implemented. In this section, the penalty function form in the objective function of the NMPC algorithm will be firstly discussed. Specifically, for the single-hormone and multiple-hormone system, the NMPC algorithm with different combinations of penalty functions will be tested in the closed-loop to compare the control effect. In addition, the NMPC algorithm will be applied to 10 virtual patients generated by the Hovorka Model and its parameter distribution to test the effectiveness and robustness of the algorithm. The parameters of 10 virtual patients is shown in Table A.1 and A.2 in Appendix A. The parameters of extended MVP model for 10 virtual patients is shown in Table 3.5 in section 3.

In the closed-loop simulation test, the single-hormone system and the multiple-hormone system use different extended Hovorka models (SDE version) as simulation models, and the Euler-Maruyama method with a fixed step size is used as a numerical integration method. The simulation time length is 24h, and the sampling time is 5min. The sampling time of the NMPC algorithm is also 5 minutes, and the control horizon is 6 hours, which is 72 sampling times. The meal size and time of the virtual patient are:

**Table 4.1:** Size and time of meals.

	Breakfast	Lunch	Dinner
Time	6:00	12:00	18:00
Carbohydrates(g)	50	60	70

The size of the meal is only informed to the NMPC algorithm at the time of meal intake. In the closed-loop simulation test of the multiple-hormone system, the exercise model is included in the simulation model. In addition to food intake, the increase in heart rate caused by exercise is also included as disturbance in the simulation. In the 24-hour simulation, the basic heart rate is 70 beats per minute(BPM), and the heart rate of 150 beats per minute is designed to be maintained between 15:00 and 15:45.

### 4.5.1 Comparison of closed-loop test of different forms for objective function

In order to achieve the best control performance, taking virtual patient 1 as an example, NMPC algorithms using different forms of objective functions are tested and compared in closed-loop simulations. For the single-hormone system, the penalty term of the controlled variable  $z$  in the objective function is as described above, and the penalty term of the manipulated variable  $u$  is divided into four cases. In addition, the weight matrix of each case is selected to achieve the best control effect through tuning.

(a)

$$\Phi_u = \Phi_{ui} = \frac{1}{2} \|\Delta u_{ba,k}\|_{2,Q_{\Delta uba,2}}^2 \quad (4.35a)$$

$$= \frac{1}{2} \|\Delta u_{i,k}\|_{2,Q_{\Delta ui,2}}^2 \quad (4.35b)$$

Where

$$\Delta u_{i,k} = \begin{bmatrix} \Delta u_{ba,k} \\ \Delta u_{bo,k} \end{bmatrix} \quad Q_{\Delta ui,2} = \begin{bmatrix} Q_{\Delta uba,2} & 0 \\ 0 & Q_{\Delta ubo,2} \end{bmatrix} = \begin{bmatrix} 150 & 0 \\ 0 & 0 \end{bmatrix}$$

(b)

$$\Phi_u = \Phi_{ui} = \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_{2,Q_{uba,2}}^2 \quad (4.36a)$$

$$= \frac{1}{2} \|u_{i,k} - \bar{u}_{i,k}\|_{2,Q_{ui,2}}^2 \quad (4.36b)$$

Where

$$u_{i,k} = \begin{bmatrix} u_{ba,k} \\ u_{bo,k} \end{bmatrix} \quad \bar{u}_{i,k} = \begin{bmatrix} \bar{u}_{ba,k} \\ \bar{u}_{bo,k} \end{bmatrix} = \begin{bmatrix} u_{ba,ss} \\ 0 \end{bmatrix} \quad Q_{ui,2} = \begin{bmatrix} Q_{uba,2} & 0 \\ 0 & Q_{ubo,2} \end{bmatrix} = \begin{bmatrix} 150 & 0 \\ 0 & 0 \end{bmatrix}$$

(c)

$$\Phi_u = \Phi_{ui} = \frac{1}{2} \|\Delta u_{ba,k}\|_{2,Q_{\Delta uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}} \quad (4.37a)$$

$$= \frac{1}{2} \|\Delta u_{i,k}\|_{2,Q_{\Delta ui,2}}^2 + \|u_{i,k} - \bar{u}_{i,k}\|_{1,Q_{ui,1}} \quad (4.37b)$$

Where

$$\begin{aligned}
 \text{a) } \Delta u_{i,k} &= \begin{bmatrix} \Delta u_{ba,k} \\ \Delta u_{bo,k} \end{bmatrix} & Q_{\Delta ui,2} &= \begin{bmatrix} Q_{\Delta uba,2} & 0 \\ 0 & Q_{\Delta ubo,2} \end{bmatrix} = \begin{bmatrix} 150 & 0 \\ 0 & 0 \end{bmatrix} \\
 \text{b) } u_{i,k} &= \begin{bmatrix} u_{ba,k} \\ u_{bo,k} \end{bmatrix} & \bar{u}_{i,k} &= \begin{bmatrix} \bar{u}_{ba,k} \\ \bar{u}_{bo,k} \end{bmatrix} = \begin{bmatrix} u_{ba,ss} \\ 0 \end{bmatrix} & Q_{ui,1} &= \begin{bmatrix} Q_{uba,1} & 0 \\ 0 & Q_{ubo,1} \end{bmatrix} =
 \end{aligned}$$

(d)

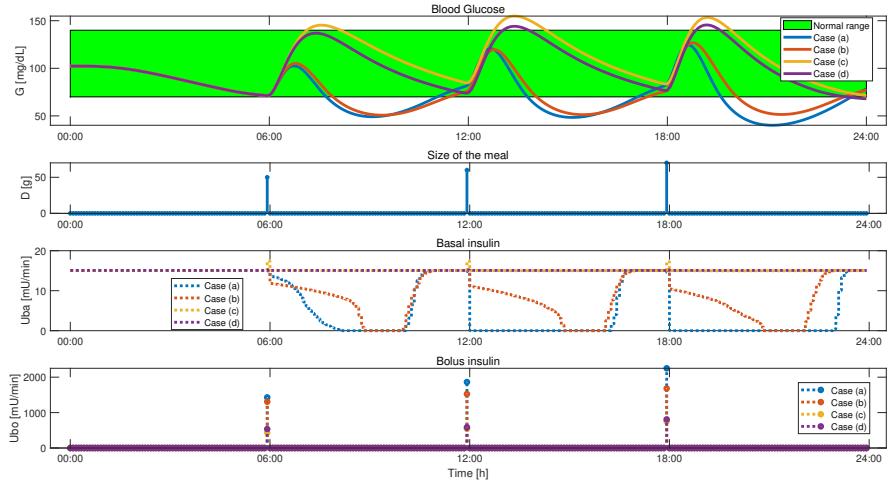
$$\Phi_u = \Phi_{ui} = \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_{2,Q_{ui,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ui,1}} \quad (4.38a)$$

$$= \frac{1}{2} \|u_{i,k} - \bar{u}_{i,k}\|_{2,Q_{ui,2}}^2 + \|u_{i,k} - \bar{u}_{i,k}\|_{1,Q_{ui,1}} \quad (4.38b)$$

Where

$$\begin{aligned}
 \text{a) } u_{i,k} &= \begin{bmatrix} u_{ba,k} \\ u_{bo,k} \end{bmatrix} & \bar{u}_{i,k} &= \begin{bmatrix} \bar{u}_{ba,k} \\ \bar{u}_{bo,k} \end{bmatrix} = \begin{bmatrix} u_{ba,ss} \\ 0 \end{bmatrix} & Q_{ui,2} &= \begin{bmatrix} Q_{uba,2} & 0 \\ 0 & Q_{ubo,2} \end{bmatrix} = \\
 &\begin{bmatrix} 150 & 0 \\ 0 & 0 \end{bmatrix} \\
 \text{b) } Q_{ui,1} &= \begin{bmatrix} Q_{uba,1} & 0 \\ 0 & Q_{ubo,1} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 100 \end{bmatrix}
 \end{aligned}$$

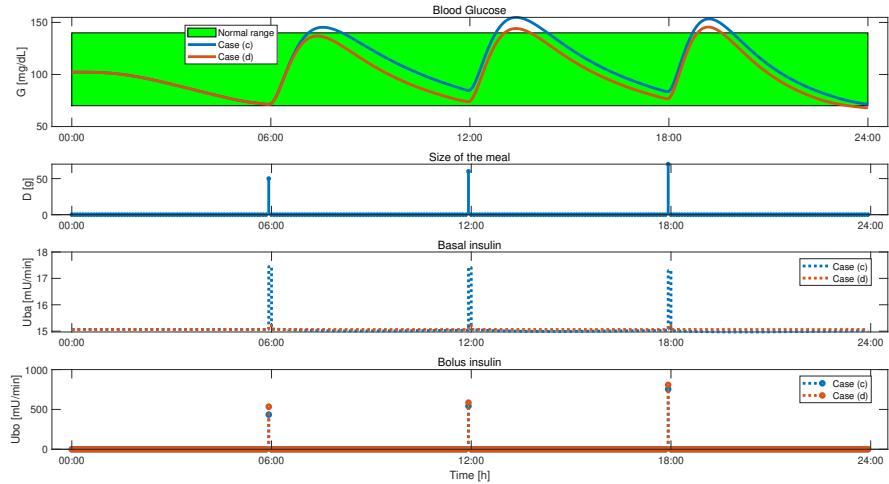
Since the randomly generated process noise and measurement noise can affect the results, the ODE version of the simulation model without measurement noise is used in closed-loop simulation. The Figure 4.5 shows the control performance of the four cases of NMPC of single-hormone for virtual patient 1.



**Figure 4.5:** Closed-loop test result of the four cases of NMPC of single-hormone for virtual patient 1 .

Compared with case (a), (b), case (c) and (d) have better control effect. Hypoglycemia occurred many times in case (a) and (b). On the contrary, The blood glucose concentration under the control of case (c) and (d) has been stable within the normal range. Since the objective function of case (a) and (b) only penalizes the basal insulin, when a large amount of meals are taken, bolus insulin is infused in large dose, which leads to the occurrence of hypoglycemia. When the  $\ell_1$  norm penalty of bolus insulin is added to the objective function of case (c) and (d), the dose of bolus insulin will not be too large, which avoids hypoglycemia. Therefore, the  $\ell_1$  norm penalty for bolus insulin is essential in the objective function.

In addition, cases (c) and (d) with better control effects apply different forms of penalty for basal insulin. The Figure 4.6 shows the controlled blood glucose in the closed-loop simulation test of case (c) and case (d) of single-hormone.



**Figure 4.6:** Closed-loop test result of the case (c) and (d) of NMPC of single-hormone for virtual patient 1.

Although both cases (c) and (d) have a good control effect, and there is no hypoglycemia. Compared with case (c), case (d) has better control effect, and there are fewer moments of hyperglycemia. In addition, the basal insulin calculated by case (d) has a smaller change rate, and an appropriate amount of bolus insulin is also infused at the moment of meal intake. In this thesis, the penalty form used by case (d) is  $\frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|^2$ , and the steady-state value of the basal insulin calculated in section 2.2.5 is used as the value of  $\bar{u}_{ba}$ . The accuracy of  $\bar{u}_{ba}$  determines the effect of the penalty form. Therefore, compared with the penalty form  $\frac{1}{2} \|\Delta u_{ba,k}\|^2$  used in case (c), when a more precise  $\bar{u}_{ba}$  value is given, the penalty form  $\frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|^2$  is expected to have a better control effect.

Therefore, for the single-hormone system, a better penalty function for the manipulated variable  $u$  is

$$\Phi_u = \Phi_{ui} = \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_{2,Q_{uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}} \quad (4.39)$$

Where  $\bar{u}_{ba,k} = u_{ba,ss}$ ,  $Q_{uba,2} = 150$ ,  $\bar{u}_{bo,k} = 0$  and  $Q_{ubo,1} = 100$ .

