Derivative-Free Optimization to model Continuous Glucose Monitoring in Type 2 Diabetes

A Thesis

submitted to
Indian Institute of Science Education and Research Pune
in partial fulfillment of the requirements for the
BS-MS Dual Degree Programme

by

Durga M Parkhi

Indian Institute of Science Education and Research Pune
Dr. Homi Bhabha Road,
Pashan, Pune 411008, INDIA.

April, 2019

Supervisor: Prof. Pranay Goel

Certificate

This is to certify that this dissertation entitled Derivative-Free Optimization to model Continuous Glucose Monitoring in Type 2 Diabetestowards the partial fulfilment of the BS-MS dual degree programme at the Indian Institute of Science Education and Research, Pune represents study/work carried out by Durga M Parkhi at Indian Institute of Science Education and Research under the supervision of Prof. Pranay Goel, Associate Professor, Department of Biology , during the academic year 2018-2019.

Prof. Pranay Goel

Committee:

Prof. Pranay Goel

Prof. Santhanam

This thesis is dedicated to everyone suffering from diabetes and its complications.

Declaration

I hereby declare that the matter embodied in the report entitled Derivative-Free Optimization to model Continuous Glucose Monitoring in Type 2 Diabetes are the results of the work carried out by me at the Department of Biology, Indian Institute of Science Education and Research, Pune, under the supervision of Prof. Pranay Goel and the same has not been submitted elsewhere for any other degree.

Durga M Parkhi

Acknowledgments

I would like to take this opportunity to thank everyone who has contributed in making this study successful.

First of all, I would like to express my deep gratitude to my guide, Professor Pranay Goel, who has mentored me for the past 3 years and has groomed me to be a research student. I am indebted to him for teaching me to learn on my own, and in particular, through my mistakes. He always urged me to discern my mistakes and correct them myself. He gave me a vast exposure to the real scientific research environment. The learnings from his emphasis on the quality of work and perfection will be very valuable in the future, too. I feel blessed to have had him as my guide.

Next, I would like to thank Dr. Anil Vaidya and Dr. Mita Shah for providing the clinical data for this study.

Arjun and Sandra, who are Ph.D. students from our lab, have constantly supported me all throughout this work and have been a source of motivation and inspiration. Sandra's relentless efforts in helping me write a final report on modeling for gastroparesis are truly commendable.

At a broader level, I would like to thank IISER for providing a platform to get a glimpse of scientific research, from all the semester projects and this final year project, that has been included in our curriculum. The entire IISER fraternity was always ready to help, whenever needed.

Last but not the least, my family has always been extremely supportive during all

my endeavors.

Abstract

Continuous Glucose Monitoring (CGM) is a novel technique to monitor plasma glucose levels in response to food and other daily activities of people diagnosed with Type 2 Diabetes (T2D). In an earlier study (Goel et al., 2018), we proposed a new dynamic diagnostic test, called 'Isolated Liquid Meal Tolerance Test' (ILMTT) to obtain personalized estimates of clinically important parameters like Insulin Sensitivity index (S_I) and maximum Insulin Secretion (I_{max}). This is achieved through the fitting of the CGM data, with isolated liquid meals, to the model glucose graph by following a standardized optimization procedure. The formal optimization is performed in Matlab, using the function fmincon. However, as CGM data is noisy and contains discrete points, it is not smooth. Here, we develop an optimization code based on derivative free methods, (directional direct search method) and test it on more CGM data sets to obtain personalized S_I and I_{max}. In addition to this, we use derivative-free optimization to model glucose and insulin data obtained from mixed meal tolerance test (MMTT) of a person diagnosed with gastroparesis. In the long term, the aim of this study is to develop a robust algorithm that should give out the indices S_I and I_{max} when the CGM data is given as input, for as many people as needed, in a very short time. Thus, we present a novel approach for modeling CGM data in type 2 diabetes using derivative-free optimization. We propose that our technique will be useful for developing a personalized medicine in T2D: Personalized analysis of CGM will not only help in prescribing the diet for T2D patients but also be useful in testing the exact effect of drugs.

Keywords: continuous glucose monitoring, minimal model, type 2 diabetes, insulin secretion, derivative free methods

Contents

Introduction

What is Derivative-free optimization (DFO)?

Derivative-free optimization focuses on developing algorithms to optimize a function without having the analytical knowledge of that function [1]. Most of the optimization algorithms that have been developed till now are based on the information of the derivatives of the function. For instance, Taylor models have been widely used in derivative-based optimization. However, it always may not be possible to find the derivative of the function to be optimized. For the purpose of optimizing such functions, a considerably new field, called derivative-free optimization, is emerging. There are mainly three reasons for requiring to use derivative-free optimization [2]:

1. The objective function to be optimized may not be known in the explicit functional form. In this case, since the functional form is not known, computing its derivative is not possible.
2. The explicit functional form of the objective function may be known, but it may be too expensive to compute its derivative. This case arises especially when the objective function is very complex.
3. The objective function may have a simple functional form, but it may be very noisy. In this case, the computed derivatives of the objective function may not be accurate.

Derivative-free optimization can be used in unconstrained as well as constrained optimization. However, the field of constrained derivative-free optimization is very new.

We will only look at unconstrained derivative-free optimization and its applications as a part of this work. Both smooth and non-smooth problems can be addressed using derivative-free optimization [3].

Why is DFO used in this study?

The number of people getting affected by diabetes is continuously rising over the years. At the same time, a lot of efforts are being directed to using technology in the study of diabetes. Continuous glucose monitoring (CGM) is one such novel way for measuring blood glucose at regular intervals, and gives us a broad insight into the glycemic excursions after meals and other regular activities throughout the day [4]. In an earlier study, we have already modeled the glucose pulsatility observed in CGM data of a non-diabetic and diabetic person, and recovered the model parameters, which can potentially be useful in developing personalized medicine for type 2 diabetes [5]. The optimization was carried out using derivative-based methods like $fmincon$ and *patternsearch*, in Matlab. However, since CGM data is very noisy, and consists of discrete points, it is not smooth. Therefore, using derivative-based methods for optimization of CGM data may not yield the best results. Therefore, we propose to extend the work from [5] by using derivative-free optimization.

Goal of this study

To the best of our knowledge, this study is a first attempt at using derivative-free optimization for modeling Continuous Glucose Monitoring data in type 2 diabetes. Towards the end, we will also use derivative-free optimization to model both glucose and insulin data simultaneously, obtained from mixed meal tolerance test (MMTT) of a patient diagnosed with gastroparesis. In general, the goal of this study is to attempt to solve the problem: $\arg\min_{x\in\mathbb{R}^n} f(x)$, where $f(x)$ is the objective function to be optimized and x is the set of parameters over which the derivative-free optimization will take place.

Chapter 1

Preliminaries

1.1 Introduction

Insulin sensitivity index (S_I) and maximum insulin secretion index (I_{max}) are the two major indices in the study of type 2 diabetes. S_I describes how sensitive our body is to the effects of insulin and I_{max} is the maximum capacity of the β cells to secrete insulin in order to lower the blood glucose concentration. Hence, when insulin sensitivity is lowered, the β cells get a false feedback that blood glucose concentration is still higher than normal due to insufficient secretion of insulin. In order to compensate for this condition, the β cells tend to secrete even more insulin. This is known as 'hyperinsulinemia' which occurs in the early stages of type 2 diabetes. As this condition progresses, even the maximum secretory capacity of the β cells starts deteriorating. Thsis leads to the phenomenon known as 'β cell exhaustion' in the later stages of type 2 diabetes. As most of all these stages of type 2 diabetes can be described with the help of the indices S_I and I_{max}, they play a crucial role in the study of type 2 diabetes.

It is already known that the indices S_I and I_{max} vary from person to person. However, to the best of our knowledge, there is no study yet published to estimate the actual values of S_I and I_{max} for a particular person. The novel idea of this research is that we can obtain a good estimate of the actual values of S_I and I_{max} for a particular person as output when the Continuous Glucose Monitoring (CGM) data of that person is provided as input to our model. Inside the model: The model glucose obtained from the glucose differential equation is dependent on the model parameters. The aim is to obtain the optimized set of model parameters such that the model glucose curve best fits the input CGM data. As S_I and I_{max} are two of the model parameters, we obtain their optimized values as the output of this fitting process.

Through our analysis, it has been observed that liquid meals from the CGM data can be captured well by our model, and hence, we proposed the 'Isolated Liquid Meals Tolerance Test' (ILMTT). ILMTT essentially means that during the course of CGM measurement, the person should intake liquid meals, which are separated from each other by a sufficient time interval. The estimation of the indices S_I and I_{max} can potentially lead to the development of personalized medicine for type 2 diabetes.

1.2 Continuous Glucose Monitoring (CGM)

CGM is a novel technique used to measure blood glucose levels at regular intervals. The sensor, which is attached to either the arm or the abdomen of the person, measures blood glucose at every 15 minutes for a period of 14 days. The time series obtained from CGM data helps us get broad insight into the variation of blood glucose in a day, following meals and other regular activities. It can be observed that the average blood glucose is much higher in the diabetic CGM data (around 200 mg/dl) than that in the non-diabetic CGM data (around 100 mg/dl). Also, the width and the amplitude of the glucose pulses in the diabetic CGM time series is much larger than that in the non-diabetic CGM time series. CGM not only depends on the food intake the person has during the course of measurement, but it also depends on the individual physiology of the person. Hence, a complete dynamic modeling of the CGM time series is difficult at this point of time. In this study, our aim was to model the 'glucose pulsatility' involved in CGM time series. Glucose pulsatility is the variation of glucose following a meal. In other words, we tried to model the glucose peaks observed in the CGM data following breakfast, lunch, dinner, etc.

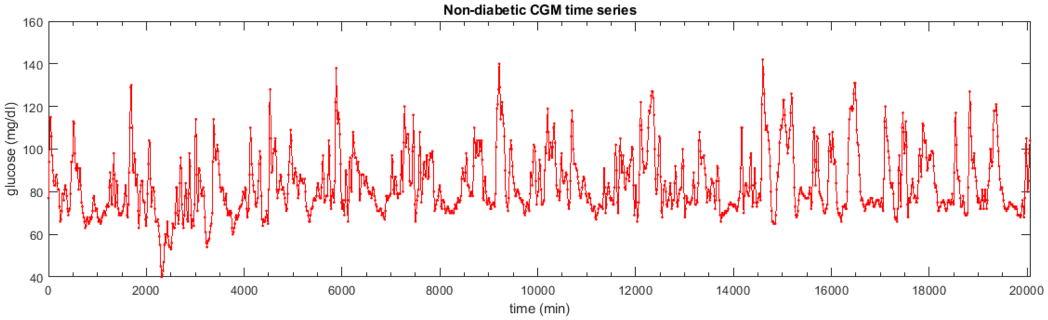

Figure 1.1: The figure shows the Continuous Glucose Monitoring (CGM) time series of a non-diabetic person. It can be seen from the time series that the average blood glucose is approximately 100 mg/dl. Also, the width and the amplitude of the glucose pulses appears to be smaller than in the diabetic CGM time series.

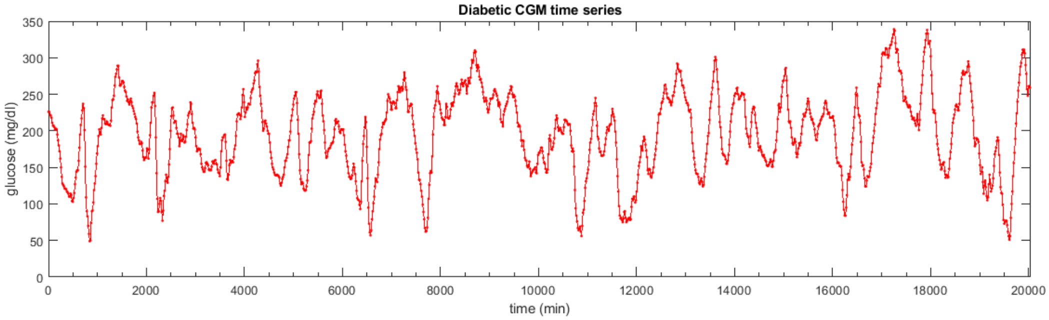

Figure 1.2: The figure shows the Continuous Glucose Monitoring (CGM) time series of a diabetic person. It can be seen from the time series that the average blood glucose is approximately 200 mg/dl. Also, the width and the amplitude of the glucose pulses appears to be larger than in the non-diabetic CGM time series.

1.3 The CGM model

For this purpose, we adopted the Topp model for glucose-insulin dynamics and developed a two compartmental food dynamics model. The stomach and the gut are the two compartments.

The CGM Model is given as follows:

$$\frac{dG}{dt} = R_0 - (E_{G0} + S_I I)G + k_{gut} q_{gut} \tag{1.1}$$

$$\frac{dI}{dt} = \frac{I_{max}G^2}{\alpha + G^2} \tag{1.2}$$

$$\frac{dq_{sto}}{dt} = -k_{sto}q_{sto} \tag{1.3}$$

$$\frac{dq_{gut}}{dt} = k_{sto}q_{sto} - k_{gut}q_{gut} \tag{1.4}$$

Food enters the stomach and leaves to enter the gut with rate constant k_{sto}. It leaves the gut with rate constant k_{gut} and then enters the bloodstream. The glucose differential equation in the Topp model for G-I dynamics contains terms for constant rate of increase of glucose (R_0), insulin-independent disposal of glucose from the blood ($E_{G0} \cdot G$), insulin-dependent disposal of glucose from the blood ($-S_I \cdot I \cdot G$) and a term responsible for the the rate of increase of glucose in the blood due to the intake of food ($k_{gut} \cdot q_{gut}$). In the insulin differential equation, the rate of increase of insulin in the blood is given by the term $\frac{I_{max}G^2}{\alpha+G^2}$, which is a sigmoidal function and the clearance of insulin from the blood is given by the term $-k_I \cdot I$.

