Multimodel Approaches for Plasma Glucose Estimation in Continuous Glucose Monitoring

Development of New Calibration Algorithms

Author:
Fátima Barceló Rico

Supervisors:
José Luis Díez Ruano
Jorge Bondia Company

UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

April, 2012

Agraïments

Han sigut quatre els anys dedicats a este treball i han passat moltes coses durant este temps. Moltes són les persones que han estat al meu costat en esta part del meu camí i és complicat anomenar a tot el món que d'alguna forma a contribuït a este treball. Igualment, és quasi impossible que puga transformar a paraules tot l'agraïment que sent i els dec.

Sense Jorge i José Luis este treball no seria com és. En els inicis, em va eixir la possibilitat de treballar amb ells i vaig acceptar per estar en un àrea que m'agradaba i m'agrada, sense saber com resultaria. I la veritat, difícil seria que haguera eixit millor. Gràcies per la vostra dedicació a este treball i gràcies per la vostra consideració en tantes coses.

També estan els meus companys de treball, els que ja s'han anat i els que queden. Sense ells no haguera sigut igual anar a treballar i no ho haguera fet amb les mateixes ganes. Gràcies pels cafés per començar el dia. Gràcies per tots els moments fora de la sala. Especialment a David que ha sigut el meu company de camí durant molts anys i a Laguna, que sempre ha estat disponible, sobre tot en els moments de major frustració.

Els meus amics han sigut una peça fonamental per a desconectar. Gràcies per tots els moments. Braulio, gràcies per totes les converses profundes i trivials.

A la meua familia (Aitor, Silvia, Sergio, Mari, Pedro, Sergio, Iaia, Abuela) li dec molt més que agraïments. Sempre han estat al meu costat i espere que sempre seguisquen estan. Gràcies per tot l'ànim que m'heu donat. Gràcies per ser com sou.

Carol, muchas gracias por ayudarme a ver que este camino puede ser, y es, mucho más agradable y menos complicado de lo que yo imaginaba.

A la meua germana li he d'agraïr una infinitat de coses, sobre tot el seu suport incondicional. Tata, ets un dels pilars que aguanta la meua estructura, ara i sempre. Gràcies per tot. A Arantxa li done les gràcies per aparèixer un dia i quedar-se. Gràcies a les dos.

Finalment dir que és als meus pares als que dec en gran part estar on estic. Sempre heu cregut en el que he fet i en la importància d'este treball. Estes línies mai seran suficient per expressar tot el que vos agraïsc. Gràcies per totes les coses que heu fet per mi i que seguiu fent cada dia que m'han ajudat a estar ací. Gràcies.

En definitiva, gràcies a cadascun de vosaltes. Gràcies per estar amb mi.

GRÀCIES A TOTS.

***A**l club de Fa*

Abstract

Diabetes Mellitus (DM) embraces a group of metabolic diseases which main characteristic is the presence of high glucose levels in blood. It is one of the diseases with major social and health impact, both for its prevalence and also the consequences of the chronic complications that it implies.

One of the research lines to improve the quality of life of people with diabetes is of technical focus. It involves several lines of research, including the development and improvement of devices to estimate "online" plasma glucose: continuous glucose monitoring systems (CGMS), both invasive and non-invasive. These devices estimate plasma glucose from sensor measurements from compartments alternative to blood. Current commercially available CGMS are minimally invasive and offer an estimation of plasma glucose from measurements in the interstitial fluid

CGMS is a key component of the technical approach to build the *artificial pancreas*, aiming at closing the loop in combination with an insulin pump. Yet, the accuracy of current CGMS is still poor and it may partly depend on low performance of the implemented Calibration Algorithm (CA). In addition, the sensor-to-patient sensitivity is different between patients and also for the same patient in time.

It is clear, then, that the development of new efficient calibration algorithms for CGMS is an interesting and challenging problem.

The indirect measurement of plasma glucose through interstitial glucose is a main confounder of CGMS accuracy. Many components take part in the glucose transport dynamics. Indeed, physiology might suggest the existence of different *local behaviors* in the glucose transport process.

For this reason, *local modeling techniques* may be the best option for the structure of the desired CA. Thus, *similar* input samples are represented by the same local model. The integration of all of them considering the input regions where they are valid is the final model of the whole data set.

Clustering is the technique chosen in this application for local modeling, as it is able to find automatically knowledge inherent to a database.

The general goal of this work is to design a new calibration algorithm capable of improving the accuracy of plasma glucose estimations of current devices. The proposed CA is a dynamic local-model-based clustering algorithm designed according to the system requirements.

Compensation of sensor sensitivity variations is included in the calibration algorithm through an adaptive scheme. The algorithms developed are validated with data from several clinical trials and in silico studies.

Resum

Diabetes Mellitus (DM) abarca un grup de malalties metabòliques, sent la seua característica principal la presència d'alts nivells de glucosa en sang. És una de les malaties amb major impacte social i econòmic, tant per la seua prevalència com per les conseqüències de les complicaciones a què pot portar.

Una de les línies d'investigació per a millorar la qualitat de vida dels pacients amb diabetes és d'enfocament tècnic. Esta inclou diverses sublínies com és el desenvolupament de dispositius per estimar "en línia" la glucosa plasmàtica (GP): sistemes de monitorització continua de glucosa (SMCG), invasius i no-invasius. Estos dispositius estimen la GP des de mesures del sensor en compartiments alternatius a la sang. Els actuals SMCG comercials són mínimament invasius: sensor implantat en el líquid intersticial.

Els SMCG són un component fonamental en la construcció del *pancrees artificial*, amb l'objectiu de tancar el llaç en combinació amb una bomba d'insulina. No obstant, la presició dels monitors actuals és encara pobra i això pot deure's en part a la baixa actuació dels Algorismes de Calibració (AC) implementats. A més, la sensibilitat entre el sensor i el pacient varia entre pacients i també amb el temps.

Està clar, aleshores, que el desenvolupament de nous AC efficients per a SMCG és un problema interessant i un repte.

Les mesures indirectes de la GP a través de la glucosa intersticial és un dels majors factors de confusió per a la presició dels SMCG. Són molts els components presents en la dinàmica del transport de glucosa. De fet, la fisiologia pot suggerir l'existència de diferents *comportaments locals* en esta dinàmica.

Per esta raó, les *tècniques de modelat local* poden ser la millor opció per a l'estructura de l'AC desitjat. Així, entrades *similars* són representades pel mateix model local. La integració de tots ells considerant la regió d'entrada on cadascun és vàlid dòna el model final de tot el conjunt. Les tècniques d'*agrupament* són les elegides en esta aplicació per a modelat local, ja que són capaces de trobar automàticament coneixement inherent a les dades.

L'objectiu global d'este treball és el disseny d'un nou AC capaç de millorar la presició de les estimacions actuals de la GP. El AC proposat es basa en un algorisme d'agrupament amb models locals dinàmics, dissenyat d'acord amb els requeriments del sistema.

S'ha inclòs la compensació a les variacions de sensibilitat del sensor en l'AC amb un esquema adaptatiu. Els algorismes desenvolupats han sigut validats amb dades de diversos experiments clínics i també de simulació.

Resumen

Diabetes Mellitus (DM) abarca un conjunto de enfermedades metabólicas cuya característica principal es la presencia de altos niveles de glucosa en sangre. Es una de las enfermedades con mayor impacto social y económico, por su prevalencia y por las consecuencias de las complicaciones crónicas.

Una de las líneas de investigación para mejorar la calidad de vida de los pacientes con diabetes es de carácter técnico. Ésta contiene varias sublíneas, incluyendo el desarrollo y mejora de dispositivos para estimar "en línea" la glucosa plasmática (GP): sistemas de monitorización continua de glucosa (SMCG), invasivos y no-invasivos. Estos dispositivos estiman GP desde medidas en compartimentos alternativos a la sangre. Los SMCG comerciales actuales son mínimamente invasivos: situan el sensor en el fluido intersticial.

Los SMCG son un componente fundamental para la construcción del *páncreas artificial*, con el objetivo de cerrar el bucle en combinación con una bomba de insulina. Sin embargo, la precisión de los actuales SMCG es todavía pobre y eso puede deberse parcialmente a la baja actuación de los Algoritmos de Calibración (AC) implementados. Además, la sensibilidad entre el sensor y el paciente varia entre pacientes y también con el tiempo.

Por tanto, el desarrollo de nuevos y eficientes AC para SMCG es un problema interesante y un reto.

Las medidas indirectas de la GP desde la glucosa intersticial es un gran factor de confusión para precisión de los SMCG. Muchos son los componentes que contribuyen en la dinámica del transporte de glucosa. De hechom, la fisiología puede sugerir la existencia de *comportamientos locales* en este proceso.

Con esto, las *tecnicas de modelado local* pueden ser la mejor opción para la estructura del deseado AC. Así, entradas *similares* están representadas por el mismo modelo local. La integración de todos ellos considerando la región de entrada donde son válidos da el modelo final del conjunto. Las técnicas de *agrupamiento* son las elegidas en esta aplicación de modelado local, al ser capaces de encontrar automáticamente conocimiento inherente a los datos.

El objetivo global de este trabajo es el diseño del nuevo AC capaz de mejorar la precisión de las estimaciones de GP de los dispositivos actuales. El AC propuesto está basado en un algoritmo de agrupamiento con modelos locales dinámicos, de acuerdo con los requerimentos del sistema.

Se ha incluido la compensación a las variaciones de la sensibilidad en el AC con un esquema adaptativo. Los algoritmos propuestos han sido validados con datos de diversos experimentos clínicos y de simulación.

Contents

Chapter 1

Introduction

1.1 Scope of this thesis and global objective

Diabetes Mellitus (DM) embraces a group of metabolic diseases which main characteristic is the presence of high glucose levels in blood (hyperglycemia) [134]. It is one of the diseases with major social and health impact, both for its prevalence and also the consequences of the chronic complications that it implies [43].

At present there is no cure for diabetes, only treatment is possible to control glucose levels in the healthy range. This means that any advance in the treatment will be of great and significant impact. Type 1 Diabetes or **DM1** is characterized by total lack of production of the insulin hormone by the pancreas, requiring its exogenous administration for survival. DM1 represents around the 10% of cases of DM and it is the one where the control of glucose level is more difficult [43].

One of the research lines to improve the quality of life of people with diabetes is of technical focus [130]. It involves several lines of research, one of which is the development and improvement of devices to estimate "online" plasma glucose: continuous glucose monitoring systems (CGMS), both invasive and non-invasive [19]. The use of CGMS has been associated with better metabolic control in adults in several studies, although it depends on its frequency of use which tends to decline with time [112]. This fact is exacerbated in children and adolescents with whom no benefit has been demonstrated. Besides, CGMS is a key component of new technological approaches to DM1, such as the artificial pancreas aiming at closing the loop in combination with an insulin pump [19]. Several clinical studies have demonstrated the efficacy of closed-loop control in nocturnal control, although

performance after a meal (postprandial period) is still an open issue.

CGM devices, in general, offer an estimation of the glucose concentration every 5 min [16]. Current commercially available CGMS are minimally invasive (needle-type) and have the sensor inserted subcutaneously in the abdomen region. Thus, they offer an estimation of plasma glucose from measurements in the interstitial fluid. The algorithms carrying out such estimations from the sensor's measurements are called **calibration algorithms** (CA). Usually, the sensor's output is an intensity of current signal (in nA) which varies somehow with the level of glucose in the interstitial compartment. The CA must, then, include: 1) a calibration of known glucose to relate current intensity levels to plasma glucose levels; and 2) information from self monitoring blood glucose (SMBG) samples (also known as calibrations) included to correct the output as frequently as patients take regular measurements with a glucometer for control of their disease [16].

The design of a CA has some requirements. The online estimation of the glucose level is, probably, the most important one. Current commercially available monitors are not approved by the Food and Drugs Administration (FDA) to be used on their own, due to the excessive error associated to the estimations. Therefore, the accuracy of CA is to be improved to get these devices used by their own by patients, as well as its integration into an artificial pancreas being a limiting factor for its full development. Indeed, accuracy improvement of CGMS has been a specific requirement of th FP7 programme in calls related to the artificial pancreas [46]. Indeed, the development of new CAs or recalibration methods can be found in literature as it is an increasing area of research [86, 92, 42].

On the other hand, it is known that the sensor-to-patient sensitivity is different between patients (inter-patient variations) and also for the same patient in time (intra-patient variations) [49, 85]. This means that for a similar range of glucose variations the sensor's signal may span different ranges from one patient to another and also with time.

Thus, the global objective of this thesis is **the development of a new CA improving accuracy of current devices**. In next section an outline of the main hypothesis leading to the methodologies explored in this work is presented. Then, specific objectives for this thesis are defined.

1.2 Starting hypothesis and proposed methodology

It is clear that the indirect measurement of plasma glucose though interstitial glucose is a main confounder of CGMS accuracy. The use of better models

describing this relationship (beyond the linear regression used in current commercial devices) can be a significant source of accuracy improvement. However, interstitium-to-plasma glucose transport is a complex process to model as the physiology underlying it is not yet well known. A wide range of lag time among plasma and interstitial glucose has been reported, from 0 to 45 minutes [19]. Besides, some studies have reported a fall of interstitial glucose prior to that in plasma. This may be theoretically explained by high insulin levels inhibiting hepatic glucose production and increasing significantly glucose uptake, provoking a glucose gradient in the opposite direction. Thus, it may be hypothesized that the patient metabolic state may have an influence on glucose transport dynamics and thus on CGMS performance. Physiology might thus suggest the existence of different *local behaviors* in the glucose transport process.

For this reason, *local modeling techniques* may be the best option for the structure of the desired CA. Advantages of these techniques are many, being one of them the possibility of representing a complex system with a set of models with structure simpler than a global model of the whole system [77].

Among all the local modeling techniques, *clustering* is one of the most popular, as these techniques are able to find automatically knowledge inherent to a database [33]. The aim of the clustering techniques is to group data into different sets, called clusters, in which the data belonging to each one will be more similar to the elements of the same cluster than to elements of other clusters.

When **clustering** is applied to **systems modeling**, the output that describes each cluster is a local model [62]. Thus, similar objects are represented by the same local model. The integration of all of them considering the cluster they represent (the input regions where they are valid) is the final model of the whole data set.

These techniques are increasingly being applied to complex systems modeling. The use of several local models, valid for certain characteristics of the inputs, can contribute to define a flexible model. In addition, fractioning the total set in subspaces might lead to find simpler structures for each one of the clusters, which might be useful both for the accuracy of the model and also to enhance the interpretability of the dynamic changes and estimations.

Data normalization is required for a correct application of the modeling algorithm with clustering techniques [5]. All inputs considered must be rescaled to meet same characteristics to avoid the effects of different magnitudes and the units of measurement.

The structure used for each local model and the way of defining where

these models are representative are defined taking into account the system characteristics.

1.3 Specific objectives

As stated before, the general goal of this work is to design a new calibration algorithm capable of meeting the continuous glucose monitoring requirements and improving the accuracy of plasma glucose estimations of current devices.

For this reason the first particular aim of this work is to study the problem of CGMS and go through the current state of the art of calibration algorithms, used and/or developed in literature. In this way, the characteristics and requirements of the system will be detailed and considered, as well as the possible lines to improve the estimation of these devices.

Based on the starting hypothesis previously described, the second particular goal is to review the clustering techniques, including the main concepts, its applicability to systems modeling and the current state of the art of local modeling based algorithms. The study of the already existent clustering algorithms for local modeling is a prior step to any new development. The applicability of current techniques to the problem posed must be checked.

Given the non existence of a clustering algorithm for local modeling that meets the desired properties, the next goal is to design a dynamic local model based clustering algorithm according to the problem requirements to be later integrated into a calibration algorithm. It is important to analyze first the performance of this new algorithm on general benchmarks prior to its application to continuous glucose monitoring. This new algorithm will be designed for its general application to local modeling.

Once the new local modeling algorithm is designed, it will be used to build a new calibration algorithm. Several calibration algorithms can be proposed, according to different structures. Their performance has to be checked in the CGM problem, analyzing and comparing the obtained results and discussing the advantages and drawbacks of the proposed CAs. It is expected that the new proposed CA will improve the current accuracy of commercially available CGMS.

Finally, the last goal is to include the compensation to the sensor sensitivity variations in the calibration algorithm. This has to be done adaptively, as sensitivity changes with time. The previous local model-based calibration algorithm must be the base used for this adaptation. Performance of proposed adaptive calibration algorithm must be analyzed for a typical sensor's lifetime. In this way, the accuracy of previously designed CA is to be improved as it is

adapted to any new patient characteristics.

Thus, the expected results of this work are:

1. Review of diabetes mellitus disease and study of the state of art of CGM devices.
2. Review of local modeling methods and details of clustering techniques and algorithm for system modeling.
3. A new clustering algorithm for local modeling meeting properties of glucose transport system.
4. The adaptation of this algorithm to build a Calibration Algorithm, checking different configurations.
5. A method capable to *individualize* this CA for a new patient, compensating the sensitivity variations.

1.4 Layout of this work

The organization of this work is the following: in the next chapter there is a general description of the problem of Diabetes Mellitus, focused on the problem treated here: the Continuous Glucose Monitoring. This chapter also includes the state of the art of these devices. Chapter 3 describes the state of the art of the clustering techniques used with modeling purposes, including all related concepts.

Chapter 4 a new clustering algorithm for local models identification based on the requirements of the problem under study. Following, Chapter 5 shows its integration into a new CA and its validation with a clinical data set. Analysis of results and discussion of the performance are included at the end of this chapter.

Chapter 6 presents the proposed adaptive calibration algorithm, also based on local model identification. It is defined to compensate the effects of inter and intra-patient sensor sensitivity variations, using the normalization parameters of input and output signals to do this compensation. Later, in this same chapter the performance of this new adaptive calibration algorithm is checked considering the performance during a day from clinical data. This validation is complemented with an in silico study for a week, due to the lack of experimental data for long-term validations. Analysis of results and comparison between the calibration algorithm proposed in this work and others found in literature are also included in this chapter.

To end this document, the conclusions of this work are exposed and future lines of research are included.

Chapter 2

Problem statement: Continuous Glucose Monitoring

2.1 Introduction to Diabetes Mellitus

Diabetes Mellitus (DM) embraces a group of metabolic diseases which main characteristic is the presence of high glucose levels in blood (hyperglycemia) [134].

Most of the food we eat is turned into glucose, a form of sugar. We use glucose as a source of energy to provide power for our muscles and other tissues. The glucose is transported to all the body by the blood. In order for our muscles and other tissues to absorb glucose from blood, we need a hormone called insulin. Without insulin, our bodies cannot obtain the necessary energy from our food.

Insulin is made in a large gland behind the stomach called the pancreas. It is released by cells called **beta cells**. Diabetic people are so because they have certain malfunction in this production or efficiency of insulin.

The two principal causes that define the two types of diabetes are: the total lack of production of insulin hormones by the pancreas, called Type 1 Diabetes or **DM1**; and a reduction of the sensitivity of the tissues to the insulin effects, called Type 2 Diabetes or **DM2**.

As a result, people with diabetes cannot use enough of the glucose in the food they eat. This leads to an increase of the amount of glucose in the blood, and high levels of glucose in the blood can lead to serious complications.

Diabetes Mellitus is one of the diseases with major social and health

impact, both for its prevalence and also by the consequences of the chronic complications that it implies.

At present there is no cure for diabetes. Thus, any advance in its treatment will be of great and significant impact.

2.1.1 Numbers and estimations

According to the data of the *International Diabetes Federation* (IDF, [43]), diabetes affects, nowadays, to more than 285 million people, around 6.6% of the world population. The previsions estimate that the people with diabetes will reach the 438 millions in 20 years. Each year a further 7 million people develop diabetes. In Figures 2.1 and 2.2 the distribution of the disease all over the world is represented, both for year 2010 and the estimation for year 2030. An increment of the percentage of patients can be observed in almost all contries, being only a few the contries where the prevalence is below the 5%. This gives a general idea of the social impact of this disease.

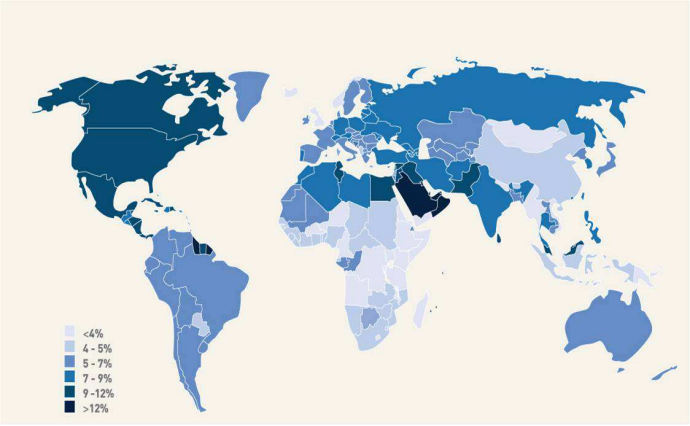

Figure 2.1: *Prevalence estimates of diabetes (29-70 years old) in 2010 [43]*

In Spain around 3 million of adults suffer this disease, according to the IDF (2010), which is around 8.7% of the country population.

Besides the social impact, another thing to take into account is its impact in health cost. The American Diabetes Association (ADA) estimated the national costs of diabetes in the USA for 2002 to be $US 132 billion, increasing to $US 192 billion in 2020. This involves not only diabetes treatments, but treatments to all complications caused directly by this disease.

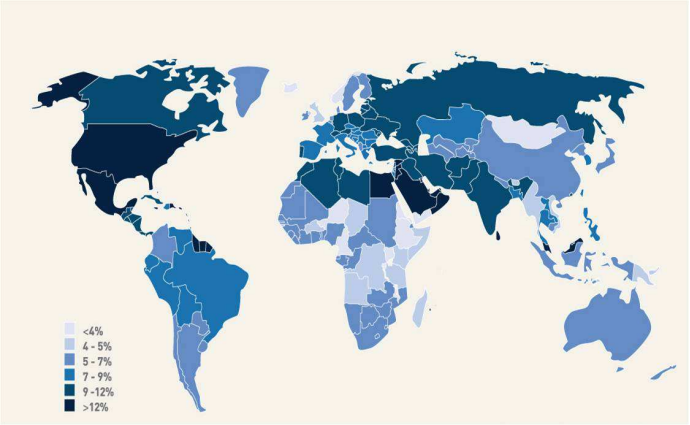

Figure 2.2: *Prevalence estimates of diabetes (29-70 years old) in 2030 [43]*

The mean cost by person with diabetes (in $US) can be seen in Figure 2.3, distributed by countries for the year 2010. The health cost by person varies from country to country and over time. The diabetes cost ratio, which is the ratio of all medical care costs for persons with diabetes is, in average, twice the cost to all medical care costs for age- and sex-matched persons who do not have diabetes.

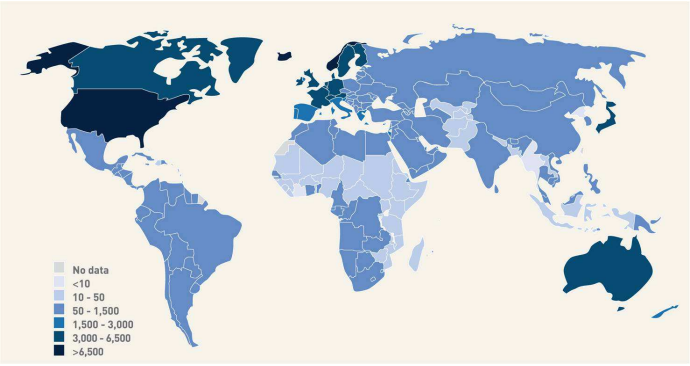

Figure 2.3: *Mean health expenditure per person with diabetes (USD), 2010 [43]*

It is very important to consider, in addition to the number of people

affected with the disease and the cost of the treatment, the tragic consequences of this disease. Figure 2.4 shows the distribution of the number of deaths caused by this disease in the year 2010. As this figure shows the total number of deaths, not the percentage, the countries where more people dies as consequence of this disease are the ones with larger population. There are countries where the number of deaths caused by this disease is larger than one hundred thousand, being this a clear indicator of how dangerous this disease can be.

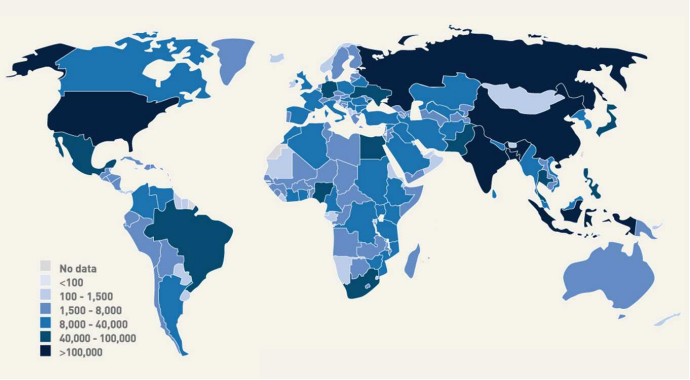

Figure 2.4: *Number of deaths attributable to diabetes (20-79 years), 2010 [43]*

All these numbers and estimations give an idea of the high impact of this disease all over the world and the importance, mainly for health reasons but also from the economic point of view, of focusing on advances in this field.

2.1.2 Glucose homeostasis

As said before the glucose taken from the food is the one that provides energy for the muscles and tissues [57]. The process of how the glucose goes from the ingested food to the organs is very complex, but here the most relevant steps and stages will be briefly described.

Metabolism is defined as the necessary energy management to keep the life conditions. In the human body this energy is used for several purposes and by different organs: muscular activity, nutrients absorption, production of cells, etc.

Digestion is the procedure to transform the food we eat into substances assimilable by the organism. These substances, called **nutrients**, are the ones

needed by the cells to do vital functions. In our diary diet we can find these types of nutrients: carbohydrates, fats, proteins, minerals, vitamins and water.

From all the substances, the **metabolism of carbohydrates** is the most relevant in this case. The metabolism of fats is also related but not in such a strong way. Before reaching the rest of the body, all the nutrients have to go through the liver. A big amount of glucose will be absorbed by the liver helped by the insulin hormone and stored in the liver cells in form of glycogen. This glycogen will be converted to glucose only if the body needs it (between meals, during the night, etc.). The rest of glucose not stored in the liver will be distributed, through the circulatory system, to the cells of the rest of the body.

For the glucose to be metabolized in the right way, the contribution of several hormones is needed, specially **insulin** and **glucagon**. The function of *insulin* is to power the absorption of the glucose by the cells and the synthesis of glycogen by the liver, reducing the concentration of glucose in blood. On the other hand, *glucagon* increases the concentration of glucose in blood, powering the conversion of glycogen into glucose.

Both hormones are produced by specials cells of the pancreas, contained in the **islets of Langerhans**, to be later released by the liver. Among other cells, these islets contain β cells, the ones that **secrete insulin**, and α cells, the ones that **secrete glucagon**. All cells, and specially these two are very interrelated, allowing for the communication between them. For example, the insulin inhibits the secretion of glucagon, as their effects are opposed.

Considering the metabolism of carbohydrates, three are the main situations where different actions are produced:

- **In resting conditions:**

  In these conditions the glucose in blood is almost constant, although the organs keep consuming it. For this, the glucose used is replaced either by the liver from the stored glycogen (glycogenolysis) or by the generation of glucose from fats and proteins (gluconeogenesis). Therefore, to keep the level constant it is needed a combined action between glucagon (to transform glycogen into glucose) and insulin (to help the distribution of the glucose to the organs through blood).

  Figure 2.5 shows the diagram of insulin and glucagon actions in resting (or fasting) conditions. Subfigure A shows the condition present for non-diabetic people. For these individuals in the fasting state, plasma glucose is derived from glycogenolysis under the direction of glucagon (1). Basal levels of insulin control glucose disposal (2). Insulin's role

in suppressing gluconeogenesis and glycogenolysis is minimal due to low insulin secretion in the fasting state (3).

◇ **In activity conditions:**

In this case the demand of glucose by the muscles increases considerably, the secretion of insulin is reduced and the glucagon increased, in order to favor the degradation of glycogen into glucose in the liver. In these conditions of physical exercise the muscles do not need insulin to absorb the glucose. Thus, in addition to the absorption of glucose dependent on the insulin, the glucose can get in the cell by other way not dependent of the insulin.

If the activity lasts a long period of time, the glucose can reach a very low level, even though it is regulated. This is due to the fact that when there are no more reserves of glycogen in the liver, the only source of glucose is the synthesis from fats and proteins (gluconeogenesis). The amount of glucose obtained by this method is not enough for covering the demand of the muscles during high activity conditions.

◇ **After a meal:**

After eating a meal rich in carbohydrates, the β cells receive signals to increase the secretion of insulin. Besides, the secretion of glucagon will be reduced. The insulin will go through the blood to the different cells, favoring the glucose transport into the cell.

The major part of glucose will be stored in the liver (glycogen), the amount exceeding the space of the liver will be transformed into fats. Once the level of glucose is stable, the β cells will stop the extra secretion of postprandial insulin.

Figure 2.5 shows the diagram of insulin and glucagon actions after a meal, or feeding conditions. Subfigure 1C shows the conditions for non-diabetic people. For these individuals in the fed state, plasma glucose is derived from ingestion of nutrients (1). In the bi-hormonal model, glucagon secretion is suppressed through the action of endogenous insulin secretion (2). This action is facilitated through the paracrine route (communication within the islet cells) (3). Additionally, in the fed state, insulin suppresses gluconeogenesis and glycogenolysis in the liver (4) and promotes glucose disposal in the periphery (5).

Metabolism is a much more complicated process, but the mentioned conditions are a brief summary of the most important situations where the

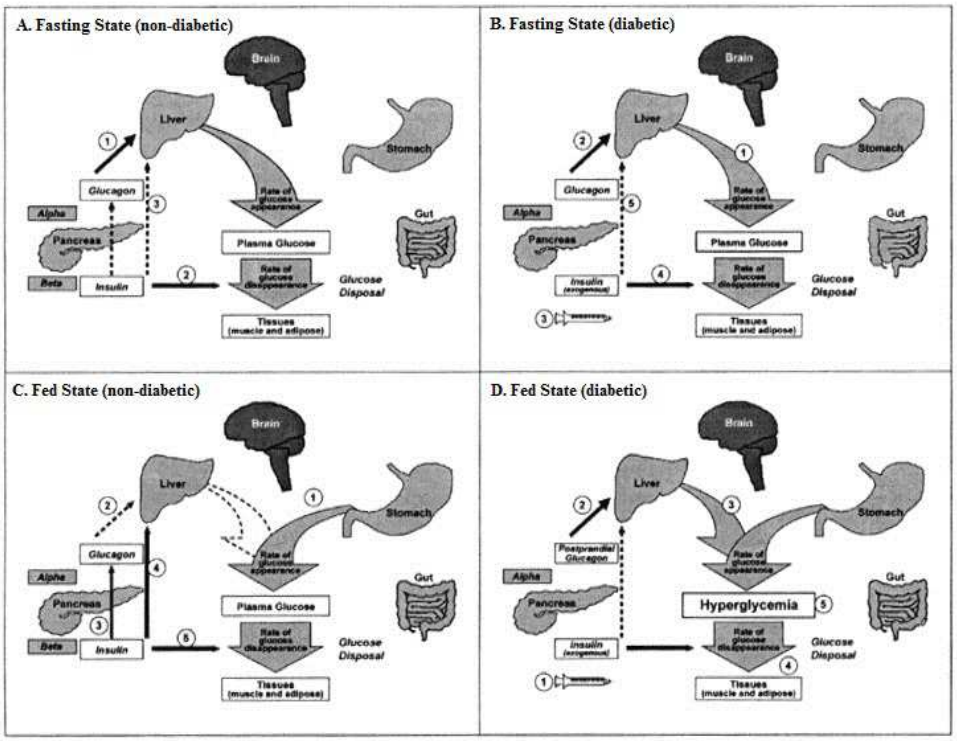

Figure 2.5: *Glucose homeostasis: roles of insulin and glucagon, comparision of non-diabetic and diabetic states, adapted from [6]*

insulin hormone works and reflect how the lack of this one would affect the normal behavior of the organs.

2.1.3 Alterations by diabetes

After the brief description of the complex system that distributes the glucose in the precise situations, it is clear that an alteration in any of the components can affect directly to glucose regulation. Therefore, in both types of diabetes (DM1 and DM2) [126], where the normal function of the insulin hormone is altered or insulin is lacking, this process is perturbed.

In the case of Type 1 Diabetes Mellitus, there is a destruction of the β cells, producing a total deficit of insulin hormone. Then, there is a total dependency on exogenous insulin for survival. The most common cause of Type 1 Diabetes

(over 90% of cases) is the T-cell mediated autoimmune destruction of the islet β cells [134]. This type of diabetes can be present at any age, starting more frequently in people under 15 years. The rate of destruction of the β cells is variable, but after certain period of time it will be complete. DM1 is also known as insulin dependent diabetes.

The alterations by DM1 in the glucose homeostasis process are also shown in Figure 2.5:

- ◇ Subfigure B shows the diagram of insulin and glucagon actions in resting (or fasting) conditions, for DM1. For individuals with diabetes in the fasting state, plasma glucose is derived from glycogenolysis and gluconeogenesis (1) under the direction of glucagon (2). Exogenous insulin (3) influences the rate of peripheral glucose disappearance (4) and, because of its deficiency in the portal circulation, does not properly regulate the degree to which hepatic gluconeogenesis and glycogenolysis occur (5).

- ◇ Subfigure D hows the diagram of insulin and glucagon actions after a meal, or feeding conditions, for DM1. For individuals with diabetes in the fed state, exogenous insulin (1) is ineffective in suppressing glucagon secretion through the physiological paracrine route (2), resulting in elevated hepatic glucose production (3). As a result, the appearance of glucose in the circulation exceeds the rate of glucose disappearance (4). The net effect is postprandial hyperglycemia (5).

Type 2 Diabetes Mellitus is the most extended case of diabetes and this classification also involves other types of diabetes. This type is present when there is a resistance to insulin effects joined to a relative deficiency in its pancreatic production. There are many factors which can potentially give rise to type 2 diabetes. These include obesity, hypertension, elevated cholesterol, etc. In general, for patients with DM2 the treatment is a healthy diet, exercise and some drugs. This is why sometimes this type is also know as non-insulin dependent diabetes.

These two types are the most commonly types of diabetes. Yet, the current classification of diabetes is based on the aetiology of the disease and two more categories can be found [134].

- ◇ *Other specific types of diabetes:* caused by such conditions such as endocrinopathies, diseases of the exocrine pancreas, genetic syndromes, etc.

- ◇ *Gestational diabetes:* defined as diabetes that occurs for the first time in pregnancy.

2.1.4 Complications of the disease

High levels of glucose in blood for a long time can cause several complications in the organism [79, 59, 134]. For this reason, in both types of diabetes, a poor control of the glucose levels can lead to several and severe problems in different organs.

For patients with DM1 the two most frequent **complications** are: **ketoacidosis** (DKA) and **hypoglycemia**. The diabetic *ketoacidosis* is a state of severe and uncontrolled diabetes, caused by an insulin deficiency and it requires an emergency treatment with insulin and fluids. This is a serious problem, with an estimated mortality between 5 and 10% in western countries. Usually DKA can be found in young patients where DM1 is still not diagnosed. Nevertheless, it can also be present in patients with DM2 during severe infections or other illnesses.

Hypoglycemia is a secondary effect very common due to the treatment with insulin. Hypoglycemia in people with DM1 arises when there is a inappropriate increment of insulin concentrations or an increment in the insulin effects. The brain depends on a continuous feeding of glucose and its interruption during a few minutes can cause damage in the central nervous system, loss of consciousness or even coma.

There are other complications caused by the long-time high levels of glucose in blood, known as **hyperglycemia**, both present in DM1 and DM2 patients. The most frequent ones are:

- ◇ *Cardiovascular problems:* Diabetic people have the same risk of mortality by a heart attack as the non-diabetic people that have already suffered one. This risk is three times higher when a diabetic patient has already had a heart attack. High levels of glucose joined to cholesterol problems favor the appearance of arteriosclerosis, which are the base to develop many heart problems.

- ◇ *Ocular complications:* These complications in diabetics affect mostly to the retina blood vessels, although diabetes also speeds up the apparition of cataracts and glaucomas. Around 5% of patiens reach a total blindness, being diabetes the first cause of blindness in several countries.

- ◇ *Kidney complications:* DM is the third cause of terminal renal insufficiency. The clinic nephropathy happens in 30-40% of the cases

with DM1. This failure provokes the presence of protein in urine. With time, this affects to the kidney functions and can reach a kidney disease in terminal state.

◇ *Neuropathy:* This is a very common complication in diabetic patients. Nerves damage are caused by a drop in the blood flow and by high levels of glucose in blood. There is a higher risk of developing it if the levels of glucose in blood are not well controlled. Usually the extremities are the first affected regions.

A common consequence is the so called **diabetic feet**. With damage to the nervous system, a person with diabetes may not be able to feel his or her feet properly. Bacterial infection of the skin, connective tissues, muscles, and bones can then occur. These infections can develop into gangrene. Because of the poor blood flow, antibiotics cannot get to the site of the infection easily. Often, the only treatment for this is amputation of the foot or leg. If the infection spreads to the bloodstream, this process can be life-threatening.

◇ *Gastrointestinal disfunction:* The most common of these diseases is the gastroparesis. This is a disorder in the stomach where this lasts a long time in getting empty. This causes problems like excessive bacteria growing. Besides, the food can harden, causing nauseas, vomits, etc.

Although these complications are the most common, they are not the only ones caused by hyperglycemia or a poor control of the glucose levels in blood. This shows the importance of having a good glucose control, not only for the daily life of patients but also for their long-time quality of life.

Therefore, any contribution in improving the quality of life of these patients will be of great importance for people with diabetes.

2.2 Research on Diabetes and its Treatment

2.2.1 The discovery of insulin

The major part of the advances leading to the discovery of the disease and development of its treatments [31] have been done in 20th century.

It was in the Greek medicine where the term *Diabetes* starts being used, in 2nd century b.C. This term refers to an urine excess, which is one of the most characteristic symptom of this disease. Other symptoms of the disease like constant thirst were also detected at that time.

It was later, in 2nd century a.C., when diabetes starts being considered like a disease from the kidney, and this idea was accepted until a few centuries ago.

In 10th and 11th centuries it was the Moorish medicine which had higher importance in this field. Moorish physicians advanced in the discovery of the symptoms and consequences of this disease [8].

Important advances in this field started in 16th century. It was detected that when urine evaporated, an strange substance was remaining. Later, it was discovered that in diabetic people, the urine has some sweet smell or taste. Finally, the rests of glucose in urine were proved at the end of 18th century. A bit later, at the beginning of 19th century, Claude Bernard determined the presence of glucose in blood.

It was in 19th and 20th centuries when the major part of advances took place. In 1889 Oskar Minkowski and Josef von Mering demonstrated that the absence of certain substance in the pancreas caused an increment of the levels of glucose in blood.

In 1869 Paul Langerhans had already discovered some group of pancreatic cells which could be separated from the rest [134]. In 1893 Edouard Laguesse suggested that these cells were the endocrine part of the pancreas, and called them *Langerhans islets*.

Later, in 1921 Frederic Grant Banting and Charles Best managed to isolate an extract of the pancreas to be used as treatment of diabetes mellitus: the **insulin**. This discovery made them win the Nobel prize [117].

With the isolation of insulin, the treatment of this chronic disease was a revolution. From that moment until these days the treatment for type 1 diabetes mellitus (DM1) is based on the exogenous replacement of the insulin hormone. There have been many advances in its dosage and composition, but the real big advance was to isolate this hormone.

2.2.2 Developing a treatment for DM1

Developing the required technology

With the discovery and purification of insulin, diabetes becomes a chronic disease with treatment instead of a deathly pathology [19]. From that moment, patients can survive, but the disease consequences start appearing. The only available technique for these patients was the test of glucose in urine. Yet, this test is only effective when the level of glucose in blood is large. At the same time, insulin was extracted from cows and pigs, which is not identical to human insulin and has impurities. At that time, patients injected the insulin

with glass needles, sterilized at home.

These are the three technology areas that take place in the treatment of diabetes: the insulin itself, the way of injecting it and the technology to measure the level of glucose. These three technologies have changed completely with time.

- ◇ Firstly, the synthesis of the human insulin was achieved. Later, this insulin was modified and insulin analogues were obtained with different pharmacokinetic profiles, with insulins of fast and slow absorption. With these, patients have more flexibility in their daily activities.
- ◇ The **measurement of glucose** has also changed completely. Firstly, a technique that used colorimetry appeared to measure glucose concentration in blood. It was only available for physicians first and later at home for patients. Years later, the first biosensor appeared, which is the technique used currently by patients at home. The first continuous glucose measurement system was developed later (end of 1990s [19]): a system that uses an inserted sensor in the subcutaneous tissue to give glucose estimations every 5min.
- ◇ The way of injecting insulin has advanced a lot since the reusable needles. The most common devices to do the injection are disposable needles and the *pens* of insulin, where the dosage can be set. Nowadays, another option is the *insulin pumps* that supply the insulin subcutaneously.

Current treatment for DM1

With these devices, the treatment of diabetes has changed very much since its first days. The therapy for diabetic patients is still based on the control of the level of glucose in blood. This means that the patients must try to have their levels of glucose all time in the appropriate range for their health. This range has been set between 70 mg/dl and 180 mg/dl. If the level is below 70 mg/dl the patient has *hypoglycemia* and if the level is over 180 mg/dl the patient is in *hyperglycemia*. Consequences of both conditions have already been described in Section 2.1.4.

In addition to the basal injection of insulin to replace the resting level of the organism, there is a treatment to control the blood glucose level for T1DM at meals. It is based in three points [41]:

1. Measurement of the current level of glucose.
2. Estimation of the ingested glucose in the meal.

3. Estimation and injection of the appropriate dosage of exogenous insulin.

Advances in the research of the treatment make it possible to find many options for each of the steps [134]. Each patient performs them in the best way for them, given their age, conditions, etc. In this section, there will not be a detailed description of all the current devices to measure the level of glucose, neither the types and composition of insulins. Here, the focus will be in the generalities of the treatment to have a global idea.

To measure the level of glucose the most popular technique is to measure the capillary glucose. This is done with a drop of blood taken from the finger tip. The device used is called **glucometer**. The capillary glucose reflects the glucose concentration at a certain moment. These measurements are very usual in the patient's therapy. They can do it by themselves, at home, which is a great advantage, as it permits the self-control. Usually, in a controlled diabetes, the patients perform around four measurements in a day (one before each meal and one before going to bed).

The most important part of these measurements is that after doing them, the patient will have to do a compensatory action, depending on the value, to get in this way a proper control. This action will be done either eating some food to increase the glucose level or injecting the proper dose of insulin.

If the level of glucose measured is below the adequate range, it must be increased by eating some food. It is very important not to increase it too much, to go over the upper limit of the healthy range. If the level of glucose is over the upper recommended limit, insulin will have to be administrated to reduce it. The amount of insulin injected depends on many factors. The most important of them, in addition to the current level of glucose, is the amount of glucose that the patient is to take. This will modify the current glucose in blood and for this reason the two things are to be considered.

The patients, by their experience, are the ones that estimate how the current level of glucose will be modified, based on the food that they will eat. They estimate the amount of carbohydrates their food has and based on their empirical knowledge they estimate how this will affect.

Once they have completed the two previous steps (measurement of glucose level and carbohydrate content estimation from the food), the patients will have to carry out the compensation action, to keep the levels of glucose in the adequate range. There are many types of insulins and also many devices to inject them. The classifications of insulin types is done attending, mainly, to its time of action and its composition. The patients will use the type or types recommended by their physician.

The most popular devices to inject the insulin are the ones called **pens**.

The insulin is measured in *units* (IU)[1]. These pens have a needle which permits the subcutaneous administration of the insulin. They also have the possibility of adjusting the dose. This dose is also calculated based on the empirical knowledge of the patient of how the insulin affects them.

The patients, when the disease is first detected, have more help from their physicians. However, with the experience, it is better for them to perform this autocontrol, which allows them to have more freedom in their daily activities.

2.2.3 Current Research

Nowadays, the research lines to improve the quality of life of type 1 diabetic patients are mainly two [130, 46]: the first of them is focused on the research of *stem cells* to replace the damaged β-cells of the pancreas (the cells that produce the insulin). The second line has a more technical character. This second approach involves several lines of research:

- ◇ The production of new *types of insulins*, more efficient and that can emulate better pancreatic secretion.
- ◇ The development and improvement of devices to *measure* the blood *glucose*: continuous glucose monitors either invasive, minimally-invasive or non-invasive [108].
- ◇ The development and improvement of *devices* to *administrate the insulin*: insulin pumps and pens.
- ◇ The development of *decision support systems* and telemedicine platforms.
- ◇ The development of *algorithms* to enable the *automatic control* of glucose to close the loop or **artificial pancreas**. A few more details about it will be given.

The automatic control of the glucose levels will enhance the quality of life of diabetic patients. This would allow them to avoid almost all the steps mentioned above, necessary for their daily control.