For the multiple-hormone system, the penalty term of the controlled variable  $z$  in the objective function is as described above, and the penalty term of the manipulated variable  $u$  is divided into four cases:

(a)

$$\Phi_u = \Phi_{ui} + \Phi_{ug} = \frac{1}{2} \|\Delta u_{ba,k}\|_{2,Q_{\Delta ui,2}}^2 + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}} \quad (4.40a)$$

$$= \frac{1}{2} \|\Delta u_{i,k}\|_{2,Q_{\Delta ui,2}}^2 + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}} \quad (4.40b)$$

Where

$$a) \Delta u_{i,k} = \begin{bmatrix} \Delta u_{ba,k} \\ \Delta u_{bo,k} \end{bmatrix} \quad Q_{\Delta ui,2} = \begin{bmatrix} Q_{\Delta uba,2} & 0 \\ 0 & Q_{\Delta ubo,2} \end{bmatrix} = \begin{bmatrix} 150 & 0 \\ 0 & 0 \end{bmatrix}$$

$$b) \bar{u}_{g,k} = 0 \quad Q_{ug,1} = 300$$

(b)

$$\Phi_u = \Phi_{ui} + \Phi_{ug} = \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_{2,Q_{ui,2}}^2 + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}} \quad (4.41a)$$

$$= \frac{1}{2} \|u_{i,k} - \bar{u}_{i,k}\|_{2,Q_{ui,2}}^2 + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}} \quad (4.41b)$$

Where

$$a) u_{i,k} = \begin{bmatrix} u_{ba,k} \\ u_{bo,k} \end{bmatrix} \quad \bar{u}_{i,k} = \begin{bmatrix} \bar{u}_{ba,k} \\ \bar{u}_{bo,k} \end{bmatrix} = \begin{bmatrix} u_{ba,ss} \\ 0 \end{bmatrix} \quad Q_{ui,2} = \begin{bmatrix} Q_{uba,2} & 0 \\ 0 & Q_{ubo,2} \end{bmatrix} = \begin{bmatrix} 150 & 0 \\ 0 & 0 \end{bmatrix}$$

$$b) \bar{u}_{g,k} = 0 \quad Q_{ug,1} = 300$$

(c)

$$\Phi_u = \Phi_{ui} + \Phi_{ug} = \frac{1}{2} \|\Delta u_{ba,k}\|_{2,Q_{\Delta ui,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ui,1}} + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}} \quad (4.42a)$$

$$= \frac{1}{2} \|\Delta u_{i,k}\|_{2,Q_{\Delta ui,2}}^2 + \|u_{i,k} - \bar{u}_{i,k}\|_{1,Q_{ui,1}} + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}} \quad (4.42b)$$

Where

$$a) \Delta u_{i,k} = \begin{bmatrix} \Delta u_{ba,k} \\ \Delta u_{bo,k} \end{bmatrix} \quad Q_{\Delta ui,2} = \begin{bmatrix} Q_{\Delta uba,2} & 0 \\ 0 & Q_{\Delta ubo,2} \end{bmatrix} = \begin{bmatrix} 150 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\text{b) } u_{i,k} = \begin{bmatrix} u_{ba,k} \\ u_{bo,k} \end{bmatrix} \quad \bar{u}_{i,k} = \begin{bmatrix} \bar{u}_{ba,k} \\ \bar{u}_{bo,k} \end{bmatrix} = \begin{bmatrix} u_{ba,ss} \\ 0 \end{bmatrix} \quad Q_{ui,1} = \begin{bmatrix} Q_{uba,1} & 0 \\ 0 & Q_{ubo,1} \end{bmatrix} =$$

$$\begin{bmatrix} 0 & 0 \\ 0 & 30 \end{bmatrix}$$

$$\text{c) } \bar{u}_{g,k} = 0 \quad Q_{ug,1} = 300$$

(d)

$$\Phi_u = \Phi_{ui} + \Phi_{ug} = \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|^2 + \|u_{bo,k} - \bar{u}_{bo,k}\| + \|u_{g,k} - \bar{u}_{g,k}\|_1 \quad (4.43\text{a})$$

$$= \frac{1}{2} \|u_{i,k} - \bar{u}_{i,k}\|_{2,Q_{ui,2}}^2 + \|u_{i,k} - \bar{u}_{i,k}\|_{1,Q_{ui,1}} + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}} \quad (4.43\text{b})$$

Where

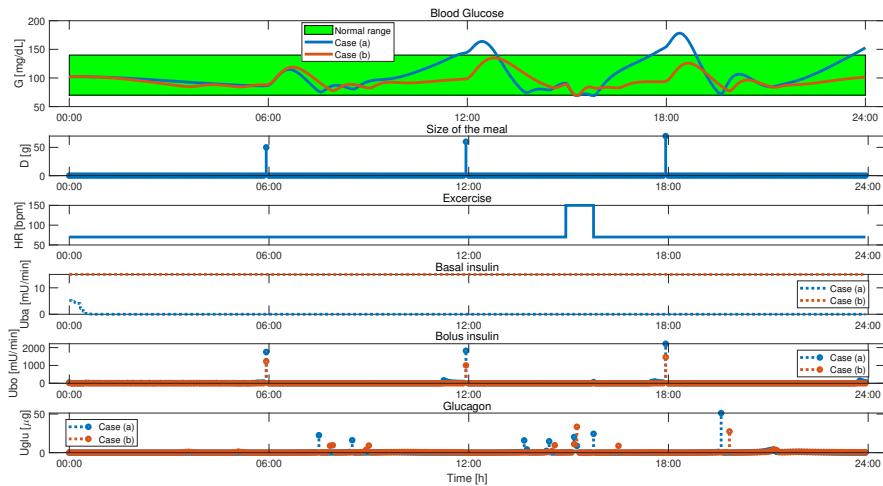
$$\text{a) } u_{i,k} = \begin{bmatrix} u_{ba,k} \\ u_{bo,k} \end{bmatrix} \quad \bar{u}_{i,k} = \begin{bmatrix} \bar{u}_{ba,k} \\ \bar{u}_{bo,k} \end{bmatrix} = \begin{bmatrix} u_{ba,ss} \\ 0 \end{bmatrix} \quad Q_{ui,2} = \begin{bmatrix} Q_{uba,2} & 0 \\ 0 & Q_{ubo,2} \end{bmatrix} =$$

$$\begin{bmatrix} 150 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\text{b) } Q_{ui,1} = \begin{bmatrix} Q_{uba,1} & 0 \\ 0 & Q_{ubo,1} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 30 \end{bmatrix}$$

$$\text{c) } \bar{u}_{g,k} = 0 \quad Q_{ug,1} = 300$$

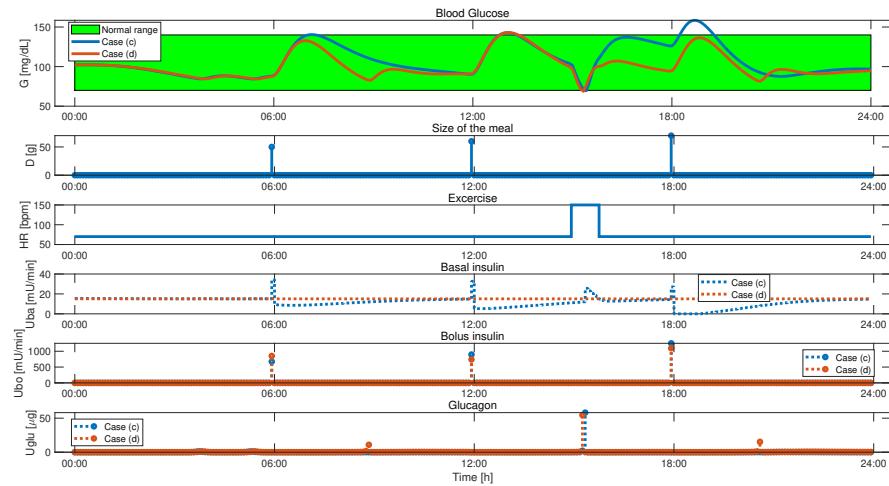
It should be noted that the objective function must include a penalty term for glucagon  $\Phi_{ug}$ . As mentioned above, in the penalty of the controlled variable  $z$ , in order to prevent the occurrence of hypoglycemia, the lower limit of the controlled variable  $z$  will be set a large penalty. If there is no penalty for glucagon, the infusion amount of glucagon will be calculated as a large value to avoid hypoglycemia. The Figure 4.7 shows the control performance of the cases (a) and (b) of NMPC of the multiple-hormone system for virtual patient 1.



**Figure 4.7:** Closed-loop test result of the case (a) and (b) of the multiple-hormone system.

The control effects of case (a) and (b) for the multiple-hormone system are very different from those for the single-hormone system. For the single-hormone system, there is no penalty for bolus insulin, which leads to multiple hypoglycemia. For the multiple-hormone system, since glucagon will be infused to prevent hypoglycemia, cases (a) and (b) can still achieve good control results despite the lack of penalty for bolus insulin. But the disadvantage is that glucagon needs to be infused frequently, which is not desirable. Regarding the form of penalty for basal insulin, compared with case (a) using  $\frac{1}{2} \|\Delta u_{ba,k}\|^2$ , case (b) using  $\frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|^2$  has a better control effect.

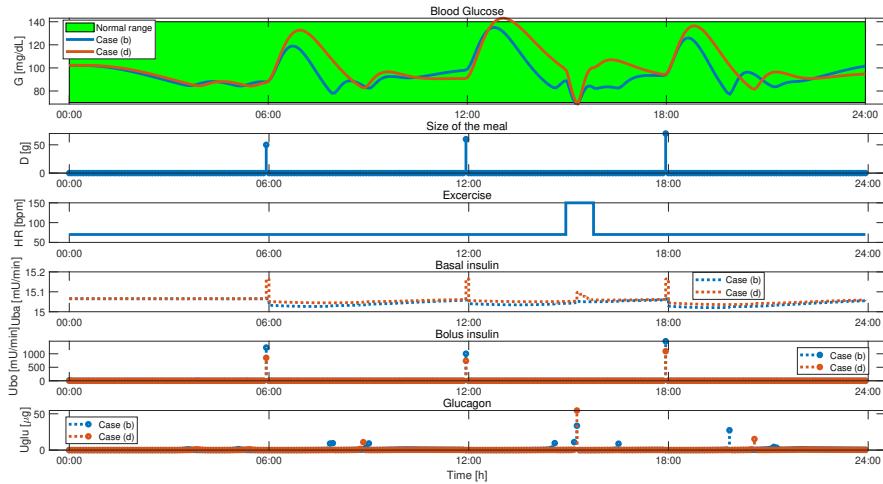
In addition, the closed-loop test results of case (c) and case (d) of the multiple-hormone system are also compared



**Figure 4.8:** Closed-loop test result of the case (c) and (d) of the multiple-hormone system.

For the multiple-hormone system, both cases (c) and (d) control blood glucose within the normal range, and glucagon is only infused when the heart rate increases due to exercise. Similar to the results for the single hormone system, compared with case (c) using  $\frac{1}{2} \|\Delta u_{ba,k}\|^2$ , case (d) using  $\frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|^2$  has a better control effect. In case (c), in response to the increase in heart rate caused by exercise at 15:00, excessive glucagon is injected and caused hyperglycemia, while case (d) kept the blood glucose perfectly within the normal range within 24 hours.

In order to explore the effect of penalty on bolus insulin in a multiple-hormonal system, the closed-loop test results of case (b) and case (d) for the multiple-hormone system are also compared



**Figure 4.9:** Closed-loop test result of the case (b) and (d) for the multiple-hormone system.

It can be seen from the closed-loop test results that case (b) controls blood glucose in a normal range better than case (d), but the larger dose of bolus insulin leads to frequent infusion of glucagon, which is undesirable.

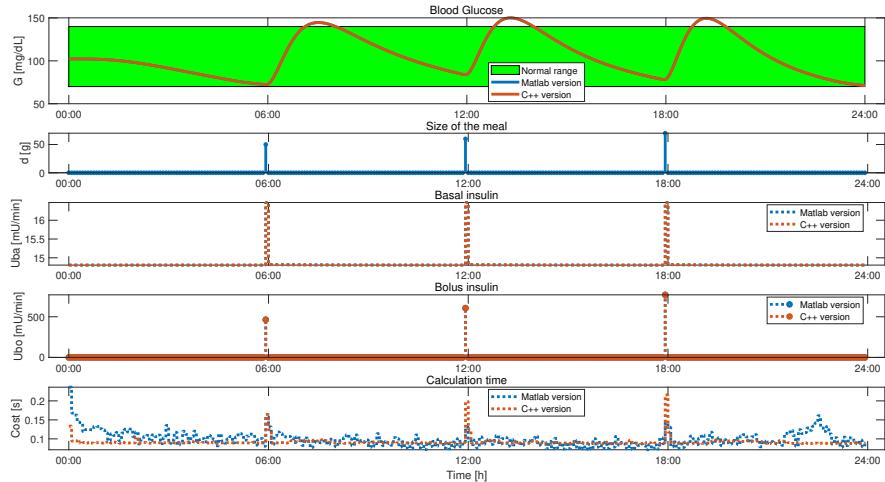
Therefore, for the multiple-hormone system, a better penalty term for the manipulated variable  $u$  is

$$\begin{aligned} \Phi_u &= \Phi_{ui} + \Phi_{ug} \\ &= \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_{2,Q_{uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}} + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}} \end{aligned} \quad (4.44)$$

Where  $\bar{u}_{ba,k} = u_{ba,ss}$ ,  $Q_{uba,2} = 150$ ,  $\bar{u}_{bo,k} = 0$ ,  $Q_{ubo,1} = 30$ ,  $\bar{u}_{g,k} = 0$ ,  $Q_{ug,1} = 300$ .

### 4.5.2 Closed-loop simulation test of virtual patient

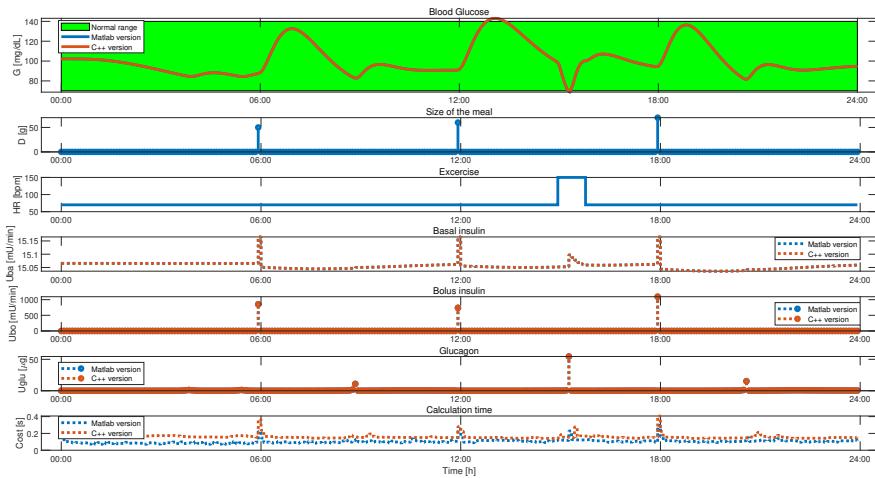
In this thesis, the Matlab version and C++ version of the same structure of the NMPC algorithm are implemented. If the NMPC algorithm is implemented correctly, the closed-loop simulation test results of the two versions of the NMPC algorithm should be the same, and only the calculation cost (calculation time) is different. As the process noise and sensor noise are implemented in different ways in Matlab and C++, in order to verify that the two versions of the NMPC algorithm are implemented correctly, the ODE version of the Hovorka model that does not contain the CGM model is used for simulation (Without process noise and Sensor noise). The figure 4.10 is a comparison of the results of the closed-loop simulation test of the Matlab version and the C++ version for the single-hormone system.