The cost function to be optimized is the sum of squared difference between CGM data and the model glucose. This is a function of the model parameters over which the optimization is carried out. The optimization was carried out in Matlab using a combination of the functions *fmincon* and *patternsearch* using suitable constraints and the tolerance was set to 10^{-6} for the termination of the code.

1.4 Results

1.4.1 Non-diabetic CGM

The figure depicts a 24-hr period non-diabetic CGM data (Red). The black curve is the model glucose curve. We can see that the first peak, which is due to a liquid meal fits well as compared to the other two peaks, which are due to mixed meals.

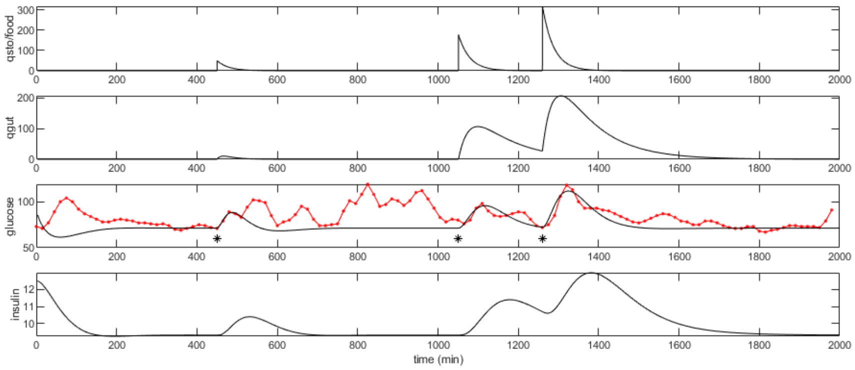

Figure 1.3: The figure shows a 24 hr period non-diabetic CGM data (red), starting at 12:00 hrs in the night, overlaid with the model simulation glucose curve (black). Out of the three selected foods, as can be seen in the figure, the first is a liquid meal and other two are mixed meals. Liquid meal can be seen to fit very well as compared to mixed meals.

1.4.2 Diabetic CGM

Similar procedure was followed for fitting the liquid meal from diabetic CGM data. We captured this liquid meal from the food diary we asked the patients to maintain during the course of CGM measurement. As it can be seen from the figure, the peak due to a liquid meal from the diabetic CGM data can be fit well by the CGM model.

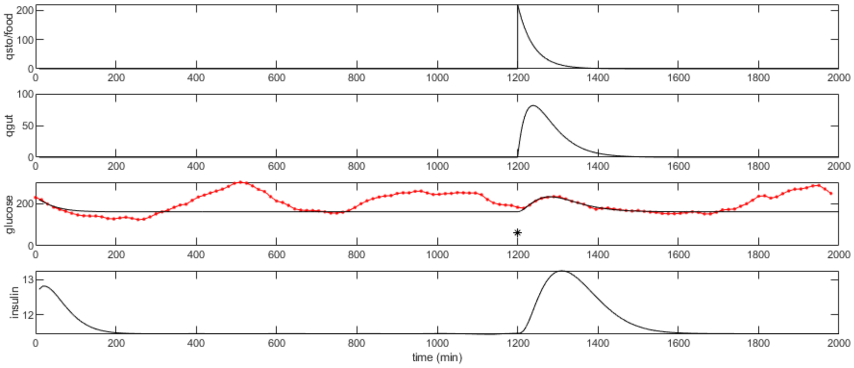

Figure 1.4: The figure shows a 24 hr period diabetic CGM data (red), starting at about 04:00 hrs in the night, overlaid with the model simulation glucose curve (black). An isolated liquid meal was selected from the diet diary of the patient and can be seen to fit very well by the model.

1.5 Objective of the current study

Further, we decided to extend this study by using derivative-free method for optimization. We decide to use derivative free methods for two reasons:

1. CGM data may involve noise.
2. CGM data is not smooth, it is discontinuous.

As a result, use of derivative-based methods like $fmincon$ and *patternsearch* may induce some errors in the optimized values of the model parameters. Therefore, we resorted to using derivative-free optimization. Derivative-free optimization and its numerous applications are described in this thesis in the chapters ahead.

The objective of this work is as follows:

1. To study derivative optimization algorithm and develop codes in Matlab to test on various applications
2. To study if derivative-free optimization can be used to recover the parameters of the Continuous Glucose Monitoring (CGM) model
3. To evaluate the model parameters of the modified CGM model as well as the model having gut dynamics given by the Lehmann-Deutsch model, to fit glucose and insulin data obtained from Mixed Meal Tolerance Test (MMTT) of a patient suffering from gastroparesis

1.6 Defining the Problem

1. For Objective 2:
   Unconstrained optimization of the cost function f, where f is the sum of squared difference between the CGM data and model glucose, at particular time values. x is the set of model parameters over which the optimization will take place.

$$\underset{x\in\mathbb{R}^n}{\arg\min} f(x) = \underset{x\in\mathbb{R}^n}{\arg\min} \quad \sum(\text{CGM} - \text{model glucose})^2 \tag{1.5}$$

2. For Objective 3: Unconstrained optimization of the cost function f, where f is the sum of sum of squared difference between observed glucose data and model glucose and observed insulin data and model insulin, at particular time values. x is the set of model parameters over which the optimization will take place.

$$\underset{x\in\mathbb{R}^n}{\arg\min} f(x) = \underset{x\in\mathbb{R}^n}{\arg\min} \quad (\sum(\text{observed glucose data - model glucose})^2 \tag{1.6}$$

$$+\sum(\text{observed insulin data - model insulin})^2) \tag{1.7}$$

Chapter 2

Derivative free Optimization

In this chapter, we will discuss part of the theory behind derivative-free optimization and see numerous applications. Our focus will be on ‘directional direct search method’ which is an important class of derivative-free methods, used to optimize a given objective function. The essence of directional direct search method is that the objective function is sampled at finite number of points in the neighborhood of the current iterate at each iteration and the next action to be taken is based on these function values at the sampled points [2].

Definition 2.0.1. *$D_{\oplus} = [I - I]$ is the Polling vector.*

Definition 2.0.2. *x_k is the current iterate*

Definition 2.0.3. *The sample points of the form*

$$P_k = \{x_k + \alpha_k \cdot d \quad | \, d \in D_{\oplus}\} \tag{2.1}$$

are the Poll points.

The algorithm samples the poll points in a predefined order and tries to decrease the value of the objective function for any one of the poll points sampled.

2.1 Directional Direct Search Method

2.1.1 Algorithm

Let the objective function to be minimized be f (as defined in Equations (1.5) and (1.6)).

1. Initialization -
   Fix $x_1 > 0$ and $\alpha_1 > 0$.

2. Poll step -
   Evaluate the objective function at the polling points in the polling set and set

$$x_{k+1} = x_k + \alpha_k d \quad \text{if} \quad f(x_k + \alpha_k d) < f(x_k) \tag{2.2}$$

$$x_{k+1} = x_k \quad \text{otherwise} \tag{2.3}$$

$$\tag{2.4}$$

3. Parameter update -

$$\alpha_{k+1} = 2\alpha_k \quad \text{if poll step is successful(Equation (2.2))} \tag{2.5}$$

$$\alpha_{k+1} = \frac{1}{2}\alpha_k \quad \text{otherwise(Equation (2.3))} \tag{2.6}$$

Note -
The poll step must terminate after a finite number of steps, since ∃ at least one descent direction d ∈ D_k where D_k is a positive basis, provided the following assumptions hold:

1. The objective function possesses some differentiability properties

2. The current iterate x_k is not a stationary point for the objective function

2.2 Global Convergence in the continuously differentiable case

In this section, we list the theorems for global convergence of continuously differentiable functions. The proofs of these theorems can be found in [1].
Following is a list of assumptions needed for the theorems.

1. Assumption 1 - The level set $L(x_0) = \{x \in \mathbb{R}^n | f(x) \leq f(x_0)\}$ is compact.
2. Assumption 2 - If $\exists \alpha > 0$ such that $\alpha_k > \alpha \forall k$, then the algorithm visits only a finite number of points.
3. Assumption 3 - Let ξ_1 and $\xi_2 > 0$ be fixed constants. The positive bases D_k used in the algorithm are chosen from the set $\mathbb{D} = \{\bar{D} positive basis | cm(\bar{D}) > \xi_1, ||\bar{d}|| < \xi_2, \bar{d} \in \bar{D}\}$.
4. Assumption 4 - When $|D| = \infty$, and ∇f is Lipschitz continuous with Lipschitz constant $\nu > 0$ in an open set containing $L(x_0)$. This assumption is not needed when $|D| < \infty$.
   Also,
   $||\nabla f(x)|| \leq \frac{\nu}{2} cm(D)^{-1} \max_{d \in D} ||d|| \alpha$.
5. Assumption 5 - $|D| < \infty$.
6. Assumption 6 - f is continuously differentiable in an open set contaioning $L(x_0)$. (This assumption is needed as ∇f is needed to be continuous).

Theorem 2.2.1. *Let Assumption 2 hold. Then,* $\liminf_{k \to 0} \alpha_k = 0$.

Corollary 2.2.2. *Let Assumptions 1 and 2 hold. Then,* $\exists$ *a point* x_* *and a subsequence* k_i *of unsuccessful iterates for which* $\lim_{i \to \infty} \alpha_{k_i} = 0$ *and* $\lim_{i \to \infty} \alpha_{k_i} = x_*$.

Theorem 2.2.3. *Let Assumptions 1, 2, 3 and 4 hold. Then,* $\liminf_{k \to \infty} ||\nabla f(x_k)|| = 0$ *and the sequence of iterates* $\{x_k\}$ *has a limit point* x_* *(as in corollary) for which* $\nabla f(x_*) = 0$.

Theorem 2.2.4. *Let Assumptions 1, 2, 5 and 6 hold. Then, the sequence of iterations* $\{x_k\}$ *has a limit point* x_* *for which* $\nabla f(x_*) = 0$.

2.3 Application of Directional Direct Search Method to test functions

In this section, examples of application of derivative free optimization to a lot of test suite problems is shown. All the functions are examples of unconstrained optimization. The function was given as input to the DFO code developed in Matlab (an example of which is given in Appendix (6.1)) and an initial guess for parameters was provided. The optimized values of the parameters obtained as output from the optimization process were compared to the known global minima for that function. The same process is followed for all functions in this section.

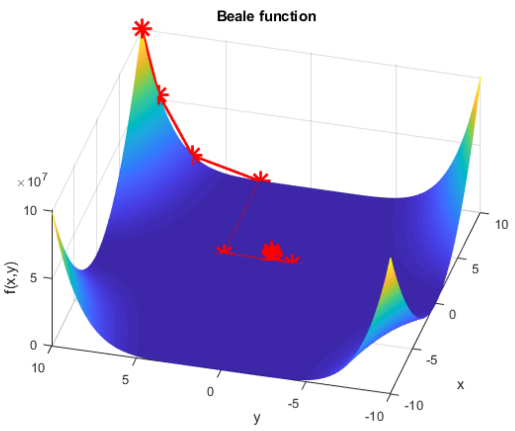

Figure 2.1: **Beale function:** The figure depicts convergence to the point of minimum from the initial guess, with every iteration.

$f(x,y) = (1.5 - x + xy)^2 + (2.25 - x + xy^2)^2 + (2.625 - x + xy^3)^2$

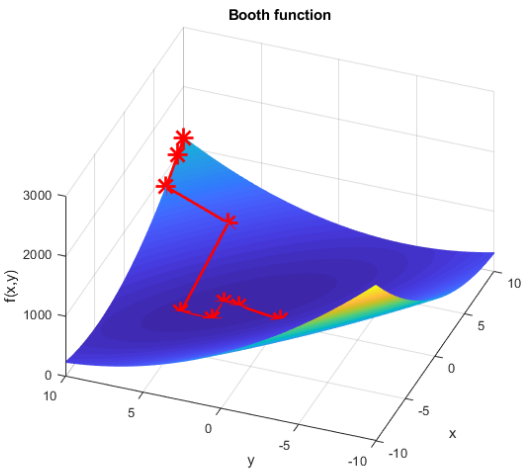

Figure 2.2: **Booth function:** The figure depicts convergence to the point of minimum from the initial guess, with every iteration.

$f(x,y) = (x + 2y - 7)^2 + (2x + y - 5)^2$

	x	y
Initial guess	10	10
Optimized value	2.999832	0.4999542
Known global min	3	0.5

Table 2.1: Table of parameters for Beale function.

	x	y
Initial guess	10	10
Optimized value	1	3
Known global min	1	3

Table 2.2: Table of parameters for Booth function.