As the regulation process done by the human body to keep the levels of glucose is very complex, the development of a system which does it automatically will be very hard.

[1]One unit (U) of insulin is equal to the amount required to reduce the concentration of blood glucose in a fasting rabbit to 0.045 per cent (45 mg/dl) within 4 hours [11].

The key elements for the development of an artificial pancreas are (see Figure 2.8):

- ◇ *Continuous Glucose Monitoring Systems:* abbreviated CGMS. In order to have a good glucose control it is important to have frequent measurements of the levels of glucose. These devices, in general, offer an estimation of the glucose every 5 min. They are inserted subcutaneously in the abdomen region. Thus, they offer an estimation of plasma glucose from measurements of *interstitial glucose*. Frequently measured and accurate samples would enable the user to have a better knowledge of how the glucose is varying with time and actions.

- ◇ *Insulin Pumps:* they are external devices that allow the administration of insulin in a continuous way. Recently, internal pumps have also been developed. The pump has a compartment to place the insulin, filled in the same way as the normal needle. This compartment is connected to a needle placed in the subcutaneous tissue, to perform the administration. These pumps are programmed to administrate a **basal** profile during 24h, emulating the insulin between meals or during the night. Besides, the patient can indicate the extra amount of insulin needed before each meal (**bolus** of insulin). Nowadays, these devices are not automatic, it is still the patient who has to program them.

  Figure 2.6 shows a graphic example of the placement of the continuous glucose monitor and the insulin pump in the human body, in the case of an integrated device from Medtronic.

- ◇ *Control Algorithm:* it defines the basis for the automatic control. This is the part that takes the "decision" of how much insulin will be administrated at each instant [48]. As said before, nowadays, the patients do it based on the information they have and their experience. With the correct design of a control algorithm it would be possible to "close the loop" and make it automatically.

  The way of closing the loop more feasible nowadays is the use of subcutaneous (*s.c*) sensor for monitoring the glucose and the use of a subcutaneous insulin pump, although other routes have been explored such as intraperitoneal insulin delivery and intravenous measurement. Yet, many challenges exist for the development of the *artificial pancreas* (AP) [19]:

  - The use of s.c pumps arises the problem that the infusion is not physiological, introducing an additional delay. In addition this

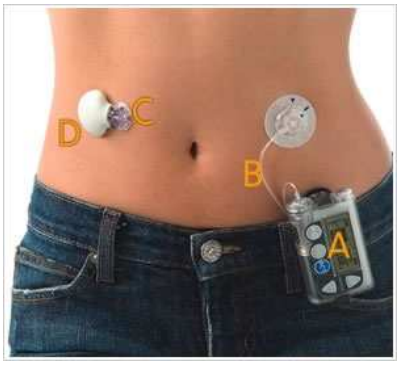

Figure 2.6: *Insertion in the human body of an insulin pump and a subcutaneous glucose monitor [131]. Elements: A) Insulin pump and CGM central unit; B) pump catheter; C) CGMS needle; D) CGMS transmission unit*

delay is variable, from 20 to 180 minutes after infusion. An injection of fast-acting insulin can make the maximum reduction of plasma glucose 80 min later [68]. This is an important challenge for the control of plasma glucose levels, since it can involve a over-action with the risk of a later hypoglycemia.

- The problem of over-acting is much more relevant in the meal compensation, where errors are larger. A meal acts as a perturbation for the control system and can be estimated but is not measurable. Clinical studies have proved the efficiency of concomitant bolus of insulin to compensate meals versus schemes based just on feedback [132].
- Another problem is the variability among patients (inter-patient) and with time for a patient (intra-patient). Variations are due to physiological aspects like stress or other physical influences. The control must be robust to act in all situations.
- The last challenge for the artificial pancreas is the large error that current monitors have in the estimations of plasma glucose. Although mean error could be acceptable, the dispersion is large and relative error can not be acceptable to close the loop efficiently.

Note that the control algorithm should have the same information as the patient does, before estimating the amount of insulin to administrate.

For this reason, not only information about the actual level of glucose will be needed, but also of how the glucose will be modified by a meal or some exercise and how the insulin affects the patient, things that patients know by their experience

Current debate is divided into two lines. On the one side, there are control algorithms directly tuned from patient usual parameters such as total insulin dose, not requiring an explicit model, i.e. *non model-based* approach, like PID controllers [67, 19]. Another approach is the design of control algorithms incorporating a patient's model for glucose predictions, i.e. *model-based* approach, like MPC controllers [94, 110]. This line can be divided in two new lines, depending on the modeling approach: the *data-driven* models and the *physiologycal models*.

In this last line, the information of the patient can be included using mathematical models of behavior determining how certain inputs (food, insulin) will act for that particular patient. Figure 2.7 represents an scheme of the models that take part in the physiology of glucose.

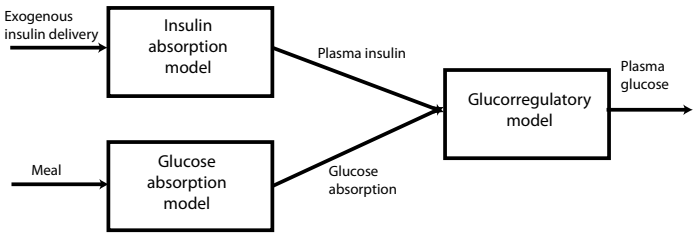

Figure 2.7: *Models and inputs of the physiology of glucose.*

The *insulin absorption model* defines how the insulin will spread in plasma once it is administrated. The *glucose absorption model* defines how the glucose intake spreads in the body. And finally, the *glucorregulatory model* will define how both signals interact and determine the level of glucose in blood at the defined conditions.

This is the translation of the information the patient "knows" by their experience in terms the control algorithm can interpret. Thus, it will have all the information needed to estimate the right amount of insulin that should be administrated preceding to disturbances.

Control algorithms have already been tested in patients [67, 110, 94]. The control of the plasma glucose levels during the night in a hospital

environment has already been achieved [68]. Other studies are currently focused on performing the control at home [110].

Thus, **current research** in the artificial pancreas is focused in **all these areas** [46]: finding right models that represent the glucose physiology, designing proper control algorithms and improving the actual monitors and insulin pumps.

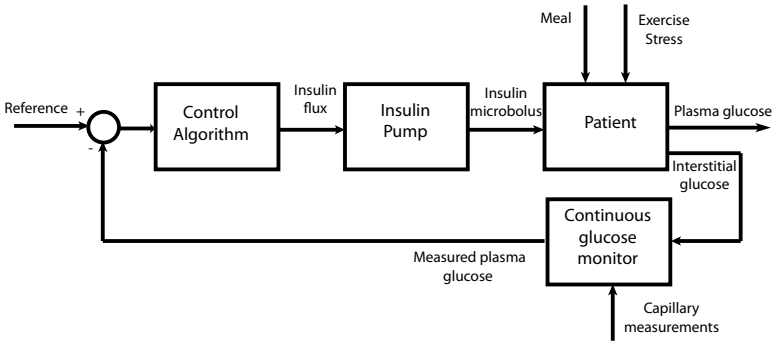

Figure 2.8: *Typical glucose closed-loop system. Adapted from [19].*

2.3 Continuous Glucose Monitoring

2.3.1 Generalities

Self monitoring of blood glucose (SMBG) is recognized as one of the fundamental steps of diabetes care since the 1980s, allowing for improvement of metabolic control especially in insulin treated patients [51, 112]. This has motivated the research of new devices, different from glucometers, for continuous glucose monitoring (CGM). This is due to the fact that there is a growing body of evidence that continuous glucose monitoring might translate into better glycemic control as compared to SMBG [124].

However, none of the devices currently available for CGM have analytical accuracy comparable to that of glucometers, especially in the hypoglycemic range where the false positive and false negative rates are unacceptably high [88, 29, 133, 112]. This is a critical issue, since accurate prediction of hypoglycemia by CGM devices is strongly needed in diabetic subjects. Nowadays, these systems are used by patients just to complement the

measurements done with the glucometers, but not to replace them. Therefore, CGM devices must be improved before these systems can be used by they own either for glucose monitoring only or with the aim of "closing the loop" for the design of the abovementioned **artificial pancreas**.

Current commercially available CGM systems (CGMS) for home monitoring are subcutaneous electrochemical sensors. All of them use the glucose-oxidase enzyme-based technology [112], which give a measurement of glucose concentration into the interstitial fluid (ISF). This measurement is expressed in terms of intensity of the current generated from the enzymatic reaction (expressed in nA) [138]. This means that the output of the CGMS is a current signal (in nA) representative of the levels of glucose in the ISF. For this reason, CGMS devices need to be calibrated with concurrent blood glucose values (from capillary measurements), thus providing an estimation of blood glucose concentration, from the generated current signal. This type of sensors are also known as minimally-invasive.

Non-invasive techniques for glucose monitoring are also a hot research topic. Nevertheless, these devices are still in a development state and they are not currently used for home monitoring. These devices measure the glucose in an indirect way from certain signals. As their development is in an early stage, their estimations are very poor, having worse accuracy than minimally invasive devices. For this reason, the regulatory agencies have not approved their commercialization yet.

In both cases, the indirect measurement of glucose levels in other compartments than plasma implies the necessity of methods to estimate the glucose levels in plasma, which is the variable of interest. These methods are also known as **calibration algorithms**, contributing to the accuracy of the estimation. To measure the accuracy of the estimations, for any device, they will have to be compared with reference measurements, also called **gold standard**, which will be samples of glucose in blood (venous, capillary or even arterial samples depending on the context).

As minimally invasive devices are in a further stage of development, there is a big interest in making improvements in this area of research. Besides, as these devices are currently commercialized, any improvement will reach faster the diabetic patient.

It is important to remark that, nowadays, the poor accuracy of the continuous glucose monitoring devices is a major constraint for its stand-alone use for therapeutical decisions, and for the development of the artificial pancreas.

2.3.2 Minimally invasive CGMS

The current intensity from subcutaneous glucose sensors is the result not only of the interstitial glucose concentration, but also of the complex interaction between the sensor and the subject. Mechanical and biological issues take part in the interaction between the sensor and the subject. Examples of these issues are: the thickness of the adipose layer where the sensor is inserted [22], the reaction of the cells where the sensor (a foreigner) is inserted [49, 85], modification of sensor response when pressure is applied due to external reasons [64], and many others that influence the sensor reaction and its output.

The accuracy of the estimation of blood glucose from the measurement in the ISF will depend, among other factors, on the calibration algorithm. Indeed, this calibration algorithm is a function that includes the relationship between plasma glucose and the sensor signal from ISF glucose, trying to minimize the error introduced by the physiological differences between the two compartments.

Unfortunately, few studies have systematically investigated this relationship [81, 7, 74, 127, 99, 20, 118], with heterogeneous results. This highlights the complexity of the plasma-to-interstitial glucose dynamics and also of the subject-to-sensor interaction, which contrasts with the rather simplistic approach of calibration algorithms currently implemented in commercial CGMS [109, 112, 16] (basically linear regression methods).

Indeed, linear regression models implemented in CGMS usually require calibration under conditions of relative glycemic stability (at "stationary" metabolic states) where equilibrium between plasma and interstitial glucose is expected. In this way the relationship between the measured current intensity and plasma glucose is considered static, neglecting any plasma-to-interstitium transport dynamics besides a pure delay. This has been recognized as a limitation of current calibration algorithms [112] but only a few studies have proposed alternative calibrations strategies [93, 92, 42].

In any case, accurate estimation of plasma glucose requires mathematical models describing the relationship between plasma glucose and the electrical signal generated by ISF glucose concentrations, both during steady state and dynamic conditions.

An alternative approach is to consider that the relationship between plasma glucose and the sensor signal from interstitial glucose is non-linear and likely depends on the metabolic state. A non-linear approach seems to be more appropriate for this system, given that the flows between compartments is a complex relationship. Thus, two relations that affect the sensor output will be included in the calibration algorithm: the one between the different

CGMS	**MARD** mean/median	**Reference Glucose**	**Bibliography**
Abbott Freestyle Navigator	12.8%/9.3%	Venous (YSI)	Weinstein et al. (2007)
Medtronic Guardian REAL-Time	19.9%/16.7%	Venous (YSI)	Mazze et al. (2009)
Medtronic Paradigm VEO	15.9%/11.6%	Capillary (Glucometer)	Keenan et al. (2010)
DexCom SEVEN	16.7%/13.2%	Venous (YSI)	Zisser et al. (2009)
DexCom SEVEN Plus	15.9%/13%	Venous (YSI)	Bailey et al. (2009)

Table 2.1: *Performance of the monitors commercially available nowadays. MARD is the mean or median* $ARD(\%) = 100 * \|gm - g\|/g$*, where* $\boldsymbol{gm}$ *is the monitor measurement and* $\boldsymbol{g}$ *is the reference glucose [19].*

compartments (ISF-plasma) and the relation between the electrical signal and the glucose concentration. Therefore, as both relations are included, the estimation of the output is to be more accurate than when one of them is ignored.

2.3.3 Calibration algorithms: State of Art

In general, the monitors available in the market do not offer detailed information about the technique used to compute the estimation of glucose. In the bibliography their performance can be found in some studies. In Table 2.3.3 the monitors commercially available nowadays are shown, as well as their performance [112, 19]. All them are minimally invasive. **YSI** refers to Yellow Springs Instrument, Yellow Springs, OH, a laboratory glucose analyzer. **ARD** is the absolute relative deviation, a measurement to quantify the accuracy of glucose estimations.

As said before, the basic linear regression models do not seem to be enough for this system. This fact and the increasing importance of having a good and frequent measurement of blood glucose has prompted to the research in this field. Here there is a collection of most of the strategies found in literature, shown from older to more recent techniques. In all them the general variables will be **y** referring to the output (**glucose**) and **x** referring to the input (output of the sensor or **current intensity**).

- ◇ **Basic linear techniques** [16]: The general equation used to compute

the output is:

$$y = m \cdot x + b \tag{2.1}$$

There are several ways of finding parameters **m** and **b**, slope of the linear equation and offset, respectively:

- **One-point calibration.** If the offset b is considered known, usually $b = 0$, then using only one sensor signal (x) - blood glucose (y) pair, taken in a certain time instant, the slope m can be computed:

$$m = \frac{(y - b)}{x} \tag{2.2}$$

  For future instants (k) with m and b the glucose level $\hat{y}_k$ can be estimated from current intensity sample x_k.
  This technique is used by the GlucoDay® CGMS (Meranini, Firenze, Italy) [12].

- **Two-point calibration.** This technique is based on two sensor/glucose pairs, and is more appropriate when the offset b is unknown:

$$\begin{aligned} y_1 &= m \cdot x_1 + b \\ y_2 &= m \cdot x_2 + b \end{aligned} \tag{2.3}$$

  where subindexes 1 and 2 represent first and second calibration. Then m and b can be found by solving the system of equations:

$$\begin{aligned} m &= \tfrac{(y_2 - y_1)}{(x_2 - x_1)} \\ b &= y_2 - m \cdot x_2 \end{aligned} \tag{2.4}$$

- **Multiple-points calibration.** When many calibration points are available, then linear regression (LR) can be used. Standard LR techniques find m and b that minimize the sum of squared errors e_k^2 (differences between measurements y_k and model predictions $\hat{y}_k = m \cdot x_k + b$, $k = 1, 2, ..., N$):

$$\min_{m,b} \sum_{k=1}^{N} e_k^2 = \min_{m,b} \sum_{k=1}^{N} \left(y_k - (m \cdot x_k + b)\right)^2 \tag{2.5}$$

In the LR case, the correlation coefficient is a measure that defines the quality of the model fit.

A related study [26], shows that using the one-point calibration method is superior to the two-point calibration. They also found that two one-point calibrations yielded nearly as good as three one-point calibrations. This results might be due to the fact that in those studies the validation was done using the Clarke Error Grid [28] to compare performances and SMBG for calibration.

On the other hand, in [84] it is shown that it was necessary that the two reference blood glucose values differ significantly when using a two point-calibration (more than *30 mg/dl*), since a larger difference reduced the bias resulting from blood glucose (BG) to interstitial glucose (IG) dynamics

Finally, to remark that calibration of CGMS by SMBG meter readings remains the major weakness of CGM technology, as the error of SMBG is carried to CMGS and it is difficult to compensate the error of these readings in the calibration.

◇ **Predictive Projections** [16]. Another way of estimating the glucose level is making projections. This is necessary for glucose estimations since the lag between compartments would require future values of the intensity of the current to estimate the current value of plasma glucose, like in the retrospective algorithms, which is not feasible for real time systems. This technique is based only on previous values of glucose and time between samples. The idea, basically, is to "find" the rate of change of glucose with time. So, knowing the time since last value of glucose and this rate of change the glucose at a certain instant k can be estimated ($\hat{G}_k$). Usually, a *window* of recent glucose measurements is used to give importance only to those few samples closer to the instant when the prediction is to be done.

It is evident that for having a model of this type many or at least several samples of known glucose are needed.

– **Linear Regression.** This analysis estimates the slope, α, and the offset, β, of the set of data y_k, $k = 1, 2, .., N$. Both parameters are found adjusting the set of data to a linear regression. Here the "independent" axis ($\mathbf{x}$) corresponds to **time**, having then a first-order polynomial-in-time model:

$$y_k = \alpha \cdot t_k + \beta \tag{2.6}$$

Considering the error of known samples to the linear regression, the *confidence interval* can also be obtained, having a measurement of the accuracy of the prediction.

The basis of this model is the consideration that the slope of the model (rate of change of glucose) is constant with time. This could be considered the main drawback of this method. Linear projections are used for short time predictions (5-10min) [103] and during steady intervals, where changes are less abrupt.

– **Other Linear Projections** have been proposed by several authors. In [25], a method is proposed that uses the extremes of the window interval to get the rate of change:

$$r_k = \frac{(y_k - y_{k-N})}{N\Delta t} \tag{2.7}$$

where N is the number of samples considered to compute the estimated rate, r_k, of variation and Δt is the time between samples.

The standard deviation (σ) of the rate is computed by the difference between the rate between samples (r_i) and the estimated rate r_k. A measure of the goodness of prediction can be made by scaling deviation, σ, by the total rate r_k. With this technique projections of 20 min into the future can be made [27].

In [24] an approach is developed using a multiple window lengths of past data for multiple linear regressions. With this technique, several predictions horizons are tested.

◇ **Dynamic Input-Output Models** [93]. Input (x) - output (y) models are very common for modeling a dynamic system. In a dynamic model the current output depends on current and past values of the input (x_k, x_{k-1}, x_{k-2}, etc) and also depends on past values of the output (y_{k-1}, y_{k-2}, etc). The number of delays considered for the input or the output will determine the order of the model.

The most used structure in these cases is a linear combination of the variables, X, known as *regressors*. Equation 2.8 shows a system modeled with a linear model, first order in the inputs and second in the outputs.

$$\hat{y}_k = a_1 \cdot x_k + a_2 \cdot x_{k-1} - b_1 \cdot \hat{y}_{k-1} - b_2 \cdot \hat{y}_{k-2} = \vec{p} \cdot X \tag{2.8}$$

Another way of expressing equation 2.8 is as follows:

$$\hat{y}(z) = \frac{a_1 + a_2 \cdot z^{-1}}{b_1 \cdot z^{-1} + b_2 \cdot z^{-2}} \cdot x(z) = \frac{A\,(z)}{B\,(z)} \cdot x(z) \tag{2.9}$$

where z^{-N} represents a delay of N time instants in the considered variable. A, B are the polynomials for the input and output.

Thus, to define this model two things have to be found: 1) the regressors (X) that best define the dynamics of the system; and 2) the regression coefficients (vector of coefficients, $\vec{p}$=[a_1 a_2 $-b_1$ $-b_2$]) that best estimate the output $\hat{y}$ (glucose) at instant k.

A common approach is to set the order of the inputs and outputs first and later find the regression coefficient vector. If the error of this structure is acceptable, then the modeling process is finished. If not, another structure has to be checked and so on.

This type of model can be used as calibration algorithm when the relation between intensity of current and plasma glucose level is introduced.

According to the nature of the variables used, dynamic input-output models can be divided into:

- **Deterministic model** is when a model uses only deterministic or "predictable" variables. This accounts for inputs and outputs (in previous instants). In general, as the output is the variable to estimate, there will not be known samples available of previous instants (y_{k-1},y_{k-2}, etc.) because they were also estimated. So, estimated samples will substitute these variables ($\hat{y}_{k-1}$,$\hat{y}_{k-2}$, etc.) to compute the output of the model.

  The linear structure is usally defined around a working point. For this reason, the offset of the system does not have to be considered in the previous description, as the origin of the axis is moved to this point, so the point (0,0) is part of the model.

  Nevertheless, an offset could be considered (a_0) in the general linear equation and would also have to be found as another coefficient.
- **Stochastic Model.** In some cases it is appropriate to consider the output is affected by some noise (w). For this reason another approach is to consider, besides the deterministic part of the model, another stochastic part, to model the noise influence.

This model can consider that the polynomial for the output is constant for both variables (noise and inputs) (ARX structure):

$$\hat{y}(z) = \frac{A(z)}{B(z)} \cdot x(z) + \frac{C(z)}{B(z)} \cdot w(z) \tag{2.10}$$

Or that dynamics of the output affecting each of them are different (Box-Jenkins structure):

$$\hat{y}(z) = \frac{A(z)}{B(z)} \cdot x(z) + \frac{C(z)}{D(z)} \cdot w(z) \tag{2.11}$$

In any case, parameters of all polynomials will have to be found to best estimate the output. This structure is used in [93], where third order polynomials are considered for the definition of the calibration algorithm.

◇ **Optimal estimation techniques** [16]. These approaches are based on the Kalman filter estimations, which are the "optimal" estimations in real-time (important fact for CGMS) for all variables of interest given all the information available, subjected to noise. This performance is due to its capabilities to systematically accommodate to new information as it develops. Since noise is considered in the information, these techniques can also be included in stochastic models category.

The underlying model is:

$$\begin{aligned} x_{k+1} &= \Phi x_k + \Gamma^w w_k \\ y_k &= C x_k + v_k \end{aligned} \tag{2.12}$$

where x is the vector of *states* and y is the vector of *outputs*, w_k is the *process noise* with covariance Q and v_k is the *measurement noise* with covariance R.

For CGMS estimations, it is quite common that one state is the output of the system g or glucose level and other is the rate of variations step to step, d, of the glucose. Thus, the Kalman filter will give at each step an estimation of the glucose level and also the change of this level at next time.

Process and measurement noise are considered stochastic processes, and the process noise covariance, Q, is known approximately and is often

used as a tuning parameter. The states are estimated using predictor-corrector equations of the form of eq. (2.13), (2.14) respectively[2].

$$\hat{x}_{k|k-1} = \Phi \hat{x}_{k-1|k-1} \tag{2.13}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + L_k \left(y_k - C\hat{x}_{k|k-1} \right) \tag{2.14}$$

where $\hat{x}$ represents an estimate of the states and the subscript $k|k-1$ means the estimate at step k is based on measurements up (and including) step $k-1$. The measurement at current step time y_k is then used to update the state based on the Kalman gain L_k.

The state estimate covariance (P_k) is determined by an update of that covariance at time $k-1$ with the state varying matrix (Φ) and also using the rest of covariances information. A correction of the propagation due to measurements update is also included:

$$L_k = P_k C^T \left(C P_k C^T + R \right)^{-1} \tag{2.15}$$

Essentially, the Kalman filter provides a trade-off between the likelihood that a change in a sensor reading is due to a real effect or to sensor noise.

Future predictions (time $k+j$) from most recent measurement at time k are given by:

$$\hat{x}_{k+j|k} = \Phi^j \hat{x}_{k|k} \tag{2.16}$$

Considering the proper states, the Kalman filter is capable of recognizing that the glucose rate of variations may change in the future, improving the accuracy of estimations.

Some works have further developed this strategy for continuous glucose monitoring systems:

- **Extended Kalman Filter for CGM** [86]. To apply the Kalman filter theory to a nonlinear process, the equations have to be linearized. Then, it is known as *Extended Kalman Filter* or EKF. The EKF provides minimum variance estimates in system

[2]This technique specifies in its equations the correction terms. Yet, any technique used as calibration algorithm uses the same correction technique when a calibration is introduced to correct the estimation given by the algorithm.

applications where the dynamic process and measurement models contain nonlinear relationships. In [86] a *5-state* Kalman filter is designed and validated on data from four patients (DirecNet [71]).

The 5 states used in the design of this extended Kalman filter are: *subcutaneous glucose levels, blood glucose levels, time lag between the sensor measured subcutaneous glucose and the blood glucose, time-rate-of-change of blood glucose level and subcutaneous glucose sensor scale factor*, which are represented respectively by: $G_B, G_S, \tau, \dot{G_S}, k_S$.

The process model used is:

$$\dot{G_S} = -\frac{1}{\tau}G_S + \frac{1}{\tau}G_B \tag{2.17}$$

This model assumes that at steady state, the subcutaneous glucose will be equal to the blood glucose.

The sensor measurement (I_S) depends on the subcutaneous blood glucose and on the sensor scale factor mainly, but also on a bias (b_S, taken as $b_S = 0$ in this case) and on error variations ε_S:

$$I_S = k_S G_S + b_S + \varepsilon_S \tag{2.18}$$

Occasional glucose measurements (capillary) will be the blood glucose plus a variation ε_B: $G_{MB} = G_B + \varepsilon_B$.

Finally some considerations are taken into account: 1) the inverse of the time lag is considered as a state $1/\tau$; 2) the variations of the sensor scale factor are considered null for a set time T and constant for time longer than T.

Authors of this work demonstrate that the 5-state Kalman filter work better than a 4-state Kalman filter, as more relevant information is included improving the accuracy of the estimations. In all cases the estimator was "tuned" for 24h or more and its performance is checked in data of later days.

– **Dual-Rate Kalman Filter** [92]. The Kalman filter in this work is applied to give an estimation of the sensor gain as well as the blood glucose at the sensor's sampling rate. Sporadic blood glucose capillary measurements are used to further improve performance.

  As the sensor gain changes with time, including noise in this variable will allow to handle this fact. For this reason, four states are considered: blood glucose (g_k) and its rate of change (Δg_k) and

also sensor gain (a_k) and its rate of change Δa_k. Both pairs follows these equations:

$$\begin{bmatrix} g_{k+1} \\ \Delta g_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} g_k \\ \Delta g_k \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w_{g,k} \tag{2.19}$$

The model has two measured outputs at different time scale: sensor output (magnitude in minutes) and capillary measurements (magnitude in hours). Thus, the model is divided into two dynamics: fast and slow. The output from the sensor (y_s) can be computed by both dynamics, while the output of capillary measurements (y_c) can only be computed with the slow model, eq. (2.21).

$$y_{s,k} = \underbrace{\begin{bmatrix} 0.5a_k & 0 & 0.5g_k & 0 \end{bmatrix}}_{C_{fast}} \begin{bmatrix} g_k \\ \Delta g_k \\ a_k \\ \Delta a_k \end{bmatrix} + v_{s,k} \tag{2.20}$$

$$\begin{bmatrix} y_{s,k} \\ y_{c,k} \end{bmatrix} = \underbrace{\begin{bmatrix} 0.5a_k & 0 & 0.5g_k & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}}_{C_{slow}} \begin{bmatrix} g_k \\ \Delta g_k \\ a_k \\ \Delta a_k \end{bmatrix} + \begin{bmatrix} v_{s,k} \\ v_{c,k} \end{bmatrix} \tag{2.21}$$

It is important to notice that matrices Φ and Γ (same as for EKF) are static, but the presence of g_k and a_k in C_{slow} and C_{fast} make them dynamic, including the degradation of sensor gain over time. Both set of equations (fast and slow) are to be updated using the predictor/corrector equations (2.13,2.14). As dynamics of both sets can be different two Kalman gains have to be defined (L_k^{fast} and L_k^{slow}).

This technique has been compared to one-point calibration strategy in simulated and experimental data, showing better performance in all data sets used.

– **Enhanced Bayesian Calibration Method**, or BCM [42]. This algorithms uses a Bayesian calibration method with EKF to improve the accuracy of glucose estimations. It was tested on simulated data for 10 patients during 7 days [89], with considerations trying to make these data "realistic". The

starting point is the work in [86] and their approach performs a enhanced calibration which works in cascade to the standard device calibration.

This is one of the first works that gives an equation (triple integration of zero mean white noise (w_α)) to define the variations of the sensor gain (α) over time:

$$\alpha_{k+1} = 3\alpha_k - 3\alpha_{k-1} + \alpha_{k-2} + w_{\alpha,k} \tag{2.22}$$

The model chosen to describe the blood glucose (G_B) is a second integrator, based on a preliminary analysis of data:

$$G_{B,k+1} = 2G_{B,k} - G_{B,k-1} + w_{G,k} \tag{2.23}$$

Besides, this work includes the dynamic relation between plasma and interstitial fluid (like [86], eq. (2.17)), consideration which is important but it is not taken into account in many works (like [92]).

Finally, six states are considered: the blood glucose at current instant, the blood glucose at previous instant, the subcutaneous glucose, the sensor gain at this instant, the sensor gain at previous instant and the sensor gain at two instants before ($x_{1,k} = G_{B,k}$; $x_{2,k} = G_{B,k-1}$; $x_{3,k} = G_{S,k}$; $x_{4,k} = \alpha_k$; $x_{5,k} = \alpha_{k-1}$ and $x_{6,k} = \alpha_{k-2}$).

The outputs of the system are two: the G_B signal and the CGMS (G_M), given that in the simulator the monitor has to be also computed. G_M is defined as the product of sensor gain α and interstitial glucose G_S with addition of a zero mean noise. Thus, output G_B is linear but output G_M is non-linear in the state space. For this reason Extended Kalman Filter has to be developed.

In this work, the authors propose an improvement of the basic EKF. As their model uses six states but only three variables and only two variances have to be computed (w_G and w_α)), the model is simpler than the one in [86].

To apply the algorithm, the authors propose a new calibration strategy, based on basal and peak instants for breakfast and dinner, trying to include the maximum information of the variation of glucose when calibrating.

This work shows how this algorithm performs better than a two-point calibration algorithm and the five state KF in [86]. Besides,

the proposed BCM shows to be more robust (performance was similar in all cases) when calibration are changed.

The main drawback of these techniques is the complicated initialization of the estimation algorithms, that involves the formulation of several extra equations (see Appendix 1 of [86]).

- ◇ **Other Methods** have been proposed in this research line. **Non-linear** techniques are gaining importance as the complexity of the problem may require more powerful techniques than linear regressions, reason why KF is also very popular. Within the non-linear theory, a very powerful technique also used by some authors for the estimation of glucose levels are the artificial neural networks (ANN). This is due to the flexibility of these models, capable of adapting to any set of data.

  [105] is an example of the application of non-linear techniques in glucose estimations. This work does not presents exactly a calibration algorithm. Instead it defines a technique that founds patterns in the glucose concentration variations and designs a NN to predict them.

2.3.4 Difficulties for advancing in the field

It is worthwhile to mention the difficulties arising in conducting academia research in this field. The main drawbacks are:

- ◇ *Inter and intra patient variability.* As it has been already mentioned this is a complex system. This is caused by the fact that human body is a complicated system. Each person metabolism is different from other person's one, and what makes things worse (from the modeling point of view), the metabolism of a person varies too, depending on the time of the day, the external conditions, the physiologic state and many other things. This difficulties to obtain a calibration algorithm efficient for everybody and every time.

- ◇ *Difficulties to develop a clinical trial to get data:* This is due to the fact that humans have to be involved. Therefore, it has to be designed the trial and also a clinical protocol which meets the ethic restrictions.

- ◇ *Expensive trials:* this is because to the cost of sensors needed in a clinical trial. When research is done by universities, it is not easy to get funding and this makes difficult to afford this type of trials. This involves that many times the developed algorithms are validated with in-silico data,

where results might not be the same as in a real application. This is a major limitation for obtaining conclusions of the studies performed.

- ◇ *Not much data available:* because getting data is so difficult, obtaining it is a precious information. Therefore, the different research groups do the most of it and avoid sharing it with other groups. This means that each group only possesses data from few trials, making it more difficult to get conclusive advances.

  There exist public data (like DirecNet), but it is difficult that data meets the required characteristics for the development of a calibration algorithm, such as frequent reference glucose measurements.

These are the main difficulties found in this research area. Given the variability present in the system, to obtain a representative calibration algorithm, for the population data from many patients would be required. Yet, obtaining data is difficult due to the presented reasons. These difficulties are quite important and have to be taken into account for the actual state of the development of this field, and also in this work.

This means that with data from few patients, the results presented can only be considered as a proof of concept, raising a hypothesis that should be validated with a representative population.

2.4 Discussion and conclusions

This chapter goes very briefly through the basic concepts of the diabetes disease, specially Type 1 diabetes mellitus. The relevance of this disease in the current world and the future evolution of the number of patients are also mentioned. There is a brief description of how the disease was discovered and diagnosed through history and also the development of its treatment based on the insulin isolation.

In addition to this, the basic characteristics of glucose homeostasis are introduced, and the deleterious complications related to the disease to see the social and economical impact of this disease and the importance of working on this area.

The second part of the chapter is dedicated to the current treatment of the disease and the technologies that take place in it. Some notes are pointed out on each area that research on this field includes. Special emphasis is given to the Continuous Monitoring of Glucose in the last section, which is the base of this work. Its importance comes from the fact that it is a bottleneck in the development of the artificial pancreas.

Accurate estimations of plasma glucose are required to be able to close the loop. It can be considered the first step to solve before the artificial pancreas can be developed.

Current state of art of calibration algorithms is reviewed in this chapter and also the problems found. Simple approaches do not reach the required accuracy while complex approaches might not be feasible in the CGM devices. Thus, it is extremely important to improve the accuracy of calibration algorithms with the design of a method that can be applied in commercially available monitors.

The idea that different dynamics can exist arises studying the characteristics of the system. The existent inter and intra-patient variability may imply that different local dynamics are present in the relation between the sensor output (intensity of current) and the level of plasma glucose. This hypothesis is the base of the approach presented in this work.

Indeed, another idea that arises from the review of the state of current calibration algorithm is that other inputs additional to the intensity of current might help in the estimation of plasma glucose levels, since in the glucose homeostasis many things are related. For example, the insulin level affects to the glucose distribution and therefore, might also contribute to modify the direction of the glucose gradient among plasma and interstitial compartments especially with high levels of insulin affecting measurement lag. Thus, the consideration of additional inputs is a line to be studied.

In next two chapters it will be introduced the methodology proposed for the design of a new calibration algorithm that takes into account the possible existence of local dynamics. The aim of this calibration algorithm is to improve the current state of the CGM devices. Firstly the general description of the methodology will be described in next chapter, and following that, the proposed approach to improve CGMS accuracy.

Chapter 3

Clustering techniques for modeling local behavior

3.1 Introduction

3.1.1 Advantages of local modeling

Advantages of local modeling are many. The main one is the possibility of finding several models with simpler structure instead of a unique complex model of the system. This enhances, many times, the interpretability of the model and the system.

Besides, this approach makes easier the procedure of finding a model for any complex system, as the problem can be divided into regions similar to simpler models instead of finding the right complex mathematical function for the whole set.

Other advantage is that the local models can have different structures, adjusting better different regions, although this technique is not as straightforward to use as the same structure for all local models and is advisable only when information of the system is well known [38].

The existence of several local models enables the possibility of finding similar regions and characterize them: similar dynamics, similar characteristics etc. Thus, the system response could be better understood with this approach.

Given the starting hypothesis raised in the previous chapter that glucose transport dynamics may depend on the metabolic state of the patient, for instance for high insulinemia, or on other characteristics of the population, this approach seems sensible to address accuracy improvement of calibration algorithms for CGMS.

3.1.2 General concepts

The general goal of local modeling techniques [77] is to find a set of c **Local Models**, LMs, that after integration in a **Global Model**, GM, represent the output of the system. In this way, each local model is simpler than the global model and can be analyzed individually. The key, therefore, consist on defining the correct parameters that allow for this partition. On the other hand, a supervising element is required capable of the integration of all local models in the final global model.

Each local model will be representative of some inputs or a region of the input space. This representativeness will define the validity region of each local model. It can be introduced as a weighting factor in the range [0,1] that indicates the grade of validity of each local model. This validity function are also called *membership functions* (MFs) and are represented by μ.

A Global Model is formed by the addition of the c weighted local models (WLM), as shown in next equation,

$$GM = \sum_{i=1}^{c} WLM_i \tag{3.1}$$

where each weighted local model is obtained by the result of the local model scaled by the membership of objects to that cluster:

$$WLM_i = \mu_i \cdot LM_i. \tag{3.2}$$

The structure of each local model and each membership function are to be set for each application, in addition to the proper number of local models.

3.1.3 Local Modeling technique selection

There are different approaches for the identification of the local models of the system. One of the first developed was the *gain scheduling*. In general, gain-scheduling encompasses the attenuation of nonlinear dynamics over a range of operations, the attenuation of environmental time-variations or the attenuation of parameter variations and uncertainties [100]. Classical gain scheduling involves offline linearization of nonlinear system dynamics at multiple operating conditions [33].

Gain scheduling technique was born with the aim of designing several controllers for different working points. For the design of the controllers, the necessity of having a model for each one of the operating points arose, *i.e.* a local model network (LMN) [77].

Another technique commonly used for local modeling is the application of the fuzzy-set theory and fuzzy logic [136]. Examples of fuzzy systems are *rule-based* fuzzy systems [136] and fuzzy linear regression models [125]. These fuzzy rules are of the type:

If *antecedent proposition* **then** *consequent proposition.*

The antecedent is a set of fuzzy propositions and, according to [9], there are three types of models, depending on the consequence proposition: 1) *Linguistic fuzzy model* [136, 97], 2) *Fuzzy relational model* [106] or 3) *Takagi-Sugeno (TS) fuzzy model* [123].

In all cases the antecedent propositions are the ones that make the selection of the rule to apply. As they are fuzzy sets, the change of rule is supposed to be smooth if the input also changes smoothly.

In both techniques, LMN and fuzzy rule-based models, the most common way of defining the local models is to identify several local models in different working points and later integrate them defining the validity regions of each of those models. Therefore, the identified local models depend in a high way of the working points defined for the process or by the user [96].

Another technique that can be used to extract local models from data is **clustering** [33]. Clustering refers to the division of data into groups of similar objects, where each group is called *cluster*. Each cluster consists of objects that are similar to the objects in the same cluster and dissimilar to objects in other clusters. In the same way, each cluster can be represented by a local model and the integration of all them, weighted by their validity, results in the final global model (see equations (3.1)-(3.2)).

Clustering is among the so-called *data mining* group, which is a group of techniques that work with sets of data and try to get some knowledge embedded within them and hidden from simple looks. Clustering is widely used in several disciplines. The increasing number of applications where this technique [58] is applied shows its potential.

The main advantage of clustering techniques versus the other possible techniques for local modeling is that clustering algorithms find *automatically* the groups of *similar* behavior. This similar behavior is defined depending on the application or the system, and the automatic detection of these groups makes the resulting global model be formed by the *best* local model that adjust the proposed structure.

Clustering techniques allow for the identification of flexible structures, as the definition of *similarity* can be defined according to the system. All these advantages make the clustering techniques the most interesting option for

local modeling of a system where each local model is desired to have certain properties.

3.2 Clustering technique review

In this chapter a general overview of concepts related to this technique is given, as well as its particularization for systems modeling applications. The main algorithms developed in this line will be described to see the state of the art of this technique and the point where this work starts.

3.2.1 Clustering Steps

Given that clustering techniques have to determine which objects are *similar* to which others, this suggests that clustering is an iterative process [4]. The following steps describe the process of applying this technique:

- ◇ **Data Collection**: Refers to the careful extraction of relevant data objects from the underlaying data sources. Here, data objects are distinguished by their individual values for a set of parameters considered, called *attributes*.
- ◇ **Initial Screening**: Refers to the first treatment of the data after its extraction from the source, in order to see whether the data collected is suitable for clustering.
- ◇ **Data pre-processing**: Includes the proper preparation of the data in order to become suitable for the clustering algorithm. Here it is where the similarity measure is chosen (or defined) and the characteristics, including the dimensionality of the data, are examined.
- ◇ **Clustering Tendency**: Checks whether the data in hand has a natural tendency to cluster or not. Most of the times this step is ignored.
- ◇ **Clustering Strategy**: Involves the careful choice of the clustering algorithm, initial parameters and its immediate application.
- ◇ **Validation**: This is one of the last steps. It can include visual techniques if the input dimension is low. However, as the amount of data and its dimensionality grow, the manual and visual techniques become useless. A validity measure has to be defined to check the performance of the algorithm according to the application purpose.

◇ **Interpretation**: The idea is to draw conclusions from the clustering results. Therefore, it is highly desired to have an algorithm that provides interpretable results that can be used for conclusions or further analysis.

3.2.2 Notation and Concepts

In order to clarify those terms mentioned before such as *object*, *attribute*, etc. and used all along this document we will offer a brief explanation on notation. Consider a data set or input matrix $X = \{x_1, x_2, ..., x_n\}$ consisting of n data points which will be referred to as *objects* and may represent people, things, transactions, etc. Each object can be represented in the following way $x_i = (x_{i1}, ..., x_{iD})$, where each component from 1 to D will represent an attribute from the attribute space A, $x_{il} \in A_l$, $l = 1 \dots D$. Attributes may represent characteristics, variables, dimensions, fields, etc. This *object by attribute* data format corresponds to an $n \times D$ matrix and is used by most of the clustering algorithms.

An important feature of clusters are their prototypes or also called *centers*. To refer to them c_i will be used, where each of the centers will have D coordinates (as many as the column dimension of the input matrix). In many cases, the centers of the clusters c_i will act as a simplification of the data set, where all the objects will be represented by only the centers and the belonging of each object to a cluster. This belonging to a cluster is better known as *membership*.

U and μ represent the matrix of membership values (μ_{ij}), which size is same number of rows as objects and same number of columns as number of clusters, $n \times c$. The set of all memberships of all objects to a cluster can also be referred to as *membership function* .

3.2.3 Data Types

All the objects within the database are represented by their attributes which represent the characteristics or features considered relevant to do the clustering of these objects. These characteristics are an important factor for the definition or selection of the clustering algorithm.

According to the value by which attributes are represented, two types of data can be differentiated: **numerical** and **categorical**, [72]. **Numerical** refers to an attribute that can be fully represented by a number. This leads to another subdivision: *continuous* and *discrete*. An attribute is continuous if its domain is uncountably infinite, i.e. *temperature, length, etc.*. On the other hand, a *discrete* attribute is so if its domain corresponds to a finite

set. This means that such attributes can adopt only some of the values of the interval and between any two values the number of values is finite, i.e. *number of children, age, etc.*. Numerical attributes have the inner property of being ordered by its magnitude.

Categorical attributes are those that can not be fully represented by a number, but they have to be represented by a category. These categories are most of the times nominal labels which are assigned to the attribute to define the particular characteristics it has. One of the main characteristics is that these categories are independent from one another and there is not an absolute way of ordering them, i.e. *brands, colors, etc.*.

Gradable attributes are a particular type of categorical attributes. Each category is a grade: bad, regular, good, etc. Although they are categorical, the different categories are dependent and have an inner order. This means that, in this point, they are more similar to numerical attributes than to categorical and their treatment could be done as if they were discrete numerical values.

Attributes that are numerical but divided into intervals are also an exception, given that they correspond neither to continuous numbers nor to discrete number, but to a group of numbers which can be considered as categories which are dependent and can be ordered. Their treatment can be done in a similar way to gradable attributes.

One particular case is that of *binary* attributes, when they can adopt only a pair of values, either numerical or categorical. There are multiple examples for this case, like: *yes/no*, *female/male*, *black/white*, etc. This type of attributes are more similar to categorical than to numerical, meaning **presence/no presence** of something, but they can easily be converted to numerical values **0/1**.

So, despite the different types of attributes, they can be divided into two types: ones that can be treated like numbers, i.e. *numbers, gradable and binary attributes*, and another one that are independent, i.e. *categorical attributes*.

The type of attributes is very important when clustering, as it aims at finding *similar* objects. This similarity is defined by a measure or metric that will have to take into account the nature of the attributes to be able to handle them.

3.2.4 Similarity/Dissimilarity Measures

It is clear, therefore, that similarity/dissimilarity among the objects of the database has to be quantified somehow. However, there does not exist an inherent way of measuring this, and that is why similarity/dissimilarity measures have to be defined [72]. The quantification of similarity is done by

looking and comparing the attributes that define two involved objects [5].

After choosing or finding the most relevant characteristics for the data set, a measure to determine their similarity is to be considered. The measure has to take into account that the more two objects resemble each other the larger their similarity is and the smaller their dissimilarity. For this reason, in the numerical input space, the most common way of measuring this similarity is using a distance measure, also called *distance metric*. These metrics are based on geometry properties, inherent to numbers, so they cannot be used with categorical attributes.

Given three objects (formed each one by D attributes) x, y and z a distance metric d should satisfy the following properties:

1. $d(x, y) \geq 0$: non-negative value;
2. $d(x, y) = 0$ **if** and **only if** $x = y$: distance of any object to itself is 0;
3. $d(x, y) = d(y, x)$: symmetry;
4. $d(x, z) \geq d(x, y) + d(y, z)$: triangle inequality.