**Figure 4.10:** Comparison of the results of the closed-loop simulation test of the Matlab version and the C++ version for the single-hormone system.

It can be seen from the closed-loop test results that the control effect of the NMPC algorithm of the Matlab version and the C++ version is exactly the same, and both stabilize the blood glucose within the normal range very well. From the results of the calculation time, both the Matlab version and the C++ version of NMPC can control the calculation cost below 0.3s in each iterator. The C++ version has a slightly longer calculation time at the three moments of meal, and the overall calculation time is about the same as the Matlab version.

A comparison of the results of the closed-loop simulation test of the Matlab version and the C++ version for the multiple-hormone system is also shown in Figure 4.11

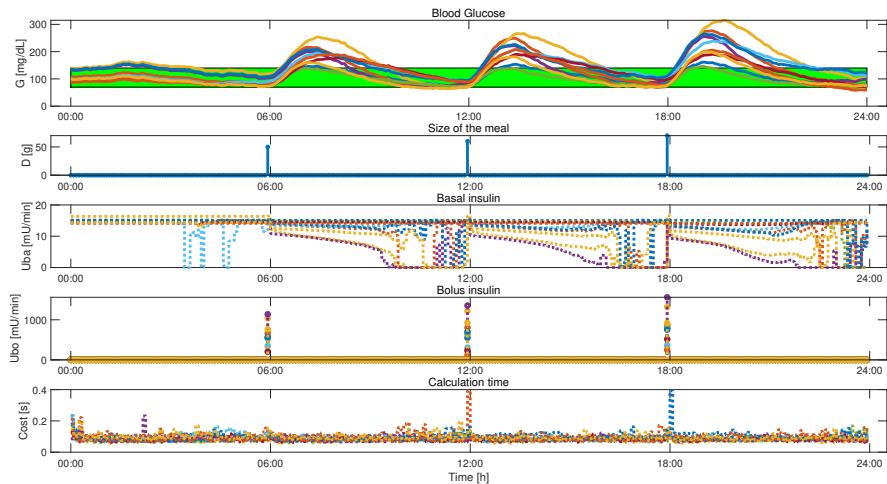


**Figure 4.11:** Comparison of the results of the closed-loop simulation test of the Matlab version and the C++ version for the multiple-hormone system.

From the comparison results, similar to the results of the single-hormone system, when dealing with the disturbance of meal intake and increased heart rate caused by exercise, the calculation time of both versions will increase, and the peak value does not exceed 0.4 seconds. During the fasting period, the calculation time of both versions remained below 0.2 seconds. In the whole control process, the calculation time of the Matlab version is less than that of the C++ version.

The reason for the similar calculation time of the two versions is also related to the working mode of CasADI. In the implementation of the Matlab version of NMPC, the NLP problem is constructed by the rules in the previous section and passed into the caller function of the solver. The automatic differentiation, calling the solver, etc. required for the solution in CasADI are implemented in C language, which is the same as in the C++ version of NMPC. It also indicates that the MPC algorithm built using the CasADI framework has less dependence on the programming language.

In order to test the effectiveness and robustness of the NMPC algorithm, the designed and implemented NMPC algorithm is used to perform a 24-hour closed-loop simulation test on 10 virtual patients generated by the Hovorka model and his parameter distribution. Below are the test results of the single-hormone system



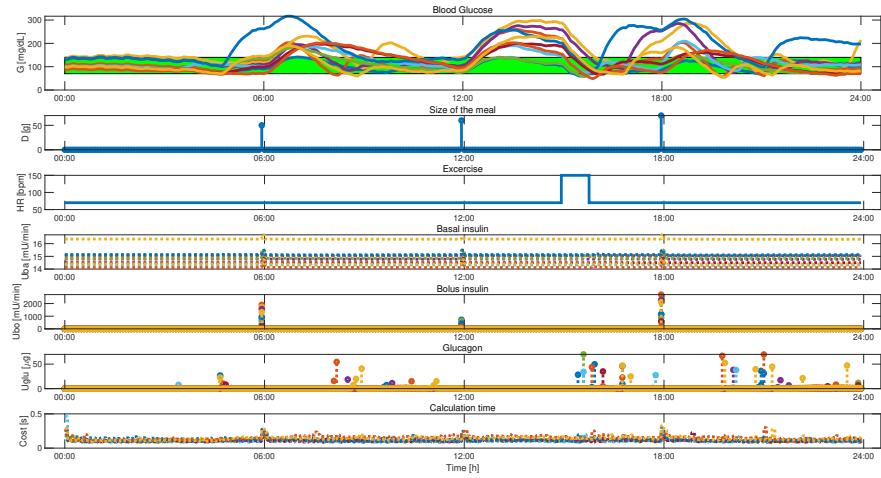
**Figure 4.12:** Closed-loop test result of NMPC algorithm of the single-hormone system on 10 virtual patients .

From the closed-loop simulation results of 10 virtual patients, despite the occasional occurrence of hyperglycemia, the NMPC algorithm of the single-hormone system successfully stabilized the blood glucose concentration within the normal range and avoided the occurrence of hypoglycemia. For the meal intake at the three time points, both basal insulin and bolus insulin are calculated as appropriate doses and infused, so that the rising blood glucose drops to the normal range as soon as possible. The dose of basal insulin remains almost unchanged. Only during a period of time after a large amount of bolus insulin is infused, in order to avoid the occurrence of hypoglycemia, the infusion rate of basal insulin is reduced to 0. The calculation of bolus insulin also meets the desired injection strategy. The appropriate dose is calculated only when meal intake, and the infusion rate of bolus insulin during the fasting period is set to 0. From the calculation cost(calculation time), except for one or two closed-loop simulation tests where the calculation time increased at the time of meal intake (but also under 0.4 seconds), the NMPC algorithm of single-hormone can keep the time cost of a single calculation below 0.2 seconds.

From the comparison of the results of the closed-loop simulation test of 10 virtual patients, the change trend of blood glucose under the control effect of the NMPC algorithm of single-hormone is similar. During the fasting period, NMPC mainly re-

lies on the smooth infusion of basal insulin to keep the blood glucose concentration within the normal range. After infusing an appropriate dose of bolus insulin at the time of meal intake, the blood glucose concentration will first rise and then drop to the normal range. However, in the control results of some virtual patients, the blood glucose concentration will rise to a concentration higher than the normal range. This situation is considered to be related to the effect of parameter identification. When the parameters are accurately identified, the control model can be very good matching the physiological model of the virtual patient(the extended Hovorka model is used in this thesis), bolus insulin will be calculated more correctly to prevent the blood glucose from rising as soon as possible and avoid the occurrence of hyperglycemia.

The test results of the NMPC algorithm of multiple-hormone system on 10 virtual patients is



**Figure 4.13:** Closed-loop test result of NMPC algorithm of the multiple-hormone system on 10 virtual patients .

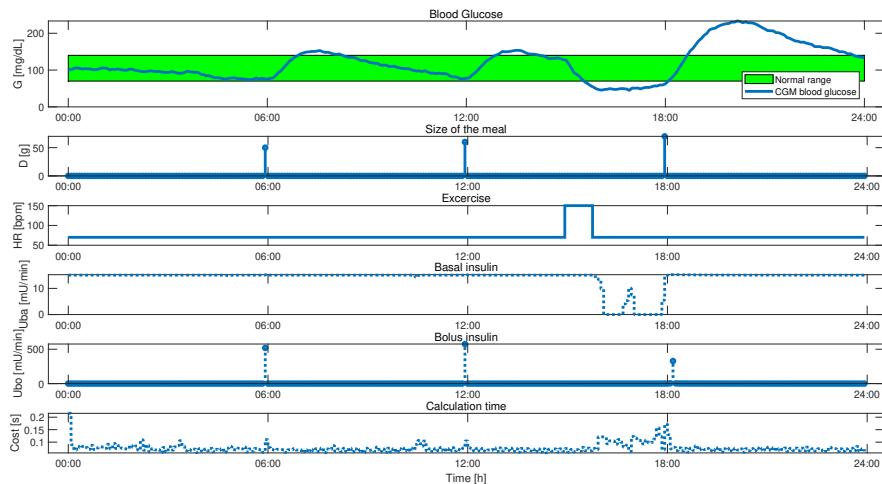
From the closed-loop simulation test results of 10 virtual patients, despite the occurrence of several hyperglycemia, the multiple-hormonal NMPC algorithm also controlled the blood glucose concentration within the normal range for most of the time and avoided the occurrence of hypoglycemia. In addition, the calculated doses of basal insulin and bolus insulin also meets the desired injection strategy, but at the 12:00 of meal intake, the calculated dose of bolus insulin is low, which is considered to be related to the infusion of insulin and glucagon in the morning. After a large amount of bolus insulin is infused at the 6:00 of meal intake, in order to avoid the occurrence of hypoglycemia, glucagon is infused to make blood glucose rise to a certain extent, so the dose of bolus insulin calculated later is lower than desired. This also reflects the cooperative relationship between glucagon and bolus insulin in the NMPC algorithm of multiple-hormone system. During the 45 minutes after 15:00, the increase in heart rate caused by exercise will make blood glucose drop. It can be seen that a large amount of glucagon is infused at this time to prevent the occurrence of hypoglycemia.

From the comparison of the closed-loop simulation test results of 10 virtual patients, similar to the single-hormone NMPC, the change trend of blood glucose under the control effect of the multiple-hormone NMPC is similar, but the control effect for individual virtual patients is not very ideal. Although hypoglycemia is avoided, hyperglycemia occurs frequently. This situation is considered not only related to the

effect of parameter identification, but also related to the control of glucagon. The higher penalty for hypoglycemia makes the injection of glucagon very sensitive. This makes the NMPC controller infuse excessive glucagon in order to prevent the occurrence of hypoglycemia, which leads to the frequent occurrence of hyperglycemia.

Compared with the closed-loop simulation test of NMPC of single-hormone, except for the increase in blood glucose caused by meal intake. NMPC of multiple-hormonal also faces increased heart rate caused by exercise, which leads to a drop in blood glucose. The NMPC of single-hormone can only infuse insulin(basal insulin and bolus insulin) but does not contain the manipulated variable glucagon which promotes the increase of blood glucose, so it cannot cope with the risk of hypoglycemia caused by unknown disturbance (increased heart rate caused by exercise).

In order to verify the effectiveness of the NMPC algorithm of the multiple-hormone system on disturbance of exercise, the closed-loop simulation test results of the NMPC algorithm of the single-hormone system under disturbance of exercise are also shown as follows.



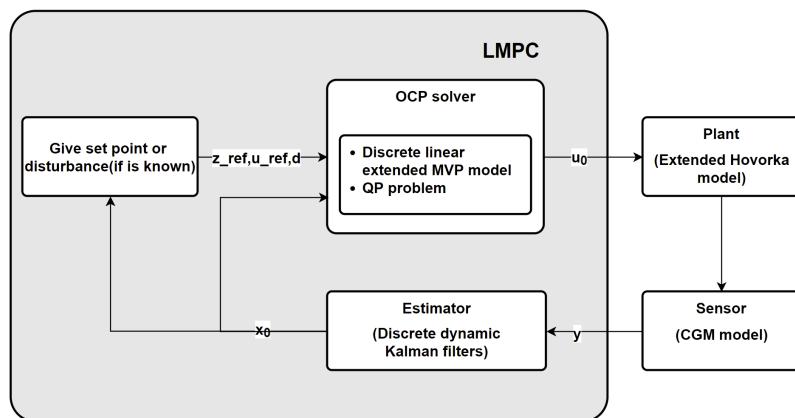
**Figure 4.14:** Closed-loop test result of NMPC algorithm of the single-hormone system under disturbance of exercise.

It can be seen from the Figure 4.14 that during the 45 minutes from 15:00, the increase in heart rate caused by exercise make blood glucose drop. Although both basal insulin and bolus insulin are calculated as 0, hypoglycemia still occurs. Therefore, compared with the NMPC of single-hormone , the NMPC multiple-hormonal can cope with the unknown disturbance of lowering blood glucose and prevent the occurrence of hypoglycemia.

# CHAPTER 5

# Linear Model Predictive Control for artificial pancreas

In this chapter, linear model predictive control(LMPC) is used in the artificial pancreas problem. Figure 5.1 shows the block-diagram of closed-loop simulation with LMPC.