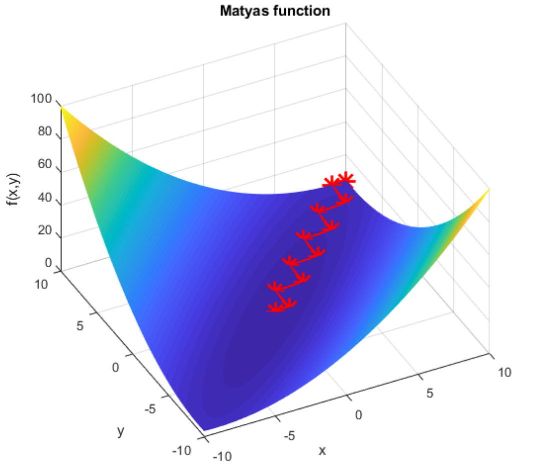

Figure 2.3: **Matyas function:** The figure depicts convergence to the point of minimum from the initial guess, with every iteration.

$$f(x, y) = 0.26(x^2 + y^2) - 0.48xy$$

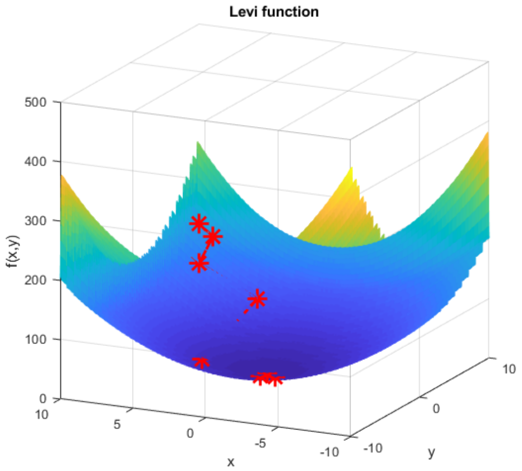

Figure 2.4: **Levi function:** The figure depicts convergence to the point of minimum from the initial guess, with every iteration.

$$f(x, y) = sin^2(3\pi x) + (x - 1)^2(1 + sin^2(3\pi y)) + (y - 1)^2(1 + sin^2(2\pi y))$$

	x	y
Initial guess	10	10
Optimized value	0	0
Known global min	0	0

Table 2.3: Table of parameters for Matyas function.

	x	y
Initial guess	10	10
Optimized value	1	1
Known global min	1	1

Table 2.4: Table of parameters for Levi function.

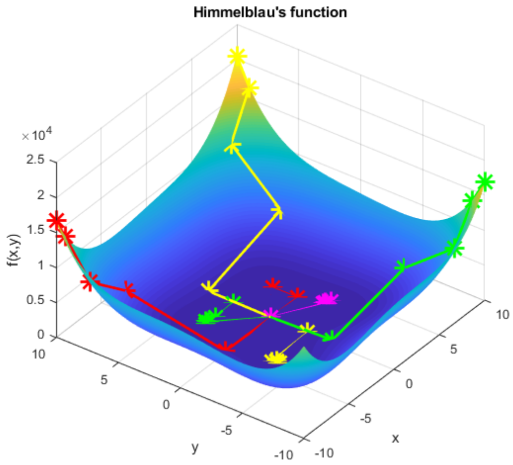

Figure 2.5: **Himmelblau function:** The figure depicts how the algorithm converges from different initial guesses to different points of minimum with every iteration.

$f(x,y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2$

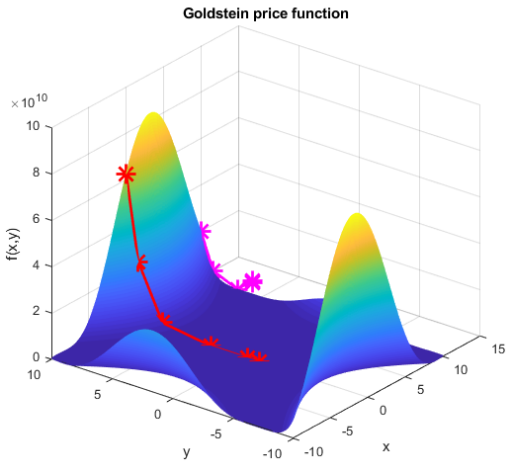

Figure 2.6: **Goldstein price function:** The figure depicts how the algorithm does not converge to the point of global minimum (shown in red) with every iteration, on starting with a different initial guess (shown in purple).

$f(x,y) = [1 + (x + y + 1)^2(19 - 14x + 3x^2 - 14y + 6xy + 3y^2)][30 + (2x - 3y)^2(18 - 32x + 12x^2 + 48y - 36xy + 27y^2)]$

	Initial guess	Optimized value	Known Global Minima
(x,y)	(-10,10)	(3,2)	(3,2)
(x,y)	(10,-10)	(-2.805115,3.131317)	(-2.805118,3.131312)
(x,y)	(10,10)	(-3.779312,-3.283188)	(-3.779310,-3.283186)
(x,y)	(4,-2)	(3.584427,-1.848129)	(3.584428,-1.848126)

Table 2.5: Table of parameters for Himmelblau's function.

	Initial guess	Optimized value	Known Global Minima
(x,y)	(0,10)	(0,-1)	(0,-1)
(x,y)	(10,10)	(1.800217,0.2001495)	-

Table 2.6: Table of parameters for Goldstein price function.

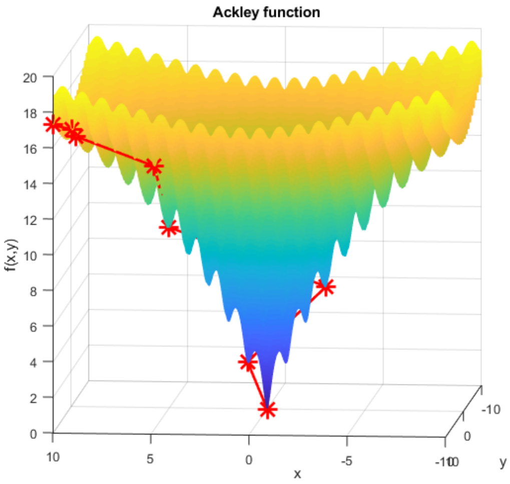

Figure 2.7: **Ackley function:** The figure depicts how the algorithm converges from the initial guess to the point of global minimum with every iteration, inspite of various local minimas in between..

$$f(x,y) = -20e^{[0.2\sqrt{0.5(x^2+y^2)}]} - e^{[0.5(cos(2\pi x)+cos(2\pi y))]} + e + 20$$

	x	y
Initial guess	10	10
Optimized value	0	0
Known global min	0	0

Table 2.7: Table of parameters for Ackley function.

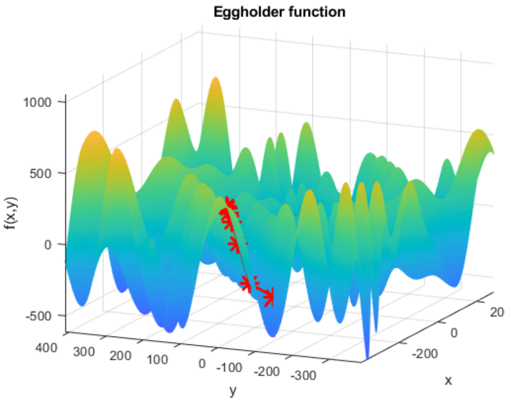

Figure 2.8: **Eggholder function:** The figure depicts how the algorithm converges from the initial guess to the point of local minimum with every iteration. The algorithm is unable to converge to the global minimum even for an initial guess in a small neighborhood of the global minimum.

$$f(x,y) = -(y+47)\sin\sqrt{\left|\frac{x}{2}+(y+47)\right|} - x\sin\sqrt{\left|x-(y+47)\right|}$$

	x	y
Initial guess	-350	0
Optimized value	-408.7198	-156.1012
Known global min	512	404.2319

Table 2.8: Table of parameters for Eggholder function.

2.4 Applications of Directional Direct Search Method: Linear Regression

The directional direct search method algorithm was applied to obtain the best fit line through the points (1,1), (2,4) and (5,8) in the cartesian plane. For this linear regression problem, the cost function can be shown to be:

$$f = 30x^2 + 3y^2 + 16xy - 98x - 26y + 81 \tag{2.7}$$

Derivative free optimization of this cost function yields the following results:

Table 2.9: Parameter Comparison Table.

	slope (x)	intercept (y)
Initial guess	5	0
Point of minima	1.6539	-0.0769

Thus, the equation of the best fit line through the points (1,1), (2,4) and (5.8) is:

$$y = 1.6539 \cdot x - 0.0769 \tag{2.8}$$

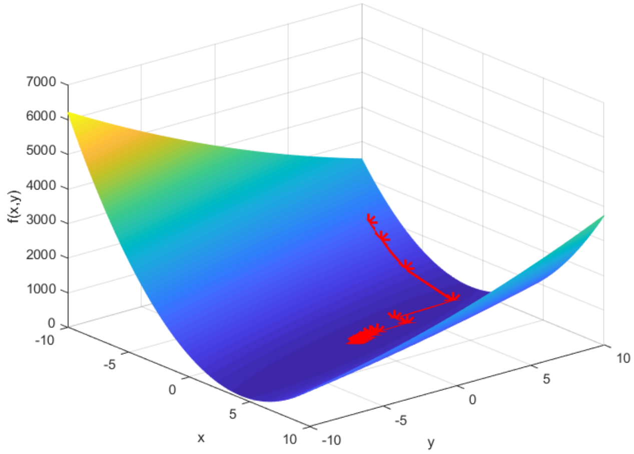

Figure 2.9: The figure depicts how the algorithm converges from the initial guess to the point of minimum with every iteration.

2.5 Applications of Directional Direct Search Method: Decay fit

The directional direct search method was applied to a hypothetical simple exponential decay system. The decay constant (par1) and the step height (par2) were the parameters optimized in this case. Data corresponding to par1 = 0.1 and par2 = 20 was generated and the parameters par1 and par2 in the ode were optimized using derivative free optimization to obtain the best fit of the generated data. The following table summarizes the output obtained:

Table 2.10: Parameter Comparison Table.

	par1	par2
Data values	0.1	20
IC	0.9	1
Optimized values	0.1000169	20.00854

It took 8950 iterations to reach these optimized values for the tolerance of 1e-06.

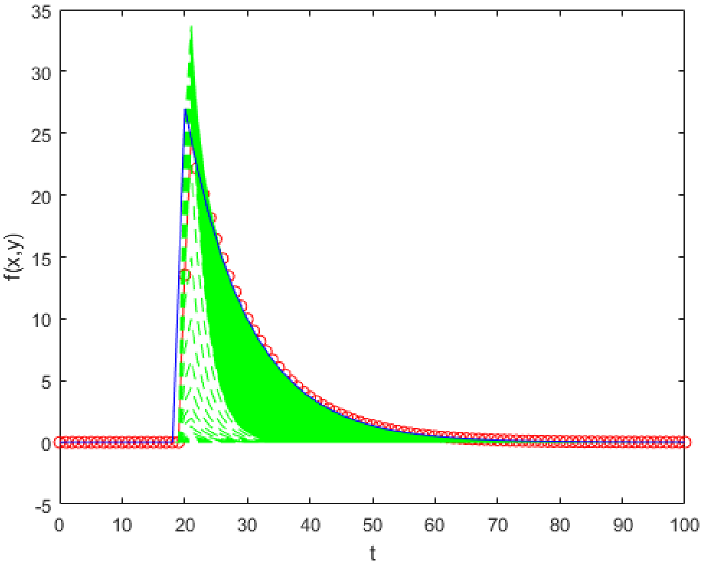

Figure 2.10: The figure depicts how the algorithm converges from the initial guess to the point of minimum of the cost function, corresponding to the best fit of the generated data, with every iteration.

2.6 Applications of Directional Direct Search Method: Non-Diabetic CGM fit

The directional direct search method was applied to fit the CGM data of a non-diabetic subject. R_0 and S_I were the parameters from the CGM model (Equations (1.1) to (1.4)) optimized in this case to obtain the best fit of the three selected foods. The following table summarizes the output obtained:

Table 2.11: Parameter Comparison Table.

	par1	par2
Optimized values from literature	2.1	3.06
IC	10	10
Optimized values	3.000147	3.959921

The code was run for 10000 iterations.

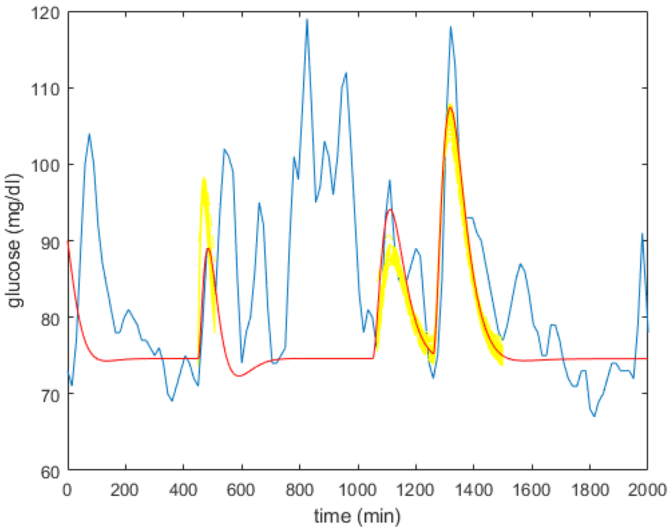

Figure 2.11: The figure depicts how the algorithm converges, with every iteration, from the initial guess to the point of minimum of the cost function, corresponding to the best fit (red) of the CGM data (blue).

Note - We note here that the first peak fits almost perfectly to the CGM peak. The second and the third peaks being double peaks, are not expected to fit accurately.

Chapter 3

Applications of Directional Direct Search Method: Diabetic CGM fit

3.1 Introduction

If we do not eat food for some time, the glucose concentration in the blood starts decreasing. A gradual rise in the blood glucose concentration is seen as soon as we eat food. The glucose concentration in the blood reaches a peak value and it again starts declining once the insulin sets into action. The rise and fall of the blood glucose concentration, after having food, together form a 'glucose peak'. We will use the term 'fall of glucose before the intake of food' for the decrease in blood glucose concentration before having food and 'fall of glucose after the intake of food' for the decrease in blood glucose concentration after having food. The term 'rise of glucose after the intake of food' will be used to describe the increase in blood glucose concentration after having food, till the insulin action begins.