A description of some measures will be done for each of the attributes types.

Numerical metrics

Many distance metrics can be found for general geometric purpose and used for clustering. Here there are some of the most used ones, most of them based on the **q-norm** of the difference vector:

$$d(x, y) = \|x - y\|_q \tag{3.3}$$

◇ *Minkowski Distance* is another expression for (3.3), and is defined as:

$$d(x, y) = \left(\sum_{i=1}^{D} |x_i - y_i|^q \right)^{1/q} \tag{3.4}$$

where q is a positive integer ($q \geq 1$).

◇ *Euclidean Distance* is defined as:

$$d(x,y) = \sqrt{\sum_{i=1}^{D} (x_i - y_i)^2} \tag{3.5}$$

Note that this equation is a particularization of (3.4) with $q = 2$.

◇ *Manhattan Distance* is defined as:

$$d(x,y) = \sum_{i=1}^{D} |x_i - y_i| \tag{3.6}$$

Note that this equation is a particularization of (3.4) with $q = 1$.

◇ *Maximum Distance* is defined as:

$$d(x,y) = \max_{i=1...D} |x_i - y_i| \tag{3.7}$$

Note that this equation is a particularization of (3.4) when $q \to \infty$.

◇ *Mahalanobis Distance* is defined as:

$$d(x_j, c_i) = (x_j - c_i)^T \Sigma_i^{-1} (x_j - c_i) \tag{3.8}$$

where Σ_i is the covariance matrix of the cluster. Note that this equation is like Euclidean distance, but the covariances of the objects have been included or, what is the same, the Euclidean distance corresponds to the Mahalanobis distance but with the identity matrix (I) as covariance.

Figure 3.1 shows a representation of the shape of the resulting cluster using Mahalanobis and Euclidean distances. In the first case (a), the shape is an ellipse while in the second the cluster shape is a circle.

It is important to see that all these metrics describe a *dissimilarity* measure, as they give a larger value when more different are the two considered objects. Thus, to measure **similarity** with these metrics the **smallest** value the best.

Although the most common distance measures are these ones, they are not the only ones for numerical attributes. For example, it can also be found the *sample correlation coefficient* in some applications as similarity measure for quantitative (numerical) data [15]. The sample correlation coefficient is

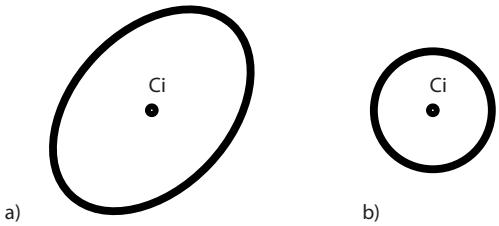

Figure 3.1: *Shape of resulting cluster when using a) Mahalanobis distance and b) Euclidean distance*

a similarity measure used to quantify the lineal dependency between two characteristics or dimensions.

Sometimes it is desirable or appropriate to use a more complex distance as it can be the ones proposed by **Gowda and Diday** [53] and **Kullback-Leibler** [91], when the application requires certain properties in the *similar* objects. For this reason, it is very common to find many metrics defined for a particular application with particular characteristics. Nevertheless, the basic distance metrics are, still, the most used ones for general clustering purposes with numerical data.

Binary metrics

For this type of attributes the *matching coefficients* (special numerical metrics) can be defined, which are based on the number of common elements between the two objects that are being compared. As they are binary attributes their characteristics are only defined by two values. Let's use **1** to describe one of the options and **0** to describe the other one whichever their actual categories are, in order to simplify the description of the measures. Now four parameters are defined to express matching and not matching:

- $\diamond$ α is the number of attribute values that are equal to 1 in both objects.
- $\diamond$ β is the number of attribute values that are equal to 1 in x but 0 in y.
- $\diamond$ γ is the number of attribute values that are equal to 0 in x but 1 in y.
- $\diamond$ δ is the number of attribute values that are equal to 0 in both objects.

◇ τ is the total number of attributes and then, $\tau = \alpha + \beta + \gamma + \delta$

From this definition we can express the two most used distance measures for binary attributes:

◇ *Simple Matching Coefficient* is defined as:

$$d(x, y) = \frac{\alpha + \delta}{\tau} \tag{3.9}$$

This index accounts for the number of attributes that are equal in both objects. This coefficient is used in most of the situations when there are binary attributes.

◇ *Jaccard Coefficient* is defined as:

$$d(x, y) = \frac{\alpha}{\alpha + \beta + \gamma} \tag{3.10}$$

This index disregards the number of 0-0 matches. This is used when the state described as 1 has a higher weight or influence than the other (described as 0). This one is not commonly used and only applied in singular cases.

Categorical metrics

Although in the case of numerical and binary attributes the distance measures are very well-known and widely used, in the case of categorical attributes there is no a distance measure that has become of widespread use for its great performance and variety of possible applications. Indeed, the most generalized situation is that for any new work of clustering with categorical information the requirements or the properties of the data make necessary a design of a new similarity measure, to cope with the expectations. Nevertheless, one measure which is well-known is:

$$d(x, y) = \sum_{i=1}^{D} \delta_{xy}^{i} \tag{3.11}$$

where for each attribute i in x and y, the parameter δ_{xy}^{i} is defined as:

$$\delta_{xy}^{i} = \begin{cases} 0 & if \quad x_i = y_i \\ 1 & if \quad x_i \neq y_i \end{cases} \tag{3.12}$$

In this way when both attributes are equal they do not affect to the total distance and when they are different they contribute by one unity. This means that (3.11) measures dissimilarity between a pair of objects.

3.2.5 Types of Membership

In classical clustering an object belongs just to a cluster. This is also called exclusive membership or **hard membership**. In this case, the membership of an object to each cluster is either 1 if the object belongs to that cluster or 0 if the object does not belong to a particular cluster. The resulting clusters have the following properties:

$$P_i \cap P_j = \emptyset, \quad 1 \leq i \neq j \leq c \tag{3.13}$$

$$\emptyset \subset P_i \subset Z, \quad 1 \leq i \leq c \tag{3.14}$$

$$\bigcup_{i=1}^{c} P_i = Z. \tag{3.15}$$

where P_i and P_j are two subgroups (clusters) of objects of the total group of objects Z, and c is the total number of clusters.

Equation (3.13) means that there are no elements in common between any two clusters, equation (3.14) means that no cluster is the empty space and finally, equation (3.15) indicates that the union of the objects from all clusters gives the whole database.

A relaxation of this condition was later applied. This was called non-exclusive membership or **fuzzy membership**. Non-exclusive membership indicates that the objects belong to more than one cluster with different membership degrees to each of them.

The idea of non-exclusive membership algorithms comes up immediately if it is presented a situation like the following one. Let's have a database where a hard clustering is performed, and when the clustering is performed there is an object which is equally distanced from the centers of two clusters. Then, which cluster must be this object assigned to? This situation makes that those points equally distanced to more than one cluster are not represented properly if they are only assigned to any of those clusters.

Fuzzy clustering relaxes the requirement that data points have to be assigned to one and only one cluster. In these algorithms data points can belong to more than one cluster and even with different levels of membership,

not just half and half. These non-exclusive cluster assignments can represent the database structure in a more natural way, specially when clusters do not have a perfect boundary, or what is the same, when clusters overlap. At these overlapping boundaries the fuzzy membership can indicate the ambiguity of the cluster assignment. In this type of clustering the total membership of an object (to all the clusters) is always the unity. The membership is usually represented by μ. μ_{ik} represents the membership of object k to cluster i. Equation (3.16) represents the restriction that the total membership of a cluster has to be positive, which means that no cluster can be empty. Equation (3.17) shows the restriction of the fuzzy membership to add in total the unity.

$$\sum_{j=1}^{n} \mu_{ij} > 0 \quad \forall i \in \{1, ..., c\} \tag{3.16}$$

$$\sum_{j=1}^{c} \mu_{ij} = 1 \quad \forall j \in \{1, ..., n\} \tag{3.17}$$

A particular case of non-exclusive membership is the **possibilistic membership**. This is particularly interesting when the clusters are defined attending to independent characteristics. In the possibilistic approach the restriction of the total membership of each object to be the unity is avoided.

Two examples will clarify fuzzy and possibilistic approaches. Let's consider a data set where the height of children is stored. If three clusters are considered: small, medium and tall. Each object (children) will belong to one of the cluster or will be between two, sharing the membership. On the other hand, a data set where the ability to play sports is stored. If three clusters are considered: soccer, basketball and handball, then each object can belong with different levels of membership to each cluster, and this level of membership is not defined by the rest of the clusters.

The possibilistic approach is very useful to detect objects that do not meet the characteristics of any model, i.e. **outliers**. With this approach an object can have a very low membership value for all clusters, indicating that this object is not well represented by any of the prototypes and therefore, not by that global structure. This is not possible in fuzzy clustering, where all objects have a total membership equal to unity, even when they are very dissimilar to all prototypes.

Therefore, hard membership is useful when the boundaries of the clusters can be well defined. Fuzzy membership is useful when the clusters are dependent, while possibilistic membership is useful when clusters are independent and the rescaling of the membership for each element can mislead the fitting of the object by each cluster.

3.2.6 Normalization

The attributes considered in the database can be of very different nature, not only for the type of data they belong to, but also for the units they are expressed in, their magnitudes and other characteristics. Therefore, to make all attributes have the same importance it is advisable to normalize all them before the clustering algorithm is applied [5, 73]. TAs result, the units with which the attributes are measured will not affect, nor the magnitude of them.

The most important property of normalization is that after it, all variables have to be equally important for the clustering algorithm. The normalization step is essential for a proper clustering process, much more if the clustering algorithm is a numerical algorithm. Different **ways** of normalizing the variables can be considered. Two commons ones are:

- **Within an interval.** In this type of normalization, the attributes will be normalized so that all of them, after normalization, are defined in the same interval. The normalized interval is, usually, defined by the user. This normalization is based on a linear transformation of the data in the defined range $[y_{min}, y_{max}]$. Two common intervals are: [-1,1] and [0,1]. The equation for the new normalized values is:

$$y_{norm} = \frac{\Delta y}{\Delta x}(x - x_{\min}) + y_{\min} \tag{3.18}$$

  where y refers to the normalized variable while x to the non-normalized one. Δy is the length of the interval for the normalized attribute and Δx the length of the interval of the attribute before normalization.

- **Statistical properties.** In this other type of normalization, the variables will be normalized in order to meet some properties equal for all of them. The most common type is the normalization to have certain statistical properties in the variables, like null mean and unitary variance (following a normal distribution).

  The equation to normalize the variable in this way is:

$$y_{norm2} = \frac{(x - \bar{x})}{\sigma_x} \tag{3.19}$$

  where $\bar{x}$ is the mean of the non-normalized variable and σ_x its standard deviation. With this type of normalization all attributes will be, in the statistical sense, balanced and will have the same effects in the clustering process.

Apart from the normalization used, it is necessary to account the origin of the inputs and therefore, how to consider them for the normalization. Each input can correspond to just one or to several experiments, which could be batches or different subjects. If this is the case, the variables can be, if convenient, treated in an special way. Thus, the different **types** of normalization can be:

- ◇ **By the set.** Whatever the type of normalization taken, each variable is normalized taken the parameters of the full set. That is, the interval or statistical properties are obtained for the total vector. This normalization is to be applied when the input is obtained just from one experiment or several experiments have been performed but some homogeneity condition holds. In Figure 3.2 a diagram of this type of normalization using statistical properties is represented, where Psn represents the full matrix of normalized variables using set parameters.

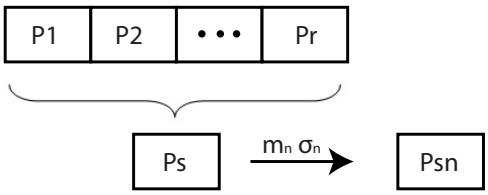

Figure 3.2: *Schematic of set normalization*

- ◇ **By subject.** This type of normalization is only available when each input is composed by data of different heterogeneous experiments or subjects. Then, inputs of each experiment can be normalized with its own parameters. So, the final normalized input vector will be formed by several vectors normalized by they own. This can be very useful when ranges of each experiment are different but the input-output dynamics is the same for all of them. In Figure 3.3 a diagram of this type of normalization using statistical properties is represented, where Pin represent the final matrix of inputs where each subject has been normalized by its own parameters and the total matrix is formed by latter joining all normalized vectors. This type of normalization is also known as **individual normalization**.

It is important, and somehow obvious, that the resulting output of the clustering will be in *normalized* dimensions and, therefore, to have it

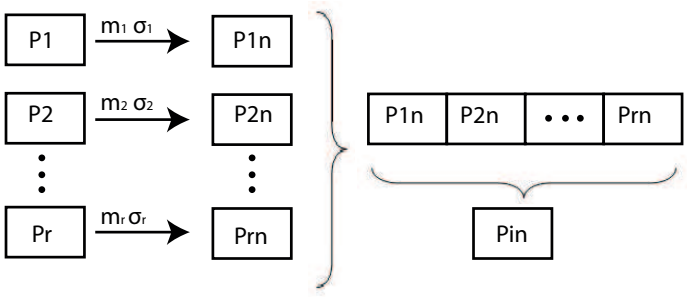

Figure 3.3: *Schematic of individual normalization*

in natural dimension it will have to be denormalized accordingly to the normalization applied.

3.2.7 Classification of clustering

There exists a large amount of applications for clustering, and therefore many algorithms have been developed to solve the requirements of each case. There are different techniques underlying the clustering algorithm. Considering the application, the technique used, the type of output, and many other factors, the different algorithms can be classified. Each class is done according to one criteria and as these criteria do not consider the same factors, this yields to an overlap between the several classes, being it possible that one algorithm belongs to two groups if the classification criteria is different in both.

The main divisions of the algorithms are:

- ◇ By the clusters division: **Partitional/Hierarchical**. Hierarchical techniques are further divided into *agglomerative* and *divisive*. While hierarchical algorithms assemble (if they are agglomerative) or disassemble (if they are divisive) points into clusters, partitional algorithms learn clusters directly. Usually, this last type of algorithms tries to discover clusters by iteratively relocating points between clusters, but sometimes the technique used is different, like finding highly populated regions, dividing the space into segments or many others.

- ◇ By its membership: **Exclusive/Non-exclusive**. Exclusive membership means that the objects belong or not to a particular cluster. This is the most traditional way of clustering attributes and many times

it is denoted as *hard* clustering. However, if the membership is non-exclusive then the objects belong to more than one cluster with different membership degrees to each one. This is denoted as *fuzzy* clustering and it is quite interesting in the cases where the database is not very uniform. Another case is the *possibilistic* clustering, which is a particularization of the fuzzy clustering, where another degree of freedom is included.

- ◇ By the type of inputs: **Numerical/Categorical/Mixed**. The description of the types of attributes has been done in Section 3.2.3. According to the type of data which with the algorithm is able to work with, it will be placed in one of these three classes.

Other divisions, not as largely used as the previous ones but also to be considered, are:

- ◇ By criteria of clustering: **Proximity/Density** based clustering. Density-based algorithms group objects according to a specific density objective function, instead than by proximity. Density is usually defined as the number of objects in a particular neighborhood of a data object. In these approaches a given cluster continues growing as long as the number of objects in the neighborhood exceeds some parameter (threshold).

- ◇ By its representatives: **Single point/Model**. Model-based clustering algorithms find, instead of a single point to represent a particular cluster, the parameters of a function (given its structure a priori) that best represent each cluster. Thus, each cluster will be represented by a model. This division is not largely used as it is the application which defines the type of representative and there are no many options of selecting another type. Nevertheless, it is very important to have clear this type of classes as both of them are largely required in clustering applications.

- ◇ By the division of the space: **No-Division/Grid**. Most of the algorithms do not perform any division of the space and allow the clusters to be placed anywhere. On the other hand, the aim of *Grid-Based* algorithms is to quantize the data set into a number of cells (of the input space) and then work with objects belonging to these cells. They do not relocate points, but rather build several hierarchical levels of groups of objects.

These are some of the criteria used to classify clustering algorithms. It can be seen how some classifications overlap between them, as they use different criteria. Some algorithms in each class are:

- ◇ Hierarchical algorithms: **BIRCH** [137], **CURE** [55], **CHAMELEON** [78], **COBWEB** [44], etc.
- ◇ Partitional algorithms: **k-means** [61], **PAM**, **CLARA** [80], etc.
- ◇ Exclusive membership: **k-means**, **BIRCH**, etc.
- ◇ Non-exclusive membership: **FCM**, **PCM** [2], etc.
- ◇ Categorical algorithms: **k-modes** [70], **ROCK** [54], **STIRR** [50], **CACTUS** [47], etc.
- ◇ Mixed algorithms: **k-prototypes** [70], **dSQUEEZER** and **usmSQUEEZER** [63], etc.
- ◇ Density-based algorithms: **DBSCAN** [75], **GDBSCAN** [40], **DENCLUE** [66], etc.
- ◇ Model-based algorithms: **FCRM** [62], **AFCR** [114], **TSP** [35], **Expectation-Maximization** [21], etc.
- ◇ Grid-Based algorithms: **STING** [129], **WAVECLUST** [116], **CLIQUE** [3], etc.

3.2.8 Definition of the cost index

In order to find the clusters embedded in a data set, it is necessary to define the methodology used by the clustering algorithm. There are many clustering algorithms and many clustering tendencies already developed. Nevertheless, most of them have a characteristic in common: the clustering is performed by the optimization of a cost index.

In this cost index the essential properties desired for the clusters are included in a mathematical way and later the clusters are obtained by finding the best solution that performs the best in the defined index, resulting in the output of the clusters. For that reason, the first thing to be set is what is wanted as the output of the data set after the clustering is done. This could be the center of each cluster, the number of clusters, the size of them and so on, depending on the purpose of the clustering and the desired structure for the resulting groups of objects.

Once the output is set the cost index can be defined by translating the properties of the resulting clusters in a mathematical way. Some of these properties are: the distance from each objects to the center of each cluster, type of membership, shape of the clusters, number of clusters and so on.

When the cost index is defined, two other things have to be fixed. The first of them is finding what is the best solution for a particular index. In most of the cases, the function includes distances or errors, thus, the index will have to be **minimized** in order to find the output of the clustering algorithm. In other cases, the index might need to be **maximized** to find the optimal solution for the set.

The second thing to fix, once it is clear what we are looking for in that function, is the procedure to find this solution. The minimization (or optimization) of the function can be done using many **optimizers**, which, at the same time, can be local or global ones. Choosing one solver depends on many things: the application, the data set, etc. Two factors of high importance are the computational time allowed and the precision required in the solution.

Nevertheless, the **key** of clustering algorithms is how to **translate** those desired **properties** of the clusters and its outputs into a **mathematical** language to be treated by numerical ways.

3.3 Relevant algorithms for clustering

There has been a few clustering algorithms that have meant a revolution or a significant advance in clustering theory. Most of them use a relatively easy principle, but their importance comes from this fact. They will be described here and used as a few examples of how a clustering algorithm performs the clustering process. The detailed algorithms have been divided into three groups, according to the type of data that can handle: numerical, categorical or both (mixed data algorithms).

3.3.1 Numerical algorithms

k-means

This is a partitional numerical algorithm, with exclusive membership.

This algorithm [61] is by far the most popular clustering tool used widely in the past and nowadays. The goal in k-means is to produce k clusters from a set of n objects, so that the objects of all clusters minimize the squared error objective function, i.e., the addition of all Euclidean distances, eq. (3.5), from all the objects to the center of the cluster where they have been assigned to:

$$J_{index} = \sum_{i=1}^{k} \left(\sum_{j=1}^{n} d\left(x_j, c_i\right) \right) \tag{3.20}$$

where n is the number of objects, k the number of clusters, c_i the center of cluster i, $d(x, c_i)$ is the distance of the element x to the center of the cluster it has been assigned i.

This equation can be expanded as:

$$J_{index} = \sum_{i=1}^{k} \left(\sum_{j=1}^{n} \sqrt{\sum_{v=1}^{D} (x_{j,v} - c_{i,v})^2} \right) \tag{3.21}$$

where now D is the number of attributes. So, the total distance will be the root square of the addition of the squares of the difference of each attribute.

The k-means algorithm has as input parameter k, the number of clusters, and as an output the algorithm returns the centers or *means* of all clusters c_i. The algorithm procedure is as follows:

1. Select k objects as initial centers,
2. Assign each data object to the closest center,
3. Recalculate the centers of each clusters,
4. Repeat steps 2 and 3 until centers do not change.

This algorithm works very well if and only if the parameter k chosen is the proper one. Otherwise, the results will not be interpretable and then this algorithm will lose its great advantage. The main disadvantage is that because the similarity measure is a distance metric, it cannot be used for categorical attributes, only for numerical.

It also happens for the k-means algorithm that it is able to detect well distributed and spherical-shaped groups of data. When the clusters have different sizes or their shapes are not spherical-like, then this algorithm does not provide efficient results.

FCM: Fuzzy C-Means

This is partitional numerical algorithm, with non-exclusive membership function.

This algorithm allows gradual membership which will be measured in the interval [0,1]. This makes the data model much more detailed and allows the total model to express how ambiguous or definite the database is. Given that now memberships are fuzzy, they cannot be expressed with only one value or label. Now they have to be a vector for each point $x_j \in X$ the length of which is k the number of clusters:

$$\mu_j = (\mu_{1j}, ..., \mu_{kj})^T \tag{3.22}$$

where μ_{ij} represents the membership to each cluster for object j.

Matrix **U** will be $n \times k$ and it is called the **fuzzy partition matrix**. Because this algorithm is probabilistic, it will be called the probabilistic cluster partition of X. It has to meet two properties, reflected in equations (3.16) and (3.17). These equations mean that all data are equally included and receives the same weight as all other data, although the distribution of this weight among the clusters differs from one object to another. As consequence, no cluster can contain all data and the membership values have to be normalized for each object.

Obviously, the closer a data point lies to the center of a cluster, the higher its degree of membership should be to this cluster. Now the problem of finding the best partition of the data set not only relies upon the optimization of a cost index computing the sum of all distances from the point to their cluster center, but it is also desired to maximize the degrees of a membership. Now the general cost index will be:

$$J(X, U, C) = \sum_{i=1}^{k} \sum_{j=1}^{n} \mu_{ij}^m \cdot d_{ij} \tag{3.23}$$

where μ_{ij}^m is the element of the fuzzy partition matrix that is related to the element x_j and all the clusters i from 1 to k and d_{ij} is the distance of the point x_j to the center of the i-th cluster c_i. In k-means algorithm the Euclidean distance is used. The parameter m ($m > 1$) is called *fuzzyfier* or weighting exponent. With higher values for m, the boundaries between clusters become softer. Usually $m = 2$ is chosen.

Because the cost function now depends on two parameters it is iteratively optimized. This means that first the membership degrees are optimized for fixed cluster centers, then the cluster prototypes are optimized for fixed membership degrees, which are the *optimum* values obtained in the previous iteration. Equations (3.24) and (3.25) show respectively the membership and the center updates, where t refers the actual clusters, i covers all the clusters and, finally, j is the number of objects n.

$$\mu_{tl} = \frac{1}{\sum_{i=1}^{c} \left(\frac{d_{tj}^2}{d_{ij}^2} \right)^{\frac{2}{m-1}}} \tag{3.24}$$

$$c_t = \frac{\sum_{j=1}^{n} (\mu_{tj})^m \cdot x_j}{\sum_{j=1}^{n} (\mu_{ij})^m} \tag{3.25}$$

The choice of the optimal cluster center for fixed membership of the data is the same case as the k-means cost index, and that is why this algorithm is called *fuzzy c-means*. Fuzzy c-means is a stable and robust classification method, it is quite insensitive to the initialization and is not likely to get stuck in an undesired local minima.

PCM: Possibilistic C-Means

Often it is desirable to have the property of the probabilistic membership degrees, although some times it can be misleading. High values for the membership of a datum in more than one cluster mean that the point is at the same distance to those clusters. If now there is another point with similar characteristics this can suggest that both points are close, but this might not be true, as it can be seen in Figure 3.4. Both points are equally distanced to the two clusters, but they are not close.

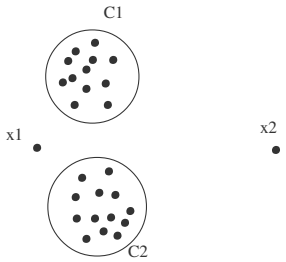

Figure 3.4: *Case when the same normalized membership value correspond to two different points*

The normalization of membership values can lead further to undesired effects in the presence of noise and outliers. The membership values affect the clustering results, since data point weights influence on cluster prototypes. A more intuitive assignment of degrees of membership can be achieve by dropping the normalization constraint, avoiding undesirable normalization effects. This

last point can be highly desirable if the clusters are considered completely independent from one another.

Now the U matrix is a possibilistic cluster partition of X and still meets the equation (3.16). Now the membership degrees of each object resemble the possibility of being a member of the corresponding cluster instead of pointing how is the proportion of their belonging to each cluster. Dropping the normalization constraint, (3.17), leads to the problem that the cost index could be minimized with all $\mu_{ij} \to 0$. In order to avoid this trivial solution a penalty term is introduced, which forces the membership degrees away from 0, eq. (3.26). In this way the desire for strong assignments is expressed and included in the cost index.

$$J(X, U, C) = \sum_{i=1}^{c} \sum_{j=1}^{n} \mu_{ij}^{m} \cdot d_{ij} + \sum_{i=1}^{c} \eta_i \sum_{j=1}^{n} (1 - \mu_{ij})^m \tag{3.26}$$

In the possibilistic approaches, the clustering methods have to learn the weights of the data points, while in probabilistic all weights are 1, understanding by this *weight* the total membership value of one point. The formula for updating the membership degrees is:

$$\mu_{ij} = \frac{1}{1 + \left(\frac{d_{ij}^2}{\eta_i} \right)^{\frac{1}{m-1}}} \tag{3.27}$$

where we can see that the new value will only depend on the distance d_{ij}, and the rest of the distances do not modify this membership.

Depending on the cluster's shape the parameters η_i have different interpretations. If some knowledge of the data set is known a priori, then η_i can be set a priori. However, this does not usually happen and these parameters have to be estimated. Good estimations can be found using a probabilistic clustering model of the given data set. Then η_i can be estimated using the probabilistic fuzzy partition matrix, as shown in (3.28).

$$\eta_i = \frac{\sum_{j=1}^{n} \mu_{ij}^{m} \cdot d_{ij}}{\sum_{j=1}^{n} \mu_{ij}^{m}} \tag{3.28}$$

The best property of this algorithm is that it is able to leave some points which are far from cluster (outliers) outside and do not assign them to any cluster.

3.3.2 Categorical algorithms

k-modes

This is a partitional categorical algorithm, with hard membership function.

This algorithm [70] was the first one oriented to categorical data. It is based on the *k-means* algorithm and uses the same structure but with a different similarity/dissimilarity measure where the main differences are that the *means* are replaced by *modes* and that the way of updating them is a *frequency based method*. The dissimilarity measure used is like equations (3.11) and (3.12). This expression counts the number of mismatches the two objects have on their attributes. All attributes are considered to have the same weight. If the frequencies of the values in a set are to be included to give the attributes different importance, the equation would be:

$$d\left(x,y\right)=\sum_{l=1}^{D}\frac{n_{x_l}+n_{y_l}}{n_{x_l}\cdot n_{y_l}}\cdot\delta_{xy}^{l} \tag{3.29}$$

where n_{x_l} and n_{y_l} are the numbers of objects in the database with attributes values x_l and y_l for attribute l respectively. The **mode** of a set is the value that appears in the majority of elements of the set. Every cluster i, $1 \leq i \leq k$ will have a mode, defined by a vector $Q^i = (x_1^i, ..., x_D^i)$, where D is the dimension of the attribute vector. The vector Q^c that minimizes the cost index, eq. (3.30), is the desired output of the algorithm.

$$E=\sum_{i=1}^{k}\sum_{x\in i}d\left(x,Q^{i}\right) \tag{3.30}$$

Given that the k-modes algorithm is based on the k-means structure, it has the same advantages and disadvantages as this one.

3.3.3 Mixed data algorithms

k-prototypes

This is a partitional algorithm for mixed data, with hard membership function.

This algorithm [70] is an integration of $k-means$ and $k-modes$, adding both dissimilarity measures:

$$s^n+\gamma s^c \tag{3.31}$$

where s^n is the dissimilarity on numeric attributes, s^c the dissimilarity on categorical attributes, and γ is a weight to balance the two parts and avoid

favoring either type of attribute. This last one is a user-defined parameter, and depends on the particular application.

Data codification

This is not an algorithm by itself, but it is an option to cluster data sets with mixed (categorical and numerical) attributes [69, 15, 14, 13, 63]. It consist on transforming (codifying) one type of data into the other type and then using a clustering algorithm capable of working with that data. The conversion of numerical data into categorical data makes the converted attributes lose some known information: categories are independent while numbers are intricately related [63]. This reason, joined to the fact that numerical algorithm are, in general term, better developed and are more efficient, make the codification of categorical into numerical data a more common option [69, 14, 13].

3.4 Clustering algorithms for systems modeling

The idea of system modeling using clustering techniques is like a combination of the classical clustering techniques with the local modeling theory. The basis are the same, but now instead of "manually" generating the antecedent and consequent propositions as in many applications of fuzzy modeling, the clustering algorithm will find similar regions to be modeled with a consequent proposition.

The use of clustering techniques for systems modeling is increasing. The reason for this is that a complex system can be modeled by the integration of several simpler systems, gaining flexibility and interpretability of the global model.

For any new case, the characteristics desired in the model have to be defined. According to these characteristics the way of modeling the system will be different. Thus, it is common that the modeling clustering algorithms can be only applied for systems with certain properties.

A description of the most relevant algorithms for this purpose is detailed here:

- **FCRM**: Fuzzy C-Regression Models, [62].

  FCRM is a model-based clustering algorithm based on the minimization of a cost function. It basically tries to find a set of fuzzy models to represent the output with a linear combination of the inputs.

  Given a data set where each independent input observation x_k has a correspondent output observation y_k, it is assumed that several linear

models, c, describe the relation between the input and the output: $\hat{y} = f_i(\mathbf{x}; \beta_i) + \epsilon_i$ where $1 \leq i \leq c$. This is known as a **switching regression model**.

If each object is described by a combination of all or some models, then the problem to solve can be divided into two: finding good estimations of parameters β_i which define $\hat{y}$ (predicted output) and finding the membership of each object to each one of those models.

In this algorithm these two problems are solved simultaneously, by an iterative approach. The membership of each object meets the restriction to be equal to the unity for the addition to all clusters. The solution will be one that minimizes a cost index defined by the summation of all errors between the real output and the output of the model weighed by the membership of that object:

$$J_{FCRM}(x, U, c, \beta) = \sum_{k=1}^{n} \sum_{i=1}^{c} \mu_{ik}^{m} \cdot E_{ik}(\beta_i) \tag{3.32}$$

where E_{ik} is the square error between the output of the object k and the output given by the local model i.

Basically, the steps to find the solution are:

1. Initialization of the variables: m, $U^{(0)}$, ϵ, etc. Choice of the error measure E_{ik}. Usually, square error function $E_{ik}^2 = (y_k - f_i(x_k, \beta_i))^2$.
2. Calculate the values for all β_i. This will be done with least squares.
3. Update the membership matrix.
4. Compare the new matrix with the last one, if the sum of changes between the two matrices is smaller than ϵ then iteration will stop. Otherwise, go back to step 2.

This algorithm is tested to work very well when the correct number of clusters, c, is chosen and when data is distributed following linear models.

As the membership of each element is only restricted by the belonging to each cluster, and not for the other elements, the membership of all elements can have a very irregular shape in the input space.

◇ **AFCR**: Adaptive Fuzzy Clustering and Fuzzy Prediction Models, [114].

This is also a model-based algorithm. It addresses the problem of the shapes of the clusters. Here they are changed dynamically and adaptively in the clustering process.

It is based on a cost index where distances to centers and modeling error are weighed, see equation (3.33), and the weight of each term changes dynamically:

$$J_{AFCR}\left(x, U, c, \beta\right) = \sum_{k=1}^{n}\sum_{i=1}^{c} \mu_{ik}^{m} \cdot Z_{ik} \tag{3.33}$$

where: $Z_{ik} = \left(1 - \alpha_i\right) D_{ik} + \alpha_i E_{ik}\left(\beta_i\right)$ and $\alpha_i = 1 - \frac{\min\limits_{j}\{\lambda_{ij}\}}{\max\limits_{j}\{\lambda_{ij}\}}$. D_{ik} refers to the distance between object k and i-th center of cluster. α_i corresponds to eigenvalues of the variance-covariance matrix, which will determine the shape of the clusters.

In that way, the index at first will have hyper-sphere clusters because the distance term will be more influential, and therefore α_i will be near to 0: the index will take into account more the distances than the output error. As the clustering algorithm progresses, the clusters will be more similar to hyper-ellipsoids and α_i will be near to 1: the index will take more into account the output error and therefore the regression parameters β_i estimation will be more accurate.

The steps to find the solution are very similar to the case of the FCRM:

1. Initialization of the variables: m, $U^{(0)}$, ϵ, etc. Choice of the error measure E_{ik} and distance measure D_{ik}.
2. Calculate the values for all β_i. This will be done applying least squares using the membership matrix.
3. Update the membership matrix. Compute α_i.
4. Compare the new matrix with the last one, if the sum of changes between the two matrices is smaller than ϵ then iteration will stop. Otherwise, go back to 2.

In this way, the shape of the cluster will be more compact in the input space.

◇ **AFCRC**: AFCR with **C**onvexity enhancement, [34].

As the name indicates, AFCRC is a modification of AFCR algorithm to include some criteria for improving μ convexity, leading to a better interpretability. Consequently, the modeling error maybe larger if the number of clusters is preserved.

Two new terms are added: first term to *penalise high membership* of points far from the prototype (J_{far}), and a second term to *penalize low membership* of points *close* to prototype (J_{near}).

$$J_{far}\left(U,C\right)=\varphi_f\sum_{i=1}^{c}\sum_{k=1}^{n}\mu_{ik}^{m}\cdot e^{\left(1-\frac{L_{ik}^2}{2\sigma_f^2}\right)} \tag{3.34}$$

$$J_{near}\left(U,C\right)=\varphi_n\sum_{i=1}^{c}\sum_{k=1}^{n}\left(1-\mu_{ik}\right)^{m}\cdot e^{\left(\frac{L_{ik}^2}{2\sigma_n^2}\right)} \tag{3.35}$$

$$L_{ik}^2\left(Z\right)=\left(z_k-c_i\right)^T B\left(z_k-c_i\right) \tag{3.36}$$

where c_i are the coordinates of the cluster centroid, B is fixed matrix, φ_f and φ_n are weights of those terms, and σ_f and σ_n are related to cluster size, to maximum and minimum respectively. Finally the total index cost for AFCRC (J_C) is the addition of J_{AFCR} with J_{far} and J_{near}, with the consideration that the term α_i will split into the three terms ($\alpha_i/3$): $E_{ik}\left(\beta_i\right)$, J_{far} and J_{near}.

Minimization of J_C leads to smoother clusters. The procedure is similar to AFCR: derive updating expressions for prototypes c_i, memberships μ_{ik} at each iteration. Consequent parameters β_i adjusted by least squares. For each iteration the rest of parameters are considered constants and only the one updated is considered variable. Steps are:

1. Initialize used defined parameters (σ_f, σ_n, φ_f and φ_n).
2. Initialize randomly U.
3. Calculate α_i and C prototypes.
4. Determine β_i coefficients by least squares.
5. Update U matrix.
6. Compare the new U with the previous one and repeat from step 3 if changes significantly.

Parameters φ_f and φ_n are to be computed iteratively and should make that contributions are around 10% and 1% respectively. Selection of parameters can be complicated to make the resulting clusters to have the desired properties, but in the work some guidelines are given to facilitate this procedure. Yet, the selection of those parameters is rather complicated and depends, in some cases of a correct visualization of clusters shape, thing that is not easy for cases with high input dimension.

◇ **TSP**: **T**arget-**S**haped **P**ossibilistic clustering algorithm, [35].

This is a possibilistic algorithm and consists in a modification of FCRM [62]. The main idea is to modify the cluster shape taking into account the distance from cluster centers in input space, in addition to modeling error. Two terms are added to equation 3.32: a first term that favors membership of points that should belong to a cluster (J_f) and a second term that penalizes membership of points that should not belong to a cluster (J_p), similarly to eqs. (3.34) and (3.35). The points desired in one cluster are those that fit an expected membership shape. The cost index is thus:

$$J_f(U,C) = \sum_{i=1}^{c} \varphi_{fi} \sum_{k=1}^{n} \mu_{ik}^{m} \delta_f(L_{ik}) \tag{3.37}$$

$$J_p(U,C) = \sum_{i=1}^{c} \varphi_{pi} \sum_{k=1}^{n} (1-\mu_{ik}^{m})\, \delta_p(L_{ik}) \tag{3.38}$$

where L_{ik}, p_i and B are like in [34], δ_f and δ_p are user defined functions of the distances to prototypes (L_{ik}) in input space (Z), and φ_f and φ_p are weighting terms for J_f and J_p respectively in the total index $J_{TSP} = J_{FCRM} + J_f + J_p$.

Note that the centroid parameters, p_i, only appear in the new terms. Basically δ_f should increase with distance, like (3.34), while δ_p is like (3.35).

The minimization of J_{TSP} for $m = 2$ can be done in a similar way to [114]. The steps are the same as in AFCRC, but applied to this case:

1. Initialize used defined weights (φ_{fi} and φ_{pi}) and functions (δ_f and δ_p).
2. Initialize U and C.

3. Calculate C prototypes, with the the derivative of J_{TSP} with respect to centroid projections p_i (it is an iterative minimization).
4. Determine β_{ij} coefficients. The problem is like a weighted least squares for each cluster i with matrix of membership U [9].
5. Update U matrix, similar to FCRM but using the unconstrained formula.
6. Compare the new U with the previous one and repeat from step 3 if it changes significantly.

As in AFCRC φ_{fi} and φ_{pi} have to be computed iteratively and should make that contributions are around 10% in total index. The higher those parameters are, the more modeling error the local models will have, in exchange for a smoother and more readable shape of the membership functions.

The selection of δ_f and δ_p is done in [35] in the following way:

$$\begin{array}{c} \delta_f(d) = (q \times d)^p \\ \delta_d(d) = \frac{1}{\delta_f(d)} \end{array} \tag{3.39}$$

where d is the distance from the point to the centroid, p is related to the softness of the cluster (softer as p decreases) and q to the size (bigger as q decreases). The tuning of those parameters is, in general, the larger drawback of this algorithm.

◇ **NN + Fuzzy Clustering**: Identification systems based on neural networks and fuzzy clustering algorithms, [38].

This is one of the most recent algorithms for multimodel identification using clustering techniques.

As most of the current clustering algorithms depend on the setting of the right number of clusters, this work offer an approach which first step is to determine c with a two-layer neural network and Rival Penalized Competitive Learning (RPCL). RPCL is an unsupervised learning strategy that automatically determines the optimal number of nodes. Referring to the learning results, the number of clusters is determined manually: the number of clusters could be equal to the number of units retained.

The second step is to apply a clustering algorithm to determine the operating clusters. This approach offers the possibility to choose between

FCM and $k-means$ algorithms, depending on the nature of the clusters, if there exist some overlap or not, respectively. After applying the algorithm, the c clusters are formed, using input-output information.

The next step is to find the model order. Usually this step is done with information of the system, but the method here can be useful if no information is available. It is based on *determinants' ratio test*. For every m matrices Q_m and Q_{m+1} have to built and RDI has to be computed:

$$Q_m = \frac{1}{N_H} \sum_{k=1}^{N_H} \begin{bmatrix} u_k \\ u_{k+1} \\ \cdots \\ u_{k+m-1} \\ u_{k+m} \end{bmatrix} \begin{bmatrix} y_{k+1} & u_{k+1} & \cdots & y_{k+m} & u_{k+m} \end{bmatrix} \quad (3.40)$$

$$RDI\,(m) = \left| \frac{\det\,(Q_m)}{\det\,(Q_{m+1})} \right| \quad (3.41)$$

The best structure is the one for which RDI quickly increases for the first time. After the structure is found, the parameters of the model are computed using Recursive Least Squares (RLS).

The final step is to find the validity functions (membership). In this case membership values are estimated with the residues's approach, based on the inverse of the distance between real and estimated outputs. To have a total membership of one, this membership are normalized by the total membership.

Although other algorithms can be found in literature [101, 128, 76], the above described ones are the most relevant in the field for the aim of this work.

3.4.1 Advantages and drawbacks of these algorithms

In the review of the state of art of clustering algorithms for systems modeling, the advantages and drawbacks have been briefly mentioned. Here these points are highlighted to have them clear.

Advantages:

1. All these algorithm use **linear structure** for the local models. Linear models are easy to interpret, even in high dimension (hyperplanes).

Besides, only one parameter by each input plus the independent term have to be found.

2. They use an iterative scheme to find the optima of the problem. This makes the algorithms to be **fast** computing the parameters of the problem.

3. The flexibility of the membership values helps to reduce the error of the model.

Drawbacks:

1. Most of the algorithms are fuzzy. Possibilistic strategy is not well explored yet.

2. Membership functions are **so flexible** in some cases that can model the process by themselves, instead of with conjunction to the local model structure. This is not the desired case.

3. This **irregularity** of MFs makes them not very interpretable, as many changes of value can be present.

4. All these methods assure convergence to a solution, but can get stuck in **local minimum**.

5. The iterative optimization method, even though fast, is **not the optimal** method.

The advantages of local modeling mentioned in 3.1.1 and the advantages of these algorithms indicate that the use of linear structures are powerful and would enhance flexibility and interpretability of the resulting model.

On the other hand, the iterative optimization is not the best option when a very accurate solution is desired, as the result can be a local minimum. Also, the flexibility of the value of MFs in each point does not help the interpretability of the total model, as MFs can have a very irregular shape. When certain properties are set for the MFs (like in [34, 35]), there might exist a small lose of accuracy, but gaining interpretability of the system.

3.5 Conclusions

Given the hypothesis that the glucose transport between interstitial and plasma fluid can be defined by several local dynamics, the local modeling

theory is to be reviewed, see previous Chapter, Section 2.4. Thus, this chapter goes through the advantages of local modeling and the possible techniques for performing it.

Clustering is chosen due to its capabilities of finding automatically groups with similar characteristics. This chapter then reviews the basic and general concepts of clustering techniques, some necessary steps for data processing have been defined and a few relevant algorithms have been described. In addition to this, some relevant algorithms for clustering are described.

Later, this chapter is focused on the application of clustering techniques to the concerned case: the local modeling of systems. Basic concepts in this applications are included along with the state of art of clustering algorithms for systems modeling. Advantages and disadvantages of existing algorithms are analyzed.

Some of the drawbacks of the existing algorithms are the lack of interpretability of the resulting MFs and that solution found might not be the best one as they can get stuck in a local minimum in the optimization of the cost index.

In Chapter 4 a new algorithm is developed accounting for the desired characteristics for the glucose transport modeling, which is the focus of this work.

To conclude this chapter the power of clustering technique for modeling complex system has to be remarked. Also, that it is very common, that for a new problem where new specifications are required the existing algorithms do not perform exactly as required, and therefore, a new algorithm has to be designed, implemented and validated.

Chapter 4

A new approach for local modeling: PNCRM

4.1 Introduction

The aim of this thesis is, as expressed in the introduction, to design a methodology to improve the current accuracy of Continuous Glucose Monitoring devices. Nowadays, the CGMS devices available in the market do not have enough accuracy to be used by they own, losing much of the potential they could have in the diabetes treatment.

As reviewed in the previous section, clustering techniques are a powerful tool capable of extracting the most important features of a data set. Regarding to modeling applications, clustering is also a powerful and flexible tool which allows to represent the relationship between inputs-output of the system with several simpler structures avoiding a more complex unique model.

For this reason, clustering has been considered as an appropriate approach for the CGMS case, which is considered complex system, with unknown relationship between inputs and output (glucose estimation levels), where local behaviors due to different metabolic states or population characteristics may be suspected.

In the concerned case, the aim is to model the interstitium-to-plasma glucose transport and the later application of this model as calibration algorithm and its inclusion on a CGM device.

A desired characteristic in this model is the *interpretability* of the global model and also of the local behaviors identified, to be able to interpret the underlaying physiological process that is taking place. This interpretability can be translated into local models that are defined in a certain region of the

input space and that are independent (each one defined by some characteristics of the inputs). On the other hand, independent models can be translated in a possibilistic clustering algorithm where membership values are not restricted.

In Section 3.4 the most relevant of the existing clustering algorithms for system modeling has been reviewed. Those algorithms were designed with a general purpose and each of them has particular characteristics, see Table 4.1.