**Figure 5.1:** Block-diagram of closed-loop simulation with LMPC.

In LMPC, the solver of the optimal control problem(OCP) is the main part of the execution. In each solution, the solver obtains the initial value of the control model from the estimator, at the same time, the set point of the controlled variable  $z$ , the target value of the manipulated variable  $u$  and the disturbance that can be given are also provided to the solver. The discrete linear model derived from the extended MVP model mentioned in Chapter 3 near normoglycemia is used as the control model. Therefore, discrete dynamic Kalman filters(KF) are designed to be the estimator and the OCP is discretized to a quadratic programming(QP) problem by using single shooting method. The solution of OCP is constructed and written using the framework of CasADi. More specifically, the solution of the QP optimization problem is also solved by using CasADi to call qpOASES. The specific description of CasADi is in section 4.4.

## 5.1 Mathematical model for LMPC

In Chapter 3, an identifiable, continuous-discrete and non-linear extended MVP model is proposed as the control model, and for different virtual patients, the parameters of the control model are identified to match the physiological model of the virtual patient (the extended Hovorka model is used in this thesis). However, the control model used by LMPC must be a discrete linear model. Therefore, the nonlinear MVP model identified cannot be directly used as the control model of LMPC. In this chapter, we linearize and discretize the nonlinear MVP model near normoglycemia. The control model of the LMPC algorithm based on the single-hormone model contains 7 states (6 states in the MVP model + 1 state in the CGM model), The control model of the LMPC algorithm based on the multiple-hormone model contains 9 states (6 states in the MVP model + 1 state in the CGM model + 2 states in the glucagon model). First, the steady-state value of the MVP model near normoglycemia, the state  $x_{ss}$ , the manipulated variable  $u_{ss}$ , the disturbance  $d_{ss}$ , the controlled variable  $z_{ss}$  and the measurement  $y_{ss}$ , are calculated according to the description in chapter 2.

### 5.1.1 Linearization

The ODE version of nonlinear MVP model is expressed as

$$\dot{x}(t) = f(x(t), u(t), d(t), p) \quad (5.1a)$$

$$y(t) = g(x(t)) \quad (5.1b)$$

$$z(t) = h(x(t)) \quad (5.1c)$$

Can be approximated by the system matrix ( $A$ ,  $B$ ,  $E$ ,  $C$ ,  $C_z$ ) as

$$\dot{X}(t) = AX(t) + BU(t) + ED(t) \quad (5.2a)$$

$$Y(t) = CX(t) \quad (5.2b)$$

$$Z(t) = C_z X(t) \quad (5.2c)$$

Where

$$X(t) = x(t) - x_{ss} \quad (5.3a)$$

$$U(t) = u(t) - u_{ss} \quad (5.3b)$$

$$D(t) = d(t) - d_{ss} \quad (5.3c)$$

$$Y(t) = Y(t) - y_{ss} \quad (5.3d)$$

$$Z(t) = Z(t) - z_{ss} \quad (5.3e)$$

The system matrix can be calculated as

$$A = \frac{\partial f}{\partial x}(x_{ss}, u_{ss}, d_{ss}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} (x_s, u_s, d_s) \quad (5.4a)$$

$$B = \frac{\partial f}{\partial u}(x_{ss}, u_{ss}, d_{ss}) = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_m} \end{bmatrix} (x_{ss}, u_{ss}, d_{ss}) \quad (5.4b)$$

$$E = \frac{\partial f}{\partial d}(x_{ss}, u_{ss}, d_{ss}) = \begin{bmatrix} \frac{\partial f_1}{\partial d_1} & \cdots & \frac{\partial f_1}{\partial d_m} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial d_1} & \cdots & \frac{\partial f_n}{\partial d_m} \end{bmatrix} (x_{ss}, u_{ss}, d_{ss}) \quad (5.4c)$$

$$C = \frac{\partial g}{\partial x}(x_{ss}) = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \cdots & \frac{\partial g_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial g_n}{\partial x_1} & \cdots & \frac{\partial g_n}{\partial x_n} \end{bmatrix} (x_{ss}) \quad (5.4d)$$

$$C_z = \frac{\partial h}{\partial x}(x_{ss}) = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial h_n}{\partial x_1} & \cdots & \frac{\partial h_n}{\partial x_n} \end{bmatrix} (x_{ss}) \quad (5.4e)$$

### 5.1.2 Discretization

Assuming the sampling time  $T_s$  and start time  $t_0$ , let  $\mathcal{N} = \{0, 1, \dots, N-1\}$  and  $t_k = t_0 + kT_s$  with  $k \in \mathcal{N}$ . Therefore, Manipulated variables  $u$  can be discretized by applying zero-order-hold

$$U(t) = U_k \quad t_k \leq t < t_{k+1} \quad k \in \mathcal{N} \quad (5.5a)$$

$$D(t) = D_k \quad t_k \leq t < t_{k+1} \quad k \in \mathcal{N} \quad (5.5b)$$

The state differential equations can be written as

$$\dot{X}(t) = AX(t) + BU(t) + ED(t) \quad X(t_k) = X_k \quad t_k \leq t < t_{k+1} \quad k \in \mathcal{N} \quad (5.6)$$

The solution is

$$X_{k+1} = X(t_{k+1}) = e^{AT_s}X_k + \int_0^{T_s} e^{A\tau} d\tau BU_k + \int_0^{T_s} e^{A\tau} d\tau ED_k \quad (5.7)$$

Further expressed as

$$X_{k+1} = FX_k + GU_k + G_dD_k \quad (5.8a)$$

$$Y_k = CX_k \quad (5.8b)$$

$$Z_k = C_z X_k \quad (5.8c)$$

Where the matrix exponential is used to calculate

$$\begin{bmatrix} F & G \\ 0 & I \end{bmatrix} = \exp \left( \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix} T_s \right) \quad (5.9)$$

$$\begin{bmatrix} F & G_d \\ 0 & I \end{bmatrix} = \exp \left( \begin{bmatrix} A & E \\ 0 & 0 \end{bmatrix} T_s \right) \quad (5.10)$$

## 5.2 Discrete dynamic Kalman filters

The discrete dynamic kalman filters is introduced for the linear discrete system used in LMPC. Based on the discretized linear model and given measurement, the state of the model can be predicted and estimated by the discrete dynamic kalman filters.

$$X_{k+1} = FX_k + GU_k + G_dD_k + G_w w_k \quad (5.11a)$$

$$Y_k = CX_k + v_k \quad (5.11b)$$

$$Z_k = C_z X_k \quad (5.11c)$$

Where process noise  $w_k \sim N_{iid}(0, Q)$  and measurement noise  $v_k \sim N_{iid}(0, R)$  are independently normally distributed.

### 5.2.1 Filtering

Given predicted state and its covariance  $\hat{x}_{k|k-1}$ ,  $\hat{P}_{k|k-1}$  and discrete measurement  $y_k$

$$R_{k|k-1} = C \hat{P}_{k|k-1} C' + R_k \quad (5.12a)$$

$$K_k = \hat{P}_{k|k-1} C' R_{k|k-1}^{-1} \quad (5.12b)$$

The innovation is calculated by

$$e_k = y_k - C\hat{x}_{k|k-1} \quad (5.13)$$

The estimated state and its covariance are obtained by

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k e_k \quad (5.14a)$$

$$\hat{P}_{k|k} = \hat{P}_{k|k-1} - K_k R_{k|k-1} K'_k \quad (5.14b)$$

### 5.2.2 Prediction

Given estimated state and its covariance  $\hat{x}_{k|k}$ ,  $\hat{P}_{k|k}$ , Compute  $\hat{x}_{k+1|k}$ ,  $\hat{P}_{k+1|k}$  by

$$\hat{x}_{k+1|k} = F\hat{x}_{k|k} + Gu_k + G_d d_k \quad (5.15a)$$

$$\hat{P}_{k+1|k} = F\hat{P}_{k|k}F' + G_w Q G'_w \quad (5.15b)$$

$u_k$  indicates the insulin infusion rate and the glucagon infusion rate calculated by the LMPC and  $d_k$  is the announced oral meal intake.

## 5.3 Optimal control problem formulation

LMPC obtains the input  $u$  and state  $x$  trajectory that minimizes the objective function in the prediction time domain by solving an optimal control problem(OCP). LMPC uses a linear model, and the form of the objective function is quadratic, the optimal control problem(OCP) can be discretized into a quadratic programming(QP) problem. A general OCP formulation for LMPC is

$$\min_u \quad \Phi = \int_{t_0}^{t_f} g(z(t), u(t), p) dt + \lambda(z(t_f), p) \quad (5.16a)$$

$$\text{s.t.} \quad x(t_0) = x_0 \quad (5.16b)$$

$$\dot{x}(t) = Ax(t) + Bu(t) + Ed(t) \quad (5.16c)$$

$$z(t) = C_z x(t) \quad (5.16d)$$

$$u_{\min} \leq u(t) \leq u_{\max} \quad (5.16e)$$

There are two terms: a stage cost term  $g(z(t), u(t), p)$  and a stage-to-go term  $\lambda(z(t_f), p)$  in the objective function. In addition, the objective function of LMPC can be written in a specified form represented only by the matrix.  $\dot{x}(t) = Ax(t) + Bu(t) + Ed(t)$  and  $z(t) = C_z x(t)$  represents the continuous linear model equations.  $t_0, t_f, x_0$  are known parameters.

In the artificial pancreas(AP) problem, for the single-hormone system,  $u(t)$  represents the basal insulin infusion rate and bolus insulin infusion rate at time  $t$  and  $d(t)$  represents the carbohydrates (CHO) intake rate at time  $t$ . For the multiple-hormone system,  $u(t)$  represents the basal insulin infusion rate, bolus insulin infusion rate and glucagon infusion rate at time  $t$  and  $d(t)$  represents the carbohydrates (CHO) intake rate and increased heart rate caused by exercise at time  $t$ . Equations  $\dot{x}(t) = Ax(t) + Bu(t) + Ed(t)$  and  $z(t) = C_z x(t)$  is the linear model derived from the extended MVP model. Since the proposed model is in continuous-discrete form,  $z(t) = C_z x(t)$  can be directly replaced by discrete  $z_k = C_z x_k$ .

### 5.3.1 Objective function

In the LMPC algorithm for the artificial pancreas(AP) problem, in order to achieve the control effect, the manipulated variables including basal insulin  $u_{ba}$ , basal insulin  $u_{bo}$  and glucagon  $u_g$ (Single-hormone system only infuses insulin, multiple-hormone system infuses insulin and glucagon) and the controlled variable  $z$  need to be penalized. In addition, the upper and lower limits of the controlled variables  $z$ , blood glucose, need to be constrained to keep in the normal range. Generally, in the design of optimization problems, if a large number of strict constraints need to be imposed on non-decision variables, instead of using direct inequality constraints, implementing constraints as a form of soft constraints in the objective function is a method that can improve computational efficiency and effectiveness. For example, in the NMPC

algorithm for artificial pancreas problems, the upper and lower limits of blood glucose are formed as a nonlinear function and added to the objective function as a soft constraint. However, in the LMPC algorithm, the objective function must be expressed in the form of  $\Phi = \frac{1}{2}u' Hu + g'u$ . In order to improve the calculation efficiency, the slack variables  $\eta_{min}$  and  $\eta_{max}$  of the upper and lower limits of blood glucose are introduced and punished to improve the tolerance of constraints.

For the single-hormone system, the objective function can be expressed as

$$\begin{aligned}\Phi &= \Phi_z + \Phi_u(\Phi_{ui}) + \Phi_\eta \\ &= \underbrace{\int_{t_0}^{t_f} \rho_z(z(t)) dt}_{\text{BG penalty terms}} + \underbrace{\int_{t_0}^{t_f} \rho_{ui}(u_{i,k}) dt + \sum_{k=0}^{N-1} \rho_{\Delta ui}(\Delta u_{i,k})}_{\text{Insulin penalty terms}} + \underbrace{\int_{t_0}^{t_f} \rho_\eta(\eta(t)) dt}_{\text{Slack penalty terms}}\end{aligned}\quad (5.17)$$

For the multiple-hormone system, the objective function can be expressed as

$$\begin{aligned}\Phi &= \Phi_z + \Phi_u(\Phi_{ui} + \Phi_{ug}) + \Phi_\eta \\ &= \underbrace{\int_{t_0}^{t_f} \rho_z(z(t)) dt}_{\text{BG penalty terms}} + \underbrace{\int_{t_0}^{t_f} \rho_{ui}(u_{i,k}) dt + \sum_{k=0}^{N-1} \rho_{\Delta ui}(\Delta u_{i,k})}_{\text{Insulin penalty terms}} + \underbrace{\int_{t_0}^{t_f} \rho_\eta(\eta(t)) dt}_{\text{Slack penalty terms}} \\ &\quad + \underbrace{\int_{t_0}^{t_f} \rho_{ug}(u_{g,k}) dt + \sum_{k=0}^{N-1} \rho_{\Delta ug}(\Delta u_{g,k})}_{\text{Glucagon penalty terms}}\end{aligned}\quad (5.18)$$

The penalty term of the controlled variable  $z$  includes the penalty term of deviation from the target value  $\bar{z}$

$$\rho_z(z(t)) = \frac{\alpha_{\bar{z}}}{2} \|z(t) - \bar{z}\|_2^2 \quad (5.19)$$

Where the blood glucose set point is  $\bar{z} = 108$  mg/dL and  $\alpha_{\bar{z}}$  are the weight values of the penalty items. Through tuning, the weight value is set to  $\alpha_{\bar{z}} = 0.01$ . The penalty for the rate of change in the insulin penalty and glucagon penalty is discrete. The discrete method will be described in section 5.3.2. Therefore, only the penalty for the controlled variable  $z$  is introduced in this section. The penalty terms of slack variables  $\eta$  and manipulated variables  $u$  will be introduced after being discretized in section 5.3.2.