3.2 Singularly perturbed system

In a coupled differential equation system, when the solution does not change on changing the initial condition of one variable, the system is called ‘singularly perturbed’. The glucose insulin subsystem in the CGM model is a singularly perturbed system as the solution does not change on changing the initial condition of insulin. In other words, insulin relaxes to the slow manifold, given by $\frac{dI}{dt} = 0$. As a result, the information of the initial condition of insulin is lost and the only information that remains is that obtained from the surface of the manifold to which insulin relaxes. This can be numerically explained as:

$$\frac{dI}{dt} = \frac{I_{max} G^2}{\alpha + G^2} - k_I \cdot I \quad \text{Insulin dynamics from CGM model} \tag{3.1}$$

$$\frac{dI}{dt} = 0 \quad \text{gives} \tag{3.2}$$

$$I^* = \frac{I_{max} G^2}{k_I(\alpha + G^2)} \tag{3.3}$$

The following plot is obtained by superimposing the insulin dynamics from equation (3.1) with I^* from equation (3.3).

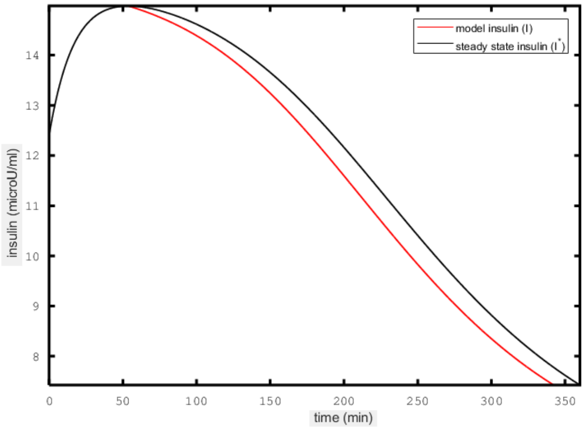

Figure 3.1: Plot of insulin (microU/ml) vs tim (min). The figure shows that irrespective of the initial condition, insulin dynamics (I) closely follows the trajectory of the steady state insulin (I^*).

It can be seen from figure (3.1) that irrespective of the initial condition, insulin follows the trajectory of the steady state insulin I^*. Therefore, we can replace the insulin from the glucose-insulin subsystem in the CGM model by I^* and reduce to a linear ordinary differential equation in one variable, G. This allows us to derive an exponential decay form of the glucose dynamics during fall before and after the intake of food. This is described in the next section.

3.3 Fall of glucose follows standard exponential decay dynamics

We observed that the fall of glucose before and after intake of food can be fit by an exponential function $ae^{-bt} + c$. The exponential decay can also be derived by solving the glucose differential equation in the CGM model analytically, assuming a constant amount of insulin I^* (from equation (3.3)) in the blood. The derivation is given below as:

$$\frac{dG}{dt} = R_0 - (E_{G0} + S_I \cdot I^*)G \tag{3.4}$$

We solve analytically the linear ordinary differential equation (3.4) in one variable G. Rearranging the terms and integrating on both sides,

$$\int \frac{1}{R_0 - (E_{G0} + S_I \cdot I^*)G} = \int dt \tag{3.5}$$

$$\frac{-\ln(R_0 - (E_{G0} + S_I \cdot I^*)G)}{E_{G0} + S_I \cdot I^*} + K = t; \quad \text{where K is a constant} \tag{3.6}$$

$$-\ln(R_0 - (E_{G0} + S_I \cdot I^*)G) + (E_{G0} + S_I \cdot I^*)K = (E_{G0} + S_I \cdot I^*)t \tag{3.7}$$

$$\ln(R_0 - (E_{G0} + S_I \cdot I^*)G) = (E_{G0} + S_I \cdot I^*)(K - t) \tag{3.8}$$

$$R_0 - (E_{G0} + S_I \cdot I^*)G = e^{(E_{G0} + S_I \cdot I^*)(K-t)} \tag{3.9}$$

$$G = \frac{R_0}{(E_{G0} + S_I \cdot I^*)} - \frac{e^{(E_{G0} + S_I \cdot I^*)(K-t)}}{(E_{G0} + S_I \cdot I^*)} \tag{3.10}$$

$$G = \frac{R_0}{(E_{G0} + S_I \cdot I^*)} - \frac{e^{(E_{G0} + S_I \cdot I^*)K} e^{-(E_{G0} + S_I \cdot I^*)t}}{(E_{G0} + S_I \cdot I^*)} \tag{3.11}$$

Rearranging the terms,

$$G = \frac{-e^{(E_{G0}+S_I \cdot I^*)K}}{(E_{G0} + SI \cdot I^*)} e^{-(E_{G0}+S_I \cdot I^*)t} + \frac{R_0}{(E_{G0} + S_I \cdot I^*)} \tag{3.12}$$

Equation (3.12) is the exponential decay form of the glucose dynamics seen during the fall before and after the intake of food. On comparing the standard exponential decay equation $ae^{-bt} + c$ to equation (3.12),

$$a = \frac{-e^{(E_{G0}+S_I \cdot I^*)K}}{(E_{G0} + S_I \cdot I^*)} \tag{3.13}$$

$$b = E_{G0} + S_I \cdot I^* \tag{3.14}$$

$$c = \frac{R_0}{(E_{G0} + S_I \cdot I^*)} \tag{3.15}$$

In equations (3.13) to (3.15), I^* is a constant. Thus, equations (3.13) to (3.15) give an algebraic relationship between the standard exponential equation parameters a, b, c and the parameters R_0, E_{G0}, S_I of the derived exponential glucose dynamics, given by equation (3.12).

3.4 Designing the procedure to fit the model glucose to CGM data

Our aim is to obtain the best values for the model parameters R_0, E_{G0}, S_I so that the model glucose best fits the glucose peak after the intake of food as well as the fall of glucose before the intake of food. Since derivative-free optimization identifies the local minimum, it is observed that the optimized values of the model parameters R_0, E_{G0}, S_I change on changing their initial guesses. Thus, the problem now reduces to finding a good approximation for the initial guesses of these parameters.

3.4.1 Fitting the standard exponential decay equation ae^{-bt+c} to the fall of glucose before and after intake of food

The optimized values of the parameters a, b, c corresponding to the fall of glucose before and after intake of food can be obtained by fitting the standard exponential decay equation $ae^{-bt} + c$ to the fall of glucose, before and after intake of food, from CGM data. Let us denote these optimized parameter values by a_{opt}, b_{opt}, c_{opt} for the fall of glucose before food.

3.4.2 Obtaining initial guesses of the model parameters R_0, E_{G0}, and S_I

After obtaining a_{opt}, b_{opt}, c_{opt}, we can now proceed to find the initial guesses of the model parameters R_0, E_{G0}, and S_I. From equation (3.15),

$$R_0^{ig} = b_{opt} \cdot c_{opt}, \tag{3.16}$$

where $R_0^{ig}$is the initial guess for R_0.
Similarly, from equation (3.14),

$$E_{G0}^{ig} + S_I^{ig} \cdot I^* = b_{opt}, \tag{3.17}$$

where E_{G0}^{ig}, S_I^{ig} are the initial guesses for E_{G0}, S_I respectively.
Equation (3.17) is a constraint equation. In other words, if we fix E_{G0}^{ig}, we can make an appropriate initial guess for S_I and if we fix S_I^{ig}, we can make an appropriate initial guess for E_{G0}. We choose to set E_{G0}^{ig} to 2.5e-03, as known from the literature for diabetic data, and find S_I^{ig} from the constraint equation (3.17). So,

$$S_I^{ig} = \left| \frac{b_{opt} - E_{G0}^{ig}}{I^*} \right|, \quad \text{since } S_I \text{ is not negative} \tag{3.18}$$

where usually, $E_{G0}^{ig} = 2.5e - 03$. We can state this as the following theorem:

Theorem 3.4.1. *We can find the initial guess values for the model parameters R_0 and S_I, of the reduced glucose dynamics equation (3.4), provided we fix an initial* *guess value for E_{G0}. Moreover, $R_0^{ig} = b_{opt} \cdot c_{opt}$ and $S_I^{ig} = \left|\frac{b_{opt} - E_{G0}^{ig}}{I^*}\right|$.*

3.4.3 Optimizing R_0, E_{G0}, and S_I to fit the model glucose to CGM data

Once we provide the initial guesses of the model parameters from equations (3.16) and (3.18) to the derivative-free optimization code in Matlab, we obtain their optimized values as the output. The optimized model parameters correspond to the best fit of the model glucose to CGM data.

We can now summarize the entire process of fitting the model glucose to CGM data, as discussed above.

1. Step 1- Fit the standard exponential decay equation $ae^{-bt} + c$ to the fall of glucose before and after intake of food and obtain the optimized values of the parameters a, b, and c, which we denote by a_{opt}, b_{opt}, c_{opt}, respectively.

2. Step 2- Obtain a good approximatiion for the initial guesses of the model parameters R_0, E_{G0}, and S_I by using equations (3.16) and (3.18).

3. Step 3- Use the initial guesses from Step 2 to obtain the optimized values of the model parameters R_0, E_{G0}, and S_I by fitting the reduced glucose dynamics $\frac{dG}{dt} = R_0 - (E_{G0} + S_I \cdot I^*)G$ to CGM data. These model parameters correspond to the best fit of the model glucose to the CGM data.

3.5 Results

We describe the results obtained after applying this procedure to a few examples:
Notation: We will use the term ‘algebraic parameters’ for the parameters a, b, c of the standard exponential decay equation $ae^{-bt} + c$ and the term ‘ode parameters’ for the model parameters R_0, E_{G0}, S_I of the reduced glucose dynamics (equation (3.4)).

3.5.1 Example 1: Demonstration of postprandial hypoglycemia

We fitted the data of fall of glucose before food and fall of glucose after food, separately, and obtained optimized values for the algebraic as well as the ode parameters. It can be seen from figures (3.11) and (3.12) that the glucose settles at a lower concentration (70.6 mg/dl) for the fall after food and at a higher concentration (114.3 mg/dl) for the fall before food. This observation is suggestive of the phenomenon of 'postprandial hypoglycemia'. In other words, hypoglycemic event is seen to occur some time (around 2 hours) after a diabetic patient has food. That is why, a diabetic patient should eat small meals at regular intervals of around 2 hours during the day.

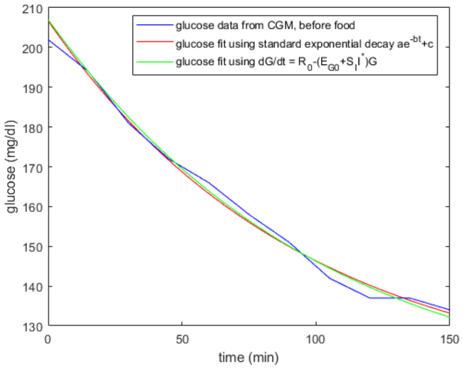

Figure 3.2: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose before food (blue) by algebraic (red) and ode (green) method is seen from this figure. Glucose achieves steady state concentration of 114.3 mg/dl.

a	b	c
92.4	0.01	114.3

Table 3.1: Optimized algebraic parameters for the fall of glucose before food

R_0	E_{G0}	S_I
0.62	0.00173	0.000456

Table 3.2: Optimized ode parameters for the fall of glucose before food

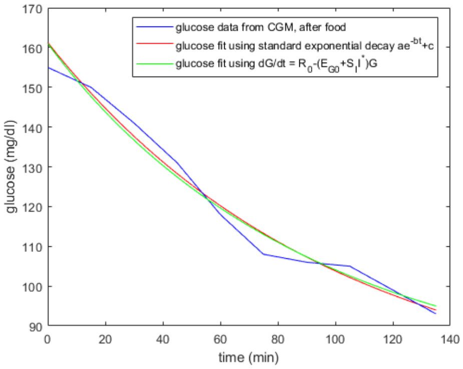

Figure 3.3: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose after food (blue) by algebraic (red) and ode (green) method is seen from this figure. Glucose achieves steady state concentration of 70.6 mg/dl.

a	b	c
90.6	0.01	70.6

Table 3.3: Optimized algebraic parameters for the fall of glucose after food

R_0	E_{G0}	S_I
0.38	0.00285	0.00045

Table 3.4: Optimized ode parameters for the fall of glucose after food

3.5.2 Example 2: Fitting diabetic CGM data

This is an example of fitting the model glucose to the experimental CGM data by following the procedure described above. Initially, the fall of glucose before food was fit using the standard exponential decay $ae^{-bt}+c$, shown in figure (3.4) and a_{opt}, b_{opt}, and c_{opt} were obtained (table (3.5)).

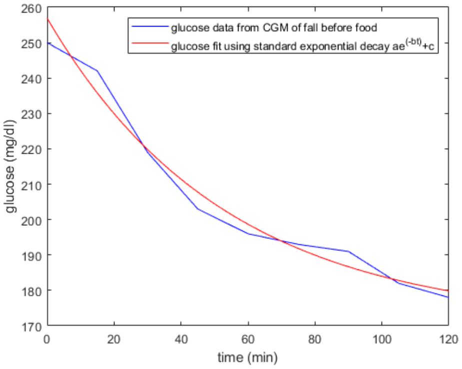

Figure 3.4: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose before food by standard exponential decay $ae^{-bt}+c$ can be seen from the figure.