Table 4.1: *Characteristics of already designed clustering algorithms for local modeling*

Algorithm	**FCM**	**PCM**	**FCRM**	**AFCR**	**AFCRC**	**TSP**	**NN+ Fuzzy**
Prototype	Point	Point	Linear Regression	LR	LR	LR	LR
Type of membership	Fuzzy	Possib.	Fuzzy	Fuzzy	Fuzzy	Possib.	Fuzzy
Interpretability	Medium	Medium	Low	Low	Medium	Large	Low
Consideration of Local error in cost index	No	No	No	No	No	No	No

If the characteristics shown in Table 4.1 are studied, it is clear that there is a gap in algorithms with linear regression models (capable to detect dynamics of inputs) as representatives of the clusters with low local error, high interpretability of local models and that use independent membership functions, which are the desired characteristics in the model of glucose transport. In addition, any of the reviewed algorithms does not directly consider the local error for the identification of the set of local models. This means that each of the local model found might not be, by itself, representative of the region where it is valid, not contributing to the interpretability of the local model. For these reasons, in this work the design of a new clustering algorithm for modeling is proposed. The motivation for this design is to develop an algorithm specific for glucose transport, taking into account the main properties known about this system.

This chapter goes over all the design steps. Firstly, the characteristics of the glucose transport are considered, defining in more detail the desired properties of the clustering algorithm. Later, there is a description of the mathematical building of the cost index of the clustering algorithm, which includes the desired characteristics. Finally, the general performance of the index is tested and analyzed with a set of benchmarks. Integration of the technique into a calibration algorithm will be the topic of next Chapter.

4.2 Characteristics desired to be included in the glucose transport model

Interstitium-plasma glucose transport is a physiological process not yet well known. For this reason its modeling is so difficult. Local modeling by means of clustering is the chosen approach to model it, thus, the global model will be formed by several local models. These local models can correspond, it is not known yet, to different levels of glucose, different rates of change, or other things.

To include more flexibility in the system, a **non-exclusive membership** of the objects seems to be more adequate, see Section 3.2.5. Indeed, as the criteria to separate the clusters is also unknown, a general approach is desired, where the membership value of each object to a particular cluster is not dependent on the membership values (MVs) to the other clusters. Thus, the **most general case** will be included, which is to consider that the criteria to classify each local model (LM) is independent and an object can meet more than one criteria with different or the same level of importance. This is what is called **possibilistic** clustering.

The computed output will be a combination of the outputs of the local models, in a similar way to fuzzy clustering. However, now the addition of weights for each cluster (membership) will not necessarily be restricted to adding the unity. Thus, an extra degree of freedom exists for the MVs of each object, being able to detect independent clusters and outliers.

The construction of the output by the integration of several local models does not mean that each local model is valid by itself, because it may happen that the total integration accurately describes the output yet does not necessarily include good local models. In glucose transport modeling it is desired that besides the good estimation of the global model, each one of the LMs is able to estimate with enough accuracy the output of the system in the region where it is representative for interpretability of the underlying process.

Imposing a particular shape for the membership bounds the region where the model is valid to just one particular vicinity of the input space, acquiring this desired local validity and interpretability of the model. Thus, in this case the **interpretability** means that local models are valid in compact regions, given that in that way the characteristics of the region will be easier to understand than if points are dispersed. This characteristic is described in more detail in the next section.

Another drawback of some of the existing algorithms for CGMS is the difficult set up they have (see Appendix 1 of [86]). This is, in most cases, due

to the large amount of parameters to be initialized. If an algorithm is to be implemented in a CGM device, it has to be easy to tune and implement. The proposed algorithm will have just few parameters to be defined, and its fixed value (for all cases) will be set in this work, avoiding the complexity of the set up.

4.2.1 Pre-fixed shaped membership function

Many clustering algorithms have MFs computed according to the characteristics of the index. These can be: distance to center, output error, a combination of both, and so on. This means that an object can have a certain value of its membership for each cluster and the object next to it can have a quite different membership value. This means that the membership functions may have very non-convex irregular shapes.

Although this high flexibility makes the error of the clustering process small, the interpretability of the models and the validity region of clusters are quite complex to understand [34]. As a result, finding a real interpretation of the functions when they are so irregular is very difficult, because the models (or the spaces) the user can understand are simpler (geometrical models or bounded regions).

Another problem associated with highly flexible MFs is that they can model the process by themselves, instead of only reflecting the validity of each local model. Much more in the possibilistic approach in which more flexibility is included.

In order to add interpretability to the division carried out by the algorithm and the shapes of the MFs, a predefined shape for the MFs will be adopted. This also avoids the problem of modeling the process through the MFs themselves.

Gaussian-like functions

The shape of the functions could be any, but a good approach might be [107] to consider them as "normal distributions"[1], i.e:

[1]Normal distributions are so called in statistics. In this case the function used is the same, but it is not used as population distribution. Calling it *distribution* is an abuse of language for this case. For this reason it will be renamed as *gaussian-like functions*. As it is not a statistical distribution the use of the terms *mean* and *variance* is not rigorous in this case. Yet, abusing of language, these terms will be adopted to define the parameters of the function.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(-\frac{(x-m)^2}{2\sigma^2}\right)}, \qquad x \in \mathbb{R} \tag{4.1}$$

which is one of the most general and well known distributions. This equation represents the unidimensional case, shown in Figure 4.1 (example of gaussian-like functions with null mean and unity variance), where x is the new datum, and m and σ^2 are the mean and the variance of the distribution, respectively.

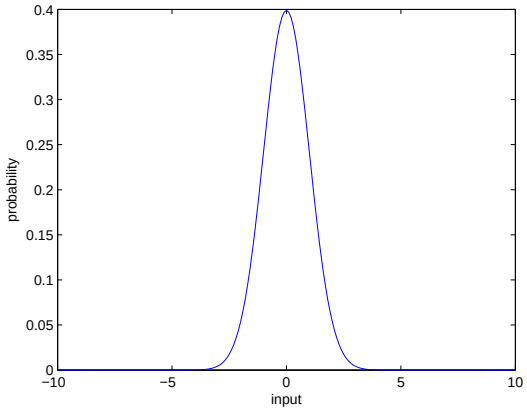

Figure 4.1: *Example of Gaussian-like function with null mean and variance unitary*

There are many reasons to choose this type of functions:

- ◇ The Gaussian-like function offers good mathematical properties: namely, continuity in the input space which involves differentiability.
- ◇ The function has different regions, mainly two, one with a high value and another with a low value. This could represent the object's levels of membership. Transition between two regions can represent elements that have a middle status.
- ◇ Only two parameters need to be defined σ and m.
- ◇ The shape is fixed but flexible: various regions can be covered by changing σ and m.

- ◇ The transition zone is smooth, which suggests that elements close to the high membership will belong to that cluster in a similar way to the neighboring objects. This makes sense in real applications where a *close* input has a *close* output, which is the fundamental idea of clustering techniques.

- ◇ The Gaussian-like function is very easy to extend to the multidimensional case:

$$f\left(\vec{x}\right) = \frac{1}{(2\pi)^{D/2}\left|\vec{\Sigma}\right|^{1/2}} e^{\left(-\frac{1}{2}(\vec{x}-\vec{m})^T\vec{\Sigma}^{-1}(\vec{x}-\vec{m})\right)}, \qquad \vec{x} \in \mathbb{R}^D \tag{4.2}$$

  where D is the number of dimensions, $\vec{\Sigma}$ is the matrix of variances, where each element of the diagonal (Σ_{ii}) corresponds to σ_i^2 and the rest are zeros, $(\vec{x} - \vec{m})$ is a vector of length D, and $\left|\vec{\Sigma}\right|^{1/2}$ is the square root of the determinant of the matrix of variances (i.e. product of variances).

- ◇ The Gaussian-like function is continuous but can be defined only for certain points, so it can take a certain value for each object of the database. In this way, it is not necessary to interpolate the signal for points not defined within the interval, because its value can be computed directly.

Gaussian-like function, both in its unidimensional and multidimensional versions, has a multiplying factor for the exponential term. This factor depends on the variance and makes that the total area covered by the function is equal to the unity. This is because in statistical analysis it is important that the total probability is equal to one. Thus, the wider the function (larger σ), the shorter the peak will be. Figure 4.2 shows a normal distribution with $m = 0$ and $\sigma = 3$. It can be seen how the peak changes when the adopted σ of Figure 4.1 is modified.

In the considered case, however, this type of function is used to represent the membership of a cluster. It is necessary that their values span from zero (object does not belong to that cluster) to one (objects perfectly meets the characteristics described by that group). In this way, the Gaussian-like function will be normalized, meaning that the multiplying factor is removed from the original equations. The final equations for the unidimensional and the multidimensional cases are then:

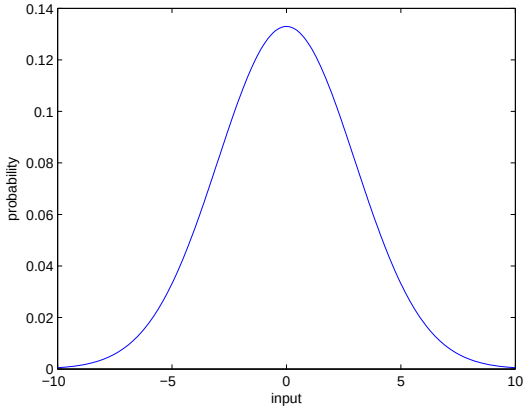

Figure 4.2: *Gaussian-like function with null mean and* $\sigma = 3$

$$f(x) = e^{\left(-\frac{(x-m)^2}{2\sigma^2}\right)}, \qquad x \in \mathbb{R} \tag{4.3}$$

$$f(\vec{x}) = e^{\left(-\frac{1}{2}(\vec{x}-\vec{m})^T \vec{\Sigma}^{-1}(\vec{x}-\vec{m})\right)} = \prod_{i=1}^{D} e^{\left(-\frac{1}{2}\frac{(x_i-m_i)^2}{\sigma_i^2}\right)}, \qquad \vec{x} \in \mathbb{R}^D \tag{4.4}$$

Figure 4.3 shows the normalized version of the MF in Figure 4.1, with $m = 0$ and $\sigma = 1$.

Once the membership functions are similar to this Gaussian-like function they will lose some detail and accuracy, but they will describe **compact regions** where a model is valid, regions where the model does not describe the objects, and in between, a smooth transition in level of description.

Transition zone

Figures 4.1 and 4.2 show that, depending on the variance, the transition zone is sharper or softer. This can greatly affect the membership function. A sharper transition means that the cluster covers only a small region and its capability of description immediately disappears. A softer change means that the objects next to the high values of the function are also to some degree described by that particular cluster prototype, covering wider regions.

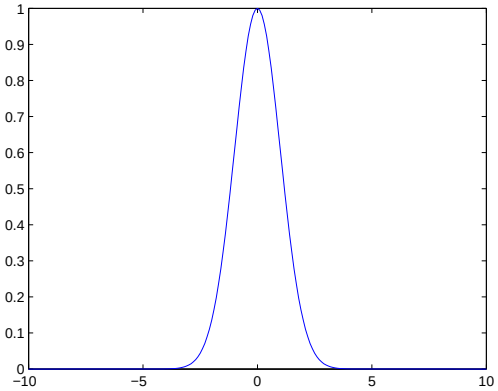

Figure 4.3: *Normalized Gaussian-like function with* $m = 0$ *and* $\sigma = 1$

It has already been pointed out that the transition depends on how large σ is for that Gaussian function and this will force some characteristics in the resulting clusters. Therefore, it would be a good idea to introduce a term in the function to obtain the desired characteristic of compact clusters.

The concept of **kurtosis** [32] is related to sharpness of the transition zone. There are positive and negative kurtosis. *Positive* kurtosis means that the transition of the function is more abrupt compared to a Gaussian function, while *negative* kurtosis means a smoother transition.

To modify the transition zone of the MF, only a parameter (H) in the exponent of the exponential function must be introduced, i.e. in the unidimensional case:

$$f(x) = \mu = e^{-\frac{1}{2}\cdot\left(\frac{(x-m)^2}{\sigma^2}\right)^H}, \qquad x \in \mathbb{R} \tag{4.5}$$

For the multidimensional case the term H is applied to the resulting term of all additions of unidimensional cases:

$$f(\vec{x}) = \mu = e^{\left(-\frac{1}{2}\left((\vec{x}-\vec{m})^T\vec{\Sigma}^{-1}(\vec{x}-\vec{m})\right)^H\right)}, \qquad \vec{x} \in \mathbb{R}^D \tag{4.6}$$

If $H > 1$ kurtosis will be positive. Sometimes, the resulting function is called *hyper-Gaussian*. While when $1 > H > 0$ kurtosis will be negative or *sub-Gaussian*. Figure 4.4 shows how the transition of the function with $m = 0$ and $\sigma = 1$ varies for different values of $H \in \{0.1, 0.5, 1, 2, 8\}$ for the

unidimensional case. Note that given that all terms of the multidimensional function are quadratic terms, the introduction of H to each of them and the addition or the application of H to the added results will result in both cases in a positive kurtosis.

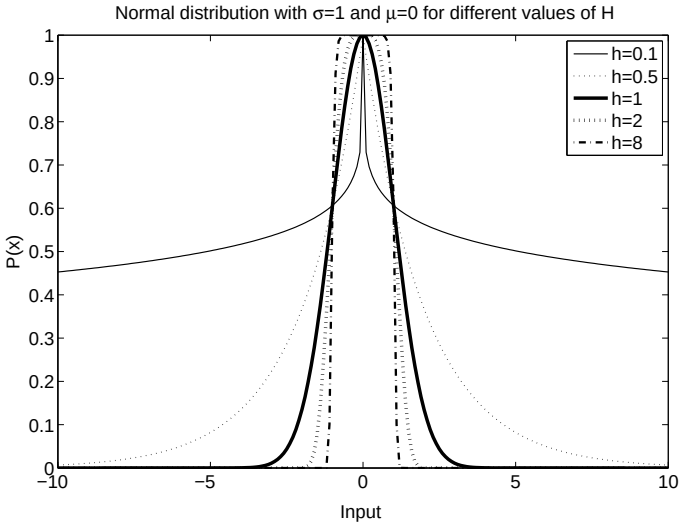

Figure 4.4: *Normalized hyper-Gaussian function with $m = 0$ and $\sigma = 1$ for different values of H*

The introduction of parameter H in the normalized Gaussian-like function means the transition of the membership of the clusters can be modified. If the clusters are then more independent a positive kurtosis will make their membership functions valid in a more compact region.

Besides, the larger the kurtosis is, the wider the peak will also be. Thus, a large H value will introduce a wider covered region, for a fixed σ value.

4.3 New cost index definition

As said before, the cost index has to reflect, in a mathematical way, the characteristics desired for the clusters or local models in this case. For this reason the new cost index has been built taking into account all the considerations about the shape of the MFs, the type of local models and their integration exposed as the desired properties of the resulting models.

The output of the global model (GM) will be a weighed combination of the linear local models. Many model structures could be adopted for each LM, but linear models have been chosen due to their interpretability, even when the dimension of the input space is high. To define each linear LM a vector of regression parameters ($\vec{\beta}_i := (\beta_0, \beta_1, ..., \beta_D)_i$, where D is the dimension of the input vector) that best defines the output from the inputs will have to be found. The form of these models is:

$$f_i\left(\vec{x}_k; \vec{\beta}_i\right) = LM_i = \left(\vec{\beta}_{1D}\right)_i \cdot \vec{x}_k + (\beta_0)_i\,, \qquad i = 1, ..., c \tag{4.7}$$

where $\vec{\beta}_{1D} := (\beta_1, ..., \beta_D)$, while β_0 means the independent term or coefficient 0 of the regression vector. Eq. 4.7 shows the estimated output of the object k for cluster i. The parameter vector length is $D + 1$, where D is the length of the input vector.

For each cluster i, parameters $\vec{\beta}_i$ will have to be found to describe the output of that cluster properly. So, for the whole data set, a set of c linear models will be defined, where c is the number of clusters. In addition, for every object x_k, the degree of membership for each model will be computed. Thus, the global output for each object $\vec{x}_k$, $\hat{y}_k$, will be the addition of all the estimations of local models, weighted by the value of the membership function for that object to each cluster:

$$\hat{y}_k = \sum_{i=1}^{c} \mu_{ik} \cdot \left(\left(\vec{\beta}_{1D}\right)_i \cdot \vec{x}_k + (\beta_0)_i\right) \tag{4.8}$$

where μ_{ik} denotes the membership value of object k to cluster i.

The goal of most indexes in regression applications is to minimize the global error of the model [62]. In this case, however, the goal is slightly different. Obviously, the desire is to make the global error of the total model very small, but a small error for every local model is also desired. This is because the index is desired to identify interpretable models (valid regionally by themselves) to explain the physiological process.

Thus, in the index, a term will be included to consider the difference between the actual output and each local model, i.e. **local error**, and another term to include the difference between the real output and the global prediction of each object, i.e. **global error**. As the index will be possibilistic, there is no extra term regarding any restriction on the total value of the membership functions. This is contrary to the fuzzy approach [2]. Given that the **cost function** is composed of errors, it will have to be **minimized**.

In clustering applications, it is very important to consider the closeness of objects. In many indexes [56, 90] a term in the index is included to penalize long distances and so boost the formation of compact clusters. In the considered case, although close objects are desired in the input space, there is no need to add any extra term. The shape of the membership functions considers the distance by itself: objects close to the center will have high membership ($e^0 = 1$) while objects that are far from the center of the cluster will have almost null membership.

Finally, a factor for modifying the importance of each term in the index is considerer, γ. This factor will have to be tuned depending on the application to give higher importance to the local error ($\gamma > 1$), to the global error ($0 < \gamma < 1$) or to both ($\gamma = 1$).

In equation 4.9 the index including the two errors and the tuning factor can be seen (coined as **P**ossibilistic **N**ormalized **C**-**R**egression **M**odels or **PNCRM**). Function f_i represents the function of the linear model for cluster i, always for an object k, (equation 4.7); while $\vec{X}$ is the matrix of inputs where each row is an object and each column is one dimension. n denotes the number of objects, so $\vec{X}$ is a matrix $n \times D$.

$$\begin{aligned} J_{PNCRM}\left(\vec{X};\vec{\sigma},\vec{p},\vec{\beta}\right) = & \sum_{k=1}^{n}\left(y_k - \sum_{i=1}^{c}\mu_{ik}\left(p_i,\sigma_i^2\right)\cdot f_i\left(\vec{x}_k;\vec{\beta}_i\right)\right)^2 + \\ & +\gamma\cdot\sum_{k=1}^{n}\sum_{i=1}^{c}\mu_{ik}\left(p_i,\sigma_i^2\right)\cdot\left(y_k - f_i\left(\vec{x}_k;\vec{\beta}_i\right)\right)^2 \end{aligned} \tag{4.9}$$

where: $\vec{x}_k$, $k = 1, 2, ..., n$, are the objects (row k in $\vec{X}$) and p_i, σ_i^2 and $\vec{\beta}_i$ are the center, the variance and the vector of regression coefficients for cluster i.

When the normal multidimensional distributions are introduced in the previous equation, the final index is obtained:

$$\begin{aligned} & J_{PNCRM}\left(\vec{X};\vec{\Sigma},\vec{p},\vec{\beta}\right) = \\ & \sum_{k=1}^{n}\left(y_k - \sum_{i=1}^{c} e^{-\frac{1}{2}\cdot\left((\vec{x}_k-\vec{p}_i)^T\vec{\Sigma}_i^{-1}(\vec{x}_k-\vec{p}_i)\right)^{H_i}}\cdot f_i\left(\vec{x}_k;\vec{\beta}_i\right)\right)^2 + \\ & +\gamma\cdot\sum_{k=1}^{n}\sum_{i=1}^{c} e^{-\frac{1}{2}\cdot\left((\vec{x}_k-\vec{p}_i)^T\vec{\Sigma}_i^{-1}(\vec{x}_k-\vec{p}_i)\right)^{H_i}}\cdot\left(y_k - f_i\left(\vec{x}_k;\vec{\beta}_i\right)\right)^2 \end{aligned} \tag{4.10}$$

These equations can be summarized as:

$$J_{PNCRM}\left(\vec{X};\vec{\sigma},\vec{p},\vec{\beta}\right) = \sum_{k=1}^{n}\left(y_k - \sum_{i=1}^{c}\mu_{ik}\cdot LM_{ik}\right)^2 + \\ +\gamma\cdot\sum_{k=1}^{n}\sum_{i=1}^{c}\mu_{ik}\cdot(y_k - LM_{ik})^2 \tag{4.11}$$

This defines the final cost index to be minimized for the new approach PNCRM.

Both x_k and y_k are known for the process, while p_i, σ_i^2 and $\vec{\beta}_i$ are the parameters to be found to characterize the model of a particular system.

Depending on the dimension of the input vector, $\mathbb{R}^D$, each center, variance, and regression vector will have a different length. The length of the first two factors is D while the length of the third is $D+1$ (always including an extra coefficient for the independent term). Therefore, given an application with c clusters and D inputs, the total number of variables to be found is $nv = c\cdot D + c\cdot D + c\cdot(D+1) = c\cdot(3D+1)$.

4.3.1 Parameters tuning

Before the application of the proposed algorithm to any systems, the defined general parameters will have to be set. In this case, only two parameters have been defined: the MF transition H and the balancing factor between errors γ. These parameters have been set to meet the desired characteristics of the problem under consideration:

- ◇ *MF transition* (H): This parameter defines the sharpness of the membership functions. It can be set individually for each μ (H_i) or globally. In this case it will be considered as a global parameter, constant for all μ, and it has been set to $H = 6$. In this way, the local models will be valid in a well-defined compact region of the input space, i.e. **interpretable**, which is one of the desired characteristics for the resulting models.

- ◇ *Weighting factor* (γ): in this study, the output of the index should have a set of local models that are valid in a region by themselves. For this reason, this parameter will be set to $\gamma = 5$, to give the term of local errors higher importance given the desire of having **independent** clusters (meaning that one LM can describe by itself a region of the system, without being combined with other local models). This will also help the interpretability of the resulting global model.

For other applications, different values could be used. Nevertheless, in this work, these values will be constant.

A summary of the main characteristics of PNCRM algorithm is pointed out in Table 4.2.

Table 4.2: *Main properties of PNCRM*

- ◇ Application of **possibilistic** approach.
- ◇ Use of **pre-fixed** shape for **membership** functions.
- ◇ Use of **linear** structure for **local models**.
- ◇ Consideration of **both**, global and local **errors** for model fitting.
- ◇ Parameters set to found independent models for **compact regions**.

4.4 Iterative index optimization

Once the index has been defined, the method by which it will be minimized must be considered. In many applications [114], [61] the procedure to find the minimum of the cost index is to randomly select an initial point and iteratively find the optimum value for the different unknown variables. The search is often divided into the different variables that have to be found and each is independently computed, keeping the others in the optimum value taken from the previous iteration. In [17] it is shown how this procedure converges to **a** minimum of the function.

To perform this iterative search the index has to be differentiated with regard to each of the unknown variables. In some cases [114], the clusters are placed and then the parameters of the models are found by least squares taking into account the objects belonging to each cluster. In this way, some parameters (β vectors) are not optimized, but computed.

In PNCRM, it is desired to found **all** optimal **parameters**, for this reason the optimization of the index will be done with regard all parameters that define the model: **centers**, p_i, and **variances**, σ_i^2 for membership functions and **regression coefficients**, $\vec{\beta}_i$, for local models.

Thus, if the iterative procedure applied in [114] is followed for this case, then equation (4.10) must be differentiated with regard to each of these

independent variables. Derivatives with respect to the centers p_i and variances σ_i^2 for the unidimensional case result in:

$$\frac{\partial J}{\partial p_i} = 0 = \cdot \sum_{k=1}^{n} \left(\begin{array}{l} \mu_{ik} \cdot H \cdot \frac{(x_k - p_i)^{2H-1}}{\sigma_i^{2H}} \cdot \\ \cdot \left\{ 2 \cdot f_{ik} \left(\sum_{i=1}^{c} (\mu_{ik} \cdot f_{ik}) - y_k \right) + \gamma \cdot (y_k - f_{ik})^2 \right\} \end{array} \right) \tag{4.12}$$

$$\frac{\partial J}{\partial \sigma_i} = 0 = \frac{1}{\sigma_i^{2H+1}} \cdot \cdot \sum_{k=1}^{n} \left(\begin{array}{l} \mu_{ik} \cdot H \cdot (x_k - p_i)^{2H} \cdot \\ \cdot \left\{ 2 \cdot f_{ik} \cdot \left(\sum_{i=1}^{c} (\mu_{ik} \cdot f_{ik}) - y_k \right) + \gamma \cdot (y_k - f_{jk})^2 \right\} \end{array} \right) \tag{4.13}$$

where μ_{ik} is the membership function, Equation (4.5), and f_{ik} is the local model of cluster i for object k, Equation (4.7).

A third step would be to differentiate equation (4.10) with regard to the vector of regression coefficients ($\vec{\beta}_i$). Yet, from (4.13) it can be seen how there is no a direct solution for obtaining the parameters σ_i, even in the unidimensional case. Therefore, it is not possible to obtain the optimum of the system in this iterative way.

Thus, for this index a different way of finding the optimum value should be defined. For this purpose, a review of the optimization methods defined in literature is done in this section.

4.5 Optimization methods

In general terms, **optimization** might be defined as the science of determining the **best** solutions to certain mathematically defined problems, which are often models of physical reality [45, 115]. These problems can include constraints or not. Constraints are defined as some requirements, usually boundaries, the inputs or the outputs of the system have to meet. Optimization can refer to the search of a maximum or minimum of the function.

It involves the study of optimality criteria for problems, the determination of algorithmic methods of solution, the study of the structure of such methods, and computer experimentation with methods both under trial conditions and on real life problems. There is an extremely diverse range of practical applications.

Before 1940 relatively little was known about methods for numerical optimization of functions of many variables. Some of the first developed

methods were the *least squares* calculation of optimum variables, *the steepest descent* and the *Newton method*. Nonetheless anything of any complexity demanded armies of assistants operating desk calculating machines. The 1940s and 1950s saw the introduction and development of the very important branch of the subject known as linear programming. All these methods however had a fairly restricted range of application and rely on a special structure in the problem.

After the 1960s research on this area started focusing on solving complex problems, as the reality showed that the real cases where, most of the times, complex cases to optimize, with several variables and not always objective function has a determined structure.

The applicability of optimization methods is widespread, reaching into almost every activity in which numerical information is processed: science, engineering, mathematics, economics, commerce, etc.

It is very important that the user discovers the structure of the problem and then the appropriate optimization method to implement on a computer.

One of the first thing to consider is if the problem has several objective functions or only one. In the case of multi-objective problems the optimization is much more complex and different strategies have to be used, called evolutionary algorithms: *generic algorithms*, *swarm theory*, *evolution strategy* and so on [10].

Another thing to consider is whether the problem is constrained or not. A function with no constrains can happen not to have a global optimum as it can constantly increase or decrease. However, if only a region of the input space is considered then, the function for sure will have an optimum value. Constraints can be present in the value of the inputs (interval of possible values), or in the output or even can reflect relations between variables. The common case in optimization is to have a problem with restrictions.

In mathematical term, optimum is defined as point where the first and second derivative of the function is equal to zero, which means that the slope of the function at that point is zero and it is not a saddle point. However, this description correspond to local minimum (or maximum) of the function and this may not be a global minimum (or maximum).

This fact is very present in optimization problems. Many algorithms are useful to converge to the minimum value of the function, starting from a random point. However, the valley found might not be the one containing the global optimum. The only simple advice in practice (not guaranteed to work) is to solve the problem from a number of different starting points and take the best local solution that is obtained [45, 83].

Local optimizers use much simpler methodologies than global optimizers, where not only the direction that minimizes the function is followed (as in the local optimizers) but also some method of searching the rest of the space. Here, the attention will be focused on the local optimizers. The reason for this is that these algorithms are less complicated to implement than the global optimizers, that perform better but need more powerful systems to work and longer time to compute the optima of the objective function.

Local search algorithms are faster to find the solution and need less requirement in software for running. The price is the issue of getting stuck in a local minimum. This can be solved, or at least in many cases, with the initialization of the function in several points and the selection of the one that performs better.

4.5.1 Computational optimization techniques

The first algorithms in this field used the strategy of generating points (randomly or with some sort of pattern) and compare the value of the function at each of them, choosing the one with minimum value, [122]. The problem of this type of methods is that the amount of effort required to implement these methods goes up rapidly (typically as 2^n, with n number of variables). The most successful of the methods which merely compare function values is that known as the **simplex method**. Another simple method which readily suggests itself is the **alternating variables method**, although in practice is very inefficient. The **conjugated direction methods** have result more efficient than the ones mentioned before.

However, in this line of research soon started the development of **iterative optimization methods**, [45]. Most of the them (if not all) are based on following the direction where the derivative of the function is negative and, therefore, the objective function tends to a minimum, which is what is desired. The typical behavior of an algorithm which is regarded as acceptable is that the iterates x_k move steadily towards the neighborhood of a local minimizer x^* and converges, finally, to it.

The steps for each iteration are three:

1. Determine direction of search s_k
2. Find the proper update step α_k.
3. Set $x_{k+1} = x_k + \alpha_k \cdot s_k$

Different approaches have been proposed for step 1. Some of the most important are:

- ◇ The **Newton's** method use information of the first and second derivative.
- ◇ The **Quasi Newton's** methods use information of first derivative and approximation of the second.
- ◇ The **Gradient** based methods use information of first derivative only.

The main advantage of Newton's methods is that they are not sensitive of poor data scaling, and the main drawback is that they need more computations to be implemented.

For step 2 also some propositions have been made. The value is usually set as a trade-off between getting a substantial reduction of the objective function f and not spending too much time making the choice. Some metrics have been defined to set α_k, like *Wolve's* condition, *Goldstain's* condition, etc [102]. However, in many cases, the value of the adaptation step, α, has to be set according to the system properties where it will be used.

4.5.2 Global optimization

The variable-iterative method mentioned before and the local optimization techniques are quite fast, yet the solution they offer may be a local minimum. This is be due to the random initialization of the values and the non-convexity of the index to be optimized.

In a real problem to find the global solution of the problem is highly recommendable, specially in the interstitium-to-plasma glucose transport case. For this reason and because in this case the iterative method to solve the problem is not possible, as shown in the previous section, a global solver has been selected to find the optima of the index.

The fact that non-linear functions are included (gaussian-like membership functions) and that multiplication of square terms are included (MFs are weighting LMs) make the index non-convex. For this reason local optimization is not possible if the global minimum is the target of the optimization.

The most relevant state-of-the-art algorithms for global optimization can be considered five: ED [120], CMAES [60], SSMopt [36], DeMat [119], PSO [82]. In the corresponding references, a description of the theory underlying this algorithms can be found. Global optimization is quite time consuming and so it will be applied off-line.

To chose the best algorithm for global optimization for the concerned case, a comparative study has been done with the already mentioned global

optimization algorithms. Thus, a total of five algorithms have been studied in two case-studies. Comparison is done in next section.

4.6 Selection of global solver for PNCRM

The global optimizers chosen for this comparative study are: *ED*, *CMAES*, *SSMopt*, *DeMat* and *PSO*. The parameters to do the comparison are:

- ◇ Difficulty to run the optimizer algorithm.
- ◇ Numerical value of the obtained optimum of the cost function.
- ◇ Time to reach that value.

Case 1 corresponds to a system with two inputs vectors. The size of the input matrix is 2422×2 and the number of clusters is $c = 3$, which means a total of $nv = 21$ parameters. On the other hand, *Case 2* corresponds to a recursive system with three inputs vectors, where one of the inputs is the estimated output in previous instant. The size of the input matrix is 195×3, and the number of clusters is $c = 2$ which means a total of $nv = 20$ parameters. In Tables 4.3 and 4.4 comparative results among these optimizers are shown for both studied cases.

Table 4.3: *Comparative study for optimizers. Case 1*

Param	**ED**	**CMAES**	**DeMat**	**SSMopt**	**PSO**
Difficulty	Medium	Medium	Large	Medium	Large
Minimum	177.4	176.9	178.32	165.7	Not reached
Time (s)	167.1	139	1544	3232.7	–

Table 4.4: *Comparative study for optimizers. Case 2*

Param	**ED**	**CMAES**	**DeMat**	**SSMopt**	**PSO**
Difficulty	Medium	Medium	Large	Medium	Large
Minimum	192.4	30.07	190.51	29.87	Not reached
Time (s)	5131	94.14	4526	3760	–

The **PSO** algorithm is very complex to run. After many trials, it was not possible to get this algorithm performing properly any minimization of even simpler indexes. This complexity is the reason to discard this algorithm as the global optimizer for this problem.

DeMat algorithm is a version of original **DE** algorithms, based on *differential evolution* search method. This algorithm is also complex to run, as it has some parameters (weight of seed, cross-over importance, etc.) that are difficult to tune. In the studied cases, it can be seen how this algorithm performs quite well in *Case 1*, reaching a value quite close to the minimum value obtained for all algorithms. The problem is that the algorithm gets stuck in many of the attempts to get the minimum and this value was obtained after running the experiment many times. Besides, the computation time is quite long. In *Case 2*, despite the long computation time, the algorithm do not reach a value close to the global minimum obtained. In this case, all runs done reached a value close to the obtained one.

ED algorithm is also a version of **DE** strategy. It performs better than the previous ones in *Case 1*, with low computation time and a value close to the minimum one. Nevertheless, in *Case 2*, this algorithm does not reach a good value for the objective function, in spite of the long computation time. This might be due to the fact that *Case 2* is much more non-linear than *Case 1*.

Finally, in both cases **CMAES** and **SSMopt** algorithms performed much better, specially in *Case 2*. From these results it can be deduced that **CMAES** algorithm reaches a value near the optimum value in a short time, thing that is an advantage as, in general, global optimizers do need a large time to reach the global optimum. On the contrary, **SSMopt** needs a larger time to reach the optimum, but the reached value of the cost function is smaller than with the other algorithms, an important point as the total global minimum is the target of this optimization.

For these reasons, and based on these results. The final optimizer method chosen is a combination of these two algorithms. Firstly, **CMAES** will be run and its results will be taken as an starting point for **SSMopt**, run secondly. As results of the first algorithm are close to the real global optimum, much computing time will be saved. This is the best optimization structure to get the best the running time and the best value of the optimization. A description of both methods is included here.

With this combined structure, the optimal solution can be found in spite of the extended computing time. This index will be applied to off-line applications, and so the high computational cost of this type of optimization is not a limitation. Therefore, using the index defined in the previous section together with this global optimization method, the best parameters for the **centers**, **variances** and **regression coefficients** can be found to model a particular data set.

Once the optimal parameters are found, the model can be applied on-line as the output in a particular time instant is fast to compute when inputs and parameters of the model are known.

In the following the basic features of CMAES and SSMopt algorithms are included.

CMAES algorithm

Its name comes from **C**ovariance **M**atrix **A**daptation **E**volution **S**trategy [60, 98].

Search steps are taken by recombination of already evaluated search points and mutation. The mutation is usually carried out by adding a realization of a normally distributed random vector. A dynamic control of certain parameters of the normal distribution is of major importance in the evolution strategy (ES), this is called *self-adaptation*. The main objective of this self-adaptation of the mutation parameters is to achieve some invariance against certain transformations of the search space. In [60], it is shown that invariance is a fundamental condition for a successful and reliable adaptation. The parameters most generally used for the control of the mutation are the complete *covariance matrix*.

The steps of this algorithm are:

1. Sample a maximum entropy distribution: multivariate **normal distribution**.
2. **Rank solutions** according to their fitness: invariance to order-preserving transformations.
3. **Update mean** and **covariance matrix** by natural gradient ascendant, improving the "expected fitness" and the likelihood for good steps.
4. **Update step-size** based on non-local information: exploit correlations in the history of steps. This is to avoid premature convergence and to allow fast convergence to a minimum.
5. Repeat steps 3 and 4 until a tolerance of the minimum is achieved or the maximum number of iterations is reached.

CMAES is a very popular optimizer, and more details about it strategy can be found in [60].

SSMopt algorithm

This algorithm is based on **S**catter **S**earch **M**ethod [37, 36]. The scatter search method is a hybrid method that combines a global search and an intensification search (local). It is based on five steps:

1. Generate a collection of diverse trial solutions.
2. Transform a trial solution into one or more enhanced trial solutions.
3. Keep a certain number of the "best" solutions.
4. Work on the solutions to combine them and create new subsets.
5. Transform the subset into one or more combined solution vectors.

A more detailed description of the method, a schematic representation, and the advantages of this global solver are described in [36].

4.7 General performance of PNCRM

Before applying the new algorithm (PNCRM) to the modeling of the interstitium-to-plasma glucose transport process, its general performance will be checked on several systems and compared with other algorithms with similar target and applicability: AFCRC [34] and TSP [35].

The reasons to compare the new algorithm with these two algorithms is that, firstly, both are designed to find a set of weighted linear local models to represent the global output (i.e. the same target as the proposed algorithm). AFCRC has been chosen for the comparative study because, as shown in [34], this algorithm performs better than others such as FCM, FCRM, and AFCR. TSP is selected because it is possibilistic as the proposed algorithm. Given than TSP is a possibilistic algorithm and AFCRC is a fuzzy algorithm, the comparison with both will offer better conclusions about the performance of the new algorithm.

One consideration is that both AFCRC and TSP find solutions using the iterative method. This means that the solution obtained could be a local minimum. For this reason, the experiments including these algorithms will be run many times and the best solution will be taken. It is not possible to apply the global optimizer to these algorithms due to the large number of parameters that would have to be obtained: matrix of membership values, centroids and regression coefficients. Therefore, the algorithms will be compared in the way they are defined, each one with its own considerations.

A total of three experiments have been performed. Firstly, a couple of data sets with **one**-input - **one**-output structure are considered. Later, an experiment with several inputs and just one output is presented.

4.7.1 Considerations before application

Before the application of the proposed algorithm to a particular system, some considerations have to be taken in order to set the proper framework.

The first consideration is the parameters tuning. For the designed algorithm (PNCRM), they have already been fixed, see Section 4.3.1, and this step do not have to be carried out.

The second consideration is data normalization, as described in Section 3.2.6. Here, normalized inputs will have all the same statistical properties (null mean and unit variance).

Given that the inputs of the algorithms are in normalized dimensions, its resulting output will also be in these dimensions. Therefore, to get the denormalized output of the model, the inverse equation of normalization must be applied to the output signal.

Note: The denormalization of the contribution of each local model is not direct, as each of them can contribute in different proportions and the possibilistic approach does not limit the total contribution. Equations for correct denormalization of each local estimation can be found in Annex A.3, at the end of this work. The denormalization of local estimations can not be done using the inverse of normalization equation directly on each local model output. This is due to the type of normalization selected and the possibilistic approach used in this methodology.

The last consideration includes two points: 1) the number of clusters, c, to form the global model and 2) the inputs to be considered for the description of the output of the system.

These two things are considered together because the aim of modeling a system using local models is always to find the minimum number of local models that describe the system and for this, the inputs with more information about the output have to be found. In this way, the global model will be easier to interpret, as it will be simpler.

For a general system these two aspects have to be defined simultaneously to find the best model structure, as both number of clusters and relevant inputs are, generally, unknown.

Yet, in the examples described below, the inputs and output of the system are known, allowing this for the search of only the proper number of clusters c and better analyze the performance. It could be a good approach to run

the experiments with different values of c. Usually, the larger the number of clusters, the lower the error. However, the c which best balances the error and the number of clusters should be taken. A very large number of clusters must be avoided as the larger c the less interpretable the global model will be. There are some measures to indicate if the correct number of clusters has been chosen: see [18] for classical clustering; or [121, 39] for fuzzy sets. In these examples, the option of trying different values for c will be adopted, as their computations are fast and this option is available.

4.7.2 Unidimensional systems

Two unidimensional experiments have been performed. Results achieved with this algorithm will be compared with those from AFCRC and TSP in both cases. The first data set is artificial while the second is real.

PARABOLA

This test signal has been obtained applying the parabola equation to the input and adding some noise to the output in order to slightly distort the perfect response. The signal is represented in Figure 4.5, where u is the input in the range $u = [-1, 1]$ and the output y, corresponding to equation $y = u^2 + \epsilon$.

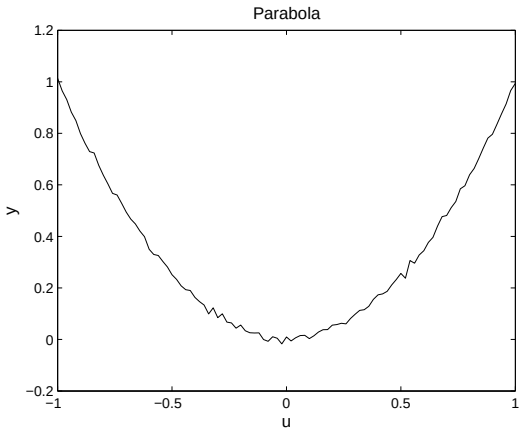

Figure 4.5: *Parabola with noise*

In this case, the number of clusters taken is $c = 3$, because it seems a good number of local models: one for the base and one for each of the branches.

Table 4.5: *MSE for parabola*

Algorithm	**PNCRM**	**AFCRC**	**TSP**
MSE	0.0013	0.0018	0.0026

This is a user decision based on knowledge of the system. Figures 4.6 to 4.8 show the results obtained by the designed algorithm. Fig. 4.6 shows both the global output of the model and the real output. Differences between both are very small, which indicates that the real output is well estimated with $c = 3$. In Fig. 4.7 each local model is separately represented in a plot. It can be seen how each LM fits the signal in a different region. Finally, in Fig. 4.8 the membership functions have been plotted: hypergaussian functions defined in different regions of the input space.

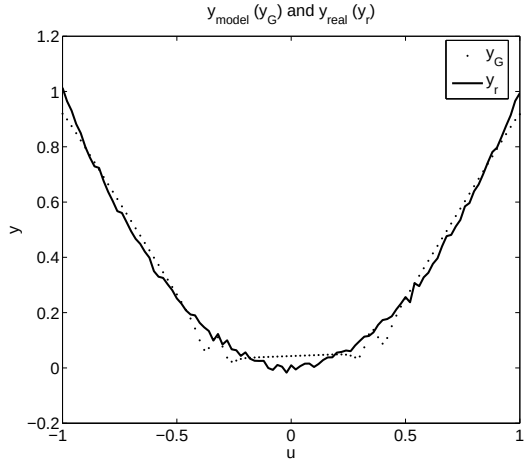

Figure 4.6: *Y_{real} and Y_{global} with PNCRM for the parabola*

The results are very similar when using AFCRC or TSP algorithms. Mean squared error (MSE) for all cases is indicated in Table 4.5. It can be observed that the error is quite small in all cases. This is because the linear models adjust the output in the three cases quite well, and the membership functions are well distributed.

MSE can be a good indicator of the error of the global model. Nevertheless, the aim of this work is to obtain a set of local models that represent the output locally. One indicator should also be included of how well these models adjust

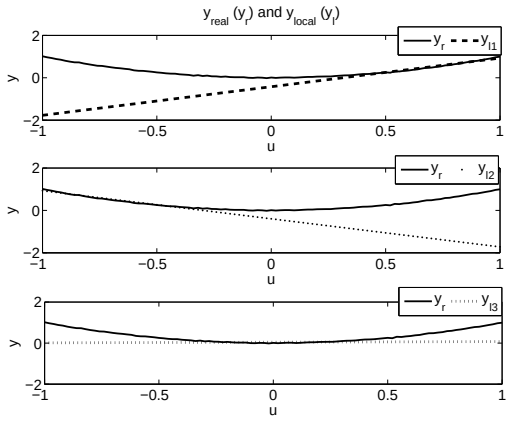

Figure 4.7: Y_{real} *and local models with PNCRM for the parabola*

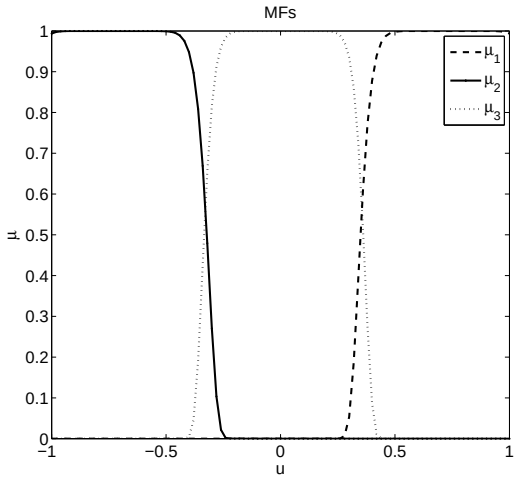

Figure 4.8: *Membership functions with PNCRM for the parabola*

the output in the region where they are valid. This region can be defined as the region where the MF has a value close to unity, which means total representativeness of the LM. In this case, the threshold $\lambda \geq 0.8$ will be taken.

The MSE can now be computed for each cluster, always considering the objects whose membership value goes beyond this threshold. Table 4.6 shows

Table 4.6: *Local MSE for parabola*

Algorithm	Charact.	Cluster1	Cluster2	Cluster3	Not Modelled
PNCRM	**LMSE**	0.0011	0.0012	0.0011	–
	%	30.7	32.7	31.7	4.9
AFCRC	**LMSE**	0.0004	0.0014	0.0013	–
	%	21.8	33.7	32.7	11.9
TSP	**LMSE**	0.0015	0.0014	0.0015	–
	%	34.7	34.7	33.7	0

MSE for local models and also the percentage of objects represented by each model ($\lambda \geq 0.8$), for PNCRM, AFCRC, and TSP.