### 5.3.2 Discretization of optimal control problem

For the LMPC of problem of artificial pancreas, direct single shooting is used to discretize the continuous OCP problem. We divide the control horizon,  $[t_0; t_f]$ , into  $N$  intervals each of length  $T_s$ . Let  $\mathcal{N} = \{0, 1, \dots, N-1\}$  and  $t_k = t_0 + kT_s$  with  $k \in \mathcal{N}$ . Manipulated variables  $u$  are discretized by applying zero-order-hold parameterization.

$$u(t) = u_k \quad t_k \leq t < t_{k+1} \quad k \in \mathcal{N} \quad (5.20)$$

The single shooting method regards the state  $x_k$  at each time point as a dependent variable. Given the initial state, the state at each time point is obtained by numerical integration of the initial state  $x_0$  and the manipulated variable  $u_k$ . For the single shooting method, the decision variable only contains the discretized manipulated variables in the control horizon.

$$W = [u_0, \dots, u_{N-1}] \quad (5.21a)$$

$$x_0 = \bar{x}_0 \quad (5.21b)$$

$$x_{k+1} = Fx_k + Gu_k + G_d d_k \quad k \in \mathcal{N} \quad (5.21c)$$

The OCP problem in discrete form can be expressed as

$$\min_{\{u_k\}_{k=0}^{N-1}} \Phi = \sum_{k=0}^{N-1} G_k(z_k, u_k, d_k, p) + \lambda(z_N, p) \quad (5.22a)$$

$$\text{s.t.} \quad x_0 = \bar{x}_0 \quad (5.22b)$$

$$x_{k+1} = Fx_k + Gu_k + G_d d_k \quad k \in \mathcal{N} \quad (5.22c)$$

$$z_k = C_z x_k \quad k \in \mathcal{N} \quad (5.22d)$$

$$u_{\min} \leq u_k \leq u_{\max} \quad k \in \mathcal{N} \quad (5.22e)$$

The discrete stage cost term is

$$G_k(z_k, u_k, d_k, p) = \left\{ \int_{t_k}^{t_{k+1}} g(z(t), u_k) dt : z_k = C_z x_k, \right. \\ \left. x_{k+1} = Fx_k + Gu_k + G_d d_k, u(t_k) = u_k \right\} \quad (5.23)$$

The discrete OCP problem can be expressed in quadratic programming(QP) problem form and solved using QP optimization methods such as the active-set algorithm and interior-point algorithm.

The general formulation of QP problem is

$$\min_w \frac{1}{2} u' H u + g' u \quad (5.24a)$$

$$\text{s.t.} \quad w_l \leq w \leq w_u \quad (5.24b)$$

$$a_l \leq Aw \leq a_u \quad (5.24c)$$

For the problem of artificial pancreas, the formulation of QP problem for the single-hormone can be further expressed as

$$\Phi = \Phi_z + \Phi_u(\Phi_{ui}) + \Phi_\eta$$

$$= \underbrace{\sum_{k=0}^{N-1} \frac{1}{2} \|z_k - \bar{z}\|_{Q_z}^2}_{\text{BG penalty terms}} + \underbrace{\sum_{k=0}^{N-1} \rho_{ui}(u_{i,k})}_{\text{Insulin penalty terms}} + \underbrace{\sum_{k=0}^{N-1} \rho_{\Delta ui}(\Delta u_{i,k})}_{\text{Slack penalty terms}} + \sum_{k=0}^{N-1} \rho_\eta(\eta_k) \quad (5.25a)$$

$$\text{s.t. } x_0 = \bar{x}_0 \quad (5.25b)$$

$$x_{k+1} = Fx_k + Gu_k + G_dd_k \quad k \in \mathcal{N} \quad (5.25c)$$

$$z_k = C_z x_k \quad k \in \mathcal{N} \quad (5.25d)$$

$$u_{\min} \leq u_k \leq u_{\max} \quad k \in \mathcal{N} \quad (5.25e)$$

$$z_k - \eta_{\max,k} \leq z_{\max} \quad k \in \mathcal{N} \quad (5.25f)$$

$$z_k + \eta_{\min,k} \geq z_{\min} \quad k \in \mathcal{N} \quad (5.25g)$$

$$\eta_{\max,k} \geq 0 \quad k \in \mathcal{N} \quad (5.25h)$$

$$\eta_{\min,k} \geq 0 \quad k \in \mathcal{N} \quad (5.25i)$$

Where

$$u_k = u_{i,k} \quad \eta_k = \begin{bmatrix} \eta_{\min,k} \\ \eta_{\max,k} \end{bmatrix}$$

The formulation of QP problem for the multiple-hormone can be further expressed as

$$\Phi = \Phi_z + \Phi_u(\Phi_{ui} + \Phi_{ug}) + \Phi_\eta$$

$$= \underbrace{\sum_{k=0}^{N-1} \frac{1}{2} \|z_k - \bar{z}\|_{Q_z}^2}_{\text{BG penalty terms}} + \underbrace{\sum_{k=0}^{N-1} \rho_{ui}(u_{i,k})}_{\text{Insulin penalty terms}} + \underbrace{\sum_{k=0}^{N-1} \rho_{\Delta ui}(\Delta u_{i,k})}_{\text{Slack penalty terms}} + \sum_{k=0}^{N-1} \rho_\eta(\eta_k) + \underbrace{\sum_{k=0}^{N-1} \rho_{ug}(u_{g,k})}_{\text{Glucagon penalty terms}} + \sum_{k=0}^{N-1} \rho_{\Delta ug}(\Delta u_{g,k}) \quad (5.26a)$$

$$\text{s.t. } x_0 = \bar{x}_0 \quad (5.26b)$$

$$x_{k+1} = Fx_k + Gu_k + G_dd_k \quad k \in \mathcal{N} \quad (5.26c)$$

$$z_k = C_z x_k \quad k \in \mathcal{N} \quad (5.26d)$$

$$u_{\min} \leq u_k \leq u_{\max} \quad k \in \mathcal{N} \quad (5.26e)$$

$$z_k - \eta_{\max,k} \leq z_{\max} \quad k \in \mathcal{N} \quad (5.26f)$$

$$z_k + \eta_{\min,k} \geq z_{\min} \quad k \in \mathcal{N} \quad (5.26g)$$

$$\eta_{\max,k} \geq 0 \quad k \in \mathcal{N} \quad (5.26h)$$

$$\eta_{\min,k} \geq 0 \quad k \in \mathcal{N} \quad (5.26i)$$

Where

$$u_k = \begin{bmatrix} u_{i,k} \\ u_{g,k} \end{bmatrix} \quad \eta_k = \begin{bmatrix} \eta_{min,k} \\ \eta_{max,k} \end{bmatrix}$$

The penalty term of the slack variable  $\eta$  includes the  $\ell_2$  norm penalty form of upper and lower limits

$$\rho_\eta(\eta_k) = \frac{1}{2} \|\eta_k\|_{Q_\eta}^2 \quad (5.27a)$$

$$= \frac{1}{2} \|\eta_{min,k}\|_{Q_{\eta,min}}^2 + \frac{1}{2} \|\eta_{max,k}\|_{Q_{\eta,max}}^2 \quad (5.27b)$$

Where

$$\eta_k = \begin{bmatrix} \eta_{min,k} \\ \eta_{max,k} \end{bmatrix} \quad Q_\eta = \begin{bmatrix} Q_{\eta,min} & 0 \\ 0 & Q_{\eta,max} \end{bmatrix}$$

Through tuning, for the single-hormone system, the weight value is set to  $Q_{\eta,min} = 10$  and  $Q_{\eta,max} = 5$ . For the multiple-hormone system, the weight value is set to  $Q_{\eta,min} = 2$  and  $Q_{\eta,max} = 10$ .

The penalty term of the manipulated variable  $u_i$ (including  $u_{ba}$  and  $u_{bo}$ ) and  $u_g$  and their change rate  $\Delta u_i$ (including  $\Delta u_{ba}$  and  $\Delta u_{bo}$ ) and  $\Delta u_g$  include the  $\ell_1$  and  $\ell_2$  norm penalty form, which can be generally expressed as

$$\rho_{ui}(u_{i,k}) = \frac{1}{2} \|u_{i,k} - \bar{u}_{i,k}\|_{2,Q_{ui,2}}^2 + \|u_{i,k} - \bar{u}_{i,k}\|_{1,Q_{ui,1}} \quad (5.28a)$$

$$\rho_{\Delta ui}(\Delta u_{i,k}) = \frac{1}{2} \|\Delta u_{i,k}\|_{2,Q_{\Delta ui,2}}^2 + \|\Delta u_{i,k}\|_{1,Q_{\Delta ui,1}} \quad (5.28b)$$

$$\rho_{ug}(u_{g,k}) = \frac{1}{2} \|u_{g,k} - \bar{u}_{g,k}\|_{2,Q_{ug,2}}^2 + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}} \quad (5.28c)$$

$$\rho_{\Delta ug}(\Delta u_{g,k}) = \frac{1}{2} \|\Delta u_{g,k}\|_{2,Q_{\Delta ug,2}}^2 + \|\Delta u_{g,k}\|_{1,Q_{\Delta ug,1}} \quad (5.28d)$$

Where

$$u_{i,k} = \begin{bmatrix} u_{ba,k} \\ u_{bo,k} \end{bmatrix} \quad \bar{u}_{i,k} = \begin{bmatrix} \bar{u}_{ba,k} \\ \bar{u}_{bo,k} \end{bmatrix} \quad \Delta u_{i,k} = \begin{bmatrix} \Delta u_{ba,k} \\ \Delta u_{bo,k} \end{bmatrix}$$

Where  $\bar{u}_{ba,k}$  and  $\bar{u}_{bo,k}$  is the target input of basal insulin and bolus insulin. The  $\bar{u}_{g,k}$  is the target input of glucagon. Since the target value of basal insulin  $\bar{u}_{ba,k}$  is the steady-state value, and the control model of LMPC is linearized around the steady-state,  $\bar{u}_{ba,k} = 0$ . In addition,  $\bar{u}_{bo,k}$  and  $\bar{u}_{g,k}$  are both set to 0.

The form of the penalty term for the manipulated variable  $u$  of the single-hormone system and the multiple-hormone system is discussed in Chapter 4, and closed-loop simulation tests are performed on the NMPC algorithm of different combinations of objective functions.

For the single-hormone system, the following two cases of penalty term for manipulated variable  $u$  can achieve the control effect

(a)

$$\begin{aligned}\Phi_u &= \Phi_{ui} \\ &= \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_{2,Q_{uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}}\end{aligned}\quad (5.29)$$

(b)

$$\begin{aligned}\Phi_u &= \Phi_{ui} \\ &= \frac{1}{2} \|\Delta u_{ba,k}\|_{2,Q_{\Delta uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}}\end{aligned}\quad (5.30)$$

For the multiple-hormone system, the following two cases of penalty term for manipulated variable  $u$  can achieve the control effect

(a)

$$\begin{aligned}\Phi_u &= \Phi_{ui} + \Phi_{ug} \\ &= \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_{2,Q_{uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}} + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}}\end{aligned}\quad (5.31)$$

(b)

$$\begin{aligned}\Phi_u &= \Phi_{ui} + \Phi_{ug} \\ &= \frac{1}{2} \|\Delta u_{ba,k}\|_{2,Q_{\Delta uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}} + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}}\end{aligned}\quad (5.32)$$

Therefore, in this chapter, the form of the penalty term for the manipulated variable  $u$  is not discussed in detail.

For the single hormone system, the applied penalty term for manipulated variable  $u$  in LMPC is

$$\begin{aligned}\Phi_u &= \Phi_{ui} \\ &= \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_{2,Q_{uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}}\end{aligned}\quad (5.33)$$

Where,  $\bar{u}_{ba,k} = 0$  and  $\bar{u}_{bo,k} = 0$ . Through tuning, the weight value is set to  $Q_{uba,2} = 100$  and  $Q_{ubo,1} = 100$ .

For the multiple-hormone system, the applied penalty term for the manipulated variable  $u$  in LMPC is

$$\begin{aligned}\Phi_u &= \Phi_{ui} + \Phi_{ug} \\ &= \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_{2,Q_{uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}} + \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{ug,1}}\end{aligned}\quad (5.34)$$

Where,  $\bar{u}_{ba,k} = 0$ ,  $\bar{u}_{bo,k} = 0$  and  $\bar{u}_{g,k} = 0$ . Through tuning, the weight value is set to  $Q_{uba,2} = 100$ ,  $Q_{ubo,1} = 100$  and  $Q_{ug,1} = 12000$ .