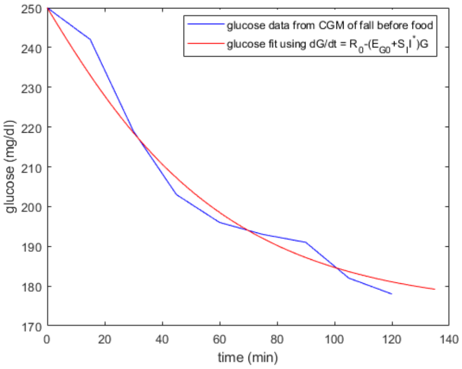

Figure 3.5: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose after food by reduced glucose dynamics $\frac{dG}{dt} = R_0 - (E_{G0} + S_I \cdot I^*)G$ given by equation (3.4) can be seen from the figure.

	a	b	c
ig	100	0.011	150
opt	86.08	0.0188	170.855

Table 3.5: Optimized algebraic parameters for the fall of glucose before food

	R_0	E_{G0}	S_I
ig	3.212	0.0025	0.001304
opt	2.22	0.0002012	0.0010735

Table 3.6: Optimized ode parameters for the fall of glucose before food

Further, the initial guesses for R_0, E_{G0}, and S_I were computed as:

$$R_0^{ig} = b_{opt} \cdot c_{opt} \qquad \text{from equation (3.16)} \tag{3.19}$$

$$= 0.0188 \times 170.855 \tag{3.20}$$

$$= 3.212 \tag{3.21}$$

$$\text{Let us fix} \quad E_{G0}^{ig} = 0.0025 \tag{3.22}$$

Therefore, from equation (3.18)

$$S_I^{ig} = \left| \frac{b_{opt} - E_{G0}^{ig}}{I^*} \right| \tag{3.23}$$

$$= \left| \frac{0.0188 - 0.0025}{12.5} \right| \tag{3.24}$$

$$= 0.001304 \tag{3.25}$$

The fit corresponding to the optimized model parameters R_0, E_{G0}, and S_I for the fall of glucose before food is shown in figure (3.5).

The optimized model parameters R_0, E_{G0}, and S_I for the postprandial glucose were adapted from the literature [5] and are mentioned in table (3.7).

	R_0	E_{G0}	S_I
opt	2.5	0.0025	0.00114

Table 3.7: Table of optimized parameters for the exponential fit for decay before meal of diabetic patient.

Thus, the complete fit corresponding to the optimized model parameters R_0, E_{G0}, and S_I from table (3.6) and (3.7) is shown in figure (3.6).

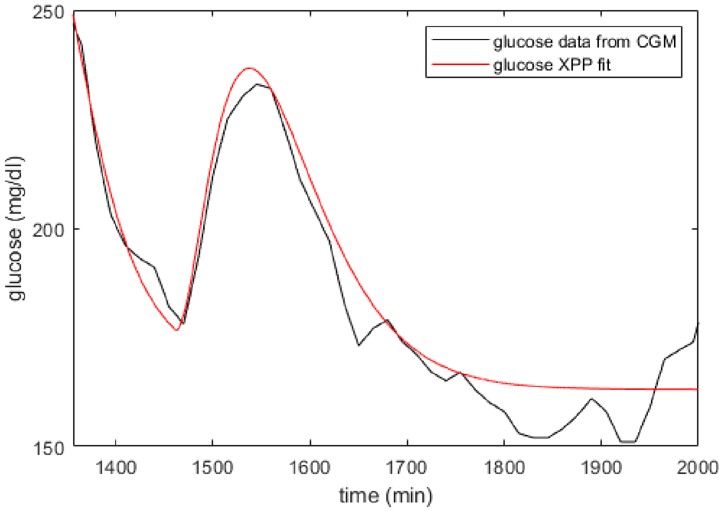

Figure 3.6: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose before food as well as the postprandial glucose peak can be seen from this figure. Only the model parameters R_0 and E_{G0} vary for the fall and peak of glucose.

By comparing the values of the optimized model parameters R_0, E_{G0}, and S_I from table (3.6) and (3.7) for fall of glucose before food and the postprandial glucose peak, respectively, we can conclude that only R_0 and E_{G0} vary for the fall and peak. All other model parameters (including those of the gut dynamics) in the CGM model remain the same. In general, we can state the following hypothesis:

3.5.3 Hypothesis

Only the parameters R_0 and E_{G0} vary for the fall and the peak of glucose in the diabetic CGM data and all other model parameters of the CGM model remain the same.

The results obtained by fitting different sets of fall and postprandial peak of glucose from diabetic CGM can be classified into the following 3 types:

Type	**Description**
Type 1 (T1)	Only the parameters R_0 and E_{G0} vary for the fall and the peak of glucose in the diabetic CGM data and all other model parameters of the CGM model remain the same.
Type 2 (T2)	Only the parameters R_0, E_{G0}, and S_I vary for the fall and the peak of glucose in the diabetic CGM data and all other model parameters of the CGM model remain the same.
Type 3 (T3)	All model parameters of the CGM model remain the same for the fall and the peak of glucose in the diabetic CGM data.

We performed exactly the same procedure described for Example 2, to fit CGM data of 3 patients (ck, hd, and ms) and classified the results as T1, T2, and T3, as described in the table above. The fits obtained, along with their classification are shown below:
(Notation: We will use the term ‘FBF’ for fall before food.)

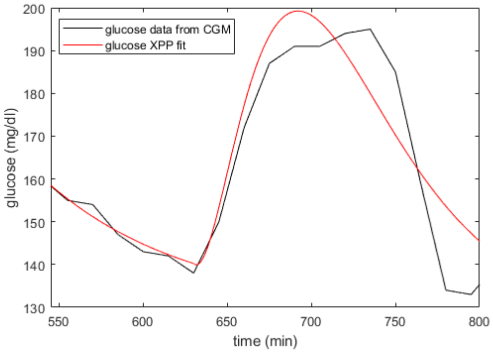

Figure 3.7: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose before food as well as the postprandial glucose peak can be seen from this figure. The fit falls in category T2.

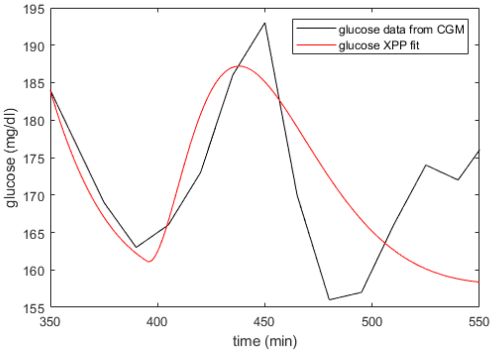

Figure 3.8: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose before food as well as the postprandial glucose peak can be seen from this figure. The fit falls in category T3.

	R_0	E_{G0}	S_I
FBF	1.155	0.0054513	0.000349
Peak	2.2929	0.001	0.0015698

	R_0	E_{G0}	S_I
FBF	6.1139	0.001	0.003812
Peak	6.1139	0.001	0.003812

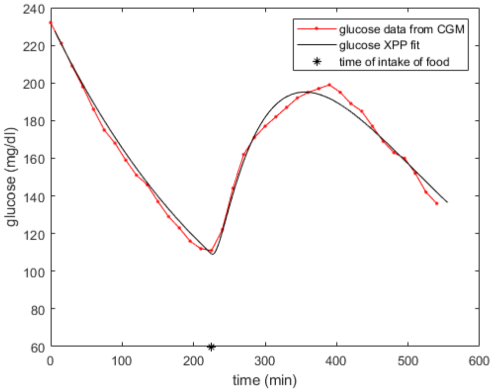

Figure 3.9: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose before food as well as the postprandial glucose peak can be seen from this figure. The fit falls in category T3.

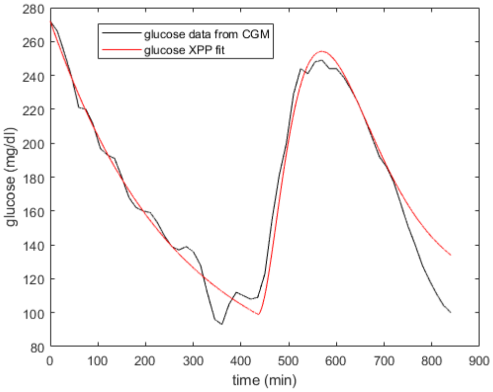

Figure 3.10: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose before food as well as the postprandial glucose peak can be seen from this figure. The fit falls in category T2.

	R_0	E_{G0}	S_I
FBF	0.001	0.001	0.0000705
Peak	0.001	0.001	0.0000705

	R_0	E_{G0}	S_I
FBF	0.1644	0.0027301	0.0000622
Peak	0.7722	0.001	0.0006701

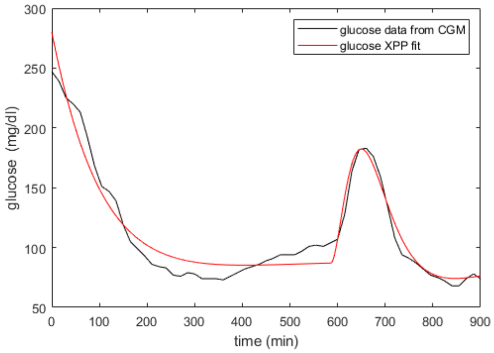

Figure 3.11: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose before food as well as the postprandial glucose peak can be seen from this figure. The fit falls in category T3.

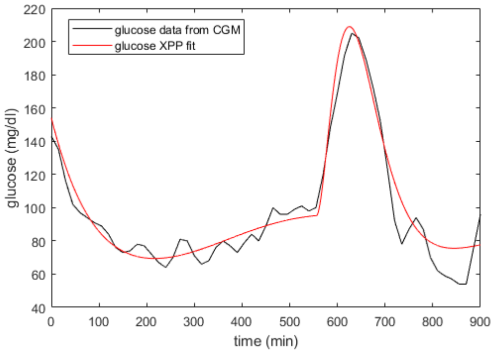

Figure 3.12: Plot of glucose (mg/dl) vs time (min). The fit of fall of glucose before food as well as the postprandial glucose peak can be seen from this figure. The fit falls in category T3.

	R_0	E_{G0}	S_I
FBF	0.628	0.004161	0.0004199
Peak	1.6745	0.001	0.0029199

	R_0	E_{G0}	S_I
FBF	0.14405	-0.00768	0.0012785
Peak	1.5121	0.001	0.0024601

3.6 Conclusion

In this chapter, we described a formal procedure, that we designed, to fit diabetic CGM data. It enabled us to fit both the fall of glucose before food as well as the postprandial glucose peak. We hypothesize that only the parameters R_0 and E_{G0} vary for fitting the fall and the peak, however this needs more elaborate mathematical explanation. In the end, we demonstrated various fits using the designed procedure and classified them into 3 types, depending on the parameters that varied for the fall and the peak of glucose.

Chapter 4

Modelling for Gastroparesis

4.1 Introduction

Gastroparesis is a disorder which is defined as a delay in gastric emptying without any mechanical obstruction in the stomach [6]. It affects people diagnosed with both type 1 and type 2 diabetes. This is because high levels of blood glucose can cause chemical changes that disturb the functioning of blood vessels that carry oxygen and nutrients to the vagus nerve, which is responsible for controlling the movement of food through the digestive tract [7]. Hyperglycemia is known to slow down gastric emptying. There is a strong correlation between gastric emptying, incretin hormones and postprandial glycemia [8]. Therefore, in order to study this relationship in more detail, it is imperative to model the postprandial glucose and insulin data from a patient diagnosed with gastroparesis. We obtained glucose and insulin data (shown in figure (4.1)) from Mixed Meal Tolerance Test (MMTT) of a person diagnosed with gastroparesis, from our collaborator at Apollo Hospital, Chennai. The data comprises of 14 glucose and insulin measurements each, taken at the same time, spread over a period of around 3 hours. The aim of this study is to model the postprandial glucose and insulin data of this patient.

Initially, we expected the CGM model that we developed in Goel et al., 2018 to fit the glucose and insulin data well. However, the observed data was not described

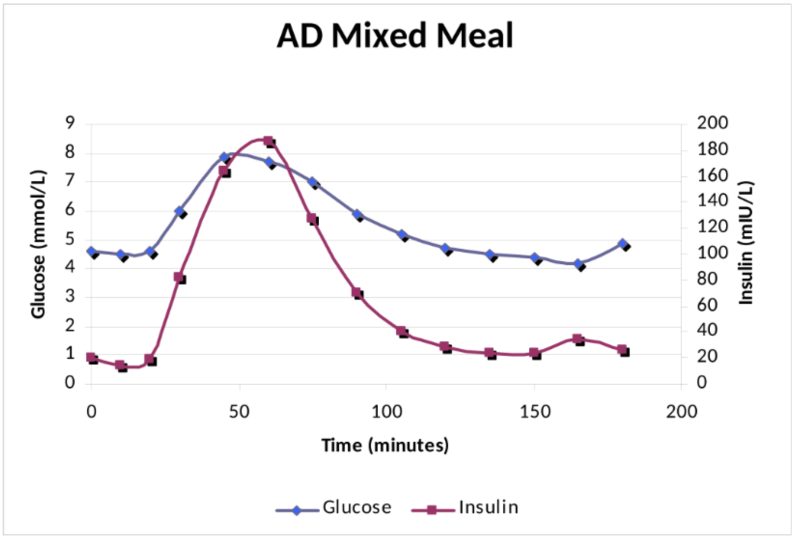

Figure 4.1: Glucose and insulin data of a patient suffering from gastroparesis, obtained from Mixed Meal Tolerance Test (MMTT). The data comprises of 14 glucose and insulin measurements each, taken at the same time, spread over a period of around 3 hours.
Courtesy: Dr.Anil Vaidya, Apollo hospital, Chennai.

satisfactorily, since the CGM model contained only 2 stomach compartments in the gut dynamics. Delayed gastric emptying is an important characteristic of gastroparesis, which was not getting accounted for in the CGM model from Goel et al., 2018. We decided to incorporate the phenomenon of delayed gastric emptying in two ways as follows:

1. Increase the number of stomach compartments in the gut dynamics
2. Use the trapezoidal gastric emptying function, given by the Lehmann-Deutsch model [9, 10], in the gut dynamics

The number of stomach compartments determines the slope of the rise and fall of the glucose and insulin in the model (Figure (4.3)). In other words, on changing the number of stomach compartments taking part in the gut dynamics, the slope of rise and fall of model glucose and insulin can be controlled.