MSE of a local model indicates how well local models represent the region where they are valid. Small local error is desired, besides small global error, for local models interpretability. Meanwhile, the *percentage of objects* represented by the LM indicates how large is the region covered by only that local model. As the characteristics desired in the resulting model are interpretability and independence, the region covered by each LM should be as large as possible, leaving the least percentage of objects represented by a combination of several LMs.

Obviously, if only a few elements are represented by a local model, then it will be easier that the model adjusts to them and the error is smaller. However, interpretability would be poor. So, it is desirable to get small errors and at the same time, large regions where the model is representative.

Local error is again very small for all algorithms, see Table 4.6. Comparing PNCRM with AFCRC, the local error is smaller for AFCRC in one of the clusters (#1) while the proposed algorithm works better in the other two (#2 and #3). Nevertheless, the percentage of objects represented with only one model is also smaller and this seems to be a reason for the error to be smaller. This means that PNCRM, although having a slightly larger error in one of the local models, represents a larger quantity of objects with only one model. Indeed, with PNCRM, only 4.9% of the objects are not fully represented by one of the local models, compared to 11.9% for AFCRC.

Comparing PNCRM with TSP, it can be observed that the local error for TSP is larger for all clusters. The percentage of objects well represented by only one cluster is also slightly larger and covers the entire set. Indeed, the total is larger than 100% because the MFs share zones with high value in several of them, as this is a possibilistic algorithm.

To better see this, in Figure 4.9 the MFs of AFCRC (right) and TSP (left)

have also been represented. It can be seen how the shape of the MF for AFCRC and TSP are quite clear and interpretable and also divided in different validity regions. Nevertheless, for the PNCRM, Fig. 4.8, the MFs are smoother and softer, and this contributes to the interpretability of the resulting model.

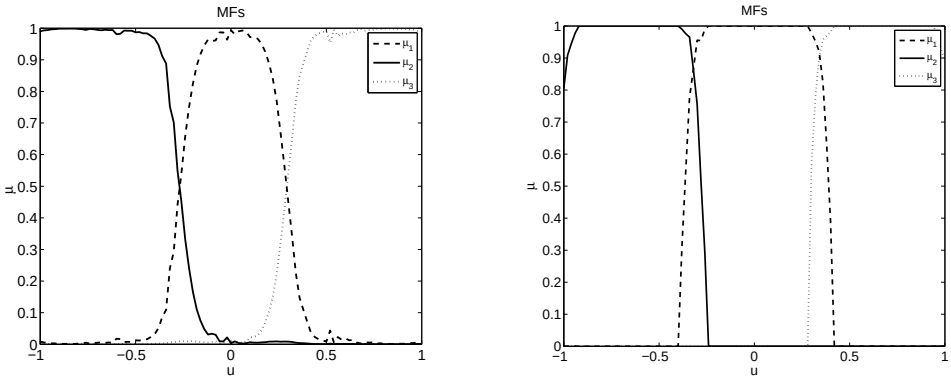

Figure 4.9: *Membership functions with AFCRC (left) and TSP (right) for the parabola*

It is important to indicate that to run AFCRC and TSP some tuning parameters must be set. For AFCRC, this is not difficult as the information is available [33]. However, for TSP the tuning parameters for this experiment are not indicated anywhere, as happens in every new experiment, and fixing them is quite difficult. Indeed, it is a trial and error process and it takes a long time to find a value which yields good performance. For this experiment the parameters [35] have been set to: $q = 3$, $p = 10$, $\gamma_f = 0.1$, and $\gamma_q = 1 \cdot 10^{-3}$.

GROWTH KINETICS

This signal represents the growth kinetics ratio of a bioreactor (units: $hours^{-1}$) which is a function of the substrate concentration (units: g/l). Figure 4.10 shows this relationship: the input u varies from 1 to 70 (g/l) and the output y is not linear in that range. When visualizing the signal, it seems proper to set $c = 3$, given the different zones that can be distinguished in the graph.

The results when applying PNCRM can be seen in Figures 4.11 to 4.13, where global output, local model responses and the MFs have been plotted, respectively. Fig. 4.11 shows how the output of the models follows quite well the trend of the signal for $c = 3$, including the most non-linear parts. Fig. 4.12 shows the regions where the three LMs are representative (low flat, medium

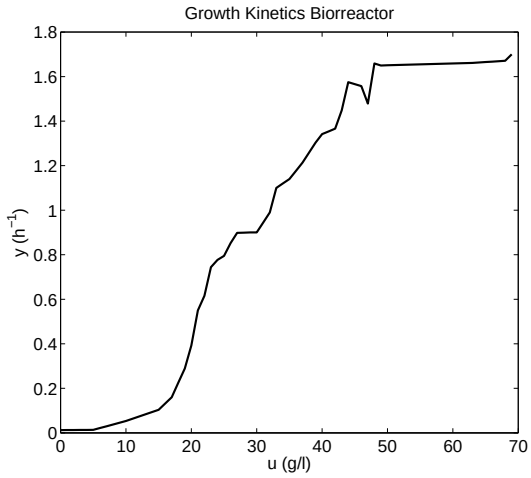

Figure 4.10: *Growth kinetics of a bioreactor*

increasing, high flat) together with Fig. 4.13 that shows the smooth MFs and their validity regions.

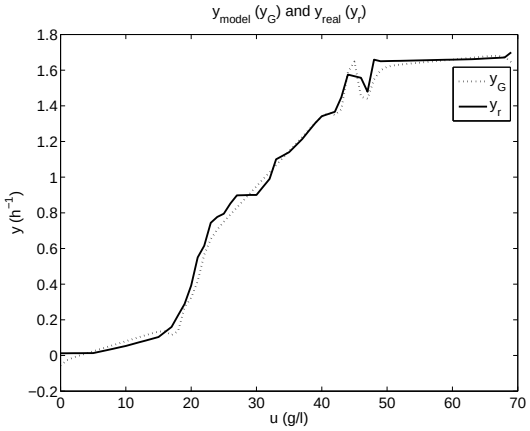

Figure 4.11: *Y_{real} and Y_{global} with PNCRM for the growth kinetics of a bioreactor*

When using the AFCRC algorithm, the results are again very similar, while TSP differs a little more than in the previous experiment. MSE for all

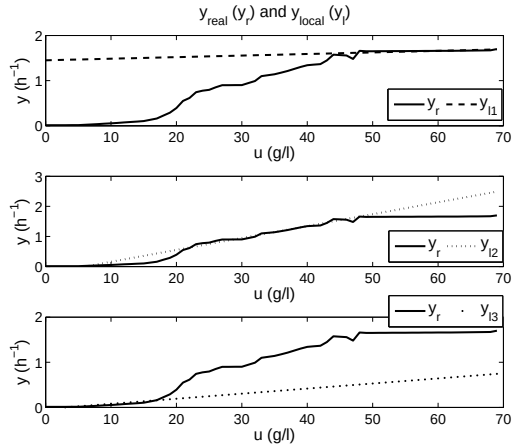

Figure 4.12: Y_{real} *and local models with PNCRM for the growth kinetics of a bioreactor*

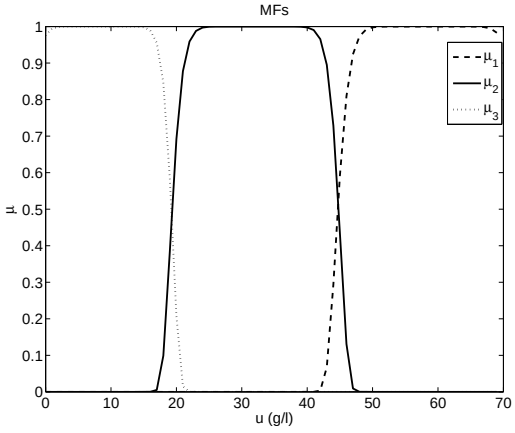

Figure 4.13: *Membership functions with PNCRM for the growth kinetics of a bioreactor*

algorithms is shown in Table 4.7. Again, the error is quite small in all cases. The error using AFCRC is half the error using PNCRM, while the error using TSP is three times higher than when using PNCRM.

Table 4.8 shows the local MSE for each cluster for all algorithms when

Table 4.7: *MSE for growth kinetics in a bioreactor*

Algorithm	**PNCRM**	**AFCRC**	**TSP**
MSE	0.0018	$9{\cdot}10^{-4}$	0.0057

Table 4.8: *Local MSE for growth kinetics in a bioreactor*

Algorithm	Charact.	Cluster1	Cluster2	Cluster3	Not Modeled
PNCRM	**LMSE**	0.0011	0.0015	0.0007	–
	%	34.3	32.9	27.1	5.7
AFCRC	**LMSE**	0.0010	0.0012	0.0008	–
	%	28.6	32.9	27.1	11.4
TSP	**LMSE**	0.0011	0.0026	0.0064	–
	%	32.9	35.7	31.4	2.9

$\lambda \geq 0.8$. As in the previous case, the local error is slightly larger for PNCRM than for AFCRC. The percentage of points represented by only one model for PNCRM is again larger than for AFCRC, leaving only 5.7% of the objects with mixed representation. Given that the local error is in both cases very small, the larger percentage of objects represented is an advantage of PNCRM over AFCRC in this case, given that the desired characteristic of the resulting model is that local models are valid in the largest region possible.

Regarding TSP, the error in two local models is larger than for the other two algorithms. As in the previous experiment, the number of objects represented is larger than when using AFCRC or PNCRM. Comparing the three algorithms it can be said that PNCRM offers the lowest error with the highest percentage of locally modeled points.

In Fig. 4.14 the MFs for AFCRC (left) and TSP (right) are represented: MFs for AFCRC are more irregular now and MFs of TSP present an overlap in some zones. This means that some objects fit well in two clusters (objects 22-24) while others are not highly described by any local model (objects 43-46). Again, MFs for PNCRM are smoother and locally independent; and therefore, more interpretable than with the other algorithms.

Accordingly, in this case, the resulting model for PNCRM will be more interpretable than the model obtained with AFCRC and TSP, and with only slightly more error than AFCRC. Moreover, the local models will be a valid by themselves, which is the goal followed by the proposed method.

As in the previous case, it is important to point out the great difficulty

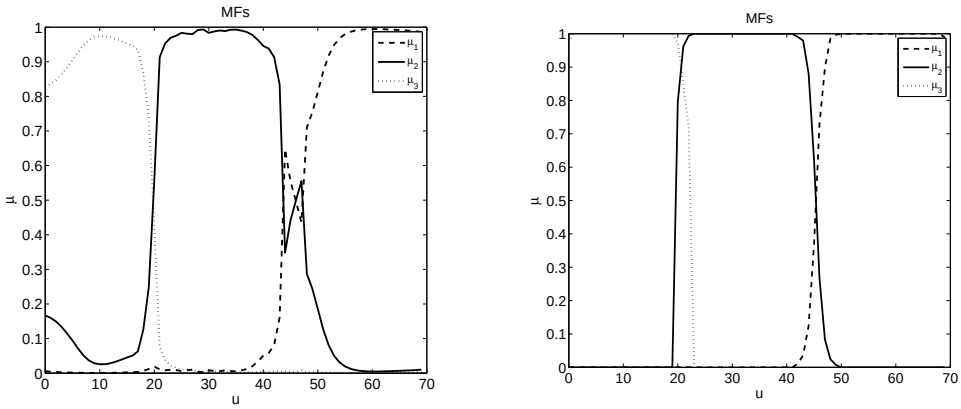

Figure 4.14: *Membership functions with AFCRC (right) and TSP (left) for the growth kinetics in a bioreactor*

in tuning the parameters of the TSP algorithm while parameters for AFCRC were again available.

4.7.3 Multidimensional System

The proposed algorithm has been used to model a laboratory setup experiment. Measurements were taken at the *Department of Systems Engineering and Control* (DISA) at the Universitat Politècnica de València.

BIOMASS SENSOR

To estimate the biomass concentration in fermenters a sensor has been developed in *DISA* based on measurements of the absorbency of a sample of the media. This device enables a variation of the gain of the system with the regulation of the intensity of the LED diode of reference. The voltage at the output is a function of the concentration of the biomass in the system.

The relationship between the output voltage of the sensor (H_r: Volts (V)) and the biomass concentration (c_H: grams/liter (g/l)) is not linear, as can be observed in Figure 4.15, for different values of the voltage of the LED, (v_l: Volts (V)), $v_l = \{0, 0.02, 0.03, 0.05, 0.1, 0.2, 0.25\}$. Therefore, an approximation by several local models can be a good approach to model this relationship which is now dependent on two input variables.

Given the highly-nonlinear behavior of the output, the number of clusters c is difficult to set. For a first approximation $c = 4$ will be set. Table 4.9

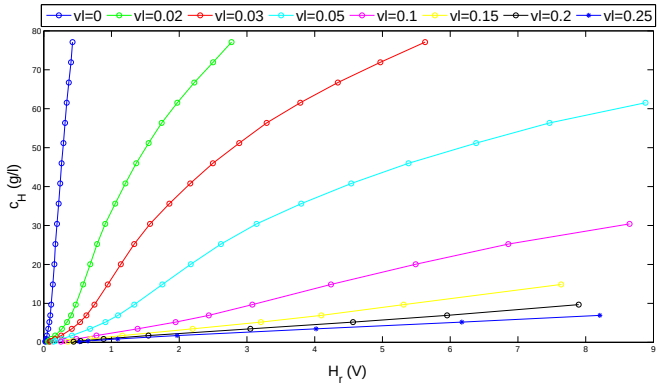

Figure 4.15: *Relation between $c_H(g/l)$ and $H_r(V)$ for different values of $v_l(V)$*

Table 4.9: *MSE for biomass sensor, c=4*

Algorithm	**PNCRM**	**AFCRC**	**TSP**
MSE	67.58	11.97	311

compares the mean squared error using PNCRM, AFCRC and TSP for $c = 4$.

It can be observed that the system is now more complex and the model is not as accurate as before. In this experiment, the error using PNCRM is 5.6 times larger than the error applying AFCRC. However the membership functions follow a hypergaussian function and are locally interpretable.

The error when using TSP is much larger, but this is mainly due to a poor setting of the tuning parameters, which were not set properly for this case due to the difficulty involved. For this reason, the results of applying the TSP algorithm will not be analyzed further. It must be pointed out that the performance of TSP can not be extensively tested due to the great difficulty in tuning the parameters, which is the main drawback of this algorithm. Only AFCRC will be used for these comparative studies.

Local errors and percentage of objects represented, Table 4.10, indicate that although the error is larger, more objects (89.7% versus 61.1%) are fully represented with just one local model for PNCRM. Moreover, it can be seen how two of the local models found with PNCRM are mainly responsible for the large MSE, because two of the models have a much lower error than the global output error.

When the MFs for the AFCRC algorithm with $c = 4$ are observed, Figure

Table 4.10: *Local MSE for biomass sensor*

Algorithm	Charact.	Clust1	Clust2	Clust3	Clust4	Not Modeled
PNCRM	**LMSE**	10.8	1.7	53.7	64.1	–
	%	30.8	6.8	29.9	22.2	10.25
AFCRC	**LMSE**	9.1	2.6	1.7	14.5	–
	%	12	18.8	20	10.3	39.3

4.16, it can be clearly seen how for this case the validity of each local model is not clear at all. Besides, each LM is only independent for a few objects and most of the objects have a mixed representation with one or several LMs. Therefore, even though the error is smaller than when using the PNCRM algorithm, the resulting model does not follow the desired characteristics. This is why the proposed algorithm is more appropriate when the system is modeled in an interpretable way.

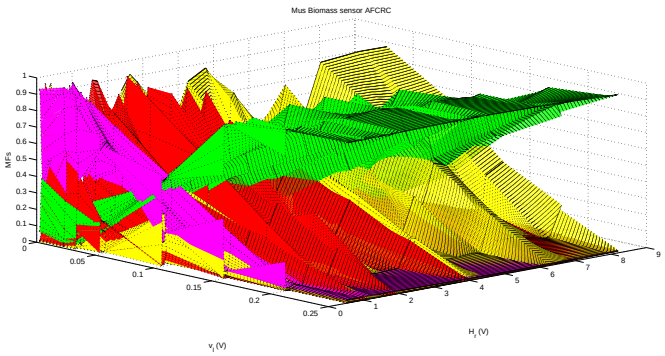

Figure 4.16: *MFs for AFCRC when c=4*

With regard to PNCRM, if a lower error is desired, then the number of clusters should be increased. Different experiments have been made setting c to different values. Table 4.11 shows the MSE for each case. It can be seen how with $c = 10$ the error for PNCRM is comparable to that of the original AFCRC experiment with $c = 4$, but it has to be remarked that they are now valid locally and highly interpretable.

Table 4.11 also shows how the error from $c = 7$ is reduced slightly with every cluster added. This might suggest that $c = 7$ is the best trade off between c and the error. Figure 4.17 shows the real output and the estimated

Table 4.11: *MSE for biomass sensor, PNCRM with increased c*

c	4	7	8	10
MSE	67.6	20.2	17.1	12.6

output for this value of c. It can be seen how the modeled output is quite close to the real output, although some points have a larger deviation. Due to the high nonlinearity of the system, the linear local models are only an approximation of the real system.

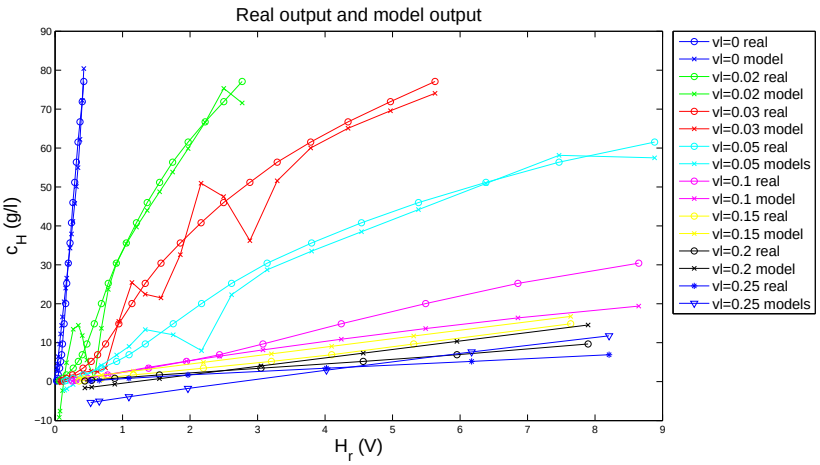

Figure 4.17: *Real and model outputs when c=7 for the biomass sensor*

This system is highly nonlinear. Therefore, to model it with linear and independent models and small errors the number of clusters must be increased until the total region is divided into subregions where a linear local model is valid and still has with smooth MFs.

The input space is now 2D, so the plotting of the MFs will be more difficult to visualize. In Figure 4.18 the shape of all the MFs can be seen. Many points are concentrated in the region near the origin. Therefore, many clusters are set by the algorithm around the most populated regions. Thus, the nonlinear regions with a high number of objects will be divided into several clusters in order to find the locally valid models and reduce the error.

The validity of each local model in this case is more interpretable compared to AFCRC, see Figure 4.16, even for a larger c ($c = 7$ vs $c = 4$).

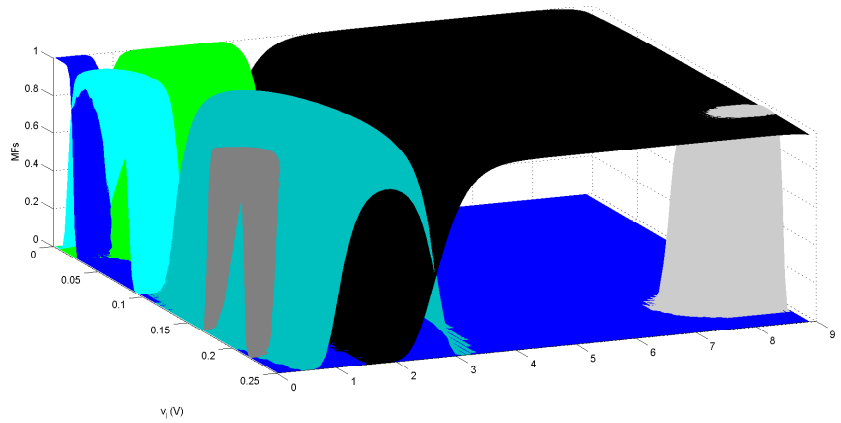

Figure 4.18: *MFs for PNCRM when c=7*

4.7.4 Analysis of the performance

With these examples, it has been studied how the proposed algorithm for modeling performs in different situations. The most important feature is the introduction of a certain shape for the membership functions, which assures a smooth and regular shape for these functions but also implies a restriction.

This restriction is the factor that causes the proposed algorithm to have a usually larger error for the same number of clusters than other similar algorithms such as AFCRC. This happens because the systems are modeled with several independent clusters that are only valid in a compact region, enhancing interpretability. In these cases, if the aim is to reduce the error then the number of clusters must be increased to find smaller compact regions where one local model is valid by itself.

Indeed, the main target of this algorithm is to get independent models with low error. For this reason if interpretable models are obtained, with small errors (global and local) then it does not matter much that other algorithm gets lower error at the expenses of losing interpretability.

The comparison with TSP can only give conclusions for cases of one-input and one-output. This is because tuning the parameters of TSP is so difficult that doing it for cases when more inputs are present has not been possible in this work. The conclusions obtained for unidimensional cases are that PNCRM works similarly to TSP, offering smaller errors and smoother membership functions.

In the three studied cases this algorithm performs very well, producing small global and local errors and with a small c. In the biomass sensor case, it was seen how a larger c was needed to obtain a small error with this method due to non-linearities, but that the model found was interpretable in all cases.

As the systems used were well-known, it has not been necessary the study of the best inputs to describe the output of each of them. Only the number of clusters had to be considered for the best fitting of the model. In a real case, where less information of the structure of the system is available, the number of clusters and the inputs to include will have to be considered simultaneously.

Then, with these studied cases it can be seen how the proposed algorithm is able to model a data base with the desired properties: independent clusters and well defined regions of validity. If the set follows this distribution a low number of clusters is expected (parabola and growth kinetics). However, if this structure is desired for clarity of the model and it is a complex system, a slightly larger number of local models is expected to obtain a small error (biomass sensor).

4.8 Conclusions

In this chapter a description of the design of a new algorithm for local modeling using clustering techniques has been detailed: **PNRCM**. This algorithm has been built taking into account the characteristics of the glucose transport system, with the aim of reducing the error of current commercially available monitors. The flexible structure of PNCRM can handle complex systems and as the model resulting is composed by several linear structures the information will easily be extracted.

This proposed method considers the restriction of a certain shape for the MF and gives more importance to the local error of the models in order to find independent clusters that are valid in a well-defined region and represented by a linear local model with little error. As such they are accurate and easy to interpret.

Another important advantage of PNCRM is that only two parameters, H and γ, have to be tuned and their values are general for all cases and given in this work. Thus, the complex set up process of some algorithms is avoided.

A review of optimization techniques is done in this chapter, as the iterative method used in other clustering algorithms cannot be used here. Finally, the characteristics of the cost index imply the use of a global optimization method.

Thus, the optimal parameters are found all together and, therefore, the best regions of validity and regression coefficients for those regions will be

found simultaneously, resulting in the best set of local models to define the system. This improves the iterative optimization method used by some of the literature clustering algorithms for modeling. The time needed for the global optimizer to find the solution is not a drawback, as this computation will be done off-line to later apply the found model on-line.

Once the parameters of the model are found, the computation of the output in a new instant will take a small amount of time, even in a commercially available microprocessor, so the on-line application of this algorithm will not be a problem.

In conclusion, a new algorithm for modeling interstitium-to-plasma glucose transport process has been developed. The characteristics of the system have been highly taken into account for its design.

Application to some test cases shows the expected performance of the algorithm. In next Chapter, the calibration algorithm based on this modeling algorithm will be defined and validated.

Chapter 5

A PNCRM-based calibration algorithm for CGMS

5.1 Introduction

In this chapter the application of the proposed modeling methodology, described in the previous chapter will be applied to the case under consideration: the estimation of plasma glucose levels for continuous glucose monitoring. Of note is that any plasma glucose estimations methods from interstitial measurements is known as **calibration algorithm** (CA).

5.1.1 Context definition

When plasma glucose is estimated from direct or indirect measurements in any compartment alternative to blood, the following information should be ideally included into the calibration algorithm in order to improve the accuracy of glucose estimations:

1. the intrinsic dynamic of the sensor;
2. glucose dynamic between plasma and the remote compartment; and
3. factors influencing the previous two dynamics.

Indeed, sensors of different nature, or even sensors belonging to the same class, may have each one a specific intrinsic dynamic. In this regard, recently it has been shown how much of the lag time of continuous glucose sensors is in fact due to the intrinsic delay in the sensor response to changes in glucose concentrations [30].

However, to date the issue of enhanced calibration algorithm has received poor attention with only a few scientific contributions. Knobbe et al. [86] were the first who considered inter-compartmental glucose dynamics and developed a five-state extended Kalman filter for the estimation of subcutaneous glucose levels. Facchinetti et al. further developed the strategy proposed by Knobbe et al. and proposed the 'enhanced Bayesian calibration method (BCM)' [42] based on an extended Kalman filter estimating interstitial glucose, plasma glucose and sensor's sensitivity along time. The method is intended to be used in cascade to any calibration algorithm built in commercial CGM devices and was validated on simulated data representative of diabetic subjects, showing improved CGM accuracy as compared to the method of [86]. However, a drawback of this validation is the use of the same model of interstitial glucose and sensor's sensitivity for data generation and state estimation; although in the first case a robustness analysis considering discrepancies in lag time estimation is conducted. Furthermore, as the authors acknowledge, application of the BCM to real data has two main limitations: first, it requires the knowledge of the variances of both state and measurement processes, which in real-life conditions are unknown; second, the existence of a burn-in period, considered as one day by the authors.

Leal et al. [93] considered sensor and glucose dynamics using a different approach. In particular, autoregressive techniques were applied to CGMS Gold (Medtronic), monitor data from a clinical study in patients with diabetes. A third order model of the relationship between current intensity given by the monitor and reference plasma glucose measurements were obtained. Predictions given by the model were corrected at every calibration point introduced by the patient, and a cross-validation analysis yielded a substantial improvement of the accuracy of glucose estimations [104]. However, a drawback of autoregressive models is that frequent recalibration may be needed to ensure a good performance.

At variance with the previously approaches the proposed calibration algorithm, thanks to its local models structure, is better suited to find and correct for all the above mentioned dynamics.

5.2 Proposed structure for CA

Figure 5.1 shows the block diagram of the local model based technique for the definition of the new CA.

The sensor is placed in a remote compartment, in this case the interstitial space. This sensor offers measurements of the intensity of current (I), which

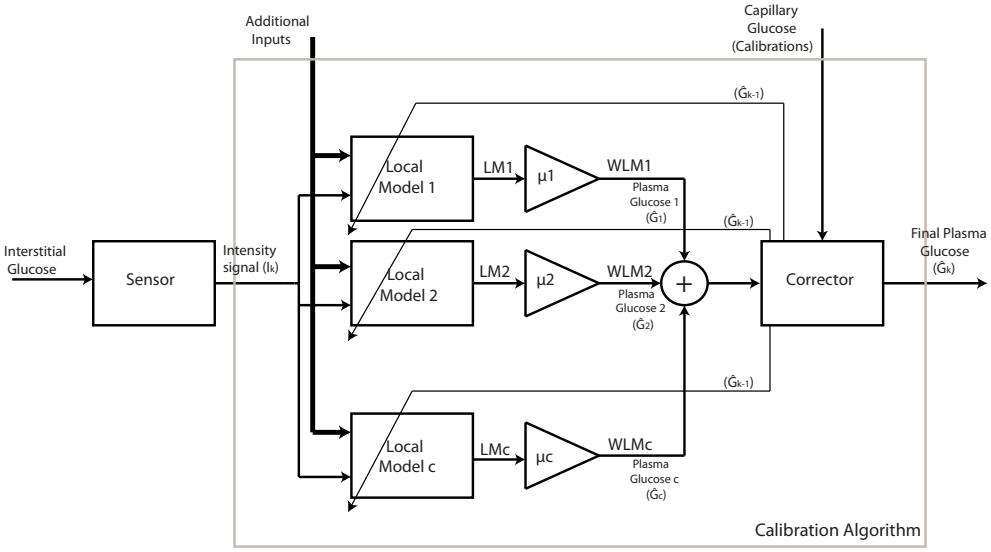

Figure 5.1: *Block diagram for the proposed new CA*

varies directly with the level of the interstitial glucose. The basic input of the model is the intensity of current from the sensor in different time instants. These inputs define which local models (LM) and to what extent they are valid (μ). The LM_i are weighted by its validity functions (μ_i) and results in a weighted local models (WLM_i) that offer a local estimation of the output ($\hat{G}_i$). The addition of all local estimations is the final global estimation of plasma glucose. Estimations of plasma glucose in previous instants ($\hat{G}_{k-1}$) are used to compute the final glucose estimation at instant k. In addition, a block to consider the calibration point(s) from capillary measurements is included to correct the estimation of the algorithm when this information is available. This proposed configuration of the calibration algorithm will be named as **basic configuration** and forms the first calibration algorithm defined and proposed in this work.

If available, additional signals can also be included as part of the information used to estimate the output. These additional inputs can be any signal related to the glucose transport process and that contributes to a better estimation of the plasma glucose level, *i.e.*: presence/absence of insulin

or glucose infusion, meal, exercise and so on.

In the case that additional signals are included the calibration algorithm changes. New calibration algorithms will be named as: **basic configuration** plus additional inputs or by the number of additional signals included.

Thus, this chapter will define several CAs and test their validity. In Section 5.4, the available data for the identification of the model embedded into the CA is described.

5.3 Adaptation of PNCRM to glucose transport modeling

PNCRM has been designed to include the important features of the CGMS system in the resulting model. Yet, there are a few characteristics that have to be taken into account for the application of the algorithm to the glucose transport process modeling. Thus, for the development of the calibration algorithm, the modeling method PNCRM has firstly to be adapted.

The first asepct is the consideration of the **relative error** in the cost function, both for local and global errors, instead of the absolute error. This is to give the same importance to the error no matter the magnitude of the sample.

The total glucose range of values can be divided into three significant regions: hypoglycemia (G≤70 mg/dl), normoglycemia (70 mg/dl<G≤180 mg/dl) and hyperglycemia (G>180 mg/dl). If absolute error is considered, then an error of, *e.g.*, 4mg/dl would be considered as important in hypoglycemic and hyperglycemic regions when, obviously, it is not. Thus, the consideration of the relative magnitude for errors will be more appropriate in the context of CGMS.

Then, the new index can be rewritten as:

$$J_{PNCRM}\left(\vec{X};\vec{\sigma},\vec{p},\vec{\beta}\right) = \sum_{k=1}^{n}\left(\frac{y_k-\sum_{i=1}^{c} WLM_i}{y_k}\right)^2 + \gamma\cdot\sum_{k=1}^{n}\sum_{i=1}^{c}\mu_{ik}\left(p_i,\sigma_i\right)\cdot\left(\frac{y_k-LM_i}{y_k}\right)^2 \tag{5.1}$$

Consequently, equations (4.9) and (4.10) are equally modified.

The second aspect to consider is the fact that glucose transport is a dynamic process. Thus, the estimation of glucose at the current instant depends on previous values of the glucose level. Actual value of these

past samples is only known at few particular instants of the day (capillary measurements), and then they can be considered by the model. The rest of time they are unknown and for this reason they are estimated. At these instants, the value for that input to the PNCRM algorithm will be taken from the past estimations of the model.

This means that a recursive strategy has to be used to compute the model. Also, this implies that the first value included has to be known to "set up" the recursive computation.

When an estimated output $\hat{y}_k$ is included as input to PNCRM, the parameters of the model are also included in the computation of the output at the next instants $\hat{y}_{k+n}$, since they were used to compute the output at k. Examples can be seen next.

Let's consider a system that has one measurable/independent input (x_1) and one input that consists on the previous value of the output ($x_2 = y_{k-1}$). Each time instant $k = 1, 2, ..., n$ will be represented as a second subindex, x_{1n} for x_1 at instant $k = n$. Then, equation (4.7) at instant k=1, will correspond to:

$$\hat{y}_1 = \beta_1 \cdot x_{11} + \beta_2 \cdot x_{21} + \beta_0 \tag{5.2}$$

As said before, x_2 is a recursive variable:

$$x_{2k} = \begin{cases} y_{k-1} & if \quad known \\ \hat{y}_{k-1} & if \quad unknown \end{cases} \tag{5.3}$$

Then, x_{21} is supposedly known, while the next samples will be supposed unknown $x_{2n} = \hat{y}_{n-1}$ for $n > 1$. For this reason, for $k = 2$ the output of the model will be

$$\begin{aligned} \hat{y}_2 &= \beta_1 \cdot x_{12} + \beta_2 \cdot (\beta_1 \cdot x_{11} + \beta_2 \cdot x_{21} + \beta_0) + \beta_0 \\ \hat{y}_2 &= \beta_1 \cdot x_{12} + \beta_2 \cdot \beta_1 \cdot x_{11} + \beta_2^2 \cdot x_{21} + \beta_2 \cdot \beta_0 + \beta_0 \end{aligned} \tag{5.4}$$

and generalized for $k = n$:

$$\hat{y}_n = \sum_{d=0}^{n-1} \left(\beta_2^d \cdot \beta_1 \cdot x_{1(n-d)} + \beta_2^d \cdot \beta_0 \right) + \beta_2^n \cdot x_{21} \tag{5.5}$$

As the number of past estimated outputs included as inputs increases, the more complex the computation of the current sample will be. In equation (5.5) it can be seen how the order of the polynomial to compute the LM increases in each sample. When a known value is included as input, then the model computation is reset to eq. (5.2).

This will affect the non-convexity of the index, making it highly non-convex, and makes more difficult its optimization. The optimization method is a global optimization algorithm and still the "global" solution could be found. Nevertheless, a much larger computation time has to be considered to reach this global solution.

A remark is that this type of systems makes the recursive optimization, as seen in Section 4.4 impossible since the *optimum* vector of regression coefficients is also to be computed recursively.

5.3.1 Measures of performance

The accuracy of the plasma glucose estimations obtained with the new algorithm (in all configurations) is defined using the ISO criteria (ISO_{ok}) [1], the Mean Absolute Relative Difference (MARD) and the Median Absolute Relative Difference (M2ARD).

ISO criteria is defined as the percentage of estimated samples, $\hat{G}$, that fulfill:

$$\hat{G}_k \quad ok \quad if \quad \begin{cases} \left|\frac{\hat{G}_k - (G_{ref})_k}{(G_{ref})_k}\right| \leq 20\% & for \quad (G_{ref})_k > 75mg/dl \\ \left|\hat{G}_k - (G_{ref})_k\right| \leq 15mg/dl & for \quad (G_{ref})_k \leq 75mg/dl \end{cases} \tag{5.6}$$

On the other hand, the MARD is defined as the mean of the absolute relative error taking into account all the estimated samples n:

$$MARD = \frac{\sum_{i=1}^{n} \left|\frac{\hat{G}_k - (G_{ref})_k}{(G_{ref})_k}\right|}{n} \tag{5.7}$$

M2ARD follows a similar equation as MARD, but computing the median value of the n samples instead of the mean, to see if the error is balanced with regard to that mean value.

Reference values are taken with a laboratory method to know the plasma glucose level at each instant. Results obtained with the proposed methods are usually compared with the commercially available monitor used in each experiment to evaluate the improvement (or not) over its performance.

All data were subjected to repeated-measures ANOVA with Huynh-Feldt adjustment for nonsphericity [135]. The ANOVA model included the test condition (manufacturer's or different configurations of the new algorithm) as the within-subjects factor. Post-hoc comparisons (Newman-Keuls test) were carried out to pinpoint specific differences on significant terms.

5.3.2 Model's saturation

The structure used for computing the global model includes several local models and their membership functions. Membership functions are indicators of the validity of each LM and they can also be called validity functions. In the proposed algorithm, PNCRM, these functions change smoothly covering a well defined region in the input space.

The identification of the model is done using training data, and the resulting model is valid for the region covered by the original information. This implies that the model is only known to work well for inputs that are covered by the validity regions. For inputs values not well represented by the original information the model estimations can be very poor since it implies the extrapolation of the identified model properties to an unknown region.

Then, to assure that the estimation of the output is done with certain validity of the model, a threshold for the validity of the model, th, has to be set. This threshold is used when the identified model is applied to new data, when used as calibration algorithm. This means that if none of the LM has a validity value larger than th the model is not representative of that input. In that case, the estimated output is replaced by the previous one (i.e., the model prediction is not updated), to avoid bad estimations of the model. Obviously, if that happens for a significant number of samples, it is an indication that new data should be collected to complete the model.

Thus, the designed PNCRM algorithm is capable of detecting inputs values not represented by the identified model, given the use of possibilistic membership functions, see section 3.2.5. This is an significant advantage of this algorithm, as it can also be used to identify not covered regions by the identified model.

5.4 Data description

The new CA (Figure 5.1) will be built using data from a clinical trial available to the group.

The data set used is the one taken from [113], where the details of the experiment, the methods and the subjects studied are described in detail. Here, only a brief description of the experimental protocol has been included.

Available data comes from a eu-, hypo- and hyperglycemic clamp, where healthy subjects wore a microdialysis-based glucose sensor (the GlucoDay®, Menarini, Italy). Briefly, after an initial period of spontaneous euglycemia, at time + 30 min an insulin infusion was started, at the rate of 1mU/kg/min and continued until 120 min. At the same time, glucose (20% dextrose wt/vol) was

infused when necessary at variable rate allowing for a controlled slow fall of plasma glucose in about 60 min., until the target of ∼50mg/dl was reached. From time +90 min. to +120 min. plasma glucose was maintained at a hypoglycemic plateau of ∼50mg/dl. At time +120 min. insulin infusion was stopped, and glucose infusion rate was increased to raise plasma glucose after 45 min. to the target of ∼165mg/dl (time +165min), which was maintained for the next 15 min. until time +180 min. Plasma glucose was measured every 6 min with a reference method synchronously with the output of the sensor, which gave a glucose estimate every 3 minutes.

According to the clamp experiment the desired and expected profile of the plasma glucose signal for the full period of time is the one shown in Figure 5.2.

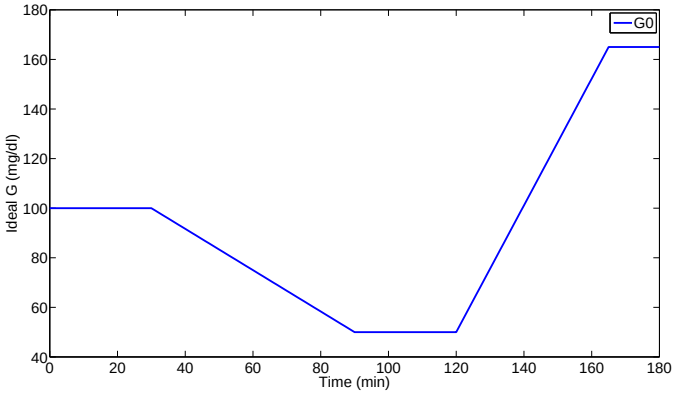

Figure 5.2: *Ideal curve of glucose profile for each patient*

A total of 8 patients were studied using this protocol. As plasma glucose was measured every 6 min during the 180 min of the study each patient has 31 valid samples. Two patients had some issues with the measurements and have less valid samples, around 14 each of them. Finally, the total set is composed by 214 paired values: sensor's electrical signal, I, and plasma glucose measurements, G.

5.5 Modeling the relationship between plasma glucose and the sensor's electrical signal

In this section the application of the proposed modeling algorithm ($PNCRM$) to the available clinical data will be described. Steps include the process of finding the proper number of LMs, c, that compose the global model and the best inputs that define the output (i.e. plasma glucose). These two parameters are dependent. Indeed, to avoid complexity, the aim is to find the minimum number of LMs that best define the output and the input signals that contain more information about the output.

The search process is an iterative process, using several combinations of inputs with different number of local models to see the one that performs best.

As described in Chapter 3 for general clustering purposes and detailed in Chapter 4 for the proposed approach, $PNCRM$ requires of a normalization of data for its correct application. This is a crucial step, and the type of normalization to apply has to be chosen according to the system characteristics.

In this case (see Section 3.2.6), the characteristics of system allow for two types of normalization: by the set or **Set Normalization** and normalization by Subjects or **Individualized Normalization**. The first type is the one applied here, considering together data from all patients, as if it were a unique input vector, and then applying normalization. The normalization with statistical properties has been applied, to get null mean and unitary variance, as in the general performance of $PNCRM$ it worked well.

It is important to remark that with PNCRM each patient will be represented by one or more LMs to a different extent, depending on the patient's inputs (signals) at each instant. The shift from one model to another will also be defined by the inputs.

5.5.1 Selection of the system inputs

After the data preprocessing using this type of normalization, $PNCRM$ is applied. A first attempt was to estimate the glucose only with current samples at the same instant (I_k), but even when the number of local models is large (i.e. $c = 8$), not good results were obtained, as expected. If previous time instants are included (I_{k-1}, I_{k-2}, etc.), which involves including information about the derivative of the signal, results improve slightly but glucose estimations are still far from the reference signal. The underlying physiology (mass transport between compartments) suggests the inclusion of information about the glucose level in addition to the sensor's intensity of current signal. In

this case information of the previous glucose level, G_{k-1}, has been included in the algorithm to include the most recent dynamics of the glucose variations. However, having information about the real glucose at every instant would be very unrealistic. It makes much more sense to include real information in G_{k-1} when this is available, through a SMBG samples (calibration points) and the rest of samples include the estimation of glucose given by the algorithm $\hat{G}_{k-1}$. At time $k = 1$, then, it is necessary to include information for the start up of the algorithm (initial conditions). Thus, the **main inputs** that can be used to compute an estimation of the current plasma glucose level are: samples of the intensity of the current, I, plasma or capillary glucose, G, and glucose estimations $\hat{G}$.

5.5.2 Outliers detection

Then, the performance of several combinations of these inputs, considered at different time instants ($k-1$, $k-2$,..., $k-n$) was checked for the available database. In all experiments performed, a set of a few samples was found characterized by a local model. These samples had a very different dynamics than the rest (a large majority).

The number of samples modeled by that particular local model were only a few (around 10 samples), and the local error of that model was very large in comparison to the rest of local models. These samples were found in patients 1, 2 and 4, at the end of the period, where both exogenous signals were stopped and response of glucose levels is $free$ (not imposed by exogenous signals). For these samples the studied dynamics was much faster. The fact that these samples are just a few and are modeled with other dynamics than the rest of the data set and with larger error suggests that they are **outliers** with regard to the total set.

In this case, these samples might correspond to a metabolic state where there are few samples to allow for its modeling, yet, the term *outliers* like in general clustering applications will be used for its definition.

The fact that only a few samples correspond to a different metabolic state than the rest of samples makes it not possible to properly identify the dynamics of these samples. For these reason they were removed from the set to be able to identify the most significant and representative samples.

For this reason, before performing the definite experiments checking different configurations, these samples were removed as they might not represent well the general behavior of the sensor.

5.5.3 Model structure identification

Thus, after the outliers removal, the process of checking different combinations of the main inputs at different time instants was repeated with the final data set. The combination of inputs that best predicted the output was [I_k, I_{k-1},G_{k-1}] (abbreviated by IIG). This means that the estimation of the actual glucose concentration at time k, $\hat{G}_k$, depends on the current intensity in the same time instant and involves a derivative indicating the trend of the signal. Using this input structure only two linear local models, $c = 2$, were needed to achieve a good estimation of plasma glucose. Thus, the global model obtained in this case has the structure reported in Eq. 4.8, with $\vec{x} = \begin{bmatrix} I_k & I_{k-1} & \hat{G}_{k-1} \end{bmatrix}$ and $c = 2$.

Of note, information about actual plasma glucose concentrations against current intensity was included only once at time $k = 1$ (G_{k-1}, calibration point). However, if more calibration points were used or needed, they would replace the glucose estimation in the previous instant ($\hat{G}_{k-1}$) to estimate plasma glucose at instant k.

This structure defines the first calibration algorithm that is designed and identified in this work. From now on this structure will be referred to as **basic configuration** or **IIG**.

5.5.4 Consideration of exogenous signals

In all calibration algorithms, the main signals are always the ones used above: I, G and $\hat{G}$. Nevertheless, the use of extra available information might help to improve the performance of the model of the system, as long as the information is related to the output.

In the clamp study described in Section 5.4 (henceforth Glucoday$^{(R)}$ study), an exogenous insulin (I_{ex}) and glucose infusion (G_{ex}) were present at some time during the 0-180 min period. Therefore, the possibility to include this information into the calibration algorithm as a **binary signal** (influence YES or NO) in addition to the other inputs already considered (I_k, I_{k-1},$\hat{G}_{k-1}$) was explored. The inclusion of this information is of particular interest since in the real-life conditions changes in the conditions of the subject (fed/fasting, insulin bolus/only basal, etc) may well influence the sensor-to-subject relationship, and feeding the calibration algorithm with this information was expected to improve its performance.