## 5.4 Matrix for constructing the QP problem

After discretizing the optimal control problem into a QP problem, the matrix for constructing the QP problem needs to be formed and passed into the QP solver for calculation. In the matrix for constructing the QP problem in the LMPC of the single-hormone system and the multiple-hormone system, the structure of the matrix for the penalty terms of the controlled variable  $\Phi_z$  and the slack variable  $\Phi_\eta$  is similar. It is only the size of the matrix is different because of the increase in decision variables (glucagon in the multi-hormonal system), so  $\Phi_z$  and  $\Phi_\eta$  will be described first. In addition, the matrix of the penalty terms of the manipulated variable  $\Phi_{ui}$  and  $\Phi_{ug}$  of the two systems will be introduced separately. Since the slack variable of the controlled variable  $z$  is introduced, the decision variable can be expressed as

$$W = [u_0, \dots, u_{N-1}, \eta_{min,0}, \dots, \eta_{min,N-1}, \eta_{max,0}, \dots, \eta_{max,N-1}] \quad (5.35)$$

Given the discrete system matrix and initial state values, the controlled variable can be expressed as

$$z_k = C_z F^k x_0 + \sum_{j=0}^{k-1} C_z F^{k-1-j} G u_j + C_z F^k G_d d_0 \quad (5.35a)$$

$$= C_z F^k x_0 + \sum_{j=0}^{k-1} H_{k-j} u_j + C_z F^k G_d d_0 \quad (5.35b)$$

Where  $H$  is the Impulse Response Coefficients of the input  $u$  and can be expressed as

$$H_i = 0 \quad i = 0 \quad (5.36a)$$

$$H_i = C_z F^{i-1} G \quad i \geq 1 \quad (5.36b)$$

So the controlled variable  $z$  of the control horizon can be expressed as

$$\begin{aligned} \underbrace{\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_N \end{bmatrix}}_Z &= \underbrace{\begin{bmatrix} CF \\ CF^2 \\ CF^3 \\ \vdots \\ CF^N \end{bmatrix}}_K x_0 + \underbrace{\begin{bmatrix} CFG_d \\ CF^2G_d \\ CF^3G_d \\ \vdots \\ CF^NG_d \end{bmatrix}}_{K_d} d_0 \\ &+ \underbrace{\begin{bmatrix} H_1 & 0 & 0 & \dots & 0 & \dots & 0 \\ H_2 & H_1 & 0 & \dots & 0 & \dots & 0 \\ H_3 & H_2 & H_1 & \dots & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots & & 0 \\ H_N & H_{N-1} & H_{N-2} & \dots & H_1 & \dots & 0 \end{bmatrix}}_{\Gamma_z} \underbrace{\begin{bmatrix} u_0 \\ \vdots \\ u_{N-1} \\ \eta_{min,1} \\ \vdots \\ \eta_{min,N} \\ \eta_{max,1} \\ \vdots \\ \eta_{max,N} \end{bmatrix}}_U \quad (5.37) \end{aligned}$$

For both single-hormone system and multiple-hormone system, the penalty term of controlled variable  $\Phi_z$  can be simply expressed as

$$\Phi_z = \frac{1}{2} \sum_{k=1}^N \|z_k - \bar{z}_k\|_{Q_z}^2 = \frac{1}{2} \|\Gamma_z U - b\|_{Q_z}^2 \quad (5.38)$$

Where

$$b = R_z - Kx_0 - K_d d_0 \quad (5.39)$$

Where  $R_z$  is a vector representing the target value of blood glucose  $\bar{z}_k$ ,  $x_0$  is the initial value of the control model, which is obtained by the discrete dynamic Kalman filter, and  $d_0$  is the meal intake to be told. For the form  $\Phi_z = U' H_z U + g_z' U + \beta_z$ , we have

$$H_z = \Gamma_z' Q_z \Gamma_z \quad (5.40a)$$

$$\begin{aligned} g_z &= -\Gamma_z' Q_z b = -\Gamma_z' Q_z (R_z - Kx_0 - \Gamma_z^d D) \\ &= M_R R + M_{x_0} x_0 + M_d D \end{aligned} \quad (5.40b)$$

$$\beta_z = \frac{1}{2} b' Q_z b \quad (5.40c)$$

Where

$$M_R = -\Gamma_z' Q_z \quad M_{x_0} = \Gamma_z' Q_z \Phi \quad M_d = \Gamma_z' Q_z \Gamma_z^d$$

The penalty term of slack variable  $\Phi_\eta$  can be simply expressed as

$$\Phi_\eta = \frac{1}{2} \sum_{k=1}^N \|\eta_{min,k}\|_{Q_{\eta,min}}^2 + \frac{1}{2} \sum_{k=1}^N \|\eta_{max,k}\|_{Q_{\eta,max}}^2 \quad (5.41a)$$

$$= \frac{1}{2} \|\Gamma_{\eta,min} U\|_{Q_{\eta,min}}^2 + \frac{1}{2} \|\Gamma_{\eta,max} U\|_{Q_{\eta,max}}^2 \quad (5.41b)$$

For the form  $\Phi_\eta = U' H_\eta U + g'_\eta U + \beta_\eta$ , we have

$$H_\eta = \Gamma'_{\eta,min} Q_{\eta,min} \Gamma_{\eta,min} + \Gamma'_{\eta,max} Q_{\eta,max} \Gamma_{\eta,max} \quad (5.42a)$$

$$g_\eta = 0 \quad (5.42b)$$

$$\beta_\eta = 0 \quad (5.42c)$$

The inequality constraint of slack variables  $\eta_{min}$  and  $\eta_{max}$  can be expressed as

$$Z_{min} - Kx_0 - K_d D \leq \Gamma U + \eta_{min} \leq +\infty \quad (5.43a)$$

$$-\infty \leq \Gamma U - \eta_{max} \leq Z_{max} - Kx_0 - K_d D \quad (5.43b)$$

For the single-hormone system, although the penalty terms  $\Phi_{ui} = \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_2^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_1$  for insulin in the manipulated variables  $u$  has been determined and applied, the construction matrix of the  $\Phi_{ui} = \frac{1}{2} \|\Delta u_{ba,k}\|_2^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_1$  form of penalty for insulin is also introduced and implemented.

The penalty of manipulated variables related to insulin  $\Phi_{ui}$  can be expressed in the form of a matrix as

$$\Phi_{ui} = \frac{1}{2} \|u_{ba,k} - \bar{u}_{ba,k}\|_{2,Q_{uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}} \quad (5.44a)$$

$$= \frac{1}{2} \|\Gamma_{ba} U - R_{ba}\|_{2,Q_{uba,2}}^2 + \Gamma_{bo} Q_{ubo,1} U - R_{bo} Q_{ubo,1} \quad (5.44b)$$

Where  $R_{ba}$  and  $R_{bo}$  are vectors representing the target value of basal insulin  $\bar{u}_{ba,k}$  and the target value of bolus insulin  $\bar{u}_{bo,k}$ , respectively. So we can have  $R_{ba} = \mathbf{0}$  and  $R_{bo} = \mathbf{0}$ .  $\Gamma_{ba}$  and  $\Gamma_{bo}$  can be expressed as

$$\Gamma_{ba} = \begin{bmatrix} \Gamma_{ba,1} & & & \\ & \ddots & & \\ & & \Gamma_{ba,N} & \\ & & & 0 \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix} \quad \Gamma_{ba,N} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad (5.45a)$$

$$\Gamma_{bo} = \begin{bmatrix} \Gamma_{bo,1} & & & \\ & \ddots & & \\ & & \Gamma_{bo,N} & \\ & & & 0 \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix} \quad \Gamma_{bo,N} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \quad (5.45b)$$

For the form  $\Phi_{ui} = U' H_{ui} U + g'_{ui} U + \beta_{ui}$ , we have

$$H_{ui} = \Gamma'_{ba} Q_{uba,2} \Gamma_{ba} \quad (5.46a)$$

$$g_{ui} = \Gamma_{bo} Q_{ubo,1} \quad (5.46b)$$

$$\beta_{ui} = 0 \quad (5.46c)$$

Another penalty of manipulated variables related to insulin  $\Phi_{ui}$  can be expressed in the form of a matrix as

$$\Phi_{ui} = \frac{1}{2} \|\Delta u_{ba,k}\|_{2,Q_{uba,2}}^2 + \|u_{bo,k} - \bar{u}_{bo,k}\|_{1,Q_{ubo,1}} \quad (5.47a)$$

$$= \Phi_{\Delta uba} + \Gamma_{bo} Q_{ubo,1} U \quad (5.47b)$$

About the penalty term  $\Phi_{\Delta uba}$ , for the form  $\Phi_{\Delta uba} = U' H_{\Delta uba} U + g'_{\Delta uba} U + \beta_{\Delta uba}$ , we have (For viewing convenience,  $Q_{\Delta uba,2}$  is represented by  $Q_\Delta$ )

$$\Phi_{\Delta uba} = \frac{1}{2} \underbrace{\begin{bmatrix} u_0 \\ \vdots \\ u_{N-1} \\ \eta_{min,1} \\ \vdots \\ \eta_{min,N} \\ \eta_{max,1} \\ \vdots \\ \eta_{max,N} \end{bmatrix}' \begin{bmatrix} 2Q_\Delta & -Q_\Delta & & & & 0 & \dots & 0 \\ -Q_\Delta & 2Q_\Delta & -Q_\Delta & & & \vdots & & \vdots \\ & -Q_\Delta & 2Q_\Delta & -Q_\Delta & & \vdots & & \vdots \\ & & \ddots & \ddots & -Q_\Delta & \vdots & & \vdots \\ & & & -Q_\Delta & Q_\Delta & 0 & \dots & 0 \\ 0 & \dots & & \dots & 0 & 0 & \dots & 0 \\ \vdots & \dots & & \dots & 0 & 0 & \dots & 0 \\ 0 & \dots & & \dots & 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} u_0 \\ \vdots \\ u_{N-1} \\ \eta_{min,1} \\ \vdots \\ \eta_{min,N} \\ \eta_{max,1} \\ \vdots \\ \eta_{max,N} \end{bmatrix}}_{H_{\Delta uba}} + \underbrace{\left( - \begin{bmatrix} Q_\Delta \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}' u_{-1} \right)' \begin{bmatrix} u_0 \\ \vdots \\ u_{N-1} \\ \eta_{min,1} \\ \vdots \\ \eta_{min,N} \\ \eta_{max,1} \\ \vdots \\ \eta_{max,N} \end{bmatrix}}_{g_{u-1}} + \frac{1}{2} u_{-1} Q_u u_{-1} \quad (5.48)$$

Therefore, the objective function of the QP problem of the single-hormone system constructed by the matrix can be expressed as

$$\begin{aligned}\Phi &= \Phi_z + \Phi_u(\Phi_{ui}) + \Phi_\eta \\ &= U' (H_z + H_{ui} + H_\eta) U + (g'_z + g'_{ui} + g'_\eta) U + \beta_z + \beta_{ui} + \beta_\eta\end{aligned}\quad (5.49)$$

For the multiple-hormone system, the penalty of manipulated variables related to insulin  $\Phi_{ui}$  is the same as the single-hormone system, and the penalty of manipulated variables related to glucagon  $\Phi_g$  can be expressed in the form of a matrix as

$$\Phi_g = \|u_{g,k} - \bar{u}_{g,k}\|_{1,Q_{g,1}} \quad (5.50a)$$

$$= \Gamma_g Q_{ug,1} U - R_g Q_{ug,1} \quad (5.50b)$$

Where  $R_g$  is a vector representing the target value of glucagon  $\bar{u}_{g,k}$ , so we can have  $R_g = \mathbf{0}$ .  $\Gamma_g$  can be expressed as  $\Gamma_g$  can be expressed as

$$\Gamma_g = \begin{bmatrix} \Gamma_{g,1} & & & \\ & \ddots & & \\ & & \Gamma_{g,N} & \\ & & & 0 \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix} \quad \Gamma_{g,N} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.51)$$

For the form  $\Phi_g = U' H_g U + g'_g U + \beta_g$ , we have

$$H_g = 0 \quad (5.52a)$$

$$g_g = \Gamma_g Q_{g,1} \quad (5.52b)$$

$$\beta_g = 0 \quad (5.52c)$$

Therefore, the objective function of the QP problem of the multiple-hormone system constructed by the matrix can be expressed as

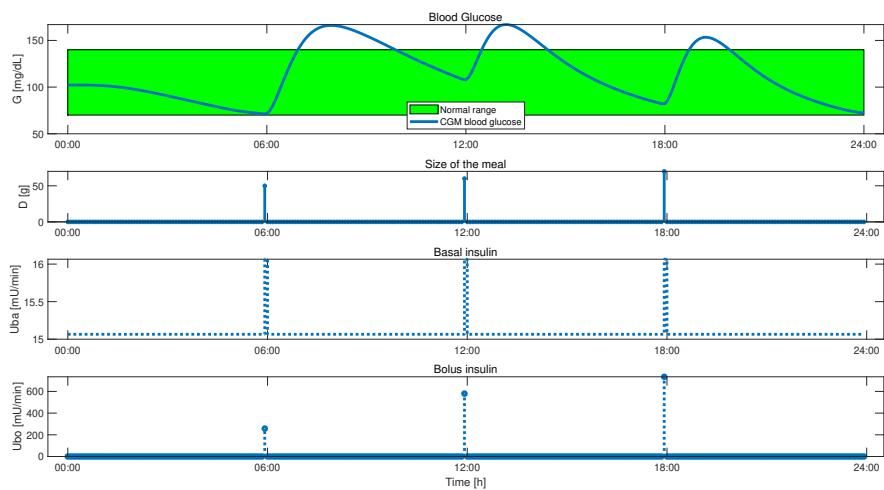
$$\Phi = \Phi_z + \Phi_u(\Phi_{ui} + \Phi_{ug}) + \Phi_\eta \quad (5.53a)$$

$$= U' (H_z + H_{ui} + H_g + H_\eta) U + (g'_z + g'_{ui} + g'_g + g'_\eta) U + \beta_z + \beta_{ui} + \beta_g + \beta_\eta \quad (5.53b)$$

## 5.5 Closed-loop simulation test

In the previous sections, we introduced and implemented the LMPC algorithm for artificial pancreas(AP) problems. The closed-loop simulation test conditions of the LMPC algorithm are exactly the same as those of the NMPC described in section 4.5. Virtual patient 1 will be used as an example to show the control effect of the LMPC algorithm of the single-hormone system and the multiple-hormone system. In addition, the LMPC algorithm will be applied to 10 virtual patients generated by the Hovorka Model and its parameter distribution to test the effectiveness and robustness of the algorithm.