We observed a linear relationship between measured glucose and insulin data and modified the insulin dynamics (4.3)) accordingly.

The modified CGM model, in which a variable number of stomach compartments were used to model the gut dynamics, is given by equations (4.1) to (4.6)). Fits corresponding to 3, 4, and 5 stomach compartments were obtained as shown in figures (4.4), (4.5), and (4.6), respectively. From the results obtained, the following hypothesis can be stated:

Hypothesis: Smaller number of stomach compartments are required to model the rise of glucose and insulin peak and larger number of stomach compartments are required to model the fall of glucose and insulin peak. However, if we were to model the rise and fall of the glucose and insulin peak using the same number of stomach compartments, $n = 4$ would be the best choice.

The model using the trapezoidal gastric function given by the Lehmann-Deutsch model is given by equations (4.32) to (4.38). The resultant fit obtained, shown in figure (4.8)), corresponding to the optimized model parameters obtained from derivative free optimization, was found to be better than the fit obtained from the modified CGM model for $n = 4$, shown in figure (4.5)).

The modified CGM model and the gut dynamics from the Lehmann-Deutsch model, that we used to fit the postprandial glucose and insulin data simultaneously, are decribed in detail in the sections ahead and the results are discussed.

4.2 Modified CGM model for Gastoparesis

4.2.1 Modifications in the CGM model

The glucose and insulin data of a gastroparesis patient was analyzed. The following two observations were noted:

1. We saw a linear relationship between insulin and glucose data as shown in figure (4.2).

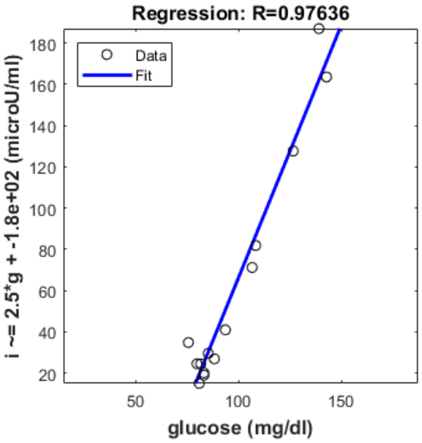

Figure 4.2: Plot of insulin data (microU/ml) vs glucose data (mg/dl). A linear relationship was observed between insulin and glucose data obtained from the Mixed Meal Tolerance Test (MMTT).

2. A better control on the slope of the rise and fall of the glucose peak can be obtained by varying the number of compartments in the stomach. It can be observed from Figure (4.3) that the slope of the rise of glucose peak decreases on increasing the number of stomach compartments. The magnitude of the slope of fall of glucose peak decreases on increasing the number of stomach compartments.

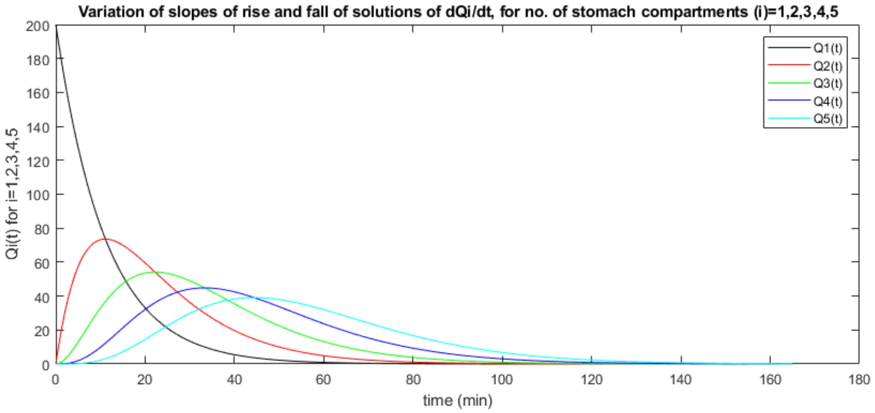

Figure 4.3: Plot of the glucose dynamics seen in different stomach compartments, as a function of time. Different curves represent glucose dynamics in different number of stomach compartments, as given in the legend. The number of stomach compartments determines the slope of the rise and fall of the glucose in the model. The slope of rise and magnitude of slope of fall of glucose peak decreases on increasing the number of stomach compartments in the gut dynamics.

We incorporated both these observations and modified our CGM model. That is, instead of 2 stomach compartments, n stomach compartments were included in the gut dynamics. Also, the sigmoidal function of glucose in the insulin differential equation was replaced by a linear function.
Thus, the modified CGM model can be given as:
Glucose-insulin dynamics is as follows:

$$\frac{dG}{dt} = R_0 - (E_{G0} + S_I I)G + kQ_n \tag{4.1}$$

$$\frac{dI}{dt} = i_{int} + i_{slpe}G - k_I I \tag{4.2}$$

$$\tag{4.3}$$

Gut dynamics:

$$\frac{dQ_1}{dt} = -kQ_1 \tag{4.4}$$

$$\frac{dQ_2}{dt} = kQ_1 - kQ_2 \tag{4.5}$$

For the n^{th} stomach compartment,

$$\frac{dQ_n}{dt} = kQ_{n-1} - kQ_n, \tag{4.6}$$

where Q_i is the i^{th} stomach compartment and n is the optimum number of stomach compartments which allow the best fit of the model glucose and insulin to the observed glucose and insulin data respectively. In the glucose differential equation, the term R_0 describes the constant rate of increase of glucose in the blood, $-E_{G0} \cdot G$ describes the rate of insulin-independent glucose disposal from the blood, $-S_I \cdot I \cdot G$ describes the rate of insulin-dependent glucose disposal from the blood, and the term $k_{gut} \cdot Q_n$ describes the rate of increase of glucose in the blood after intake of food. In the insulin differential equation, i_{int} and i_{slpe} are the parameters corresponding to the intercept and slope of the linear dependence of insulin on glucose, as observed from data and k_I is the parameter for the clearance of insulin from the blood. k is a rate constant of the gut dynamics.

4.2.2 Stomach compartments

In this section, we will describe an explicit mathematical derivation to compute the solution of the dynamics in each of the different number of stomach compartments.

Notation:
We will use the subscript 'i' for the i^{th} stomach compartment. Thus, the matrix A_5 (Equation (4.8)) represents the square matrix ($n \times n$) of type A with diagonal elements $-k$ and sub-diagonal elements k, for $n = 5$ stomach compartments. Similarly, we define the $n \times n$ square matrices B_n and I_n.

The system of differential equations representing the compartments in the stomach (for $n = 5$) can be written in matrix form as follows:

$$\begin{bmatrix} \frac{dQ_1}{dt} \\ \frac{dQ_2}{dt} \\ \frac{dQ_3}{dt} \\ \frac{dQ_4}{dt} \\ \frac{dQ_5}{dt} \end{bmatrix} = \begin{bmatrix} -k & 0 & 0 & 0 & 0 \\ k & -k & 0 & 0 & 0 \\ 0 & k & -k & 0 & 0 \\ 0 & 0 & k & -k & 0 \\ 0 & 0 & 0 & k & -k \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \\ Q_5 \end{bmatrix} \tag{4.7}$$

$$\text{Let} \qquad A_5 = \begin{bmatrix} -k & 0 & 0 & 0 & 0 \\ k & -k & 0 & 0 & 0 \\ 0 & k & -k & 0 & 0 \\ 0 & 0 & k & -k & 0 \\ 0 & 0 & 0 & k & -k \end{bmatrix} \tag{4.8}$$

Equation (4.7) can be written as

$$\frac{dQ}{dt} = A_5 Q \tag{4.9}$$

The solution to equation (4.9) is

$$Q(t) = e^{tA_5} Q(0) \tag{4.10}$$

where e^{tA_5} is computed by taking the exponential of the matrix tA_5. Q(t) is the solution matrix representing the dynamics observed in each of the stomach compartments.

Computing exponential of the matrix tA_n

$$A_5 = k\begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix} + k\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.11}$$

$$A_5 = k(-I_5 + B_5) \tag{4.12}$$

where

$$B_5 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \qquad \text{and} \quad B_5^4 \neq 0; \quad B_5^5 = 0 \tag{4.13}$$

In general,

$$\frac{dQ}{dt} = A_n Q \tag{4.14}$$

$$Q(t) = e^{tA_n} Q(0) \tag{4.15}$$

$$Q(t) = e^{tk(-I_n+B_n)} Q(0) \qquad \text{where} \quad B_n^{n-1} \neq 0; \quad B_n^n = 0 \tag{4.16}$$

For $n = 5$ in equation (4.16)

$$Q(t) = e^{tk(-I_5+B_5)} Q(0) \qquad \text{where} \quad B_5^4 \neq 0; \quad B_5^5 = 0 \tag{4.17}$$

$$e^{-tkI_5} = \begin{bmatrix} e^{-tk} & 0 & 0 & 0 & 0 \\ 0 & e^{-tk} & 0 & 0 & 0 \\ 0 & 0 & e^{-tk} & 0 & 0 \\ 0 & 0 & 0 & e^{-tk} & 0 \\ 0 & 0 & 0 & 0 & e^{-tk} \end{bmatrix} \tag{4.18}$$

$$e^{tkB_5} = \sum_{n=0}^{\infty} \frac{t^n k^n B_5^n}{n!} \tag{4.19}$$

$$= I + tkB_5 + \frac{t^2k^2B_5^2}{2!} + \frac{t^3k^3B_5^3}{3!} + \frac{t^4k^4B_5^4}{4!} \quad (\text{since} \quad B_5^5 = 0) \tag{4.20}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ tk & 0 & 0 & 0 & 0 \\ 0 & tk & 0 & 0 & 0 \\ 0 & 0 & tk & 0 & 0 \\ 0 & 0 & 0 & tk & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{t^2k^2}{2!} & 0 & 0 & 0 & 0 \\ 0 & \frac{t^2k^2}{2!} & 0 & 0 & 0 \\ 0 & 0 & \frac{t^2k^2}{2!} & 0 & 0 \end{bmatrix} \tag{4.21}$$

$$+ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{t^3k^3}{3!} & 0 & 0 & 0 & 0 \\ 0 & \frac{t^3k^3}{3!} & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{t^4k^4}{4!} & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.22}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ tk & 1 & 0 & 0 & 0 \\ \frac{t^2k^2}{2!} & tk & 1 & 0 & 0 \\ \frac{t^3k^3}{3!} & \frac{t^2k^2}{2!} & tk & 1 & 0 \\ \frac{t^4k^4}{4!} & \frac{t^3k^3}{3!} & \frac{t^2k^2}{2!} & tk & 1 \end{bmatrix} \tag{4.23}$$

Substituting equation (4.18) and (4.23) in equation (4.17), we get

$$Q(t) = \begin{bmatrix} e^{-tk} & 0 & 0 & 0 & 0 \\ 0 & e^{-tk} & 0 & 0 & 0 \\ 0 & 0 & e^{-tk} & 0 & 0 \\ 0 & 0 & 0 & e^{-tk} & 0 \\ 0 & 0 & 0 & 0 & e^{-tk} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ tk & 1 & 0 & 0 & 0 \\ \frac{t^2k^2}{2!} & tk & 1 & 0 & 0 \\ \frac{t^3k^3}{3!} & \frac{t^2k^2}{2!} & tk & 1 & 0 \\ \frac{t^4k^4}{4!} & \frac{t^3k^3}{3!} & \frac{t^2k^2}{2!} & tk & 1 \end{bmatrix} \begin{bmatrix} Q_{10} \\ Q_{20} \\ Q_{30} \\ Q_{40} \\ Q_{50} \end{bmatrix} \tag{4.24}$$

$$Q(t) = \begin{bmatrix} e^{-tk} & 0 & 0 & 0 & 0 \\ tke^{-tk} & e^{-tk} & 0 & 0 & 0 \\ \frac{t^2k^2e^{-tk}}{2!} & tke^{-tk} & e^{-tk} & 0 & 0 \\ \frac{t^3k^3e^{-tk}}{3!} & \frac{t^2k^2e^{-tk}}{2!} & tke^{-tk} & e^{-tk} & 0 \\ \frac{t^4k^4e^{-tk}}{4!} & \frac{t^3k^3e^{-tk}}{3!} & \frac{t^2k^2e^{-tk}}{2!} & tke^{-tk} & e^{-tk} \end{bmatrix} \begin{bmatrix} Q_{10} \\ Q_{20} \\ Q_{30} \\ Q_{40} \\ Q_{50} \end{bmatrix} \tag{4.25}$$

Equation (4.25) gives the algebraic solution of the dynamics in the different number of stomach compartments (1 to 5).