An infusion of exogenous insulin may be associated to high insulinemia after administration of an insulin bolus, and exogenous glucose to a postprandial period. Thus, clamp data offers us a valuable chance to analyze

the influence of different metabolic states.

In the case of subjects with type 1 diabetes, these signals are not directly measurable. Yet, information coming from the insulin pump can be used to characterize qualitatively the current metabolic state of the patients and estimate the information required by the algorithm.

In the Glucoday® study, the constant insulin infusion was in the 30-120 min interval, while the variable glucose infusion was between 30-180 min. Nevertheless, the binary signals were defined so as to identify periods of different metabolic state, in particular high insulinemia leading to hypoglycemia and the postprandial period after a recovery intake. For this reason, the interval where the glucose follows a slow decrement (30-90 min), the variable glucose infusion given by the physician to get the target decreasing glucose ramp was not considered, as it was only used to emulate the presence of effects that would have a meal in type 1 diabetic subjects.

On the other hand, the exogenous signal I_{ex} emulated the presence of effects that would have a bolus of insulin in a type 1 diabetic subjects.

The binary exogenous signals considered are shown in Figure 5.3.

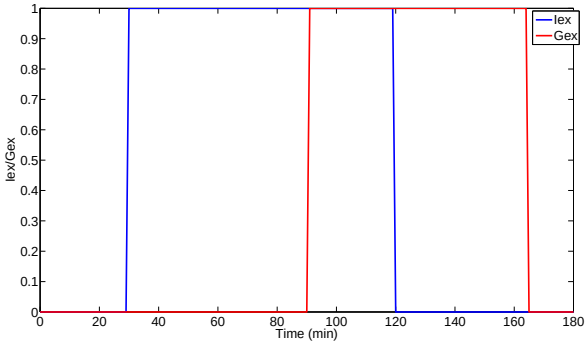

Figure 5.3: *Binary exogenous signals considered from the Glucoday$^{(R)}$ study*

With the inclusion of these signals, two new calibration algorithms are defined. Firstly, the exogenous insulin signal is added to the basic configuration yielding a second calibration algorithm called **basic configuration** plus **one exogenous signal** or abbreviated by **IIG+I_{ex}**.

Secondly, the exogenous glucose signal is included to this last configuration defining a third calibration algorithm called **basic configuration** plus **two exogenous signals** or abbreviated by **IIG+I_{ex}+G_{ex}**.

5.5.5 Results

Results are shown for each one of the calibration algorithms proposed:

CA1: Basic configuration (IIG)

Figure 5.5 shows plasma glucose estimates obtained with the new algorithm, in the basic configuration, compared with reference plasma glucose measurements. On the other hand, the output of each one of the two LMs composing the GM (already weighted by its validity function) can be seen in Figure 5.6. Figure 5.5 shows that the estimated signal is quite close to the reference glucose for all patients, indicating good performance of the algorithm even in this case with no exogenous information and only two local models. Figure 5.6 shows that each one of the two local models represents only a subset of the study subjects (LM_1 subjects 5 and 6; LM_2 the remaining subjects) while only the weighted integration of both LMs to form the GM allows for glucose estimations very close to the actual concentrations (Fig 5.5) in the whole population. It is important to remark that the validity of each local model is *optimally* found in the model identification process by the global optimizer[1].

The reason for the existence of two different local models can be easily explained by the analysis of the electric signal from the sensors. Although plasma glucose and insulin concentrations are very similar in all subjects, there are two *clusters* of current intensity, the one from sensors inserted in subjects 5 and 6 being much smaller as compared with the other. This could be probably due to differences in sensors sensitivity, which were captured automatically by the algorithm. This is an important fact. It is unknown whether the presence of clusters of sensor's sensitivity was due to "faulty" sensors or due to differences in the inflammatory response leading to such differences. Further studies should be conducted on this.

Figure 5.4 shows the histogram of sensor's sensitivities by its order of magnitude. Clearly, the sensitivities are distributed in a bimodal way for the Glucoday$^{(R)}$ study.

[1]The global optimizers do not assure the convergence to the global minimum. Yet, the results offered here are obtained running the optimizer for long enough period of time and getting the same final value for many iterations, assuming the value reached is the global minimum of the function.

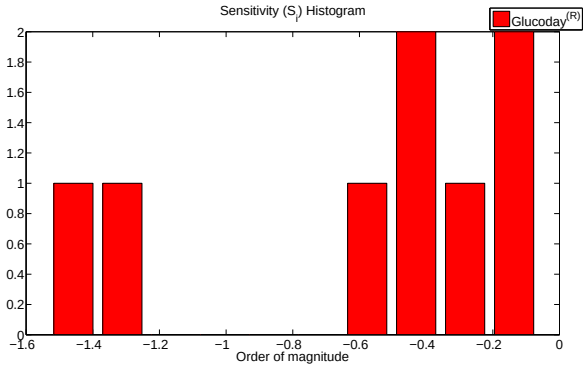

Figure 5.4: *Histogram of sensor's sensitivities by its order of magnitude from Glucoday$^{(R)}$ study*

CA2 and CA3: IIG+I_{ex} and IIG+I_{ex}+G_{ex}

Table 5.1 shows the performance of the proposed algorithm both in its *basic* configuration and considering information from binary signals, i.e. the presence (I_{ex},G_{ex}=1) or absence (I_{ex},G_{ex}=0) of insulin and/or glucose infusion effects. The comparison with the results from the one-point calibration implemented by the Glucoday, demonstrate that the new calibration algorithm allows for a significant improvement of the accuracy of the glucose estimations. In particular, the inclusion of additional information about the metabolic status further improves the accuracy of glucose estimations, reducing the MARD and M2ARD below 10%. It is worth noting that the magnitude of improvement was similar both for the whole and the hypoglycemic range. However, likely due to the small sample size, under conditions of hypoglycemia the difference did not reach statistical significance.

5.6 Inter-patient variability of sensor's sensitivity compensation

In the case of CGM systems, the sensor which provides the electrical signal related to the plasma glucose levels is placed subcutaneously in patients. Sensor's sensitivity can vary, as it is foreign body, [87, 104]. This means that the range of values of the electrical signal might differ from one patient to another for the same range of glucose levels. Analyzing the results of the

application of the proposed calibration algorithm it has been found that the main reason to find differences in the local models are the differences in the sensor's sensitivity .

Indeed, there are not only inter-patient variations in sensor's sensitivity. Intra-patient variations in the sensor's response are also present, [64]. Response of the sensor is known to change, for the same patient, with time. However, these variations are not yet well characterized.

In Section 3.2.6 it was introduced that when the input and output of a clustering application are composed by several independent subjects, the type of normalization applied can be done either by the set (seen in the previous section) or by subjects.

Then, the **individualized normalization** can be applied for the CGMS case, as signals from each patient can be considered independent from the rest of patients. With this type of transformation, data from each patient is normalized first, according to its own characteristics, and later all vectors are joined in an unique input vector, see Figure 3.3 in Section 3.2.6.

With this transformation, each patient inputs and output are rescaled according to its own ranges, taking into account (or compensating) the effects of the inter-patient sensor's sensitivity.

In the following, performance of the algorithm with individualized normalization is analyzed.

5.6.1 Data preprocessing and model structure identification

CA1: (IIG)

Once the data are normalized using this procedure, the modeling algorithm, $PNCRM$, is applied for the identification of the model. Here, the information resulting from the previous application with set normalization is used as knowledge of the system to avoid repeating the steps. Thus, the "outliers" detected in the previous case are not considered here either.

Indeed, the different combination of inputs checked with this normalization starts with the configuration that performed well in the set normalization case: the IIG structure ($\vec{x} = \begin{bmatrix} I_k & I_{k-1} & \hat{G}_{k-1} \end{bmatrix}$) with two LMs ($c = 2$).

A reduction of the number of local models to $c = 1$ has also been checked to see how it performs.

CA2 and CA3: IIG+I_{ex} and IIG+I_{ex}+G_{ex}

The inclusion of exogenous signals as binary information is also checked in this case, same as considered for the case of set normalization, for both, $c = 2$

and $c = 1$.

5.6.2 Results

Table 5.2 shows the performance of the proposed algorithm both in its *basic* configuration and considering information from binary signals, in the case of two local models. The comparison with the results from the monitor used (Glucoday), demonstrate that the new calibration algorithm allows for a significant improvement of the accuracy of the glucose estimations, specially when the binary information is used. Percentage of well estimated samples according to the ISO criteria improves around 20%, and MARD and M2ARD improve in more than 7% in all configurations and also attending to overall or hypoglycemic range.

Achieved MARD and M2ARD is equal or below 7.1% in all configurations of the algorithm. Again, the magnitude of improvement is different in the overall and the hypoglycemic range, where MARD and M2ARD are below 10% in all configurations. In this case, the difference between the original algorithm of the monitor and proposed algorithm is statistically significant in most of the cases and even in the hypoglycemic range the p-values are close to significance. Remark that the number of patients available is very small which may be the reason of such result.

Figures 5.7 and 5.8 show, respectively, the global and local models estimations of plasma glucose with the new algorithm, using the individualized normalization in the configuration where I_{ex} is included, compared with reference plasma glucose measurements. In Figure 5.8 the final output of WLMs is represented, considering the output of the linear model weighted by their validity function. Figure 5.7 shows a closer estimation of plasma glucose by the proposed algorithm, compared to the original monitor estimation. If the LMs of Figure 5.8 are analyzed, it can be seen that LM_1 detects dynamics mainly valid (in totality) when glucose levels are larger than $\sim 80 mg/dl$ while LM_2 find dynamics mostly valid when glucose levels are around the low hypoglycemic range $\sim 50 mg/dl$. In the region between them, both models contribute to the estimation of global glucose levels with different proportion (different weights).

For this type of normalization, all configurations of inputs have been checked with only one local model, $c = 1$. This is not the same as one global model, as the validity function present for the local regression also contributes to the global model, indicating the region where the model is valid. Table 5.3 shows a summary of all results. In all configuration of inputs considered, performance of the case of $c = 1$ is quite similar to the case of $c = 2$, only

slightly worse, when the individualized normalization is used.

Now that different values of patient to sensor's sensitivity are introduced in the model by means of individualized normalization, the local models found represent different dynamics of the signal or different ranges. Differences for the two models are small as the reduction from $c = 2$ to $c = 1$ does not make the performance of the calibration algorithm much worse.

Figure 5.5: *Comparison between global glucose estimation, Ge, (solid line) and paired available reference glucose, Gr, (dotted line), plotted by each patient and each 6-min sample (dots). This is for the basic input vector, called IIG, only two local models (c=2) and applying* ***set normalization****. These signals have also been compared to the original manufacturer's sensor accuracy Gs.*

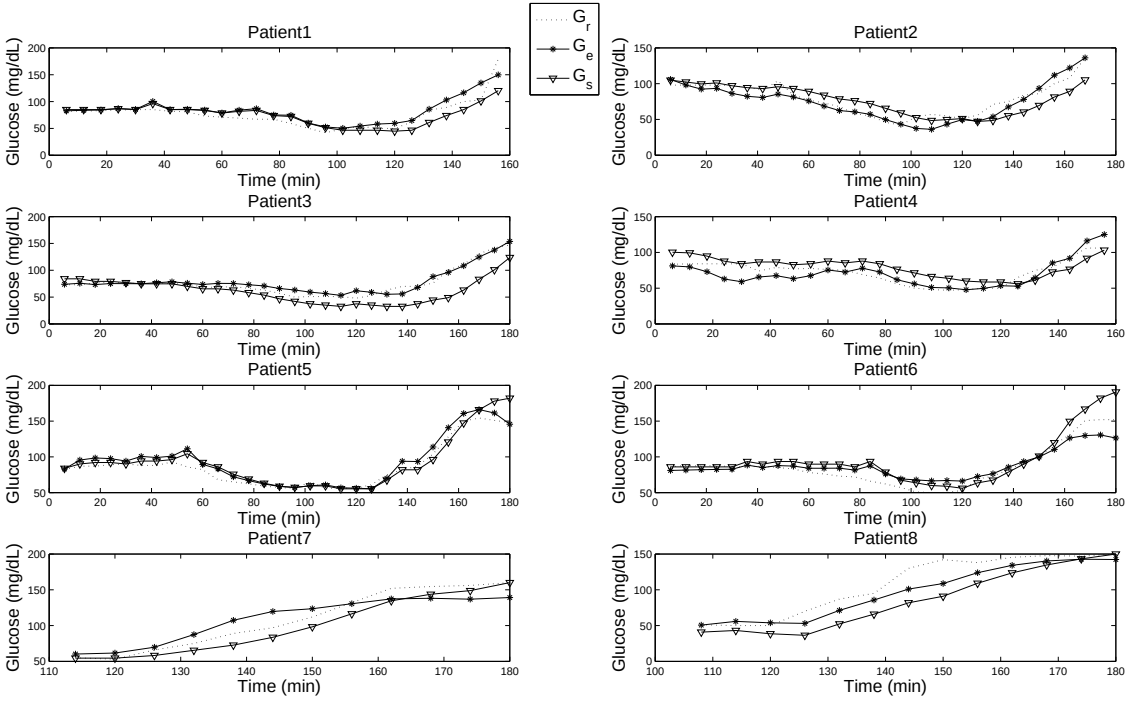

Figure 5.6: *Comparison between the reference glucose, Gr, and the estimation of glucose done only by each local model (LM_1 and LM_2). This is for the basic input vector (IIG), for c=2 and applying* ***set normalization****. It can be seen how LM_1 provides accurate plasma glucose estimations just for Patients 5 and 6 while this local estimation does not contribute at all to the accuracy of the estimation in the rest of patients. On the other hand, how LM_2 provides accurate plasma glucose estimations in all but patients 5 and 6, where its contribution to the accuracy of the estimation is nearly 0.*

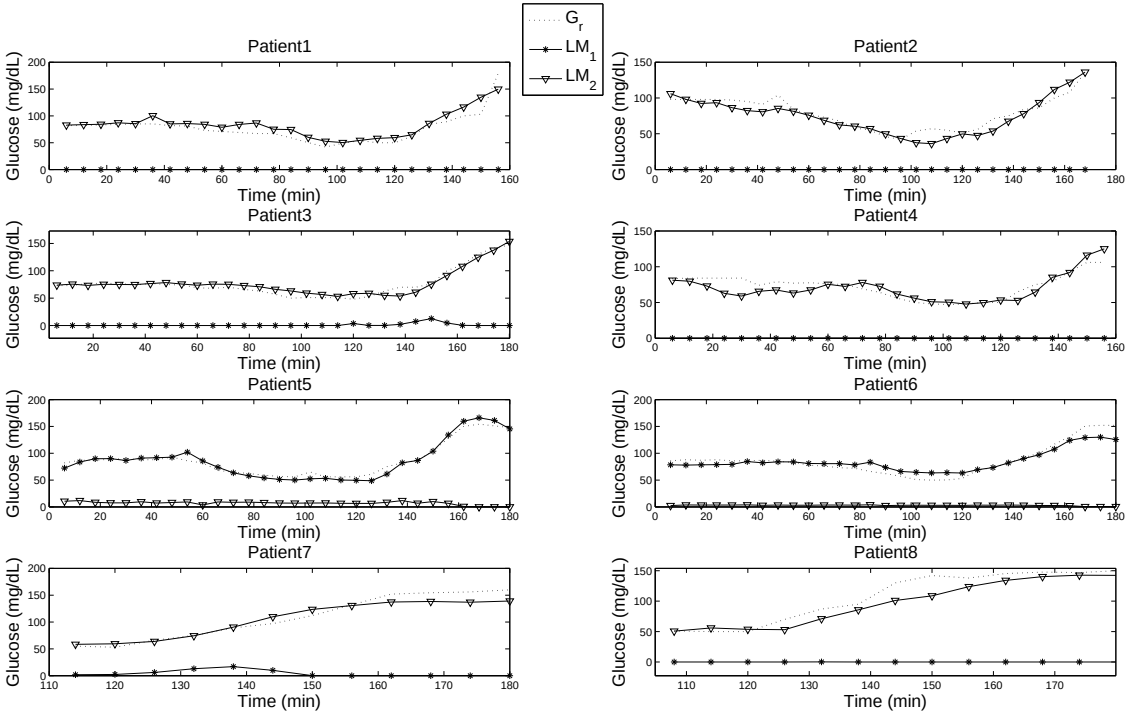

Table 5.1: *Measures of accuracy obtained with the proposed new algorithm* ***set normalization*** *are presented and compared with the Manufacturer's algorithm (**M**), both for the whole and the hypoglycemic range. Results obtained with the different configurations of the algorithm are shown: the "basic" configuration where the input is the **IIG** vector (I_k, I_{k-1},$\hat{G}_{k-1}$) noted as **CA1**; the case where insulin infusion (I_{ex}) is considered in addition to IIG (IIG+I_{ex}, noted as **CA2**); the case where both I_{ex} and the exogenous glucose infusion (G_{ex}) are used in addition to IIG (IIG+I_{ex}+G_{ex}, noted as **CA3**). Data are mean±SD (ISO_{ok} and MARD) or median and range (M2ARD). The difference between the new and the old algorithm is presented along with its confidence interval.*

Measure of accuracy	Glycemic range	Calibration algorithm							P value
		Manufacturer's (M)	Proposed algorithm: *PNCRM*			Difference [Confidence Interval]			
		Glucoday®	Configurations						
			*CA*1	*CA*2	*CA*3	*CA*1-M	*CA*2-M	*CA*3-M	
ISO_{ok} (%)	Overall	78.38% ±14.52	88.79% ±6.91	94.59% ±5.2	97.52% ±3.67	10.40 [2.08; 18.73]	16.20 [7.88; 24.53]	19.14 [10.81; 27.46]	0.008
	Hypo <75mg/dl	75.92% ±16.68	86.24% ±13.96	95.83% ±8.91	93.75% ±11.57	10.32 [-3.05; 23.70]	19.92 [6.54; 33.30]	17.83 [4.46; 31.21]	0.029
MARD (%)	Overall	14.7% ±5.35	10.8% ±1.54	8.75% ±1.02	7.8% ±2.6	-3.9 [-7.30; -0.50]	-5.96 [-9.36; -2.56]	-6.91 [10.30; -3.51]	0.012
	Hypo <70mg/dl	17.32% ±7.34	12.76% ±6.17	8.35% ±4.02	12.25% ±7.53	-4.57 [-11.72; 2.59]	-8.97 [-21.00; 2.60]	-5.07 [-12.23; 2.09]	0.11
M2ARD (%) [range]	Overall	13.2% [5.96; 20.37]	9.05% [4.67; 13.58]	7.03% [4.33; 8.62]	5.51% [3.67; 15.85]	-4.08 [-7.80; 0.35]	-6.31 [-10.03;-2.59]	-6.78 [-10.50; -3.06]	0.004
	Hypo <70mg/dl	17.63% [5.31; 29.26]	9.07% [7.49; 28.55]	6.3% [0.96; 14.54]	9.1% [1.35; 23.53]	-3.83 [-11.58; 3.92]	-8.89 [-22.00; 3.70]	-5.53 [-13.28; 2.22]	0.14

Figure 5.7: *Comparison between global glucose estimation, Ge, (solid line) and paired available reference glucose, Gr, (dotted line), plotted by each patient and each 6-min sample (dots). This is for the basic input vector plus binary signal I_{ex} (IIG+I_{ex}), only two local models (c=2) and applying* ***normalization by subject****. These signals have also been compared to the original manufacturer's sensor accuracy Gs.*

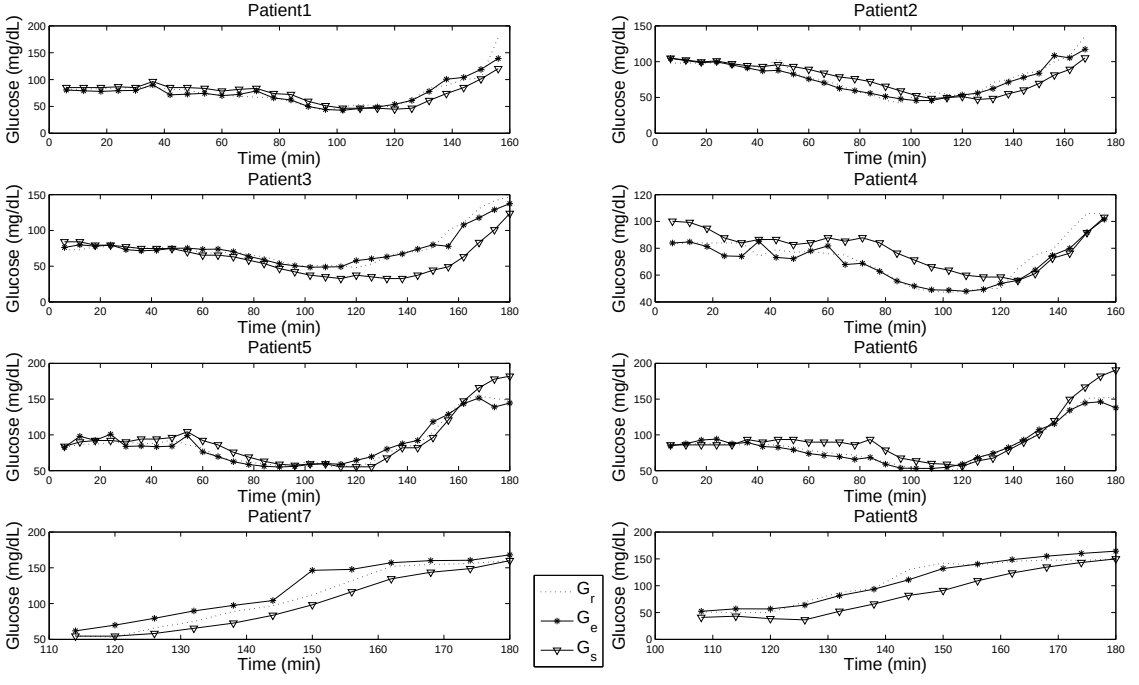

Figure 5.8: *Comparison between the reference glucose, Gr, and the estimation of glucose done only by each local model (LM_1 and LM_2). This is for the basic input vector plus binary signal I_{ex} ((IIG+I_{ex})), for c=2 and applying* ***individualized normalization****. It can be seen how LM_1 has a high validity for values of glucose larger than 80mg/dl while LM_2 has its high validity for values of glucose around sim50mg/dl. In between these values, both LM contribute with different weights to the estimation of the glucose, with smooth transition of their validity functions.*

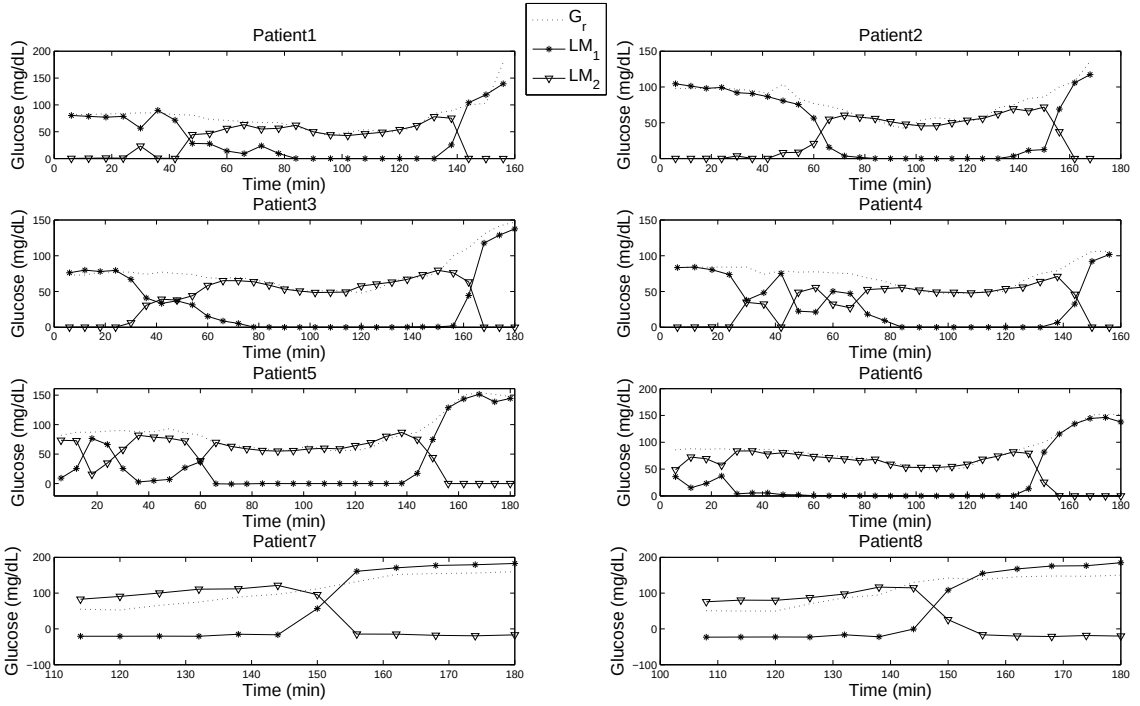

Table 5.2: *Measures of accuracy obtained with the proposed new algorithm with* ***individualized normalization*** *are presented and compared with the Manufacturer's algorithm (**M**), both for the whole and the hypoglycemic range. Results obtained with the different configurations of the algorithm are shown: the "basic" configuration where the input is the **IIG** vector (I_k, I_{k-1},$\hat{G}_{k-1}$) noted as **CA1**; the case where insulin infusion (I_{ex}) is considered in addition to IIG (IIG+I_{ex}, noted as **CA2**); the case where both I_{ex} and the exogenous glucose infusion (G_{ex}) are used in addition to IIG (IIG+I_{ex}+G_{ex}, noted as **CA3**). Data are mean±SD (ISO_{ok} and MARD) or median and range (M2ARD). The difference between the new and the old algorithm is presented along with its confidence interval.*

Measure of accuracy	Glycemic range	Calibration algorithm							P value
		Manufacturer's (M)	Proposed algorithm: *PNCRM*			Difference [Confidence Interval]			
		Glucoday®	Configurations						
			*CA*1	*CA*2	*CA*3	*CA*1-M	*CA*2-M	*CA*3-M	
ISO_{ok} (%)	Overall	78.38% ±14.52	97.44% ±4.06	97.95% ±5.76	98.46% ±5.86	19.28 [11.92; 26.64]	18.64 [9.4; 27.87]	19.05 [9.63; 28.48]	0.013
	Hypo <75mg/dl	75.92% ±16.68	97.7% ±4.04	98.85% ±8.84	98.85% ±8.84	21.9 [14.41; 29.39]	20.96 [9.47; 32.45]	20.96 [9.47; 32.45]	0.023
MARD (%)	Overall	14.7% ±5.35	7.1% ±1.75	6.71% ±2.58	6.58% ±1.87	-7.59 [-9.73; -5.45]	-7.46 [-10.8; -4.08]	-7.78 [-10.63; -4.94]	0.004
	Hypo <70mg/dl	17.32% ±7.34	9.33% ±2.65	7.17% ±5.52	7.49% ±5.09	-7.33 [-11.33; -3.33]	-8.68 [14.63; -2.72]	-8.96 [-14.66; -3.26]	0.064
M2ARD (%) [range]	Overall	13.2% [5.96; 20.37]	5.81% [2.99; 9.13]	5.27% [4.12; 10.76]	5.23% [2.68; 7.95]	-7.6 [-9.35; -5.86]	-7.31 [-9.92; -4.7]	-7.51 [-9.6; -5.43]	$7 \cdot 10^{-5}$
	Hypo <70mg/dl	17.63% [5.31; 29.26]	8.66% [5.35; 15.78]	5.07% [2.01; 20.28]	6.77% [3.04; 18.94]	-5.91 [-10.22; -1.59]	-8.19 [-14.37; -2.02]	-8.05 [-13.75; -2.34]	0.089

Table 5.3: *Comparative of all results obtained with the proposed new algorithm for for all configurations defined (**individualized normalization** with $c = 1$ and $c = 2$, and **set normalization** with $c = 2$) and the Manufacturer's algorithm (**M**), computed for the overall and the hypoglycemic range. Results are shown for the different configurations of the algorithm: the "basic" configuration where the input is the **IIG** vector (I_k, I_{k-1},$\hat{G}_{k-1}$), noted as **CA1**; the case where insulin infusion (I_{ex}) is considered in addition to IIG (IIG+I_{ex}, noted as **CA2**); the case where both I_{ex} and the exogenous glucose infusion (G_{ex}) are used in addition to IIG (IIG+I_{ex}+G_{ex}, noted as **CA3**). Data are mean±SD (ISO_{ok} and MARD) or median and range (M2ARD).*

Measure of accuracy	Glycemic range	Calibration algorithm									
		Manufacturer's (M)	Set Normalization: $c = 2$			Individualized Normalization: $c = 1$			Individualized Normalization: $c = 2$		
		Glucoday®	$CA1$	$CA2$	$CA3$	$CA1$	$CA2$	$CA3$	$CA1$	$CA2$	$CA3$
ISO_{ok} (%)	Overall	78.38% ±14.52	90.26% ±6.91	94.87% ±5.2	98.46% ±3.67	95.38% ±4.49	97.95% ±8.75	97.95% ±8.75	97.44% ±4.06	97.95% ±5.76	98.46% ±5.86
	Hypo <75mg/dl	75.92% ±16.68	87.36% ±13.96	97.7% ±8.91	97.7% ±11.57	94.38% ±8.29	96.55% ±26.52	96.55% ±26.52	97.7% ±4.04	98.85% ±8.84	98.85% ±8.84
MARD (%)	Overall	14.7% ±5.35	10.56% ±1.54	8.73% ±1.02	7.28% ±2.6	8.3% ±1.83	7.5% ±2.69	7.9% ±2.32	7.1% ±1.75	6.71% ±2.58	6.58% ±1.87
	Hypo <70mg/dl	17.32% ±7.34	13.24% ±6.17	8.34% ±4.02	9.77% ±7.53	11.27% ±3.59	9.54% ±5.36	10.08% ±5.36	9.33% ±2.65	7.17% ±5.52	7.49% ±5.09
M2ARD (%) [range]	Overall	13.2% [5.96; 20.37]	8.77% [4.67; 13.58]	7% [4.33; 8.62]	5.48% [3.67; 15.85]	6.46% [3.21; 10.5]	6.2% [3.53; 12.58]	6.71% [5.52; 13.98]	5.81% [2.99; 9.13]	5.27% [4.12; 10.76]	5.24% [2.68; 7.95]
	Hypo <70mg/dl	17.63% [5.31; 29.26]	11.09% [7.49; 28.55]	6.91% [0.96; 14.54]	8.22% [1.35; 23.53]	9.52% [6.62; 25.73]	9.02% [2.09; 23.73]	9.92% [5.41; 23.34]	8.66% [5.35; 15.78]	5.07% [2.01; 20.28]	6.77% [3.04; 18.94]

5.7 Discussion

Firstly, this study validates the feasibility of calibration algorithms for continuous glucose monitoring, **based on** a **dynamic model** which includes the relationship between plasma glucose level and measurements in a remote compartment (in this case the interstitial space). The strength and novelty of this method resides in the structure of the global model, which is composed by several local models weighted and added. Local models are defined by a validity function and a linear combination of the inputs, so that each one represents to a variable extent a different (metabolic) condition and/or sensor-subject interaction. Then, each subject will be represented by one or more LMs and the shift from one LM to another is defined by the inputs (i.e. the output of the sensor but also every other signal containing information relevant to the glucose transport dynamics). It is worth noting that each local model has a very simple structure (linear regression), favoring interpretability of the global model. Moreover, the validity function of the local models (μ_i) makes them representative of very specific and well defined regions on the input space, allowing for the identification of the regions they represent. All these may help to elucidate the relationship between the signal(s) from the remote compartment and plasma glucose levels, describing different glucose dynamics and sensors behaviors. It does mean that this calibration algorithm could be applied to any sensor that offers some indirect measurement related to plasma glucose levels, the number and parameters of each local model being determined by the particular sensor's output.

Results from the present study should be considered as a proof of concept. Indeed, in the small population studied at least two different dynamics of the sensor-to-subject interaction were found, each one detected by a different local model: LM_1 for subjects 5-6 and LM_2 for the other subjects. The reason for the existence of two different local models was easily explained by differences in the sensor's sensitivity of patients. Consideration of both dynamics by means of the two independent local models allowed for a significant improvement of glucose estimations as compared to the original algorithm implemented by the Glucoday. Additionally, the proposed algorithm admits the introduction of information about the metabolic condition of the subject as a binary signal. This further improved the accuracy of the glucose estimation and appears to be an interesting feature since any information potentially relevant to the sensor-to-subject interaction (physical activity/inactivity, fed/fasting state, etc.) can be used to feed the algorithm. This information could be easily obtained from an insulin pump or external sensors in a body area network.

However, the study has three main limitations.

1. First, the study population is composed by just eight healthy subjects and data should be considered preliminary. Thus it is well probable that the structure of the obtained algorithm is not representative of the general population of people with diabetes. Indeed, getting a calibration algorithm applicable to the general population of people with diabetes would require a specific clinical study in a representative sample, and validation of the obtained algorithm should be carried out in a different study involving larger numbers of subjects with both T1 and T2 diabetes. However, in this case the structure of the proposed algorithm is not expected to change regarding the inputs to be considered, although more local models would probably be needed to cover the greater heterogeneity of local behaviors.

2. Second, data from reference plasma glucose measurement are available for a limited time for each subject. This could in theory have reduced the possibility to identify different glucose dynamics and sensor behaviors. However, during the clamp study, glucose and insulin concentrations varied through all the clinical significant ranges of eu-, hypo- and hyperglycemia, giving a good picture of the changes observed in real-life conditions.

   On the contrary, longer studies should be performed to identify changes in the dynamics of the sensor over time, avoiding underestimation of the number c of local models needed to describe the plasma/remote compartment glucose relationship. This is particularly true for minimally invasive sensors where biofouling, inflammation and foreign body reaction (which are probably tissue and subject specific) induce changes in sensor's response to variations in glucose concentration.

3. Finally, to prove robustness of the algorithm to cope with variability, the results must be confirmed in a sample representative of the population of patients with diabetes. In this regard, from a phylogenetic point of view it is expected that the inter-subject variations in the inter-compartmental glucose dynamics, as well as the spectrum of the inflammatory responses to the sensor's insertion, are limited. However, due to changes in the microcirculation, the variability of the sensor-to-subject interaction (i.e. different sensor sensitivities, sensor's drift overtime, metabolic conditions, etc.) might be greater in patients with diabetes, especially those with microvascular complications [111]. Nevertheless, the accuracy of CGM in patients with diabetes seems to be not different as compared to healthy subjects, indicating that the

theoretical greater variability associated with the diabetes condition may have limited practical impact.

Despite its limitations, in this proof of concept study the **local models technique** using the set normalization for data is demonstrated to be an **effective approach for CGM** calibration algorithms. Indeed, aside the very good results obtained in terms of accuracy of glucose estimations, the short computational time associated with this methodology makes it feasible for real time monitoring and implementation **in every glucose sensor**.

The application of the designed modeling method ($PNCRM$) as calibration algorithm is direct using the set normalization. Results indicate that the differences in the validity of the local models are defined by differences in the magnitude of the sensors response, finding two different clusters (groups of similar sensitivities). Indeed, the interaction between a sensor and the remote compartment of measurement is likely to be specific of each sensor. In the case of minimally invasive sensors (which to date are the only commercially available) this interaction depends on the biocompatibility of the materials used and on the inflammatory reaction following its insertion. This relationship is complex, may be time-dependent (as the foreign body reaction progresses) and certainly is specific to each individual [87, 104].

For this reason, the application of normalization by subject to this case makes special sense, to see the effects of considering the sensors sensitivity in the model. Results confirm that if the effects of the sensitivity are removed from the dynamic response, then performance of the model improves with regard the original manufacturer's accuracy and also the proposed calibration algorithm with set normalization. Indeed, even when only one local model is used, $c = 1$ results are better than in both previous cases. It is very important to note that only one linear model is not the same as the global model with $c = 1$, because the global model also includes the validity function to indicate representativeness. This region is very important for the application of the model as calibration algorithm, as it will indicate if the global model is representative of the new patient or not.

However, the application of the $PNCRM$ algorithm with the individualized normalization can not be directly applied as calibration algorithm. The parameters found for the model are in normalized dimensions and, obviously, normalization parameters for a new patient can not be obtained beforehand to rescale inputs and output and remove sensitivity effects.

Thus, to use the modeling algorithm along with the individualized normalization the estimation of normalization parameters is needed. In addition to this, the estimation of parameters has to be done along with

time, as each patient's sensitivity also varies during the sensor lifetime. With this estimation of parameters the proposed modeling algorithm $PNCRM$ with individualized normalization can be used as calibration algorithm for the estimation of glucose levels. This will be the topic for next chapter.

5.8 Conclusions

In this chapter the application of the modeling algorithm proposed in Chapter 4 to the case of estimation of glucose levels for continuous glucose monitoring is described. The application was done in a data set with only 8 healthy patients with variations of glucose through exogenous infusions of insulin and/or glucose.

The use of the model as calibration algorithm (CA) results in an improvement with regard to the monitor used in the experiment to obtain the data, reaching promising results. Inclusion of exogenous binary signals indicating effects of glucose and insulin contributes to further improve the accuracy of estimations. However, the small size of the data set force these results to be only a proof study and can not be considered as conclusive.

The advantages of the proposed method are many, being the most important the improvement of glucose estimation accuracy.

Results obtained in the application of the CA indicate that the sensor's sensitivity has a large effect in the estimation of glucose, as inter-patient variations on this parameter are much influent, as the range of electrical signal can vary much between patients.

Using individualized normalization and applying PNCRM much better results were obtained, for all configurations of inputs. Nonetheless, this structure can not be directly applied as calibration algorithm unless an estimation of normalization parameters is done for the application to a new patient's data.

Chapter 6

Adaptive calibration algorithm

6.1 Motivation

In previous chapters, a new calibration algorithm (CA) for Continuous Glucose Monitoring, based on the identification of a set of local models, was introduced.

For the correct application of this new CA, input and output data must be normalized. The different performance of the algorithm when data is normalized using set parameters (set normalization) or subject parameters (individualized normalization) was shown in the previous chapter.

Performance is better in the case of individual normalization, as variations of the inter-patient sensor's sensitivity variability are compensated. However, the application of the identified model to a new patient is not possible, as their own normalization parameters can not be known a priori.

This motivates the idea of designing an adaptation scheme that allows the normalization parameters to adapt to new data. This adaptation of normalization parameters would allow the application of the computed model to data not used for the identification of the model and with different characteristics than the original data.

Starting the normalization parameters with a population value, they are adapted to make the model improve its performance with the new subject's data.

What is more, with the adaptation of normalization parameters the intra-patient sensor's sensitivity variations (variations of sensitivity with time for a subject) can also be compensated.

Thus, in this chapter details the characteristics of this new adaptive

calibration algorithm (ACA), including: the methodology chosen to perform this adaptation, the characteristics of the system and all issues related to the right design and application of this system.

Figure 6.1 shows a block diagram of the adaptive calibration algorithm, where the main block is still the calibration algorithm of figure 5.1. A new block for the estimation of normalization parameters is included, as well as two blocks for normalization and denormalization of signals, in addition to the CA block.

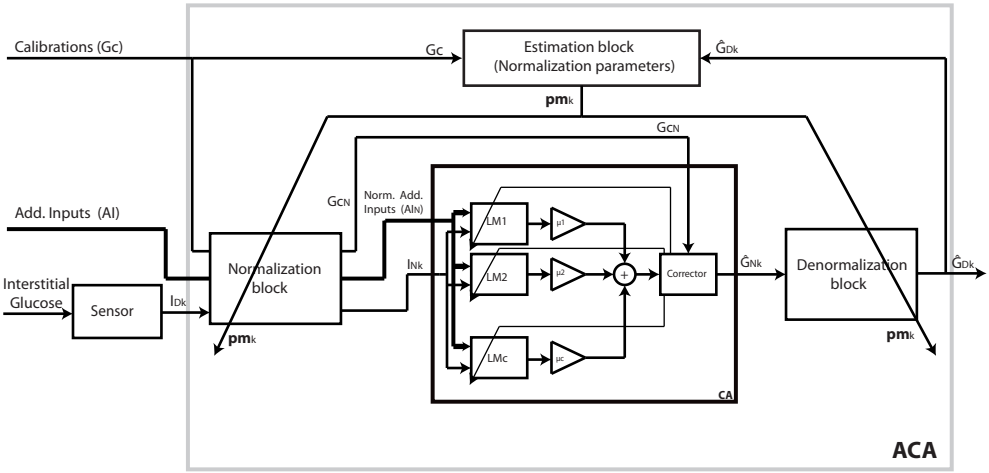

Figure 6.1: *Block diagram for the proposed new ACA.*

6.2 Data characteristics

6.2.1 Sensitivity variations

In Chapter 5, it was shown how the use of individual normalization improves the performance of the calibration algorithm, as sensor's sensitivity variations are compensated. This means, clearly, that differences in sensor's sensitivity affect very much the estimation of glucose. With a better understanding of the sensor's sensitivity variations, it might be easier to compensate them with the calibration algorithm. For this reason, an estimation of this variations is done here.

Indeed, variations on sensor's sensitivity are a fact and recent studies show

their importance [112, 64], but it is not easy to quantify them or even to define the shape. It is well known that the sensitivity of the sensor is different for each patient, but also it is different within the sensor's life time [95]. This means that the same patients will not have the same sensitivity to their sensor the first day than the fourth day after its insertion. The sensitivity variations over time is not yet well known. In general, there is a decrement of the sensitivity with time [64, 85]. In addition, there are several studies [64, 49] that have proved that with the insertion of the sensor the sensitivity also decays, as the hemorrhage produced decreases the chemical reaction effects.

So, it seems reasonable to consider that the sensitivity is a bit lower the first day or two days, to increase later to its highest value and to start decreasing after another couple of days due, e.g., to biofouling, until the end of the sensor life, which currently is around one week. The amount of this decrement is not well known, but it seems reasonable to consider it in the interval [20,50]%.

One of the few works that define an equation for these variations is [42]:

$$\delta(k+1) = 3\delta(k) - 3\delta(k-1) + \delta(k-2) + w_1(k), \tag{6.1}$$

where $\delta(k)$ represents the sensitivity variations at instant k and w_1 is a white noise with null mean. Thus, the sensitivity variations correspond to an third order integration of a white noise.

Among the diverse set of curves produced by this stochastic model, one that changes according to the estimated decay with time has been chosen to represent the sensor's sensitivity variations with time, Fig. 6.2.

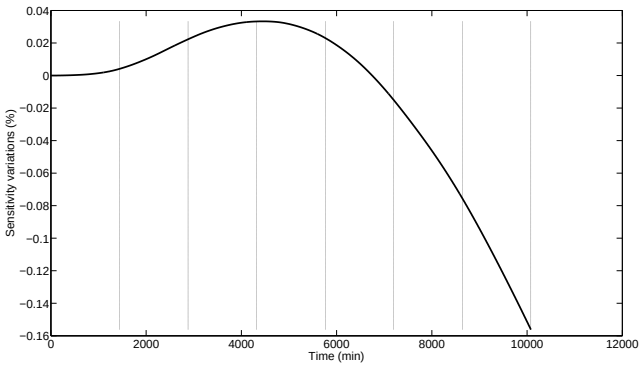

Figure 6.2: *Shape of the sensitivity variation for 7 days, in the range of 20%*

Thus, variations of sensor's sensitivity must be considered in two ways:

- ◇ **inter-patient variations:** a different constant S_p defined for each patient i (S_{pi}) defines the relation between current intensity (I) and interstitial glucose (IG)($I = S_{pi} \cdot IG$).
- ◇ **intra-patient variations:** The sensor gain S_{pi} varies with time accordingly to the curve depicted in Fig. 6.2, randomly rescaled to represent variations in the range [20,50]%. This interval seems sensible based on simulation results from [42].

6.2.2 Information available

A very important issue that has to be considered is the availability of information. When the model is identified for the set of training patients, information about the output (plasma glucose) is available, samples are frequent and regular. This is necessary to identify an accurate model of the system, as with few samples the model would be neither valid nor representative.

But in a usual case, information about the output is not frequent and that is the reason for the development of CGMS. However, adaptation can only be done with reference samples with which performing the comparison between model estimations and actual values.

In their daily life, patients with diabetes take around four control samples of glucose per day to see their level of glucose at important/critical instants. These samples are taken with glucometers in intervals of 6-8 hours. Therefore, to design the ACA with real life conditions, the adaptation of parameters can only be carried out using these control measurements as reference samples.

This means that, considering a realistic case for patients with diabetes, the adaptation will have to be done considering information included in just a few measurements per day.