The ODE version of the simulation model is used in the closed-loop simulation test of virtual patient 1. The Figure 5.2 shows the control performance of the LMPC of the signle-hormone system for virtual patient 1.



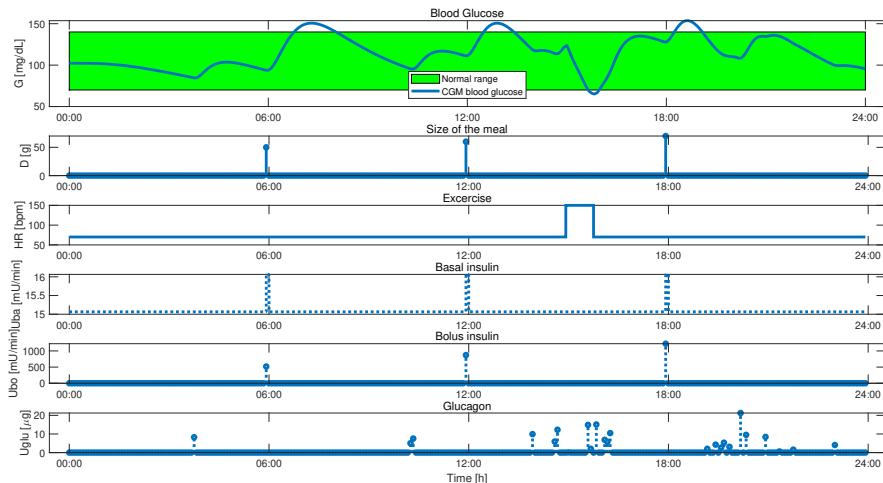
**Figure 5.2:** The control performance of the LMPC of the single-hormone system for virtual patient 1.

From the closed-loop simulation test results of virtual patient 1, during the fasting period, the LMPC algorithm of the single-hormone system stabilizes the blood glucose in the normal range through the infusion of basal insulin, and for the three-time meal intake, the LMPC also infuses the appropriate dose of bolus insulin to prevent blood glucose from rising, and there is no hypoglycemia after injection. At the same time, both basal insulin and bolus insulin meet the ideal infusion strategy.

However, at the 6:00 of the first meal intake, the calculated dose of bolus insulin is lower than the desired value, resulting in hyperglycemia. The situation may occur

because the LMPC algorithm is not sensitive to blood glucose exceeding the upper limit, which may be related to the weight value in the objective function, but considering that the weight value of the objective function has been adjusted to achieve the best control effect, the poor performance of parameter identification should be the root cause of this situation. Since parameter identification uses a nonlinear extended MVP model to match the physiological model of the virtual patient, unlike the NMPC algorithm, the LMPC algorithm needs to use a model after linearization of the nonlinear control model. Therefore, compared with NMPC, the control model used by LMPC has a higher degree of mismatch with the physiological model of the virtual patient. This can also be seen from the closed-loop test results of the NMPC algorithm of the single-hormone system on the virtual patient 1 in Figure 4.6. Compared with the LMPC, the blood glucose under the control of the NMPC algorithm hardly has hyperglycemia.

The Figure 5.3 shows the control performance of the LMPC of the multiple-hormone system for virtual patient 1.



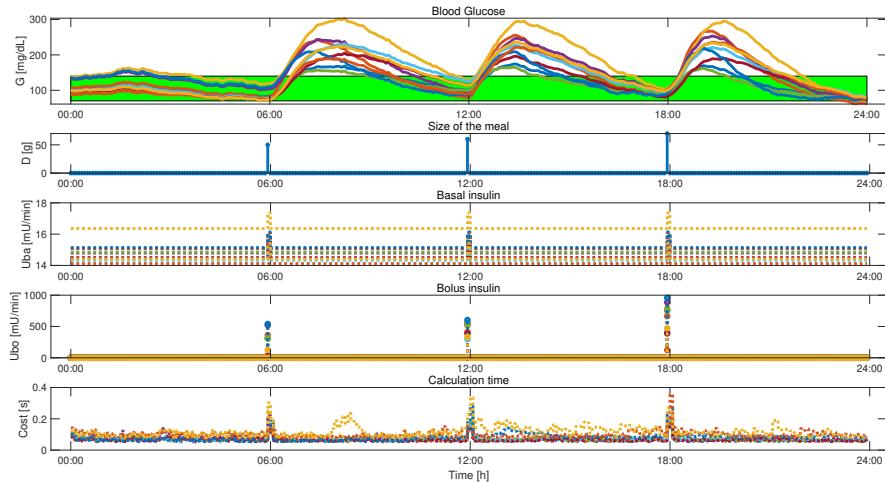
**Figure 5.3:** The control performance of the LMPC of the multiple-hormone system for virtual patient 1.

From the closed-loop simulation results in the Figure 5.3, it can be seen that during the fasting period, the LMPC algorithm of the multiple-hormone system mainly infuses basal insulin to stabilize blood glucose, and it also assists in the infusion of a small amount of glucagon to prevent blood glucose from dropping. At the time of the three meals intakes, the appropriate bolus insulin is calculated and infused to ensure that the blood glucose drops to the normal range as soon as possible. During the period of increased heart rate caused by exercise, the LMPC calculates an appropriate

amount of glucagon and infuses to prevent hypoglycemia. At the same time, basal insulin, bolus insulin and glucagon all meet the target infusion strategy.

Compared with the control effect of NMPC algorithm of the multiple-hormone on virtual patient 1 in Figure 4.9, the LMPC algorithm of multiple-hormone system is more sensitive to the control of glucagon. It can be seen that in addition to a certain dose of glucagon infusion to prevent hypoglycemia during the period of heart rate increase caused by exercise started at 15:00, there are also a small amount of glucagon infusion calculated at other times, which increases the risk of blood glucose rise. However, glucagon under the control of NMPC is only infused only when there is an unknown disturbance that causes blood glucose to drop. In other cases, the infusion of basal insulin and bolus insulin is mainly used to stabilize blood glucose.

In order to test the effectiveness and robustness of the LMPC algorithm, the designed and implemented LMPC algorithm is used to perform a 24-hour closed-loop simulation test on 10 virtual patients generated by the Hovorka model and his parameter distribution. Below are the test results of the single-hormone system.



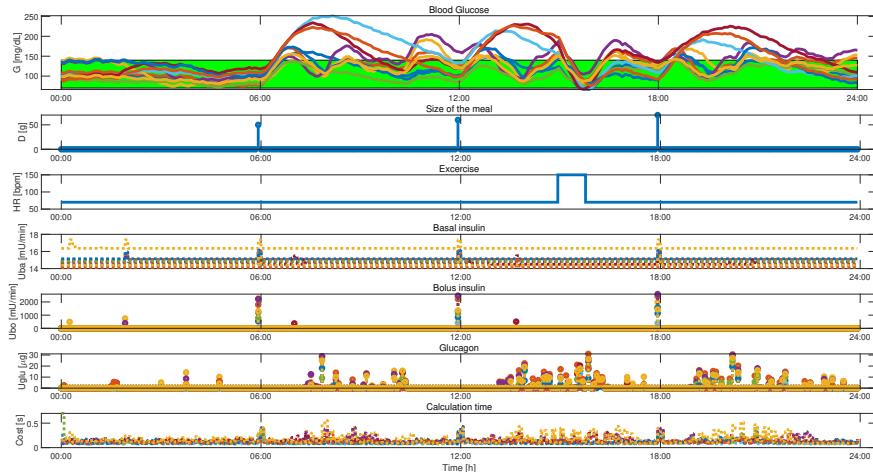
**Figure 5.4:** Closed-loop test result of LMPC algorithm of the single-hormone system on 10 virtual patients.

From the closed-loop simulation test results of 10 virtual patients, the control effect of the LMPC algorithm of single-hormone system meets the desired control strategy. However, for some virtual patients, the insensitivity to the upper limit control of blood glucose mentioned in the analysis of virtual patient 1 above increases the risk of hyperglycemia.

Compared with the results of the NMPC algorithm of the single-hormone in 10 virtual

patients, the NMPC algorithm has better stability. Calculating and infusing the correct dose of bolus insulin in response to meal intake not only reduces the risk and peak of hyperglycemia, but also avoids the occurrence of hypoglycemia. It can be seen that in addition to the difference in algorithm design, the matching degree between the used control model and the patient's physiological model greatly affects the control effect. Compared with the nonlinear control model obtained by direct parameter identification used by NMPC, the linearized control model used by LMPC lack part of the model information.

The test results of the LMPC algorithm of multiple-hormone system on 10 virtual patients is

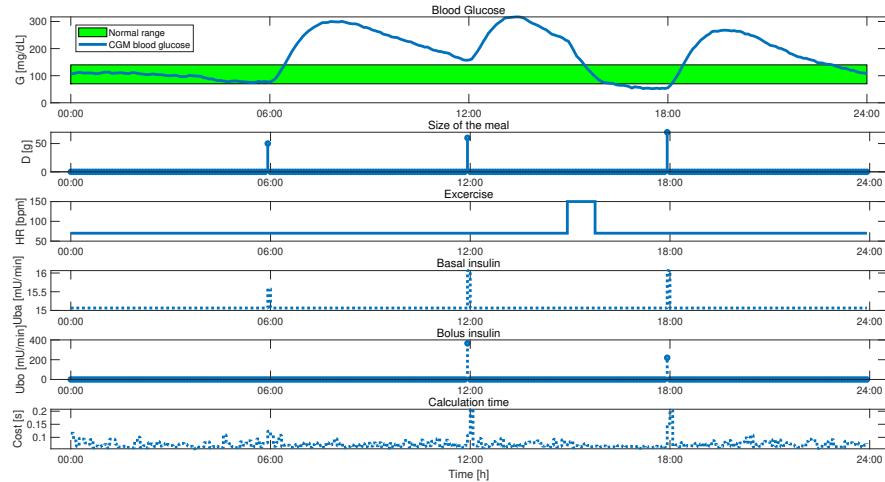


**Figure 5.5:** Closed-loop test result of LMPC algorithm of the multiple-hormone system on 10 virtual patients.

From the closed-loop simulation test results of the LMPC of the multiple-hormone system in the Figure 5.5, basal insulin, bolus insulin, and glucagon are all calculated at the appropriate dose and infused to stabilize blood glucose. Especially for the increase in heart rate caused by exercise starting at 15:00, the reasonable infusion of glucagon avoids the occurrence of hypoglycemia. In addition, the control effect of a small number of patients is similar to that of the LMPC of single-hormone system. Bolus insulin at a lower dose than the desired dose increases the risk of hyperglycemia, but at the same time, glucagon is not required. In the control process of most patients, the dose of bolus insulin for meal intake is correctly calculated and infused, but due to the high sensitivity to the lower limit of blood glucose, frequent glucagon infusions are used to prevent blood glucose from dropping.

Compared with the results of the NMPC algorithm of multiple-hormone system on 10 virtual patients, as mentioned in the previous analysis of the results of the virtual patient 1 by the LMPC of the multiple-hormone system, in addition to the unknown disturbance that causes the blood glucose to drop, a small amount of glucagon is also infused to control blood glucose when fasting and meal intake. But this kind of situation is rare in NMPC.

In order to verify the effectiveness of the LMPC algorithm of the multiple-hormone system on disturbance of exercise, the closed-loop simulation test results of the LMPC algorithm of the single-hormone system under disturbance of exercise are also shown as follows.



**Figure 5.6:** Closed-loop test result of LMPC algorithm of the single-hormone system under disturbance of exercise.

It can be seen from the Figure 5.6 that the LMPC of the single-hormone system cannot cope with the increase in heart rate caused by the exercise at 15:00, and hypoglycemia occurs. In the control of 10 patients by the LMPC algorithm of the multiple-hormone system, glucagon is calculated and infused to prevent the occurrence of hypoglycemia when the blood glucose drops.



# CHAPTER 6

# Conclusion

---

In this thesis, we designed and implemented an optimization-based control algorithm for the artificial pancreas(AP) problem. More specifically, two control algorithms, linear model predictive control(LMPC) and nonlinear model predictive control(NMPC), and their respective versions of single-hormone system and multiple-hormone system are designed and implemented by using the framework of CasADi. In addition, the implemented NMPC algorithm has Matlab and C++ versions, and the implemented LMPC has the Matlab version. In order to verify the effectiveness and robustness of the control algorithm, the implemented control algorithm is tested in closed-loop simulation on 10 virtual patients. The parameters of 10 virtual patients are generated by the extended Hovorka model and its parameter distribution. In addition, the parameters of the control model for LMPC and NMPC are obtained through parameter identification. The results of the closed-loop simulation test are compared and discussed.