4.2.3 Adding a unit food impulse

A unit impulse function is:

$$u(t) = \begin{cases} 1 & \text{if } t = 0 \\ 0 & \text{otherwise} \end{cases}$$

In our model for gastroparesis, we assume that food is of the form of a unit impulse and it enters the first compartment of the stomach.

$$\frac{dQ_1}{dt} = -kQ_1 + u(t) \tag{4.26}$$

For $t > 0$, $u(t) = 0$, and, therefore, for $t > 0$, we should solve

$$\frac{dQ_1}{dt} = -kQ_1; \quad Q_1(0) = 1 \tag{4.27}$$

For the i^{th} compartment, 2≤ i ≤ n,

$$\frac{dQ_i}{dt} = kQ_{i-1} - kQ_i; \quad Q_i(0) = 0 \tag{4.28}$$

4.2.4 Modified CGM Model for Gastroparesis

We can now state the full modified CGM model for gastroparesis as:

$$\frac{dG}{dt} = R_0 - (E_{G0} + S_I I)G + kQ_n \tag{4.29}$$

$$\frac{dI}{dt} = i_{int} + i_{slpe}.G - k_I.I \tag{4.30}$$

$$Q(t) = \begin{bmatrix} e^{-tk} & 0 & 0 & 0 & 0 \\ tke^{-tk} & e^{-tk} & 0 & 0 & 0 \\ \frac{t^2k^2e^{-tk}}{2!} & tke^{-tk} & e^{-tk} & 0 & 0 \\ ..... & ..... & ..... & ..... & ..... \\ \frac{t^{n-1}k^{n-1}e^{-tk}}{(n-1)!} & \frac{t^{n-2}k^{n-2}e^{-tk}}{(n-2)!} & & tke^{-tk} & e^{-tk} \end{bmatrix}_{n\times n} \begin{bmatrix} 1 \\ 0 \\ 0 \\ ..... \\ 0 \end{bmatrix}_{n\times 1} \tag{4.31}$$

4.2.5 Results for different number of stomach compartments

The fits obtained by using equations (4.29) to (4.31) to model the postprandial glucose and insulin data simultaneously, for $n = 3$, 4, and 5 stomach compartments, are shown below in figures (4.4), (4.5), and (4.6) respectively.

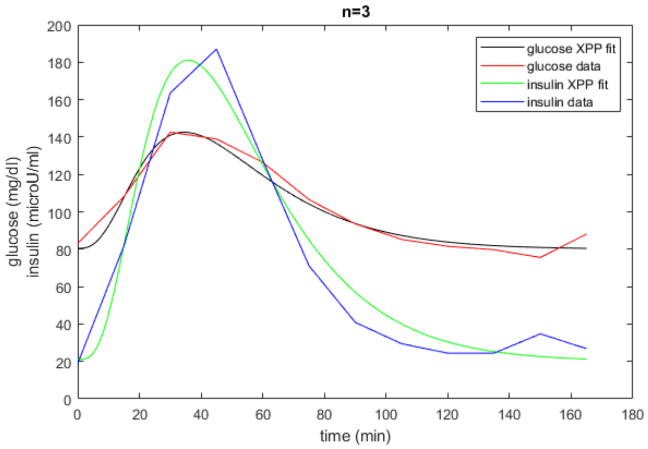

Figure 4.4: Plot of glucose (mg/dl) and insulin (microU/ml) as a function of time. The figure shows the best hand fit of model glucose and insulin to observed glucose and insulin data, obtained in XPP for $n = 3$ stomach compartments. The rise of both glucose and insulin is seen to fit well.

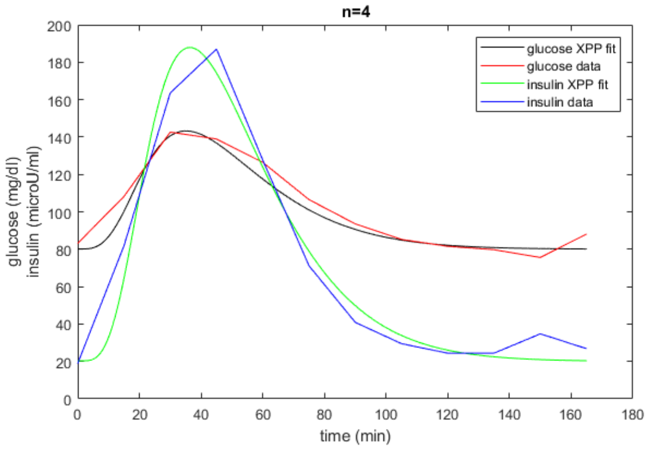

Figure 4.5: Plot of glucose (mg/dl) and insulin (microU/ml) as a function of time. The figure shows the best hand fit of model glucose and insulin to observed glucose and insulin data, obtained in XPP for $n = 4$ stomach compartments. The rise as well as fall of both glucose and insulin are seen to fit well.

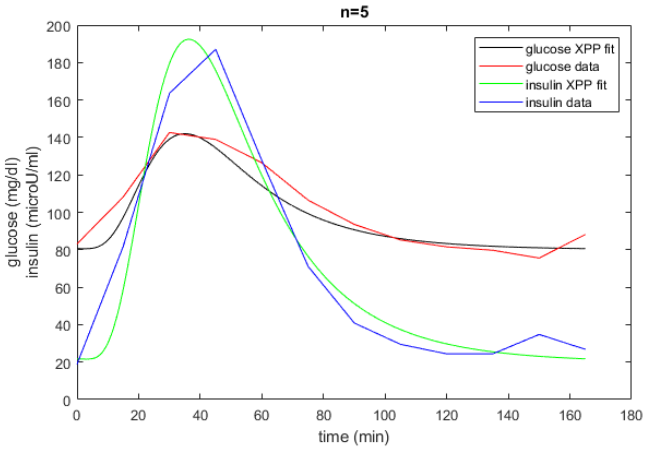

Figure 4.6: Plot of glucose (mg/dl) and insulin (microU/ml) as a function of time. The figure shows the best hand fit of model glucose and insulin to observed glucose and insulin data, obtained in XPP for $n = 5$ stomach compartments. The fall of both glucose and insulin is seen to fit well.

We can observe from the above figures that the rise of glucose and insulin peak fits better when $n = 3$ (Figure (4.4)) and fall of glucose and insulin peak fits better when $n = 5$ (Figure (4.6)). Thus, we can hypothesize that smaller number of stomach compartments can be used to model the rise of glucose and insulin peak and larger number of stomach compartments can be used to model the fall of glucose and insulin peak. However, if we were to model both rise and fall of glucose and insulin using the same number of stomach compartments, $n = 4$ would be the best choice (Figure (4.5)).

4.3 Lehmann-Deutsch model for Gastroparesis

In this section, we will use the trapezoidal gastric emptying function given by Lehmann and Deutsch to model the action of the gut on the food [9, 10].

Lehmann-Deutsch model for gut dynamics is as follows:

$$T_{max} = \frac{D - \frac{1}{2}V_{max}(T_{up} + T_{down})}{V_{max}} \tag{4.32}$$

$$G_{empt}(t) = \begin{cases} \frac{V_{max}}{T_{up}} \cdot t & \text{if } t < T_{up} \\ V_{max} & T_{up} < t \leq (T_{up} + T_{max}) \\ V_{max} - (\frac{V_{max}}{T_{down}}) \cdot (t - T_{up} - T_{max}) & (T_{up} + T_{max}) \leq t \leq (T_{max} + T_{up} + T_{down}) \\ 0 & \text{elsewhere} \end{cases}$$

$$\frac{dq_{gut}}{dt} = -k_{abs} q_{gut} + G_{empt} \tag{4.33}$$

$$R_a = f \cdot k_{abs} q_{gut}, \tag{4.34}$$

where V_{max} is the maximum velocity of gastric emptying, D is the ingested glucose dose, T_{up}, T_{max} and T_{down} are the duration of rising up, staying and dropping periods of the gastric emptying function respectively. q_{gut} is the amount of glucose in the gut, k_{abs} is the rate constant of intestinal absorption and f is the fraction of the intestinal absorption which actually appears in plasma.

The glucose-insulin dynamics equations are adopted from the modified CGM model, incorporating the linear dependence of insulin on glucose.
Glucose-insulin dynamics is as follows:

$$\frac{dG}{dt} = R_0 - (E_{G0} + S_I I)G + R_a \tag{4.35}$$

$$\frac{dI}{dt} = i_{int} + i_{slpe} \cdot G - k_I \cdot I, \tag{4.36}$$

where

$$R_0 = (E_{G0} + S_I \cdot 20) \cdot 80 \cdot 1e - 03 \tag{4.37}$$

$$k_I = \frac{(i_{int} + 80 \cdot i_{slpe})}{20} \tag{4.38}$$

in order to maintain the basal values for glucose and insulin at 80 $micro \cdot U/ml$ and 20 mg/dl respectively.

The hand fit in XPP (Figure (4.7)) and the optimized fit in Matlab (Figure (4.8)) are shown below. The corresponding optimized model parameters are as mentioned in Table (4.1).

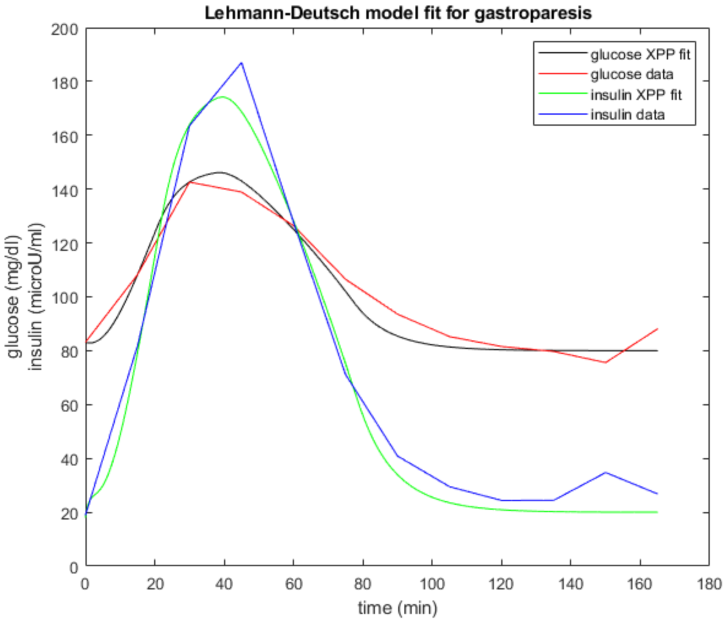

Figure 4.7: Plot of glucose (mg/dl) and insulin (microU/ml) as a function of time. XPP fit of the Lehmann-Deutsch model for gastroparesis can be seen.

Table 4.1: Parameters corresponding to the XPP hand fit and Matlab optimized fit, simultaneously for both glucose and insulin data obtained from MMTT of a patient diagnosed with gastoparesis.

Parameter	XPP hand-fit value	Matlab optimized value
T_{up}	20	28.16482
T_{down}	40	51.29878
V_{max}	9	5.024039
k_{abs}	1	144.0635
f	0.9	1.873592
D	420	405.4846
$E_{G0} \cdot 10^{-3}$	40	-
$S_I \cdot 10^{-3}$	0.22	-
i_{int}	-200	-
i_{slpe}	2.8	-
τ_i	0.86	-

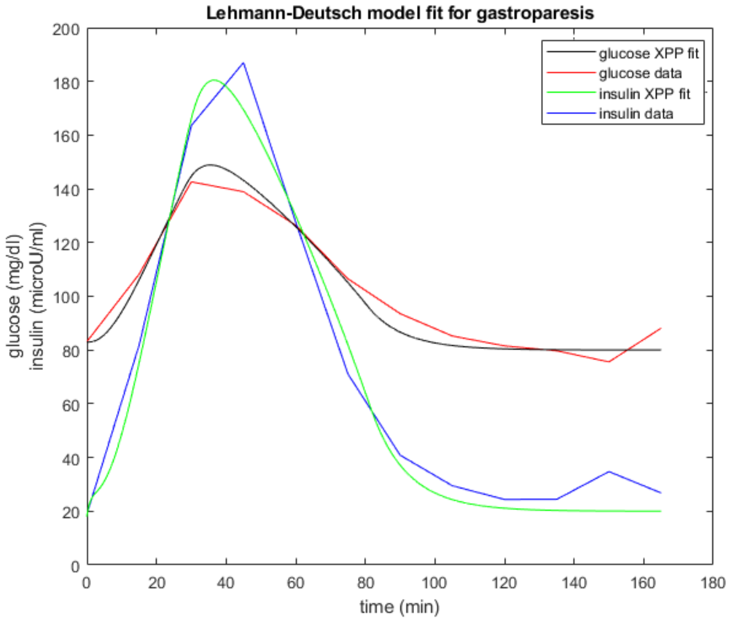

Figure 4.8: Plot of glucose (mg/dl) and insulin (microU/ml) as a function of time. Matlab fit of the Lehmann-Deutsch model for gastroparesis can be seen.

4.4 Conclusion

The phenomenon of delayed gastric emptying in gastroparesis was captured in two ways. We used two models, Model 1 (Figures 4.4 to 4.6) and Model 2 (Figures 4.7 and 4.8) to model the AD mixed meal glucose as well as insulin data. We found that Model 2 fit the data slightly better than Model 1, which suggests that trapezoidal gastric emptying function is slightly better at explaining the delayed gastric emptying than increasing the number of stomach compartments in the gut dynamics. In this way, both the models captured the presence of gastroparesis in the patient. Further, the linear relationship between AD mixed meal insulin and glucose data can be used to monitor the change in insulin sensitivity of the patient with time. Greater slope implies higher insulin sensitivity. In other words, if the slope of the regression line through insulin and glucose data of the patient decreases over time, we may conclude that it is a sign of lowering insulin sensitivity of the patient.