However, in CGM devices it is necessary that an estimated output is given every 5 min. Thus, with each new reference measurement taken by the patient, the adaptation scheme has to perform the estimation of the normalization parameters to get the estimation of the CA with the normalized signals as close as possible to the reference value. In the meantime, when no new references are measured by the patient, the adaptation can still be performed offering an estimation (suboptimal) every 5 min, but no new information is considered. Then, if the adaptation algorithm is fast enough, it is common that between reference samples, the normalization parameters do not change or not change much.

Although only 3-4 samples per day are used for adaptation, it is highly important to consider also the results of estimation between reference samples during the validation phase. If the normalization parameters are adapted in such a way that the output of the model is very close to the reference values in just a few points it is not clear that the estimation of the model is close to the real output in all points in between.

Performance of the estimation algorithm has to be evaluated in all the range, given that being only accurate in just a few points is not desired. For this reason, although the adaptation is carried out with information of few points, all range has to be used to check the performance of this scheme. Indeed, the validation, to measure properly the performance of the adaptation, will have to be done using laboratory samples, which measure plasma glucose levels with low error.

The last issue to consider in this section, is that in reality samples of capillary glucose taken by glucometers have some error with regard to the plasma glucose, which is the final reference. This error is quantified in the interval [-10,10]% and there is no way of removing it. Thus, the adaptation of normalization parameters will inherit this error, and has to offer a good estimation in spite of it.

This is not the ideal case, but it is what happens in reality, where the estimations at home of plasma glucose are given by glucometers which still have this error. The performance of the adaptation will have to be checked under this conditions, as it is what is expected in the daily life of new patients where this ACA is to be used.

6.3 Framework

As seen in Section 2.3 and in Chapter 5, the most used inputs in this system are the mentioned intensity of the current signal from the sensor in several time instants (I_k, I_{k-1}, etc.) and also some estimations of glucose in previous instants ($\hat{G}_{k-1}$, $\hat{G}_{k-2}$, etc.). As one of the inputs is the same as the output (but delayed), only four parameters are required for the normalization: *mean* and *variance* of the current signal (m_i, σ_i^2) and *mean* and *variance* of the glucose signal (m_g, σ_g^2). For the subject normalization these parameters are subject dependent, p (m_{pi}, σ_{pi}^2, m_{pg}, σ_{pg}^2).

Therefore, the generalization of the model identified for a set of subjects with individualized normalization is possible with the adaptation of normalization parameters. The adaptation of each parameter to improve adjustment of the new signal vector to the computed model can affect in

a different way and influence of each of them can be in same or opposite directions to others. For this reason, the first characteristic of this scheme is that the **adaptation** will be done **simultaneously** using the **full set of parameters**:

$$pm = \left[\begin{array}{cccc} m_i & \sigma_i^2 & m_g & \sigma_g^2 \end{array} \right] \tag{6.2}$$

If the aim of this adaptive CA is to enable the application of the model for a new subject, then it is needed that this **adaptation** can be **computed on-line**. CGM systems take a current measurement every 5 min, computing the output (glucose estimation level) at that time. This means that between measurements and estimations, there is a lap of 5 min where the adaptation will have to be done, to be able to offer a value for the output when that time elapses.

Another issue that has to be taken into account is what is expected of this adaptation. The adaptation of normalization parameters is done to re-scale the inputs/outputs in the correct range. If the normalized inputs/outputs are not in the correct range, when the identified model is applied to them the estimation of the output will have much error. On the contrary, when the inputs and outputs are correctly normalized the model will have small error.

This means that the difference between estimated and real glucose will be small when signals are normalized with correct values of parameters. The range of the error for correct normalization parameters should be similar to the error of current glucometers.

Thus, it makes sense to use the difference between the "actual" glucose and the estimation by the model applied to the normalized signals as the reference of the quality of normalization. Thus, the adaptation of normalization parameters (pm) is done to minimize the difference between both signals. A small error indicates that normalization parameters reached a value close to real one. As said before, the "reference" glucose will be taken from the glucometer measurements to perform the adaptation of parameters, given that it is the information available to patients at home.

Given that one of the common measures used for the quality of the CGM algorithm is the **MARD** (Mean Absolute Relative Deviation), it is convenient to use this measure as the **function of the error to minimize** with the adaptation of parameters:

$$MARD = \frac{\sum_{k=1}^{n} \left| \frac{y_k - \hat{y}_k}{y_k} \right|}{n}, \tag{6.3}$$

where y_k refers to reference samples taken from the glucometer while $\hat{y}_k$ refers to the estimated plasma glucose done by the model. n is the total number of samples.

Figure 6.3 shows a block diagram of the estimator of normalization parameters. It is based, as already said, in the minimization of the MARD between reference (glucometer) samples and the estimations of the model with individualized normalization. This comparison is done for the time instants where there are calibrations, Tc, and in samples within a given time window, denoted by wi. Details of the time window can be seen in Section 6.5.3 of this chapter.

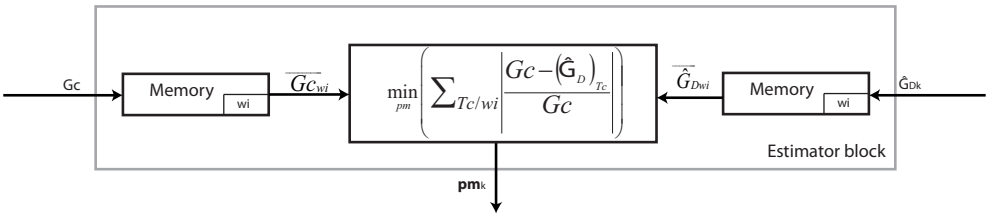

Figure 6.3: *Components of the block for estimating the normalization parameters*

6.4 Selection of the optimization technique

The best optimization technique must be chosen for the minimization of the MARD function. In Section 4.5, a review of optimization methods was done.

In the case under consideration, the problem is uni-objective, and, therefore, the techniques based on evolutionary algorithms will not be considered as their potential might exceed the complexity of the problem.

On the other hand, in the minimization of the MARD function, it is very important to offer a value every 5 min. Besides, the software where the optimization would be implemented (CGMS) is a commercially available microprocessor, so no complex requirements might be supported. For this reason, application of local optimizers makes much more sense in this case, as they are, probably, the only option for the system requirements.

Thus, according to the application requirements, the use of the *gradient based* optimization methods seems appropriate. The gradient method is a good choice, mainly because it is fast computing and can reach the (local) optimum

value in not many iterations. Besides, this application has four input variables, vector pm, which is a good number [102] for this type of methods.

Equations of parameters update is like shown before, but adapted now to the case:

$$pm_{k+1} = pm_k - \alpha \cdot G_f \tag{6.4}$$

G_f represents the gradient of the objective function, f_{MARD}, with regard to the input parameters:

$$G_f = \frac{\partial f_{MARD}}{\partial pm} \tag{6.5}$$

It is important to mention, that differentiation of the last equation with respect to the vector of independent parameters means the differentiation of the equation with regard each parameters independently. To consider all changes simultaneously, update of all parameters must be done simultaneously in each iteration.

Particularization of these equations taking into account all normalization variables, will be shown in next section.

6.5 Application of the gradient method

After the selection of the optimization method to perform the adaptation of parameters, its correct fitting for the application of the concerned case is to be done. This involves considering the specifications of the problem, tuning the parameters of the algorithm, and so on.

6.5.1 Equation to minimize

As said before, this optimization method is based on an update of the parameters based on the negative direction of the derivative of the function in each sample, following equation 6.4. In this way parameters tend to the value that gives the minimum of the function. For this reason, the first thing to do is to define the function to minimize in this problem and to compute its derivative.

The function to minimize is the MARD function, Equation (6.3), which is the absolute relative deviation between the estimated output ($\hat{y}$) and the reference output y_r. The estimated and denormalized output of the model (used for computing MARD) is renamed as $\hat{y}_D$, subindex D has been added to make clear it is the denormalized value the one considered, see Annex A. In

this application the reference output (glucose) comes from the samples taken from glucometer, as mentioned before.

Generic nomenclatures has been used: y for the output (glucose or G) and x for inputs (current intensity or I and previous glucose estimations or $\hat{G}$). Normalization variables will be generalized like: m_x, σ_x^2, m_y and σ_y^2.

Obviously, the denormalized output depends on the output normalization variables (see Annex A): $\hat{y}_D = (\hat{y}_N - off) \cdot \sqrt{\sigma_y^2} + m_y$. Substituting in 6.3:

$$MARD = \frac{\sum_{k=1}^{n} \left| \frac{y_k - \left((\hat{y}_N - off) \cdot \sqrt{\sigma_y^2} + m_y \right)_k}{y_k} \right|}{n} \tag{6.6}$$

The normalized estimation $\hat{y}_N$ is the output of the model, where the normalized inputs are included to the computation of the same. To see how they affect, the simpler case of two inputs will be considered as illustration. One of the inputs, x_1, will be an independent variable while the other, x_2, will be the same output estimation but considered in other time instants, section 5.3:

$$x_{1N} = \frac{x_1 - m_x}{\sqrt{\sigma_x^2}} + off \tag{6.7}$$

$$x_{2N} = \frac{x_2 - m_y}{\sqrt{\sigma_y^2}} + off \tag{6.8}$$

To avoid excessive complexity a system of only one local model is considered. Then the membership function, the linear regression and the total estimation of the model for each instant k are respectively:

$$\mu = e^{-\frac{1}{2}\left(\left(\frac{\frac{x_1 - m_x}{\sqrt{\sigma_x^2}} + off - c1}{\sigma 1} \right)^2 + \left(\frac{\frac{x_2 - m_y}{\sqrt{\sigma_y^2}} + off - c2}{\sigma 2} \right)^2 \right)^H} \tag{6.9}$$

$$LM = \beta_1 x_{1N} + \beta_2 x_{2N} + \beta_0 \tag{6.10}$$

$$\hat{y}_N = \left\{ \begin{array}{l} e^{\left(-\frac{1}{2}\left(\left(\frac{\frac{x_1 - m_x}{\sqrt{\sigma_x^2}} + off - c1}{\sigma 1} \right)^2 + \left(\frac{\frac{x_2 - m_y}{\sqrt{\sigma_y^2}} + off - c2}{\sigma 2} \right)^2 \right)^H \right)} \cdot \\ \cdot \left(\beta_1 \left(\frac{x_1 - m_x}{\sqrt{\sigma_x^2}} + off \right) + \beta_2 \left(\frac{x_2 - m_y}{\sqrt{\sigma_y^2}} + off \right) + \beta_0 \right) \end{array} \right\} \tag{6.11}$$

$c1$, $c2$ are the coordinates of the center of the hypergaussian function, while $\sigma 1$ and $\sigma 2$ are the variance in each direction of the distribution.

The final equation for the MARD results by substituting the estimated normalized output $\hat{y}_N$ in equation 6.6.

To derive the final MARD equation with regard to each of the normalization variables (m_x, σ_x^2, m_y and σ_y^2) is quite complex. Indeed, this is only an illustration case, where just few variables are present. Obviously, the more inputs the system has and the more LM, the more complex these equations would be. In general, the derivative of the whole MARD expression is not easy to get and implement.

Instead, it seems more appropriate to compute the numerical approximation of the derivative at the required point. These approximations are accurate enough if the considered interval is also small enough.

Numerical derivative of a function $f(x)$ ($f : \mathbb{R} \to \mathbb{R}$) is computed as the quotient between the variations of the function and the variations of the variable for an increment of the variable d:

$$\left.\frac{\Delta f(x)}{\Delta x}\right|_d = \frac{f(x+d) - f(x-d)}{(x+d) - (x-d)} = \frac{f(x+d) - f(x-d)}{2d} \tag{6.12}$$

Only two numerical values of the function have to be computed, always considering d small enough to assure the validity of this approximation.

Each of the variables will be considered independently for the computation of the numerical derivative while all the changes in variable magnitude for optimization will be considered simultaneously.

6.5.2 Parameter tuning

In the gradient method there exists an important parameter to tune: the adaptation step, α. It defines the *speed* of the adaptation and the selection of a proper value of the same depends of the convexity of the problem.

In this case, the function to minimize is highly non-convex, and this implies that the adaptation step has to be small to assure the adaptation is smooth. Numerical values of this parameter will only be assigned with trial and error, and the maximum value of α that assures convergence is the best value for the step.

If the adaptation step adopts a too large value, instead of convergence of parameters it is likely to get oscillations on their value, never approaching an stable value, whether the optimum or not. For this reason it is always better

to be slightly conservative with this assignation and assure a convergence even at the expense of a larger computational time [102].

A modification of the bare gradient method is the **steepest descent** optimization. This method is based on an adaptation of the parameter α depending on the value of the derivative or what is the same, how much the function varies around the actual point. Assigning larger values for α when the function changes a lot (large derivative) or smaller values when function is almost constant makes the method tend to the optimum value faster, and reach an stable value

The ranges of the derivative to assign larger or smaller values for the adaptation step is also case-dependent and will have to be found also by trials and error. Steep-descent method is an improvement over the simple gradient based method because it still uses the same basic methodology but helped with information of the function and the case.

Finally, the value of the deviation of adaptation parameters to compute the numerical derivative has to be considered. This value has to be small to really reflect the derivative of the function at the point selected, but if it is too small it could happen that the direction of minimum is not reflected in the resulting direction. An appropriate value of d depends on the magnitude of the parameter. Thus, a variation around the 5% is selected in many cases.

6.5.3 Window of samples

The sensor is the most important element of the CGMS devices, as it is the one that offers the measurement related to glucose levels. Research on sensor's technology has allowed to reach, currently, a working life of around a week. Works on this field are still important and the aim is to obtain long life sensors, reaching a life longer than a month [65, 52]. This is to avoid frequent re-insertation of the device on the patients, as each time the sensor is changed the device has to be inserted again.

On the other hand, the adaptation of parameters is done to adjust the estimated samples to the real ones. Obviously, the larger number of samples included for this comparison, the more information included and the better adjustment of parameters to avoid the representation of only a few samples.

However, the larger number of samples included the longer the period of time considered, and therefore, the more expected variations of the sensor's sensitivity. If the number of reference samples included in the adaptation is large, representing the sensor's sensitivity variations of the whole period with only a value would imply a generalization of the value of parameters and a loss of accuracy in the estimation of normalization parameters.

For this reason, a compromise value has to be adopted for the number of samples included for the adaptation, taking into account that the larger the number the more the accurate estimation, but also that a large number means less flexibility to counteract sensitivity variations. Fixing a determined number of samples included for adaptation means to fix a **window** of time for the adaptation. Equation of MARD 6.3 is adapted to include this window and the final variables that are compared:

$$MARD = \frac{\sum_{Tc \in [n-wi,n]} \left| \frac{Gc - (\hat{G}_D)_{Tc}}{Gc} \right|}{n}, \tag{6.13}$$

Gc refers to capillary glucose, which is the reference, $\hat{G}_D$ refers to the denormalized estimation of the model and Tc is the time vector where calibrations are measured. wi is the length of the window and n is the total number of samples from the beginning until current instants. Thus, the *window* takes into account the final wi period of time, see Figure 6.4.

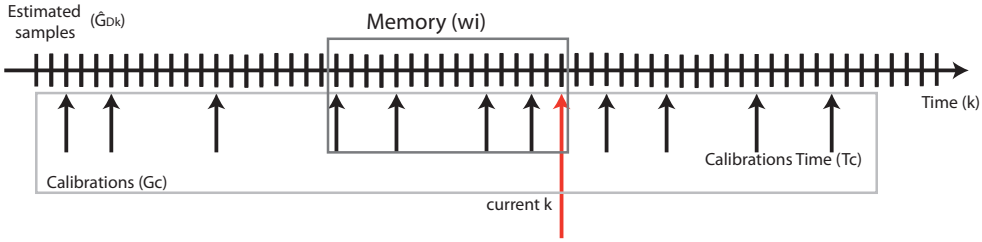

Figure 6.4: *Representation of the memory block which stores signals in the time window used for the estimation of normalization parameters*

The proper value of this window will be under study in a practical case. Nevertheless, it seems appropriate to try out values around a day, where 4 calibrations are included, because four calibrations is not a small number of calibrations and accuracy can be satisfactory. Besides, in a day time the variations of sensitivity are not expected to be extremely large and simplifying them by just a value is not a bad generalization.

6.6 Considerations for validation

In this system, the validation step includes two things. Firstly, it has to be validated that the adaptive scheme is performing adequately, and resulting parameters bring the normalized (and therefore, denormalized) estimation of glucose close to the real value of reference. The second and most important thing is that the validation has to check that normalization parameters that reflect the sensitivity for the used samples in the comparison, are also valid for the whole set of reference samples obtained from a laboratory and not used for the error minimization.

Then, although just a few samples are used for the computation of parameters, frequently taken reference samples are required to see the validity of this scheme.

The use of only a few capillary measurements to perform the adaptation over the whole signal range, make it necessary to choose the adequate samples to be able to "discover" the general behavior of the signal between samples.

The few chosen samples should include as much information as possible about the general signal variations. If measurements are all taken under similar conditions and states (i.e. glycemic levels, rate of change, etc.) it is likely that they are not representative of the full signal. The more differentiated conditions the measurements are taken, if possible under different glycemic/metabolic states, the more information included in those few samples about the general signal.

However, it could happen that even if the samples used are taken in different states (metabolic, glycemic, etc), the full signal is not represented by them. Then, adapted parameters are not valid for the rest of samples, producing a large error when compared to reference measurements.

In case the total error is larger than desired, the number of included samples should be increased until it is under the acceptable threshold. What could happen is that the number of samples that has to be included is too large to be a realistic case, and this would invalidate the whole structure.

For this reason it is very important to study the timing of capillary samples containing more information about the signal in order to include the minimum number of them, as the aim is to design a structure that can be applied and used in a realistic case. This is done in a later study, where different strategies for calibration (introduce the capillary reference samples) are tested and also different number of validation (plasma glucose) points.

Next Sections will show the application and validation of the proposed adaptive calibration algorithm (ACA) in two data sets. The first one is a data set with real information for several hours while the second one is an in

silico database, with one-week's information. Previous to the application of the ACA, the model has to be identified.

6.7 Model identification with real data set

This section describes the first step before the designed adaptive calibration algorithm is applied: the model identification, using either set or individual normalization.

It is focused on checking the performance of the proposed modeling algorithm, $PNCRM$, in the real data set described before. Both types of normalization are used, as they were applied in the previous application (see Chapter 5). First of all, the identification of the model will be done with data normalized by set parameters. This means the same parameters of normalization are applied for all patients.

Secondly, this methodology is applied when each patient is normalized by its own parameters, to compensate effects of sensor's sensitivity variations between patients. Again, the set of LMs is identified for the set of normalized patients.

Obviously, the denormalization of the output of the model (glucose estimations) in each case is done according with the normalization used for the data.

Finally, the proposed adaptive scheme is applied to the patients, using the set of LMs identified for the individual normalization, as if they were new patients (their parameters are not known), to see its performance

6.7.1 Clinical data study

17 patients with type 1 diabetes were recruited at Dr. Josep Trueta Hospital (Girona, Spain) and were asked to wear CGMS Gold (MiniMed CGMS MMT-7102; Medtronic, Northridge, CA). The device, which uses a retrospective calibration algorithm and provides plasma glucose estimates every 5 minutes, was calibrated with SMBG at least 3 times per day, following manufacturer's instructions.

Each patient underwent a 9 hours in-hospital study, where plasma glucose was measured by means of a Glucose Analyzer II (Beckman Instruments, Brea, CA). Samples were taken every 15 min for 2 h after each meal and every 30 min otherwise.

To perform the model identification, plasma glucose measurements were interpolated using a cubic method every 5 min (sample period of CGMS Gold readings). Thus, a total of 1719 paired points were obtained. Because plasma

glucose and CGMS Gold readings were obtained at different times, CGMS Gold readings were interpolated to the gold standard reference time vector to match both vectors in time.

Some gold standard samples (14 samples in 5 patients) were incorrectly measured (drastic change from previous and posterior values). In this case, they were interpolated with the other signal samples to avoid gold standard outliers.

6.7.2 Considerations

The variables used in this case as inputs are the same as in the previous experiment: intensity of current from the sensor signal, I, glucose in plasma samples (references), G, and glucose estimations given by the global model $\hat{G}$. Subindex i refers to the i-th local model, k is k-th time sample, and j refers to the input dimension. Finally the subindex N will refer to normalized signals I_N, G_N, $\hat{G}_N$, by any of the normalization methods proposed.

One thing that has to be mentioned is that it is needed the introduction of a calibration at the beginning of the computing. Thus, at time $k = 1$ $\hat{G}_0$ will be replaced by G_0. As the time computed for each patient varies between 7 and 9 hours, it makes sense to introduce another calibration, as patients usually take a calibration every 6h (4 calibrations per day). This calibration is introduced for all patients just before dinner (at 7p.m.) where all of them have a calibration point from glucometer. This point is chosen because it is taken at equilibrium conditions, where the differences between this measurement and blood glucose are minimal.

6.7.3 Model identification with set normalization

The application of this modeling methodology as calibration algorithm was previously done in Chapter 5 [12] but for healthy people. In the current study the same procedure was carried out in patients with type 1 diabetes. However, the model identified in [12] was not directly applicable to these data. This can probably be due to the small number of patients and the differences on the nature of people (here patients have type 1 diabetes), as well as the sensor used (microdyalisis versus needle-type). For this reason the structure and parameters of a new model had to be identified again.

Starting from the structure [I_{Nk} I_{Nk-1} $\hat{G}_{Nk-1}$] (called IIG) and two local models, c=*2*, several experiments were performed. Finally, it was proved that the structure [I_{Nk} I_{Nk-1} I_{Nk-2} $\hat{G}_{Nk-1}$] (called $I3G$) with c=*2* gave the best results with the minimum number of inputs considered and the lower number

of local models. Results are shown in Table 6.1, columns $N1$, $N2$, respectively.

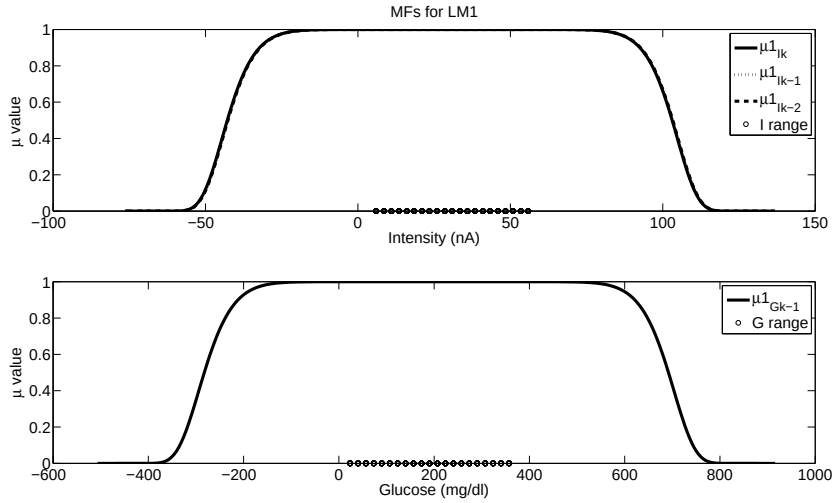

Figure 6.5: *Projected validity function of LM1*

Figs. 6.5 and 6.6 show the validity functions of each local model. There is a principal local model (LM1), descriptor of the main dynamics of the signal and a LM that contributes sometimes (LM2). This contribution only takes part when certain conditions are met. The study of the region covered by the validity function for LM2, reveals from its projections (Fig. 6.6) that this LM contributes when the glucose is in hypo or low-normoglycemic region and the current is recovering from a low level (in I_{k-2}).

In this case, identified local models do not correspond to different subsets of patients. If sensor's sensitivities are analyzed for this set of patients (see Figure 6.7), it is clear that the distribution of the sensor's sensitivities is unimodal. Thus, the different models found correspond to differences in dynamic behavior.

6.7.4 Model identification with individual normalization

As said before, the use of set normalization can hide the effects of the different inter-patient sensor sensitivities, making difficult to find the insterstitium-to-plasma dynamics. For this reason, individual normalization may be more

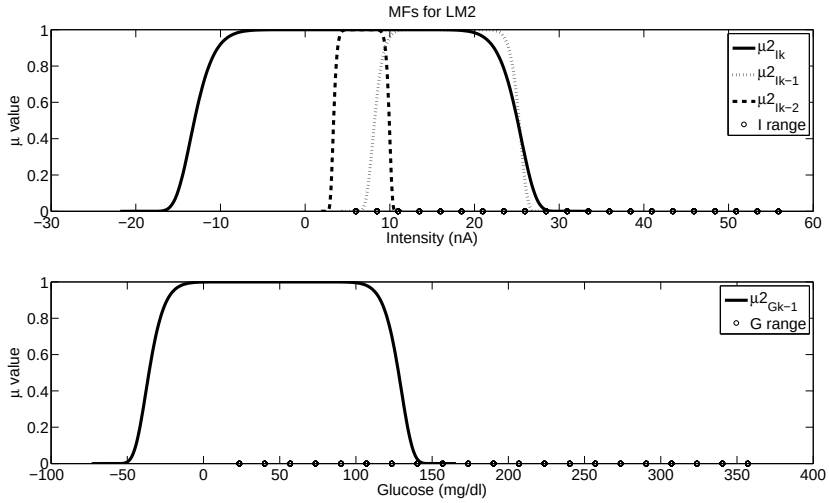

Figure 6.6: *Projected validity function of LM2*

suitable reducing the impact of inter-patient sensitivity during the modeling phase.

The identification of the model was done using the structure IIG (see Section 5.6) and *c=1*, *c=2*. Results are shown in Table 6.1, columns $N3$, $N4$, respectively.

Table 6.1: *Results: % of well estimated samples according to ISO criteria and MARD, both for the whole range and hypoglycemic range. Case **N1** is the set normalization, with IIG inputs and c=2. Case **N2** is the set normalization, with I3G inputs and c=2. Case **N3** is the individual normalization, with IIG and c=2. Case **N4** is the individual normalization, with IIG and c=1.*

	CASES				
Parameter	**CGMS Gold**	**N1**	**N2**	**N3**	**N4**
	All range				
Ok ISO	77.4%	73.4%	78.4%	90.7%	90.5%
MARD	15.1%	15.1%	14.5%	8.9%	9.7%
	Hypoglycemic range				
Ok ISO	71.9%	57%	73.3%	89.6%	91.9%
MARD	21.6%	23.9%	18.8%	11.5%	11.3%

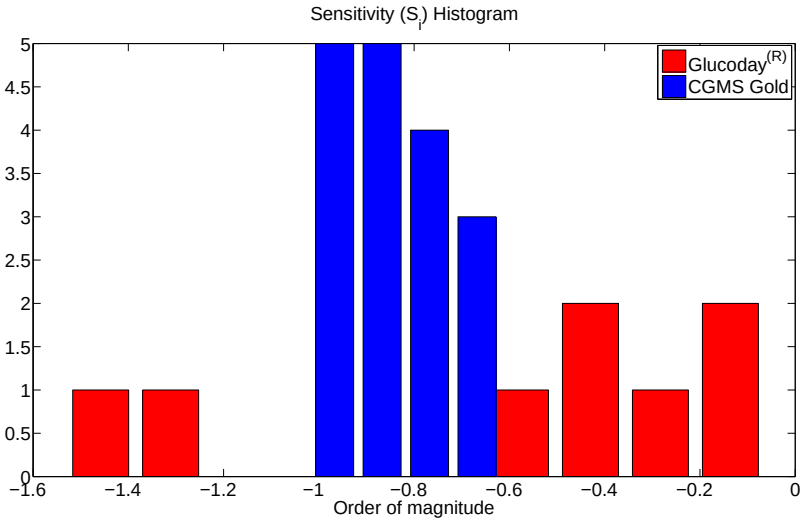

Figure 6.7: *Histogram of sensor's sensitivities by its order of magnitude from Glucoday$^{(R)}$ and from the CGMS Gold monitors.*

It is clear the improvement of the performance with regard to the set normalization, as it was already demonstrated in the previous application of PNCRM. This confirms results and conclusions obtained: when the issue of the sensitivity variations is removed and only the part of finding the dynamics between compartments is left performance drastically improves.

After the model identification, the performance of the adaptive calibration algorithm can be tested.

6.8 Validation of the adaptive calibration algorithm

This section is divided into the two cases where this adaptive calibration algorithm (ACA) is tested: the real data set and the in silico data set.

6.8.1 Application to real data

This refers to the application of the identified model for individual normalization to a new patient, where the vector of normalization parameters pm is unknown. pm was initialized using the mean of the individual parameters of the population used for the identification of the model. After

initialization, parameters were adapted to minimize the difference of the estimation of glucose and the reference measurements (calibrations). It is important to mention that although the optimization is done using only calibration points, the performance has to be tested on the full set of gold standard data. This implies that the larger number of calibrations, the more information available to perform this adaptation. Nevertheless, it is not realistic to think that any number of calibrations will be provided by the patients.

The model used is the one with just one LM, column $N4$ in Table 6.1, as its performance is very similar to the case of $c = 2$, column $N3$. Indeed, when the effects of the sensor's sensitivity are compensated through the individualized normalization, the available data is not enough to identify very different behaviors. Thus, the performance of the adaptive calibration algorithm can be properly tested in this case taking the basic structure with $c = 1$.

The study is done using the patients available. Now, their parameters are considered unknown, but the model already identified is taken and used. pm is initialized using the mean of the individual parameters of the set used for the identification of the model.

Study of calibration strategies

A study is done using different calibration strategies and different number of calibrations. All calibration strategies employed a realistic number of calibrations (3-4 per day), as using a larger number of calibrations just for one day is not realistic considering standard conditions of people with diabetes, where they take measurements with the glucometer around that number of times per day.

To see the performance of the different calibration strategies the model identified with IIG input configuration and $c = 1$ has been used and the variable compared has been the MARD for all samples. Calibrations are included in times that can be considered relevant:

1. Well spaced in time: *i.e.* a calibration every 4h. This means to include 2 calibration per patients (at T=4h,8h) as data are collected for 9h (named $CS1$).

2. The same number of calibrations but at the beginning and spaced less time: 2 calibrations included every 2h (T=2h,4h) (named $CS2$).

3. One additional calibration: 3 calibrations included every 2h (T=2h,4h,6h) (named $CS3$).

4. Consideration of calibration at important times for dynamics. At lunch one at pre-prandial time (just before eating) and another around the peak time (considered 1:30h after eating) ($CS4$).

5. Consideration of calibration at other important times for dynamics: at lunch and dinner pre-prandial times (named $CS5$).

6. Same as 4 but considering lunch and dinner. A total of four calibrations are considered (named $CS6$).

It is important to note that in all these configurations an extra calibration is included for the start-up of the algorithm. In a general case, it is only required for the first day. Results of these cases can be seen in Table 6.2.

Table 6.2: *Comparative of resulting MARD of all samples estimation given by the proposed ACA for different calibration strategies: different times and number of calibrations*

	CASES					
	CS1	**CS2**	**CS3**	**CS4**	**CS5**	**CS6**
MARD (%)	18.87	17.49	15.72	15.92	18.22	13.7

From these results several conclusions can be extracted:

- ◇ Comparing $CS1$ and $CS2$ it can be seen that with the same number of calibrations performance is better when information is introduced earlier ($CS2$).

- ◇ Comparing $CS2$ and $CS3$ it can be seen that with more calibrations included (more information) the performance improves, as expected.

- ◇ Comparing $CS4$ with $CS2$ and $CS3$ it can be seen that performance when using only two calibrations but at moments where more dynamic information is included is comparable to using three calibrations ($CS3$).

- ◇ Comparing $CS5$ with $CS4$ it can be seen that the use of two calibrations at pre-prandial times does not perform as well as when using one pre-prandial and one peak calibrations.

- ◇ Comparing $CS6$ with the rest, it can be seen that doubling the structure $CS4$ considerably improves the results as information in a wider range of the signal is included twice.

Results of ACA in real data

Finally, the one with best results and feasible for patients is the calibration strategy $CS6$, which uses a total of four calibrations per day, plus an initial calibration for the algorithm (for the start up). The four calibrations were taken at preprandial (just before they eat) and 1:30h postprandial times for lunch and dinner. Finally 15 patients were considered, as two patients did not have information for lunch time. The models considered here were IIG with just one LM, column N4 in Table 6.1, (its performance was very similar to the case of c=2, column N3), and I3G with c=1. Results are shown in Table 6.3. The time allowed for the optimization was 10s between samples, which in a processor running at 2.67GHz it is equivalent to 5 min in a conventional 100MHz microprocessor.

The improvement of the performance of the adaptive calibration algorithm with regard to the monitor used is clear and goes up to a reduction of MARD of 2% for the I3G configuration.

Table 6.3: *Results: % of well estimated samples according to ISO criteria and MARD, both for the whole range and hypoglycemic range. Case **N1** is the individual normalization model for IIG and c=1, plus the adaptive scheme (4 calibrations). Case **N2** is the individual normalization model for I3G and c=1, plus the adaptive scheme (4 calibrations).*

	CASES		
Parameter	**CGMS Gold**	**N1**	**N2**
	All range		
Ok ISO	77.4%	76.7%	78.4%
MARD	15.1%	13.7%	13%
	Hypoglycemic range		
Ok ISO	71.9%	81.6%	87.8%
MARD	21.6%	15.8%	11.7%

Graphic examples of the evolution of the estimated signals are shown in Figure 6.8, for the most representative patients (numbers 3 and 10).

6.8.2 Application to in silico data

The real data available is around 9h per patient, and full performance of the adaptive scheme cannot be tested. For this reason, this scheme will be fully tested with one-week in silico data and a larger population of 30 patients.

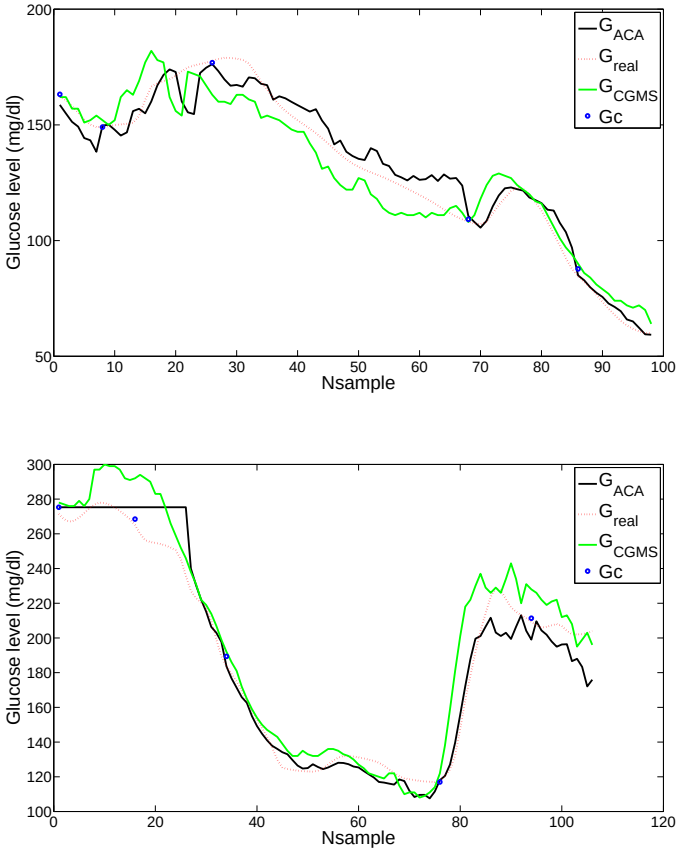

Figure 6.8: *Comparison between glucose reference signal (G_{real}), the estimation of the adaptive calibration algorithm (G_{ACA}) with input structure I3G, and the estimation given by CGMS Gold monitor. It is marked where calibration points (Gc) have been introduced. Upper graphic corresponds to Patient 3 while the graphic on the bottom corresponds to Patient 10*

The use of longer time data (one week) can contribute to check different things. Firstly, convergence of the parameters. Secondly, the performance of this scheme with the variations of sensitivity over time.

The structure applied is the same as in the real data case. The calibration algorithm used is the one that uses IIG input matrix and $c = 1$, Table 6.5 column $N2$. The model identified for the first day was tested the following days (2 to 7), emulating CGM operation.

With longer data, new aspects of the CA can be studied. On the one hand, the performance of new calibration strategies can be tested. On the other hand, the necessity and application of a windows of samples will be validated.

In silico data generation

To generate one week's data, the FDA-accepted UVa simulator was used [89]. The signals available in this simulator are: interstitial glucose (IG) and blood glucose (BG). CGMS devices do not measure directly BG. The current signal reflects IG levels, with variations of sensor's sensitivity as already described.

The sensitivity variations were reproduced following indications explained in Section 6.2.1 of this Chapter.

Thus, to generate one-week data, IG signal coming from the simulator was converted to I measurements ($I = S_p \cdot IG$), following these indications:

1. A different sensor's sensitivity (S_p) was considered for each patient (p). This sensitivity was taken randomly in the range [0.09, 0.24], obtained from the 17 patients of the previously described clinical study.

2. A degradation of this sensitivity was considered following the perturbations shown in Fig. 6.2 and scaled randomly for each patient within the range [20,50]% for each patient.

Thirty patients (adults) were considered for this in silico study, generated from the 10 adults available in the educational version of the simulator and the procedure described above to assign sensor's sensitivity (every patient was studied three times with "different sensors"). A period of one week was simulated with three meals per day: breakfast (7a.m.), lunch (12p.m.) and dinner (18p.m.). The meals were similar for all patients and days, with variations up to 40%. Four calibrations were considered per day (preprandial and 1:30h posprandial for breakfast and dinner). To emulate glucometer accuracy, a variation of ±10% was introduced to BG samples generated by the simulator.

Study of calibration strategies with in silico data

Calibration strategy that best performed in real data was using a calibration just before eating (pre-prandial) and a calibration 1:30h after eating (postprandial) both at lunch and dinner. Calibrations performed at those times include dynamic information of the signal in addition to the static value.

Other calibrations strategies are tested with this data set. The longer time data can contribute to a better study of the best moments for calibrating. Four calibrations per day will be used to do the adaptation of normalization parameters, as it is a realistic number of calibrations for people with diabetes. The new strategies tested are:

1. Covering a larger time interval: Pre-prandial and post-prandial for breakfast, pre-prandial for dinner and one more calibration at night time.

2. More distributed in time: three pre-prandial calibrations (breakfast, lunch and dinner) and one more calibration at night.

3. More distributed in time with dynamics: three pre-prandial calibrations (breakfast, lunch and dinner) and post-pandrial (breakfast or lunch or dinner).

Performance of all these configurations for calibration has been checked for ten patients. Good results are obtained with some of them, but the one that best performs is still the one with a pre-prandial and a post-prandial calibrations for breakfast and dinner. This confirms the results obtained in the real data study, where other strategies were compared.

Definition of time window for adaptation

In Section 6.5.3 the need of a time window for the definition of the minimization criteria during adaptation was explained. The sensor's sensitivity varies with time. Using individual normalization the effects of this variations can be compensated. Yet, if the time considered is long the signal will vary much. Then, simplifying all this variations by just a the value will not reflect these variations.

Besides, the minimization of the MARD function is done using a numerical approximation, for complexity reasons. In addition, the computation of the model output is recursive, because previous estimations of the output are included as inputs of the system in later instants. These two things together mean that the larger number of samples considered for the minimization of the MARD function, the larger the time needed for computation will be. The time for computation is fixed to 5 min (time between samples), then with less samples more iterations of the optimization can be performed.

The best number of calibrations that the window should include is to be found. It has to be a compromise to avoid too long time windows and too few calibrations that do not characterize the signal. Experiments with several

values for the window (wi) were performed ($wi = [4, 5, 6, 8]$), and results can be seen in Table 6.4.

Table 6.4: *Comparative of MARD of all samples estimation for different calibration strategies: different times and number of calibrations*

	Range	CASES			
		wi=4	**wi=5**	**wi=6**	**wi=8**
MARD (%)	Overall	8.63	8.79	9.04	9.64
	Hypoglycemic	13.52	14.54	13.79	14.82
Std MARD (by Patients)	Overall	3.17	3.18	3.32	3.51
Std MARD (by Days)	Overall	1.27	1.36	1.92	2.81

The minimum window considered is four calibrations, because four are the calibrations introduced per day and with $wi = 4$ information of a full day is included. Results indicate that for a larger window the MARD obtained increases compared to a smaller window. Similarities of the standard deviation computed by patients and by days indicate that results of all experiments are quite homogeneous for all values of the window. Attending to these results, four will be the number of calibrations included in the final time window ($wi = 4$), to have information of a full day dynamics. With these number of calibrations the signal characteristics are well considered, the sensitivity does not change much and an acceptable number of iterations can be made.

Results of ACA applied to in silico data

Finally, the adaptive calibration algorithm was based on four calibrations were made per day, at pre-prandial and 1:30h postprandial times for breakfast and dinner, plus an extra calibration for the first day for the start up of the algorithm. The time allowed for the optimization was 1s between samples, which in a processor running at 2.67GHz it is equivalent to 5 min in a conventional 10MHz microprocessor. A window of four calibrations (one day) was considered to do the optimization for the parameters adaptation.

Table 6.5 shows the results for the adaptation algorithm (column $N4$). Results are compared with the application of the model with set normalization (identified for the 1st day) to the rest of the week days, column $N3$. A graphic example can be seen in Figure 6.9.

Columns $N1$ and $N2$ show the results of the model identified for the first

Table 6.5: *Results: % of well estimated samples and MARD, both for the whole range and only for hypoglycemic range. All results are for 30 patietns. Case **N1** is model with set normalization, IIG inputs and nc=2 for Day1. Case **N2** is model with individual normalization, IIG inputs and nc=1. Case **N3** is the application of **N1** to Days 2 to 7. Case **N4** is the application of **N2** to Days 2 to 7 with the adaptive scheme.*

Measure of accuracy	Glycemic range	Calibration algorithm			
		N1	**N2**	**N3**	**N4**
ISO_{ok} (%)	Overall	99.6 ±0.87	100 ±0	74.7 ±8.74	87.6 ±11.95
	Hypo <75mg/dl	99.9 ±0.53	100 ±0	48.5 ±24.18	93.3 ± 9.33
MARD (%)	Overall	4.9 ± 1.69	0.2 ±0.04	16.3 ±3.9	10.9 ±5.79
	Hypo <70mg/dl	5.3 ±1.7	0.3 ±0.04	33.8 ±19.45	10.2 ±20.07
M2ARD (%) [range]	Overall	3.6 [1.33; 9.51]	0.15 [0.08; 0.27]	10.07 [7.69; 17.47]	7.4 [2.83; 17.1]
	Hypo <70mg/dl	3.32 [1.99; 4.31]	0.29 [0.23; 0.36]	28.28 [7.29; 61.07]	7.22 [1.89; 65.78]

day, with set and individual normalization respectively. It is clear that the model identified has good performance in both cases, yet the performance is better in the individual normalization case, where sensibility effects are considered through normalization. In this in silico study, the improvement of the adaptive calibration algorithm, column $N3$, versus the calibration algorithm computed using the set normalization, column $N4$, can clearly be seen, when both calibration algorithms are applied to the data of the rest of the days (days 2-7).

Improvements of performance of the adaptive calibration algorithm can be better seen in the hypoglycemic region. The adaptive algorithm takes into account the new information of inputs and calibration to update the parameters of normalization and, in this way, model is adapted to the information of the patient. In the calibration algorithm with set normalization new information is not considered and this is probably the cause that the results are much worse.

Thus, results obtained with long term data indicate, as well as the validation case of real data, that the performance of the adaptive calibration algorithm improves the use of set normalization, as new information is

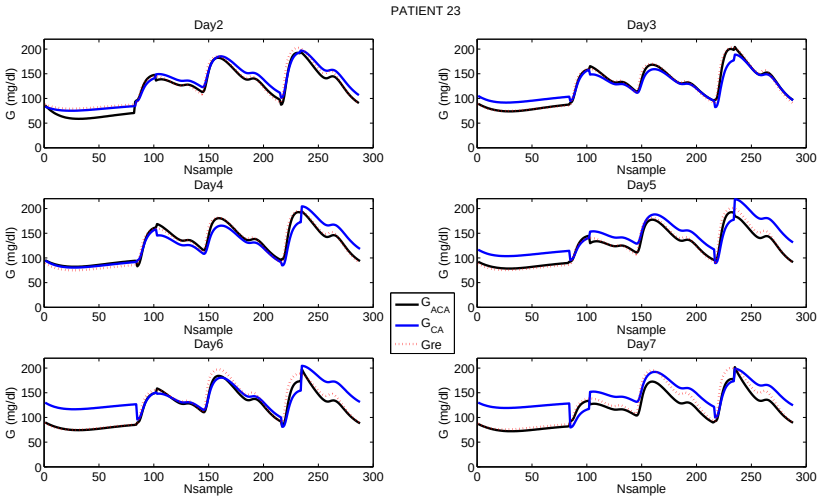

Figure 6.9: *Comparison between glucose reference signal (Gre), the estimation of the adaptive calibration algorithm (G_{ACA}), and the estimation given with the model with set normalization applied to days 2 to 7 (G_{CA}) for Patient 23*

considered for the adaptation of parameters and then, the model is adjusted to the new patient with the corresponding inputs.

6.9 Comparison of results from both studies

The MARD of the final adaptive calibration algorithm distributed by days can be seen in Table 6.6.