In order to perform a closed-loop simulation test of the artificial pancreas(AP) problem, the extended Hovorka model used for the simulation is introduced. In addition, as the control model of the MPC algorithm, the extended MVP model is introduced. Both single-hormonal and multiple-hormonal versions of both models are described.

MPC is a model-based control algorithm, and the parameter identification effect of the control model greatly affects the control effect of MPC. The experimental design to generate the data set for parameter identification is described, and the least square(LS) method and maximum likelihood estimation(MLE) method are introduced as the method of parameter identification. the stochastic differential equations(SDE)-based explicit Euler-Maruyama method with a fixed step size and the ordinary differential equations(ODE)-based explicit Runge–Kutta 4 method(ERK4) method with a fixed step size used as numerical integration methods are introduced. The maximum likelihood estimation(MLE) method with better effect in the test is used to identify the parameters of 10 virtual patients.

We designed and implemented the NMPC algorithm of single-hormone system and multiple-hormone system for artificial pancreas(AP) problem. Both versions of Matlab and C++ are implemented. More specifically, CDEKF is introduced as an estimator to observe the initial state of the control model, and the multiple shooting

method is used to discretize the optimal control problem(OCP) into a non-linear programming(NLP) problem. In addition, the form of objective function and constraints are described and discussed. Since the penalties for the manipulated variable including basal insulin, bolus insulin and glucagon have a variety of combinations to achieve the desired control effect, in order to obtain a better control effect, we especially carried out a specific analysis on the form of the penalty term of the manipulated variable to explore the influence of different penalty forms on the control effect and the penalty form with better control effect. In this thesis, both LMPC and NMPC algorithms are implemented using the framework of CasADi, Taking the NMPC algorithm of single-hormone system as an example, how to use the CasADi framework to implement the MPC algorithm is introduced, and the codes of Matlab and C++ versions are also shown.

We conducted a closed-loop simulation test on the virtual patient 1 for the different penalty forms of the NMPC algorithm of the single-hormone system and the multiple-hormone system. Through the comparison of the results, it is concluded that for the single-hormone system, the  $\ell_1$  norm penalty form of tracking the target value of bolus insulin is necessary, and for the multiple-hormone system, the  $\ell_1$  penalty form of tracking target value of bolus insulin and glucagon is necessary. Regarding the penalty form for the basal insulin of the two systems, the  $\ell_2$  norm penalty form of tracking the target value is better than the  $\ell_2$  norm penalty form of change rate. It can be inferred that using a more accurate target value of basal insulin will get a better control effect, such as improving the accuracy of parameter identification or giving a more reasonable reference value by the doctor. In addition, the Matlab version and the C++ version of the NMPC algorithm respectively performed a closed-loop simulation test without noise on the virtual patient 1 and the control effect is compared. The blood glucose and the calculated dose of insulin and glucagon under the control of the two versions of NMPC are exactly the same, which verifies the correctness of the algorithm implementation, In addition to the correctness, the calculation time of the two versions is roughly the same, indicating that the MPC algorithm built using the CasADi framework has less dependence on the programming language. Finally, the NMPC algorithm of the single-hormone system and the multiple-hormone system is tested in closed-loop simulation on 10 virtual patients. From the results, it can be seen that the implemented NMPC algorithm can effectively control the blood glucose in the normal range and has good robustness. Some patients have hyperglycemia during the control process, which is considered to be related to the mismatch between the control model and the patient's physiological model. In addition, the calculated dose of the manipulated variables also meets the desired infusion strategy.

We designed and implemented the Matlab version of LMPC algorithm for the single-hormone system and the multiple-hormone system for the artificial pancreas(AP) problem. More specifically, discrete dynamic Kalman filters is introduced as an estimator to observe the initial state of the control model. The single shooting method is used to discretize the optimal control problem into a quadratic programming(QP)

problem, the form of the objective function and constraints and how to construct them using matrices are described and discussed. Finally, the LMPC algorithm of single-hormone system and multiple-hormone system is tested in closed loop simulation on 10 virtual patients. From the results, in response to the meal intake and increased heart rate caused by exercise, the appropriate doses of basal insulin, bolus insulin and glucagon are also correctly calculated to stabilize blood glucose. However, LMPC cannot achieve the same good control effect and robustness of NMPC. For the single-hormone system, hyperglycemia occurs more and the peak value is higher. For the multiple-hormone system, some patients need to infuse glucagon during non-exercise periods due to the inaccurate calculation of bolus insulin during the control process. We speculate that this is related to the control model used by LMPC is linearized by the nonlinear control model used by NMPC. Linearization makes the control model more mismatched with the patient's physiological model.

The following is about future work

- As a model-based control algorithm, the control effect of the MPC algorithm has a great relationship with the accuracy of the control model. In the artificial pancreas(AP) problem, the parameter identification effect of the control model determines the accuracy. In this thesis, an experimental design for generating a data set for parameter identification is proposed, but in order to improve the accuracy and robustness of parameter identification, more experimental designs can be explored and tried.
- In this thesis, in order to compare the control effects of LMPC and NMPC, the linear control model used by LMPC is obtained by linearizing the nonlinear control model obtained by parameter identification. However, parameter identification and linearization increase the degree of mismatch between the control model and the patient's physiological model, and make the control effect worse. The linear control model of LMPC can be obtained by other methods to reduce the degree of mismatch and improve the control effect, such as Polynomial Models(ARX, ARMAX or Box-Jenkins) and etc.



## APPENDIX A

# Parameters of 10 virtual patients

---

The parameters of 10 virtual patients generated by the Hovorka model and its parameter distribution are shown in Table A.1 and Table A.2.

**Table A.1:** The parameters of 10 virtual patients.

Patient	$A_G$ [unitless]	$\tau_D$ [min]	$V_G$ [L]	$F_{01}$ [mmol/min]	$k_{12}$ [1/min]	$EGP$ [mmol/min]
1	0.953	50.532	10.602	0.786	0.0612	1.139
2	1.107	36.071	11.247	0.744	0.08492	1.139
3	1.099	48.317	14.571	0.756	0.0838	1.27
4	1.175	48.857	13.804	0.856	0.0605	1.367
5	1.042	46.723	9.217	0.613	0.0433	1.067
6	0.927	48.306	10.432	0.626	0.0803	1.174
7	0.948	49.152	9.990	0.592	0.0639	1.139
8	0.748	34.515	14.074	0.818	0.0919	1.171
9	0.823	38.182	9.187	0.636	0.0908	1.334
10	1.191	48.636	10.224	0.890	0.0561	1.423

Table A.2: The parameters of 10 virtual patients.

Patient	$\tau_S$ [min]	$V_I$ [L]	$k_e$ [1/min]	$k_{a,1}$ [1/min]	$k_{a,2}$ [1/min]	$k_{a,3}$ [1/min]
1	61.054	9.907	0.175	0.0077	0.0911	0.0273
2	72.089	8.475	0.155	0.0071	0.105	0.0277
3	73.735	9.608	0.127	0.0030	0.0185	0.0105
4	51.034	9.692	0.116	0.0027	0.0613	0.0112
5	49.538	8.0187	0.158	0.0044	0.105	0.025
6	54.532	8.436	0.128	0.0065	0.110	0.0326
7	51.388	8.501	0.156	0.0021	0.116	0.0074
8	54.598	11.608	0.124	0.0055	0.0706	0.0437
9	67.934	9.536	0.144	0.0052	0.0266	0.0377
10	56.40	9.859	0.167	0.0025	0.108	0.0237

Patient	$S_{I,1}$ [L/mU]	$S_{I,2}$ [L/mU]	$S_{I,3}$ [L/mU]	$\tau_{GI}$ [min]	$BW$ [kg]
1	$59.94e^{-4}$	$8.401e^{-4}$	$227.88e^{-4}$	6.7	84.511
2	$42.31e^{-4}$	$7.923e^{-4}$	$281.159e^{-4}$	6.7	77.772
3	$23.796e^{-4}$	$8.418e^{-4}$	$309.495e^{-4}$	6.7	81.407
4	$21.522e^{-4}$	$7.666e^{-4}$	$268.198e^{-4}$	6.7	76.236
5	$53.460e^{-4}$	$3.983e^{-4}$	$256.620e^{-4}$	6.7	68.202
6	$26.454e^{-4}$	$12.653e^{-4}$	$256.431e^{-4}$	6.7	74.469
7	$56.635e^{-4}$	$8.384e^{-4}$	$329.005e^{-4}$	6.7	69.304
8	$33.894e^{-4}$	$6.773e^{-4}$	$286.138e^{-4}$	6.7	93.651
9	$77.286e^{-4}$	$13.727e^{-4}$	$288.630e^{-4}$	6.7	76.272
10	$55.998e^{-4}$	$12.472e^{-4}$	$298.804e^{-4}$	6.7	80.218



## APPENDIX B

# Source Code

---

The source code of the implemented algorithm and closed-loop simulation test for artificial pancreas(AP) can be found in the zip file attached to the report. Please check the “ReadMe.txt” for specific information.



# Bibliography

---

- [And+18] Joel Andersson et al. “CasADI: a software framework for nonlinear optimization and optimal control.” In: *Mathematical Programming Computation* 11 (July 2018).
- [And13] Joel Andersson. “A General-Purpose Software Framework for Dynamic Optimization.” PhD thesis. Arenberg Doctoral School, KU Leuven, October 2013.
- [BJ18] Dimitri Boiroux and John Bagterp Jørgensen. “Nonlinear Model Predictive Control and Artificial Pancreas Technologies.” In: *2018 IEEE Conference on Decision and Control (CDC)*. 2018, pages 284–290.
- [BK08] Marc Breton and Boris Kovatchev. “Analysis, Modeling, and Simulation of the Accuracy of Continuous Glucose Sensors.” In: *Journal of Diabetes Science and Technology* 2.5 (2008), pages 853–862.
- [Boi+11] Dimitri Boiroux et al. “Insulin Administration for People with Type 1 diabetes.” In: 2011.
- [Boi+19] Dimitri Boiroux et al. “Efficient Computation of the Continuous-Discrete Extended Kalman Filter Sensitivities Applied to Maximum Likelihood Estimation.” In: 2019.
- [Boi12] Dimitri Boiroux. “Model Predictive Control Algorithms for Pen and Pump Insulin Administration.” English. PhD thesis. 2012.
- [BPC81] RN Bergman, LS Phillips, and C Cobelli. “Physiologic evaluation of factors controlling glucose tolerance in man: measurement of insulin sensitivity and beta-cell glucose sensitivity from the response to intravenous glucose.” In: *The Journal of clinical investigation* 68.6 (December 1981).
- [Cho+18] N.H. Cho et al. “IDF Diabetes Atlas: Global estimates of diabetes prevalence for 2017 and projections for 2045.” In: *Diabetes Research and Clinical Practice* 138 (February 2018).
- [Fac+14] Andrea Facchinetto et al. “Modeling the glucose sensor error.” English (US). In: *IEEE Transactions on Biomedical Engineering* 61.3 (March 2014), pages 620–629. ISSN: 0018-9294.

- [Fer+14] H.J. Ferreau et al. “qpOASES: A parametric active-set algorithm for quadratic programming.” In: *Mathematical Programming Computation* 6.4 (2014), pages 327–363.
- [Gal+09] Federico Galvanin et al. “Optimal Design of Clinical Tests for the Identification of Physiological Models of Type 1 Diabetes Mellitus.” In: *Industrial & Engineering Chemistry Research* 48.4 (2009), pages 1989–2002.
- [Hov+04] Roman Hovorka et al. “Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes.” In: *Physiological Measurement* (July 2004).
- [Kan+09] Sami S. Kanderian et al. “Identification of Intraday Metabolic Profiles during Closed-Loop Glucose Control in Individuals with Type 1 Diabetes.” In: *Journal of Diabetes Science and Technology* 3.5 (2009), pages 1047–1057.
- [KD15] Akram Kharroubi and Hisham Darwish. “Diabetes mellitus: The epidemic of the century.” In: *World journal of diabetes* 6 (June 2015), pages 850–67. DOI: [10.4239/wjd.v6.i6.850](https://doi.org/10.4239/wjd.v6.i6.850).
- [Mag+09] L. Magni et al. “Model predictive control of glucose concentration in type I diabetic patients: An in silico trial.” In: *Biomedical Signal Processing and Control* 4.4 (2009), pages 338–346.
- [Mar55] G. Maruyama. “Continuous Markov processes and stochastic equations.” In: *Rendiconti del Circolo Matematico di Palermo* 4 (1955), pages 48–90.
- [Rei+14] Hanna Reiband et al. “Dual-hormone treatment with insulin and glucagon in patients with type 1 diabetes.” In: *Diabetes/Metabolism Research and Reviews* 31 (December 2014).
- [SMM09] Irma Y. Sánchez Chávez, Rubén Morales-Menéndez, and Sergio O. Martínez Chapa. “Glucose optimal control system in diabetes treatment.” In: *Applied Mathematics and Computation* 209.1 (2009), pages 19–30.
- [Wat+11] E.M. Watson et al. “A new general glucose homeostatic model using a proportional-integral-derivative controller.” In: *Computer Methods and Programs in Biomedicine* 102.2 (2011), pages 119–129.
- [WB06] Andreas Wächter and Lorenz Biegler. “On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming.” In: *Mathematical programming* 106 (March 2006), pages 25–57.