Chapter 5

Conclusion

Derivative-free optimization (DFO) is an emerging field used to optimize functions when the analytical information of the function is not known. The objective function to be optimized can be smooth or non-smooth, and DFO can be applied to constrained as well as unconstrained problems. In this study, we applied DFO to unconstrained problems and almost all the objective functions were non-smooth. In particular, we studied the directional direct search method algorithm in detail and saw its several applications.

After studying the directional direct search method algorithm, we developed our own codes in Matlab (provided in Appendix 6.1) and tested them on a variety of test suite problems. The results of the optimization matched exactly to the known global minima, for almost all the functions. In cases of multiple minima, the algorithm converged to each of them, on giving appropriate initial guesses. However, since the algorithm yields the local minimum, it was observed that different optimization outputs were obtained on changing the initial guesses of the parameters. There was only one function for which the optimization did not yield results equal to the known minimum, inspite of starting from initial guesses within a small neighborhood of the known minimum. We suspect that the algorithm might be getting stuck in the numerous local minima of the function, and hence fails to reach the known global minimum. All in all, the optimization codes we developed were successfully validated from the results obtained on testing the codes on the test suite problems.

As a next step, we successfully implemented derivative free optimization to recover the parameters from the CGM model for non-diabetic CGM data. However, to model diabetic CGM data, using derivative-free optimization, we designed a systematic procedure based on the finding that insulin dynamics is independent of its initial condition (in other words, the system is singularly perturbed). This procedure enabled us to model both, the fall of glucose before intake of food as well as the glucose post-prandial glucose peak, for a variety of CGM data sets.

Lastly, we used derivative-free optimization to model both glucose and insulin data simultaneously, obtained from mixed meal tolerance test of a patient diagnosed with gastroparesis. For this purpose, we developed two different gut dynamics models to capture the ‘delayed gastric emptying’ phenomenon of gastroparesis. Comparing the results obtained using both these models, we concluded that the trapezoidal gastric emptying function in the gut dynamics yields better fit to the data than using variable number of stomach compartments.

In our view, we have shown a proof of concept that derivative-free optimization can be successfully implemented to model CGM data in type 2 diabetes. Despite being an emerging field, it has the potential to optimize a variety of problems, as we have demonstrated for a bunch of unconstrained, non-smooth functions. We hope that this work will prove to be helpful for a further advanced study in this respect.

Chapter 6

Appendix

6.1 Matlab code for DFO of algebraic equation: Ackley function

```
a = linspace(-10,10,1e+03);
b = linspace(-10,10,1e+03);
[A,B] = meshgrid(a,b);
fun = -20.*exp(-0.2.*(0.5.*(A.^2+B.^2)).^(1/2))-exp(0.5.*(cos(2.*pi.*A)+cos(2.*pi.*B)))+2.71828+20;
mesh(A,B,fun)
hold on;

f = @(x,y) -20.*exp(-0.2.*(0.5.*(x.^2+y.^2)).^(1/2))-exp(0.5.*(cos(2.*pi.*x)+cos(2.*pi.*y)))+2.71828+20;

par_no = 2;

dp = eye(par_no);
dn = -eye(par_no);

D = [dp dn];
```

```

n = 100000;

p = zeros(par_no,2.*par_no);
q = zeros(1,2.*par_no);

x = zeros(par_no,n);
alpha = zeros(1,n);

par_opt = zeros(par_no,1);

x(:,1) = [10 10];
alpha(1) = 1;

for j = 1:n
    fprintf('Iteration no = ');
    fprintf('%d\n',j)

    index = 0;
    found = 0;

    s = f(x(1,j),x(2,j));

    for i = 1:2*par_no
        p(:,i) = x(:,j) + alpha(j).*D(:,i);
        q(1,i) = f(p(1,i),p(2,i));

        if q(1,i) < s
            s = q(1,i);
            index = i;
            fprintf('index = ');
            fprintf('%d\n',index);
            found = 1;
        end
    end

    if found == 1
        x(:,j+1) = x(:,j) + alpha(j).*D(:,index);
        fprintf('x = ');
        fprintf('%d\n',x(:,j+1));
        alpha(j+1) = (2/1).*alpha(j);
        fprintf('alpha = ');
        fprintf('%d\n',alpha(j+1));
        fprintf('Iteration successful\n');
    else
        x(:,j+1) = x(:,j);
        fprintf('x = ');
        fprintf('%d\n',x(:,j+1));
        alpha(j+1) = (1/2).*alpha(j);
        fprintf('alpha = ');
        fprintf('%d\n',alpha(j+1));
        fprintf('Iteration unsuccessful\n')
    end

    if alpha(j+1) < 1e-05
        for i = 1:par_no
            par_opt(i) = x(i,j+1);
        end
        fprintf('par_opt(1) = ');
        fprintf('%d\n',par_opt(1));
        fprintf('par_opt(2) = ');
        fprintf('%d\n',par_opt(2));
        break
    end
end

xval = x(1,:);
yval = x(2,:);
```

```
plot3(xval(1:100),yval(1:100),f(xval(1:100),yval(1:100)),'r*-', 'LineWidth', 2, 'MarkerSize', 15)
```

6.2 Matlab code for DFO of gastroparesis data fit using Lehmann-Deutsch gut model

6.2.1 Function for the odes

```
function dy = cgmodes(t,y,par1,par2,par3,par4,par5,par6)

dy = zeros(3,1);

EG0=40;
SI=0.22;
iint=-200;
islpe=2.8;
toui=0.86;

Tup=par1;
Tdown=par2;
Vmax=par3;
kabs=par4;
f=par5;
D=par6;

R0=(EG0+SI*20)*80*1e-03;
kI=(iint+80*islpe)/20;

Tmax = ((D-(1/2)*Vmax*2*(Tup+Tdown))/Vmax);

Gempt = (Vmax/Tup)*t*heaviside(Tup-t)+...
      Vmax*heaviside(t-Tup)*heaviside(Tup+Tmax-t)+...
```

```
        (Vmax-(Vmax/Tdown)*(t-Tup-Tmax))*heaviside(t-(Tup+Tmax))
            *...
        heaviside(Tup+Tmax+Tdown-t);

%yp(3)=qgut
dy(3) = -kabs*y(3)+Gempt;

Ra=f*kabs*y(3);

%yp(1)=G
dy(1) = R0-(EG0*1e-03+SI*1e-03.*y(2)).*y(1)+Ra;

%yp(2)=I
dy(2) = toui*(iint+islpe*y(1)-kI*y(2));
```

6.2.2 Function for the optimization

```
function [T,Y] = fitcgm

t0 = 0;
tfinal = 165;
y0=[83,18,0]';

fdtimes = 1e-06;
fdtimes = [t0 fdtimes tfinal];

T=t0;
Y=y0';

gdata = dlmread('g.dat');
gdata = gdata(1:end);
tdata = 0:15:15*length(gdata)-1;
time = linspace(1e-06,165,2001);
vq1 = interp1(tdata,gdata,time);
```

```
plot(time,vq1,'r')
hold on;

idata = dlmread('i.dat');
idata = idata(1:end);
vq2 = interp1(tdata,idata,time);
plot(time,vq2,'b')

par_no = 6;

dp = eye(par_no);
dn = -eye(par_no);

D = [dp dn];

n = 100000;

p = zeros(par_no,2.*par_no);
q = zeros(1,2.*par_no);

x = zeros(par_no,n);
alpha = zeros(1,n);

par = zeros(par_no,n);
par_opt = zeros(par_no,1);
f = zeros(1,n);

g1 = zeros(length(fdtimes)-1,2001);
g2 = zeros(length(fdtimes)-1,2001);
i1 = zeros(length(fdtimes)-1,2001);
i2 = zeros(length(fdtimes)-1,2001);

t = zeros(2001,2);
% t(:,1) = linspace(0,1200,2001);
```

```
t(:,2) = linspace(1e-06,165,2001);

% gdata1 = interp1(tdata,gdata,t(:,1));
gdata2 = interp1(tdata,gdata,t(:,2));
idata2 = interp1(tdata,idata,t(:,2));

x(:,1) = [20 40 9 1 0.9 400];

alpha(1) = 1;

for j = 1:n
    fprintf('Iteration no = ');
    fprintf('%d\n',j)

    index = 0;
    found = 0;

    for i = 1:par_no
        par(i,j) = x(i,j);
    end

    for k = 2:length(fdtimes)
        sol1 = ode45(@(t,y) cgmodes(t,y,par(1,j),par(2,j),par(3,j),par(4,j),par(5,j),par(6,j)) ,...
            [fdtimes(k-1) fdtimes(k)], [Y(end,1:3)]);
        func1 = deval(sol1,t(:,k-1));
        g1(k-1,:) = func1(1,:);
        i1(k-1,:) = func1(2,:);
    end

    f(j) = sum((gdata2'-g1(2,:)).^2+(idata2'-i1(2,:)).^2);

    for i = 1:2*par_no

```

```
        p(:,i) = x(:,j) + alpha(j).*D(:,i);

        for k = 2:length(fdtimes)
            sol2 = ode45(@(t,y) cgmodes(t,y,p(1,i),p(2,i),p(3,i),p(4,i),p(5,i),p(6,i)), ...
                [fdtimes(k-1) fdtimes(k)], [Y(end,1:3)]);
            func2 = deval(sol2,t(:,k-1));
            g2(k-1,:) = func2(1,:);
            i2(k-1,:) = func2(2,:);
        end

        q(i) = sum((gdata2'-g2(2,:)).^2+(idata2'-i2(2,:)).^2);

        if q(i) < f(j)
            f(j) = q(i);
            index = i;
            fprintf('index = ');
            fprintf('%d\n',index);
            found = 1;
        end
    end

    if found == 1
        x(:,j+1) = x(:,j) + alpha(j).*D(:,index);
        fprintf('x = ');
        fprintf('%d\n',x(:,j+1));
        alpha(j+1) = (1/1.005).*alpha(j);
        fprintf('alpha = ');
        fprintf('%d\n',alpha(j+1));
        fprintf('Iteration successful\n');
    else
        x(:,j+1) = x(:,j);
        fprintf('x = ');
        fprintf('%d\n',x(:,j+1));
        alpha(j+1) = (1/2).*alpha(j);
        fprintf('alpha = ');
        fprintf('%d\n',alpha(j+1));
        fprintf('Iteration unsuccessful\n')
    end

    if alpha(j+1) < 1e-03
        for i = 1:par_no
            par_opt(i) = x(i,j+1);
        end
        fprintf('par_opt(1) = ');
        fprintf('%d\n',par_opt(1));
        fprintf('par_opt(2) = ');
        fprintf('%d\n',par_opt(2));
        fprintf('par_opt(3) = ');
        fprintf('%d\n',par_opt(3));
        fprintf('par_opt(4) = ');
        fprintf('%d\n',par_opt(4));
        fprintf('par_opt(5) = ');
        fprintf('%d\n',par_opt(5));
        fprintf('par_opt(6) = ');
        fprintf('%d\n',par_opt(6));
        break
    end
end

for k = 2:length(fdtimes)
    [t2,y] = ode45(@(t2,y) cgmodes(t2,y,par_opt(1),par_opt(2),par_opt(3),par_opt(4),par_opt(5),par_opt(6)),...
        [fdtimes(k-1) fdtimes(k)], [Y(end,1:3)]);
    g = y(:,1);
    i = y(:,2);
    plot(t2,g,'k')
    plot(t2,i,'g')
    [Y] = [Y;y(end,:)];
end
```

Bibliography

1. Hare, W. and Macklem, M., 2013. Derivative-free optimization methods for finite minimax problems. Optimization Methods and Software, 28(2), pp.300-312.

2. Conn, A.R., Scheinberg, K. and Vicente, L.N., 2009. Introduction to derivative-free optimization (Vol. 8). Siam.

3. Rios, L.M. and Sahinidis, N.V., 2013. Derivative-free optimization: a review of algorithms and comparison of software implementations. Journal of Global Optimization, 56(3), pp.1247-1293.

4. Klonoff, D.C., 2005. Continuous glucose monitoring: roadmap for 21st century diabetes therapy. Diabetes care, 28(5), pp.1231-1239.

5. Parkhi, D., Shah, M., Barua, A., Ghaskadbi, S. and Goel, P., 2018. A Minimal Model Approach for Analyzing Continuous Glucose Monitoring in Type 2 Diabetes.

6. Borges, C.M.R., Secaf, M. and Troncon, L.E.D.A., 2013. Clinical features and severity of gastric emptying delay in Brazilian patients with gastroparesis. Arquivos de gastroenterologia, 50(4), pp.270-276.

7. http://www.diabetes.org/living-with-diabetes/complications/gastroparesis.html

8. Marathe, C.S., Rayner, C.K., Jones, K.L. and Horowitz, M., 2013. Relationships between gastric emptying, postprandial glycemia, and incretin hormones. Diabetes care, 36(5), pp.1396-1405.

9. Lehmann, E.D. and Deutsch, T., 1992. A physiological model of glucose-insulin interaction in type 1 diabetes mellitus. Journal of biomedical engineering, 14(3), pp.235-242.

10. Dalla Man, C., Camilleri, M. and Cobelli, C., 2006. A system model of oral glucose absorption: validation on gold standard data. IEEE Transactions on Biomedical Engineering, 53(12), pp.2472-2478.