Table 6.6: *Distribution of MARD of all samples of 30 patients for days 2-7, when the adaptive calibration algorithm is applied using the model identified for day 1 with inputs IIG, $c = 1$ and individual normalization*

	TOTAL	Days					
		D2	**D3**	**D4**	**D5**	**D6**	**D7**
MARD (%)	10.89	12.99	11.2	9.04	9.53	9.87	12.7

The worst day is the first one. This might be due to the fact that the

algorithm starts with few information and progressively includes more samples until the time window is reached. From day one, performance of the next days is better, given that once the time window is reached from that moment on it is always considered. The ACA achieves to compensate the sensor's sensitivity variations and reaches a low MARD.

From day 4 MARD starts increasing again. This might happen because the sensitivity of last days decays more than the first few days. Indeed, with larger variations, the adaptation of parameters might not estimate the normalization variables as fast as they change. For that reason, MARD of last few days increases a bit. This results indicate that when sensitivity variations are larger the adaptation is slightly worse, as it could be expected. Yet, even in the day where sensitivity varies more (day 7) MARD reached is quite good, below 13%.

With these results from the in silico study, the results from the real data set can be analyzed in a different way. MARD of all samples in real data set reaches a value similar to the one of in silico data, looking at the first day of computation. This might suggest that if longer time of real data were available, results could be comparable to the ones of in silico data. This is only an indicator of performance, as real data is only available for one day and this can not be checked.

6.10 Conclusions

This study confirms that: 1) The use of a local-model-based calibration algorithm improves the glucose estimation for continuous glucose monitoring with regard to the actual monitors; and 2) The interpatient sensitivity variations affect substantially this estimation, and using individual parameters for normalization improves the performance of the proposed algorithm.

However, perfomance of individual normalization is not attainable since it makes use of information not available during CGM operation. Thus, an adaptive scheme is presented in this Chapter aiming at the real-time tuning of normalization parameters starting from average values in the population. In this way, an individualization is performed improving accuracy and compensating sensor's sensitivity variations over time. Two data sets are used for validation: 1) a real data set with 15 patients with T1 diabetes and data for around 9 hours; and 2) a in silico data set, with data of 30 patients and a duration of one week.

The proposed ACA is based on minimization of the MARD function. Several calibration points are used to do this comparison. A study of different

calibration strategies is done for first data set. Results indicate that the best option is including four calibrations: one at pre-prandial and other at post-pradial time in lunch and dinner. In this way static and dynamic information of the signal is incorporated. Finally, results in the case of real data indicate that the adaptive calibration algorithm works, performing better than the algorithm with no adaptation and set normalization. However, the adaptive algorithm does not perform as well as the ideal individual normalization. This could be expected, as the individual normalization uses the *best* values of each case.

For the second data set the performance of the adaptive calibration algorithm when sensor's sensitivity variations are taken into account is checked. This study reflects better a realistic case:

- ◇ Inter and intra-patient sensor sensibility variations are considered.
- ◇ Identification is done with data of just the first day.
- ◇ Adaptation is applied for data of rest of the days

Adaptation of parameters using the model identified with individual normalization improves the accuracy of estimations given by the identified model with set normalization, when applied to the rest of the days. Study of MARD by days show that the adaptation is capable of following the change of sensitivity and compensate its effects. In addition to this, the adaptation of parameters helps the compensation of the glucometer errors introduced with calibrations.

In some of the most recent works the effects of the sensitivity variations are also included. In Knobbe et al. [86] the interpatient differences on sensitivity are introduced as a constant scale factor for a time shorter than T and later it decays linearly. In Kuure et al. [92] the inter and intrapatient sensor's variations are considered, but these variations depend on the process noise and are not well characterized. Finally, in Facchinetti et al. [42], the sensitivity variations are characterized with a third order filter of the noise process and also included in the model. All these studies use the Kalman filter theory to perform the estimations of glucose. The main drawbacks of these calibration algorithms are exposed in the discussion of Chapter 5. Yet, it is worthy pointing out, that the Kalman filter has a difficult tunning of parameters for application.

Contrary to these studies, the proposed adaptive calibration algorithm has been designed to avoid a difficult set up stage. For identification of the model, the structure of $PNCRM$ is well described in Chapter 4. Only two parameters

are needed: 1) transition of membership functions, H, and 2) importance of local error versus global error γ. And the value for both is fixed in this work. For the design of the adaptive algorithm the practical conditions have been taken into account. Adaptation of parameters is done with a well-known technique, easy to implement. In addition to these, time allowed for adaptation of parameters is the time between samples of the actual monitors available in the market. Only a normal microprocessor is required. These conditions make the proposed ACA feasible to be implemented in commercially available CGM devices.

Indeed, one of the advantages of this study is that it has been validated using real data of patients with type 1 diabetes.

Recent studies [65, 52, 23] show that a research line with increasing importance is the design of long-term sensors, using different technologies. In [65] they designed a sensor based on fluorescent hydrogen fibers which responded to blood glucose concentration changes for up to 140 days. In [52] the life of the sensor reaches the year, although it is only checked in animals.

For these types of sensors, the adaptive calibration algorithm would suit much better its purposes.

6.11 Discussion

This study and validation of the proposed calibration algorithm shows a clear performance improvement of the individual normalization method with respect to set normalization, see Table 6.1. It is worthnoting that the consideration of the I3G model, even with set normalization, outperforms the CGMS Gold, especially in hypoglycemia. However, much better performance is obtained with a simpler model (IIG) and individual normalization confirming the influence of inter-patient sensor variability. In this case, a MARD below 10% is obtained when the whole glycemic range is considered, reaching a MARD reduction of about 35% with respect to the CGMS Gold. Regarding the hypoglycemic range, the improvement is even better, with a MARD reduction of 47% reaching values of about 11.5%. This was expected, as inter-patient sensitivity variations were compensated by individualizing the normalization parameters, eliminating thus a confounder of the dynamics between compartments. But this case is ideal, since individual normalization parameters are considered to be known. Thus, this performance is not attainable in practice and it must be considered as an upper limit of what can be achieved.

When the adaptive calibration algorithm (ACA) is applied to individual

normalization parameters, improvement can still be observed for both models (IIG and $I3G$), as shown in Table 6.3. It must be taken into account the clear limitation of the clinical data used due to its short duration and scarcity of capillary measurements to carry out the adaptation. Nevertheless, the adaptive scheme manages to decrement the MARD about 1.5% (9% reduction with respect to CGMS Gold) during the first day and four calibrations per patient. Improvement is more evident in the hypoglycemic range, with a reduction from 25% up to 45% in the MARD, depending on the model considered. Obviously, with just four calibrations it is difficult to converge to the optimal parameters. But this result points out that with availability of longer time data, this adaptation would reach a more significant reduction in the MARD. This is demonstrated in the in silico study.

For the one-week in silico data the identified model has good performance during the first day (identification day) for both normalization methods, as shown in Table 6.5, yet the performance is better in the individual normalization case. It must be considered that the use of simulated patients make much easier the identification task, reaching MARD values of 0.5% which obviously must not be expected with real data. Thus, figures in Table 6.5 must be interpreted qualitatively instead of quantitatively. The improvement of the adaptive calibration algorithm, column $N4$, versus set normalization, column $N3$, is evident, when both calibration algorithms are applied for the rest of the days (days 2-7). A reduction in the MARD of 33% is achieved when adaptation is considered, as compared to fixed set normalization, for the whole glycemic range. In case of hypoglycemia, the reduction reaches 70%.

Thus, results obtained with long term in silico data indicate, as well as the validation case with clinical data, that the adaptive calibration algorithm outperforms set normalization, as new information is considered for the adaptation of normalization parameters allowing for the model adjustment to the new patient characteristics or sensor's sensitivity variations.

It is important to see the performance gradation for the different calibration schemes. Obviously the model using the individual normalization is the one that performs best as the issue of the sensitivity variation is overcome. On the contrary, the model using the set normalization has worse results, as a mean sensitivity is used. And in the middle, the model with individual normalization with mean initialization and the adaptive scheme offers an intermediate performance, as the issue of sensitivity variations is progressively overcome.

Finally, this study indicates that the performance of current CGM devices can be significantly improved with the incorporation of interstitium-to-plasma

dynamics into the calibration algorithm. A local models approach allows for the description of local dynamics which can further help in a more accurate characterization of this dynamics. Together with the incorporation of adaptive capabilities, a significant reduction in the MARD is achieved. This is so even in the hypoglycemic range, where the lack of accuracy is an evident problem. The promising results obtained motivate further clinical studies with larger populations.

However, the study has several limitations. 1) First, the real study population counts only for 15 patients with diabetes and the duration of the signals is only of around 9 hours. This set of patients is small to be a well representative of the general population of people with diabetes. 2) Second, the clinical study to obtain the data did not consider the best conditions of signals for the identification of the model. This means that glucose variations are not equally distributed in all glucose ranges (hypoglycemia, normoglycemia and hyperglycemia). For a model fully representative of the dynamics of the signal, all regions should be equally considered in the identification of the model. 3) Finally, application on in silico data gives indications of performance for longer time and with more patients. However, this is a simulated study an conclusions can be considered only mere indicators of the performance of this adaptive algorithm.

Chapter 7

Conclusions

7.1 Conclusions

This work describes all steps taken to design new calibration algorithms for continuous glucose monitoring. The aim of this work is offering a good estimation of plasma glucose for patients with Type 1 Diabetes Mellitus. This is to help to control the high glucose levels that people with DM1 are affected with when poor controlled.

Diabetes Mellitus is a disease largely present in the society and that directly and indirectly causes several damages. For this reason, any option that helps patients with diabetes to have a better glycemic control would be welcome by this community.

In any system where the target is to control some variable, accurate measurements of that variable are needed first. Therefore, in the aim of developing an **artificial pancreas** the first step is to have a good estimation of the plasma glucose signal. And this is what is pretended in this work.

For this reason, a new calibration algorithm is proposed, based on the clustering and local model theory. This technique is chosen after the study of the interstitium to plasma glucose transport process. Local model technique is chosen as this work is based on the hypothesis that the glucose transport system can be formed by local behaviors and therefore, the identification of a set of local models can improve the accuracy of estimation of plasma glucose.

Among all the local modeling techniques, clustering is chosen for this application given its advantages for finding automatically local behavior.

A new algorithm for identification of local models is designed attending to the system characteristics. First of all, the performance of the modeling method is checked in general benchmarks, showing that the modeled output

meets the requirements imposed.

Later, this modeling method is used to build a new calibration algorithm. It is, then, applied to its real purpose: glucose estimation for continuous glucose monitoring. Data used is a set of patients where glucose variations were obtained through a clamp study (variations of glucose level with external infusion of insulin and glucose). The main input considered is the signal coming from the subcutaneous sensor (intensity of current in nA) and the output to estimate is the plasma glucose level. The glucose has to be estimated for each time sample (k).

Different configurations for the calibration algorithm are checked. Results show that the best performance is considering as inputs the current intensity in different instants (includes information of the signal's trend) and previous estimations of glucose (to relate current and glucose levels).

Results showed that the basic configuration of the proposed CA performs better than the current monitor used in the clinical trial. Different local models valid regionally are found automatically, indicating that the application of the local model technique makes sense in this problem.

In that experiment, other configurations for the CA were checked: including binary information related to the metabolic state (fasting/feeding) and also about the insulinemia level (high/low). Performance of these CAs improved with regard to the basic configuration, indicating this that this type of information is relevant for the glucose estimation.

Analysis of local models demonstrated that the inter and intra-patient sensor sensitivity variations affect to the accuracy of the plasma glucose estimation. Inter and intra-patient sensitivity variations have been studied in some works. Yet, they are not well characterized and quantified.

The experiment done with clamp data also demonstrated that fact: when particular information of sensor to patient sensitivity is taken into account (through normalization parameters), the accuracy of plasma glucose estimations improves.

Thus, this work proposes an adaptive calibration algorithm (ACA) to compensate the sensor sensitivity variations. The proposed ACA is based on a first identification of a calibration algorithm based on the addition of several weighted local models and the later adaptation of normalization parameters to compensate the sensitivity variations. This compensation is done through the normalization parameters required to apply the model identified first.

These parameters are adapted to minimize the difference between the estimation of the model and the reference values. These samples (calibration points) are introduced several times a day (around 4), emulating the current

system where patients with DM1 measure the plasma glucose level with low error devices.

Performance of the ACA is checked with data of people with DM1. Results indicate that the calibration algorithm with adaptation of parameters estimates the plasma glucose level better than the calibration algorithm with set parameters and no adaptation. Thus, the proposed ACA can compensate the inter and intra-patient sensitivity variations.

The final conclusion of this work is that the new calibration algorithms proposed do reach the target fixed for this work: getting a **feasible algorithm** that **improves accuracy** of current continuous glucose monitoring devices.

7.2 Contributions

The main contributions of this work are:

- ◇ A **review** of treated **problem** and the considered **appropriate technique** to work on it:
  - Diabetes mellitus basic concepts, specially the Type 1 Diabetes. As well as the CGMS requirements and its actual state of art.
  - The advantages of local modeling techniques, opting for the clustering methods as best option. The characteristics of this method and the state of art for its application to system modeling is also reviewed.
- ◇ A **new algorithm** for **local models identification** using clustering techniques. This algorithms was designed with certain characteristics and it is of general application for systems with these properties.
- ◇ The **design** of a new **calibration algorithm**, testing different configurations.
  - Local models detected can correspond to different sensor to patient sensitivity groups or to different dynamics.
  - Exogenous information coming from the patient or the insulin pump might help the CGMS accuracy.
  - Consideration of sensor to patient sensitivity improves the CGMS estimation.

- The building of a **estimation** method to be able to consider the **unknown sensitivities** for new patients, based on the steepest gradient optimization technique.
  - Different sensors result in different configurations of the Calibration Algorithm
  - Using daily calibration points (4/day) this method is capable of compensating different sensor to patient (inter-patient) sensitivity variations.
  - The designed method is also capable of compensation data variations of sensitivity with time (intra-patient), showed with in-silico data.

7.3 Publications

This work has given rise to several publications in journals:

- A codification method that can be used to include categorical information as inputs (not yet applied to the CGM problem). Two publications:
  - **F. Barceló-Rico, J.L. Díez, J. Bondia**. A comparative study of codification techniques for clustering heart disease database. *Biomedical Signal Processing and Control*. Vol. 6. Pag. 64-69. Year 2011.
  - **F. Barceló-Rico, J.L. Díez**. Geometrical codification for clustering mixed categorical and numerical databases. *Journal of Intelligent Information Systems*. DOI: 10.1007/s10844-011-0187-y. Year 2011.
- A system modeling method using clustering techniques with determined properties. One publication:
  - **F. Barceló-Rico, J.L. Díez, J. Bondia**. New Possibilistic Method for Discovering Linear Local Behaviour using Hyper-Gaussian Distributed Membership Function. *Knowledge and Information Systems*. Vol. 30(2). Pag. 377-403. Year 2012.
- A local-model-based calibration algorithm. One publication:

  - **F Barceló-Rico, J Bondia, J L Díez, P Rossetti**. A multiple local models approach to accuracy improvement in continuous glucose monitoring. *Diabetes Technology and Therapeutics*. Vol. 14(1). Pag. 74-82. Year 2012

◇ An adaptive calibration algorithm for compensation of sensor sensitivity variations. One submitted publication:

  - **F. Barceló-Rico, J.L. Díez, P. Rossetti, J. Vehí, J. Bondia**. Adaptive calibration algorithm for glucose estimation on continuous glucose monitoring based on local models. *IEEE Transactions on Information Technology in Biomedicine*. Under Review.

This research has also been exposed in some congresses:

◇ The codification method:

  - **F. Barceló-Rico, J.L. Díez**. A comparative study of codification techniques for clustering Heart Disease database. *7^{th} IFAC Symposium on Modelling and Control in Biomedical Systems (MCBMS)*. Place: Aalborg, Denmark. Year: 2009.

◇ The local-model-based calibration algorithm:

  - **F Barceló-Rico, J Bondia, J L Díez, P Rossetti**. Local-model-based calibration algorithms improve CGM accuracy. *11^{th} Diabetes Technology Meeting*. Place: San Francisco, USA. Year: 2011.

◇ The adaptive calibration algorithm:

  - **F. Barceló-Rico, J.L. Díez, P. Rossetti, J. Vehí, J. Bondia**. Adaptive local-model-based approach for plasma glucose estimation in continuous glucose monitoring. *The 5^{th} Conference on Advanced Technologies and Treatments for Diabetes*. Place: Barcelona, Spain. Year: 2012.

In addition to the publications, this work has lead to a patent:

1. **J. Bondia, F. Barceló-Rico, J.L. Díez, P. Rossetti, J. Vehí, Y.T. Leal**. System and method of plasma glucose estimation. Spanish Patent application. Ref. P201130811 (19th-May-2011)

In total: **1** patent application, **3** publications in congresses, **4** publications in journals and **1** pending publication in journal.

7.4 Future Work

Most of the conclusions obtained in this work are only indicators of how the proposed algorithms can perform. This is due to the small sizes of the validation populations where they are checked. The improvement obtained is significant even in the case of a small population size, yet results of this work can not then be considered as an absolute proof.

Thus, a future work is to validate the proposed calibration algorithms in a data set where a representative population of patients with DM1 is considered.

The study done with healthy patients was the one that showed that the level of insulinemia and the metabolic state might influence in the glucose variations and, therefore, in the plasma glucose estimation accuracy. Performance of the calibration algorithm when these signals are included is clearly better than with the bare configuration. Yet, this is still to be tested in patients with DM1. Variations of glucose levels during these states might be different in patients with DM1 than in healthy people with clamp variations of glucose, as glucose metabolism is different. Therefore, this concept here introduced is to be checked in the right framework of the problem considered.

In the same line, the proposed ACA is showed to compensate the inter and intra-patients sensor sensitivity variations. Its performance is checked with data of patients with DM1. Thus, the analysis of the results obtained lead to more indicative conclusions. Yet, two issues are found in this validation: 1) the population set is, again, small and can not be considered fully representative of the population with DM1; and 2) Data information is only available for one day and longer time data used is from an in-silico study.

To proof that the proposed ACA performs as this study indicates, longer time data must be available from patients with DM1 and also from a representative population.

The identified calibration algorithm for the ACA only uses a local model for the whole population set, as no many differences are found between patients. With a larger population, in addition to check the performance better, differences between patients can be found leading this to the placement of several local models, improving the accuracy of the calibration algorithm defined.

Indeed, other input including additional information about the patient could be included. This new information can help to define these differences

between patients and their dynamics, finding differences in the defined local models. The information included could be both numerical and categorical, as there has been proposed a codification method to include this last type of information.

Finally, one improvement that might be of interest is to find a way of initializing the normalization parameters for the ACA with a value more related to the real parameters of each patient. This could be done finding a model that estimates the initial value of these parameters for the population using relevant information of each patient.

Thus, a **summary** of the **futures lines** of research to extend this work is:

- ◇ In a larger population, representative of patients with DM1:
  1. Check the local-model-based calibration algorithm.
  2. Check the adaptive calibration algorithm.
- ◇ Check the improvement of the estimation including information of insulinemia and metabolic state in patients with DM1
- ◇ Find a model for the initialization of normalization parameters.

In addition to these lines, it could be interesting to extend the new algorithm to other areas and applications, where the identification of a set of locally valid and interpretable local models might be of interest.

Appendix A

Equations for normalization and denormalization

A.1 Variables normalization

In all this work the normalization and denormalization of variables is an important step. The normalization used in all cases is the one that makes all inputs/outputs have same statistical properties. In this case, all variables are re-scaled to have a normal distribution. Either in the case of normalization for the set or for subjects, is this type that is used.

Normal distributions are characterized by null mean and unity variance. Thus, to re-scale the original vector, its mean has to be subtracted and results have to be re-scaled by its original variance:

$$G_n = \left(\frac{G_D - \bar{G}}{\sqrt{\sigma_G}} \right) \tag{A.1}$$

where:

- G_n refers to normalized G.
- G_D refers to denormalized or original G.
- $\bar{G}$ is the mean of original vector G_D.
- σ_G is the variance of original signal G_D.

Thus, G_n will have null mean and unity variance. However, this normalization makes the transformed variable to cover a range where the zero is included. This could be a problem in this type of applications as the output

of the model in a particular point could be represented by zero, if it is equal to the mean of the signal. In that case, as the result of the local model is the multiplication of the linear regression by the membership of that point, multiplying linear regression by zero would give the correct output, even if the linear model estimation is not even close to the real output.

For this reason, it is convenient to avoid include the zero in the interval. A very simple way of doing this is to shift the signal to other range, adding an offset big enough to make all normalized points bigger than zero:

$$G_n = \left(\frac{G_D - \bar{G}}{\sqrt{\sigma_G}} + off \right) \tag{A.2}$$

In this case, using $off = 3$, this issue is solved.

A.2 Output denormalization

When normalized signals are used as inputs of the model, its output is also in the normalized space and it has to be denormalized to know the real output. Denormalization of signal when equation (A.2) is used as normalization is:

$$G_D = (G_n - off) \cdot \sqrt{\sigma_G} + \bar{G} \tag{A.3}$$

It is important to remark that parameters of model are all in the normalized space, same as inputs used. Then, for the application of this model, variables will always have to be normalized first. If the model is desired to be applied to original denormalized signals its parameters have to be transformed to be equivalent to original parameters computed for the normalized space.

This has been done later in this annex, in section A.4.

A.3 Denormalization of local estimations

The global model output is an addition of the outputs of the local models weighted by their membership functions. Thus, for a case of c local models, in each time instant k the valid equation of the output is:

$$(GM_k)_n = \sum_{i=1}^{c} (\mu_{ik})_n \cdot (LM_{ik})_n \tag{A.4}$$

Subindex n refers to normalized space.

Last equation can be generalized for any time instant:

$$\forall k \quad \rightarrow \quad GM_n = \sum_{i=1}^{c} (\mu_i)_n \cdot (LM_i)_n \tag{A.5}$$

As the global model equation is non-linear, the denormalization of each local model can not be directly done like equation (A.3). To see the contribution of each LM in the original range, transformation of equation A.3 has to be applied to total global model in equation (A.5).

To simplify this process, denormalization will be applied to a particular case of $c = 2$:

$$GM_n = (\mu_1)_n \cdot (LM_1)_n + (\mu_2)_n \cdot (LM_2)_n \tag{A.6}$$

The value of the membership of each object or time instant k indicates how this local model is representative of that object. Thus, in the denormalized space this function has to keep its value for all objects:

$$\mu_k \in [0, 1] \quad \rightarrow \quad (\mu_i)_n = \mu_i \tag{A.7}$$

Applied to equation A.6:

$$GM_n = \mu_1 \cdot (LM_1)_n + \mu_2 \cdot (LM_2)_n \tag{A.8}$$

To get the total denormalized output, the normalized output of last equation GM_n is to be replaced in equation (A.3) for G_n:

$$GM_D = ([\mu_1 \cdot (LM_1)_n + \mu_2 \cdot (LM_2)_n] - off) \cdot \sqrt{\sigma_G} + \bar{G} \tag{A.9}$$

When this equation is expanded (eq. A.10) it can be seen that not exactly the same structure as A.3 is obtained for each LM.

$$GM_D = \mu_1 \cdot (LM_1)_n \cdot \sqrt{\sigma_G} + \mu_2 \cdot (LM_2)_n \cdot \sqrt{\sigma_G} - off \cdot \sqrt{\sigma_G} + \bar{G} \tag{A.10}$$

If it is desired to express the denormalized output as an addition of denormalized local models like:

$$GM_D = \mu_1 \cdot (LM_1)_D + \mu_2 \cdot (LM_2)_D \tag{A.11}$$

then, equation A.10 is to be rewritten to match this structure, where denormalized local model is weighted by each membership function μ_i.

This can be achieved if the constant terms, $\bar{G}$ and $off \cdot \sqrt{\sigma_G}$, are multiplied and divided by the total membership $\mu_1 + \mu_2$:

$$GM_D = \mu_1 \cdot (LM_1)_n \cdot \sqrt{\sigma_G} + \mu_2 \cdot (LM_2)_n \cdot \sqrt{\sigma_G} - off \cdot \sqrt{\sigma_G} \cdot \frac{(\mu_1 + \mu_2)}{(\mu_1 + \mu_2)} + \bar{G} \cdot \frac{(\mu_1 + \mu_2)}{(\mu_1 + \mu_2)} \tag{A.12}$$

Finally, this equation can adopt the structure like in A.11:

$$\begin{aligned} GM_D = \quad & \mu_1 \cdot \left[\left((LM_1)_n - \frac{off}{(\mu_1 + \mu_2)} \right) \cdot \sqrt{\sigma_G} + \frac{\bar{G}}{(\mu_1 + \mu_2)} \right] + \\ & + \mu_2 \cdot \left[\left((LM_2)_n - \frac{off}{(\mu_1 + \mu_2)} \right) \cdot \sqrt{\sigma_G} + \frac{\bar{G}}{(\mu_1 + \mu_2)} \right] \end{aligned} \tag{A.13}$$

Last equation can easily generalized for a general number of local models c:

$$GM_D = \sum_{i=1}^{c} \mu_i \cdot \left[\left((LM_i)_n - \frac{off}{\sum_{i=1}^{c} \mu_i} \right) \cdot \sqrt{\sigma_G} + \frac{\bar{G}}{\sum_{i=1}^{c} \mu_i} \right] \tag{A.14}$$

It can be seen how denormalized local models follow a similar structure as the global model, where the constant terms $\bar{G}$ and $off \cdot \sqrt{\sigma_G}$ contribute to each of them in such way as their importance is over the global model.

A.4 Denormalization of model parameters

Instead of working on the model space (which is the normalized input range) an alternative is to denormalize the parameters of the model and apply them directly to the original variables.

A.4.1 Parameters of membership functions

As expressed in equation (A.7) denormalized MF have to keep the same value for all objects as normalized MF, because they indicate validity in the range [0,1]. In this section, to simplify nomenclature normalized variables will be written with capital letters ($A, B, ..., D$ or $A_j = (I_j)_n$) while low case letters will refer to denormalized inputs ($a, b, ..., d$ or $a_j = (I_j)_D$). Subindex j refers to number of inputs $j = 1, 2, ..., d$. Input A and a are related following equation A.2:

$$A = \left(\frac{a - \bar{a}}{\sqrt{\sigma_a}} \right) + off \tag{A.15}$$

Following the membership function equation (4.5), the normalized membership of each cluster $(\mu_i)_n$ can be written like:

$$(\mu_i)_n = e^{-\frac{1}{2}\left(\sum_{j=1}^{d} \left(\frac{A_j - c_{ij}}{\sigma_{ij}} \right)^2 \right)^H} \tag{A.16}$$

where parameters c_{ij} and σ_{ij} are the coordinates of the center and standard deviation of MF for cluster i and input j. Vector of centers $\vec{c}_i$ and standard deviations $\vec{\sigma}_i$ are the parameters that define the MF in the normalized space:

$$\begin{array}{l} \vec{c}_i = \left[\begin{array}{cccc} c_{i1} & c_{i2} & ... & c_{id} \end{array} \right] \\ \vec{\sigma}_i = \left[\begin{array}{cccc} \sigma_{i1} & \sigma_{i2} & ... & \sigma_{id} \end{array} \right] \end{array} \tag{A.17}$$

These are the parameters that have to be transformed to make the denormalized MF equivalent to the normalized one:

$$\forall k \quad \rightarrow \quad (\mu_i)_n \; = (\mu_i)_D \tag{A.18}$$

$$e^{-\frac{1}{2}\left(\sum_{j=1}^{d} \left(\frac{A_j - c_{ij}}{\sigma_{ij}} \right)^2 \right)^H} = e^{-\frac{1}{2}\left(\sum_{j=1}^{d} \left(\frac{a_j - c^*_{ij}}{\sigma^*_{ij}} \right)^2 \right)^H} \tag{A.19}$$

Parameters c^*_{ij} and σ^*_{ij} form the vectors of denormalized parameters, $(\vec{c}_i)_D$ and $(\vec{\sigma}_i)_D$.

In last equation, as both terms involve an exponential with same constants, to be equivalent, the exponent has to result in the same value:

$$\sum_{j=1}^{d} \left(\frac{A_j - c_{ij}}{\sigma_{ij}} \right)^2 = \sum_{j=1}^{d} \left(\frac{a_j - c^*_{ij}}{\sigma^*_{ij}} \right)^2 \tag{A.20}$$

And for it, each input (normalized and denormalized) has to contribute in the same way:

$$\forall i, \; \forall j \quad \rightarrow \quad \frac{A - c_j}{\sigma_j} = \frac{a - c^*_j}{\sigma^*_j} \tag{A.21}$$

Substituting the normalized input, eq. (A.15), in last equation and expanding results in:

$$\frac{\left(\left(\frac{a-\bar{a}}{\sqrt{\sigma_a}}\right)+off\right)-c_j}{\sigma_j}=\frac{a-c_j^*}{\sigma_j^*} \tag{A.22}$$

$$\frac{a-\bar{a}+off\cdot\sqrt{\sigma_a}-c_j\cdot\sqrt{\sigma_a}}{\sigma_j\cdot\sqrt{\sigma_a}}=\frac{a-c_j^*}{\sigma_j^*} \tag{A.23}$$

Finally, the denormalized parameters can be found solving the relations for each generic input a, which corresponds to coordinate j:

$$\begin{array}{c} c_j^*=(c_j-off)\cdot\sqrt{\sigma_a}+\bar{a} \\ \sigma_j^*=\sigma_j\cdot\sqrt{\sigma_a} \end{array} \tag{A.24}$$

With vectors $\vec{c^*}$ and $\vec{\sigma^*}$ the denormalized inputs can be directly used to compute the membership value of objects.

A.4.2 Regression coefficients

In the denormalized space the structure of the global model is still like equation A.5:

$$\forall k \quad \rightarrow \quad GM_D=\sum_{i=1}^{c}(\mu_i)_D\cdot(LM_i)_D \tag{A.25}$$

The membership functions are equivalent in normalized and denormalized spaces, $(\mu_i)_D=(\mu_i)_n=\mu_i$. And all local models i still correspond to linear regressions:

$$(LM_i)_D=\left(\sum_{j=1}^{d}\beta_{ij}^*\cdot a_j\right)+\beta_{i0}^* \tag{A.26}$$

where superindex $(^*)$ corresponds to the denormalized parameters of the models. Last equation will be particularized for two inputs ($d=2$) to simplify the equations:

$$(LM_i)_D=\beta_{i1}^*\cdot a+\beta_{i2}^*\cdot b+\beta_{i0}^* \tag{A.27}$$

Equation A.14 shows that the each denormalized LM follows this general structure:

$$(LM_i)_D = \left[\left((LM_i)_n - \frac{off}{\sum_{i=1}^{c} \mu_i} \right) \cdot \sqrt{\sigma_G} + \frac{\bar{G}}{\sum_{i=1}^{c} \mu_i} \right] \tag{A.28}$$

The relation between normalized and denormalized regression coefficients can be found equaling last two equations:

$$\beta^*_{i1} \cdot a + \beta^*_{i2} \cdot b + \beta^*_{i0} = \left[\left([\beta_{i1} \cdot A + \beta_{i2} \cdot B + \beta_{i0}] - \frac{off}{\sum_{i=1}^{c} \mu_i} \right) \cdot \sqrt{\sigma_G} + \frac{\bar{G}}{\sum_{i=1}^{c} \mu_i} \right] \tag{A.29}$$

Normalized inputs (A, B) have to be replaced by its relations with original denormalizezd inputs (a, b):

$$\beta^*_{i1} \cdot a + \beta^*_{i2} \cdot b + \beta^*_{i0} =$$
$$= \left[\left(\left[\beta_{i1} \cdot \left(\frac{a - \bar{a}}{\sqrt{\sigma_a}} + off \right) + \beta_{i2} \cdot \left(\frac{b - \bar{b}}{\sqrt{\sigma_b}} + off \right) + \beta_{i0} \right] - \frac{off}{\sum_{i=1}^{c} \mu_i} \right) \cdot \sqrt{\sigma_G} + \frac{\bar{G}}{\sum_{i=1}^{c} \mu_i} \right] \tag{A.30}$$

The expansion of last equation results in:

$$\beta^*_{i1} \cdot a + \beta^*_{i2} \cdot b + \beta^*_{i0} = \beta_{i1} \cdot a \cdot \frac{\sqrt{\sigma_G}}{\sqrt{\sigma_a}} + \beta_{i2} \cdot b \cdot \frac{\sqrt{\sigma_G}}{\sqrt{\sigma_b}} +$$
$$+ \left[\left(\beta_{i1} \cdot \left(\frac{(-\bar{a})}{\sqrt{\sigma_a}} + off \right) + \beta_{i2} \cdot \left(\frac{(-\bar{b})}{\sqrt{\sigma_b}} + off \right) + \beta_{i0} - \frac{off}{\sum_{i=1}^{c} \mu_i} \right) \cdot \sqrt{\sigma_G} + \frac{\bar{G}}{\sum_{i=1}^{c} \mu_i} \right] \tag{A.31}$$

From that, the denormalized regression coefficients can easily be extracted. General equations for them are:

$$\forall j, \forall i \quad \rightarrow \quad \beta^*_{ij} = \beta_{ij} \cdot \frac{\sqrt{\sigma_G}}{\sqrt{\sigma_{a_j}}} \tag{A.32}$$

$$\beta_{i0}^{*} = \left[\left(\sum_{j=1}^{d} \beta_{ij} \cdot \left(\frac{(-\bar{a}_j)}{\sqrt{\sigma_{a_j}}} + off \right) + \beta_{i0} - \frac{off}{\sum_{i=1}^{c} \mu_i} \right) \cdot \sqrt{\sigma_G} + \frac{\bar{G}}{\sum_{i=1}^{c} \mu_i} \right] \quad \text{(A.33)}$$

It can be seen that denormalized regression coefficient for model i and input j, β_{ij}^{*}, only depends on the normalized parameter and standard deviation of that input and the output. Therefore, these terms are constant for all objects k.

On the contrary, the denormalized independent term of each LM, β_{i0}^{*}, depends on constants terms like means of all inputs and output, and its standard deviations, but also on variable terms for each object like the total membership of each object to all clusters. Thus, in spite of its complex equation this term is variable for each object k, fact that makes difficult its computation.

A.4.3 Observations

Equations for the denormalized parameters of the model have been found, in order to be able to apply directly the model to the original signals. Equations of the parameters of the membership functions are easy to implement, as they only depend on variables that are constant. Same happens for the parameters of the regression coefficients of local models.

However, the equation of independent term of each LM is complex and includes variables like the total membership value that would have to be computed for all objects k. This is due to the adopted possibilistic approach, where the total membership of objects do not have to add necessarily one. In the fuzzy approach, where this constraint is included, this would not happen, as the addition of all membership would be one for all objects.

Indeed, this complexity of the denormalized model, makes it easier to work in the normalized space as it only requires the step of rescale input and output variables, while working with the original signal implies the computation of complex formula, which has to be computed for every object k.

Bibliography

[1] ISO 15197:2003. In vitro diagnostic test systems - requirements for blood-glucose monitoring systems for self-testing in managing diabetes. *mellituswww.iso.org.* 5.3.1

[2] J. Abonyi, M.D. Alexiuk, P. Angelov, and B. Bird. *Advances in Fuzzy Clustering and its applications.* John Wiley & Sons, Ltd, 2007. 3.2.7, 4.3

[3] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data. *Data Min. Knowl. Discov*, 11:5–33, 2005. 3.2.7

[4] P. Andritsos. Data clustering techniques. *Toronto, University of Toronto, Dep. of Computer Science*, 2002. 3.2.1

[5] P. Andritsos. *Scalable Clustering of Categorical Data and Applications.* PhD thesis, University of Toronto, 2004. 1.2, 3.2.4, 3.2.6

[6] S.L. Aronoff et al. Glucose metabolism and regulation: Beyond insulin and glucagon. *Diabetes Spectrum*, 17:183–190, 2004. 2.5

[7] B. Aussedat et al. Interstitial glucose concentration and glycemia: implications for continuous subcutaneous glucose monitoring. *American Journal of Physiology - Endocrinology and Metabolism*, 278(4):E716–728, 2000. 2.3.2

[8] Averroes. *Libro de las generalidades de la medicina (Kitab al-kulliyyat fi l-tibb).* Editorial Trotta S.A, 2003. 2.2.1

[9] R. Babuska. *Fuzzy modeling for control*. Kluwer Academic Publishers. 3.1.3, 4

[10] T. Back, D.B Fogel, and Z Michalewicz, editors. *Evolutionary Computation 1: Basic Algorithms and Operators*. Taylor and Francis, 2000. 4.5

[11] F.G. Banting et al. The effect of pancreatic extract (insulin) on normal rabbits. *American Journal of Physiology - Endocrinology and Metabolism*, 62 (1):162–176, 1922. 1

[12] F. Barcelo-Rico, J. Bondia, J.L. Diez, and P. Rossetti. A multiple local models approach to accuracy improvement in continuous glucose monitoring. *Diabetes Technology and Therapeutics*, 14(1):74–82, 2012. 2.3.3, 6.7.3

[13] F. Barcelo-Rico and J.L. Diez. Geometrical codification for clustering mixed categorical and numerical databases. *Journal of Intelligent Information Systems*, 10.1007/s10844-011-0187-y, 2011. 3.3.3

[14] F. Barcelo-Rico, J.L. Diez, and J. Bondia. A comparative study of codification techniques for clustering heart disease database. *Biomedical Signal Processing and Control*, 6(1):64–69, 2011. 3.3.3

[15] I.J. Benitez-Sanchez. Técnicas de agrupamiento para el análisis de datos cualitativos y cuantitativos. Master's thesis, Technical University of Valencia, 2005. 3.2.4, 3.3.3

[16] B.W. Bequette. Continuous glucose monitoring: Real-time algorithms for calibration, filtering and alarms. *Journal of Diabetes Science and Technology*, 4 (2):404–419, 2010. 1.1, 2.3.2, 2.3.3, 2.3.3, 2.3.3

[17] J.C. Bezdek. *Pattern Recognition with Fuzzy Objective Functions Algorithms*. Plenum Press, 1981. 4.4

[18] J.C. Bezdek and N.R. Pal. Some new indexes of cluster validity. *IEEE Transactions on System, Man and Cybernetics - Part B: Cybernetics*, 28:301–315, 1998. 4.7.1

[19] J. Bondia, J. Vehi, C.C. Palerm, and P. Herrero. El pancreas artificial: Control automatico de infusion de insulina en diabetes mellitus tipo 1. *Revista Iberoamericana de Automatica e Informatica Industrial*, 7 (2):5–20, 2010. 1.1, 1.2, 2.2.2, 2.2.3, 2.8, 2.1, 2.3.3

[20] M.S. Boyne, D.M. Silver, J. Kaplan, and C.D. Saudek. Timing of changes in interstitial and venous blood glucose measured with a continuous subcutaneous glucose sensor. *Diabetes*, 52(11):2790–2794, 2003. 2.3.2

[21] P.S. Bradley, U. Fayyad, and C. Reina. Scaling clustering algorithms to large databases. *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 9–15, 1998. 3.2.7

[22] H. Brem and M. Tomic-Canic. Cellular and molecular basis of wound healing in diabetes. *Journal of Clinical Investigation*, 117:1219–1222, 2007. 2.3.2

[23] U. Brueggemann. Implantable sensor device useful in measuring system and medical delivery device for in vivo monitoring or measuring of glucose in diabetics, comprises arrangement of micro-sensors, control device, emitter, and power supply module, 2011. 6.10

[24] F. Cameron, G. Niemayer, K. Gundy-Burlet, and B. Buckingham. Statistical hypoglycemia prediction. *Journal of Diabetes Science and Technology*, 2(4):612–621, 2008. 2.3.3

[25] F.M. Cameron et al. Early detection of hypoglycemia combining multiple predictive methods on retrospective clinical continuous glucose monitoring data. *Journal of Diabetes Science and Technology*, 2(2):A19, 2008. 2.3.3

[26] C. Choleau et al. Calibration of a subcutaneous amperometric glucose sensor implanted for 7 days in diabetic patients. part 2. superiority of the one-point calibration method. *Biosensors and Bioelectronics*, 17(8):622–628, 2002. 2.3.3

[27] C. Choleau et al. Prevention of hypoglycemia using risk assessment with a continuous glucose monitoring system. *Diabetes Care*, 51(11):3263–3273, 2002. 2.3.3

[28] W.L. Clarke. The original clarke error grid analysis (ega). *Diabetes Technology & Therapeutics*, 7(5):776–779, 2005. 2.3.3

[29] W.L. Clarke et al. Evaluating the clinical accuracy of two continuous glucose sensors using continuous glucose-error grid analysis. *Diabetes Care*, 28(10):2412–2417, 2005. 2.3.1

[30] R.J. Davey, C. Low, T.W. Jones, and P.A. Fournier. Contribution of an instrinsic lag of continuous glucose monitoring systems to differences in measured and actual glucose concentrations changing at variable rates in vitro. *Journal of Diabetes Science and Technology*, 4(6):1393–1399, 2010. 5.1.1

[31] A. de Leiva-Hidalgo, E. Brugues-Brugues, and A. de Leiva-Pérez. From pancreatic extracts to artificial pancreas: History, science and controversies about the discovery of the pancreatic antidiabetic hormone. *Avances en Diabetologia*, 27(1):27–38, 2001. 2.2.1

[32] L.T. DeCarlo. On the meaning and use of kurtosis. *Psychological Methods*, 2:292–307, 1997. 4.2.1

[33] J.L. Díez. *Técnicas de agrupamiento para identificación y control por modelos locales*. PhD thesis, Universitat Politècnica de València, 2003. 1.2, 3.1.3, 4.7.2

[34] J.L. Díez, J.L. Navarro, and A. Sala. A fuzzy clustering algorithm enhancing local model interpretability. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 11(10):973–983, 2007. 3.4, 3.4, 3.4.1, 4.2.1, 4.7

[35] J.L. Diez, A. Sala, and J.L. Navarro. Target-shaped possibilistic clustering applied to local-model identification. *Engineering Applications of Artificial Intelligence*, 19(2):201–208, 2006. 3.2.7, 3.4, 3.4, 3.4.1, 4.7, 4.7.2

[36] J.A. Egea, M. Rodriguez-Fernandez, J.R. Banga, and R. Marti. Scatter search for chemical and bio-process optimization. *Journal of Global Optimization*, 37:481–503, 2007. 4.5.2, 4.6, 4.6

[37] J.A. Egea-Larrosa. *New Heuristics for Global Optimization of Complex bioprocesses*. PhD thesis, Universidade de Vigo, 2008. 4.6

[38] N. Elfelly, J.Y. Dieulot, M. Benrejeb, and P. Borne. A new approach for multimodel identification of complex systems based on both neural network and fuzzy clustering algorithms. *Engineering Applications of Artificial Intelligence*, 23:1064–1071, 2010. 3.1.1, 3.4

[39] M.R. Emami, I.B. Türksen, and A.A. Goldenberg. Development of a systematic methodology of fuzzy logic modeling. *Transactions on Fuzzy Systems*, 6:346–366, 1998. 4.7.1

[40] M. Ester, H.P. Kriegel, and J. Sander. *Geographic Data Mining and Knowledge Discovery*. Taylor and Francis, 2001. 3.2.7

[41] K. Becker et al., editor. *Principles and Practices of Endocrinology and Metabolism*. Lippincott Williams and Wilkins Publishers, 2001. 2.2.2

[42] A. Facchinetti, G Sparacino, and C. Cobelli. Enhanced accuracy of continuous glucose monitoring by online extended kalman filtering. *Diabetes Technology and Therapeutics*, 10(15):353–363, 2010. 1.1, 2.3.2, 2.3.3, 5.1.1, 6.2.1, 6.2.1, 6.10

[43] International Diabetes Federation. *IDF Diabetes Atlas*. 2010. 1.1, 2.1.1, 2.1, 2.2, 2.3, 2.4

[44] D. Fisher. Cobweb: Knowledge acquisition via conceptual clustering. *Machine Learning*, 2:139–172, 1987. 3.2.7

[45] R. Fletcher. *Practical methods of optimization*. Wiley, 1987. 4.5, 4.5.1

[46] Juvenile Diabetes Research Foundation. http://www.jdrf.org/. 1.1, 2.2.3, 2.2.3

[47] V. Ganti, J. Gehrke, and R. Ramakrishnan. Cactus—clustering categorical data using summaries. *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 73–83, 1999. 3.2.7

[48] M. García-Jaramillo et al. Insulin dosage optimization based on prediction of postprandial glucose excursions under uncertain parameters and food intake. *Computer Methods and Programs in biomedicine*, 105:61–69, 2012. 2.2.3

[49] M.A. Gibney, C.H. Arce, K.J. Byron, and L.J. Hirsch. Skin and subcutaneous adipose layer thickness in adults with diabetes at sites used for insulin injections: implications for needle length recommendations. *Current Medical Research & Opinion*, 26(6):1519–1530, 2010. 1.1, 2.3.2, 6.2.1

[50] D. Gibson, J. Kleinberg, and P. Raghavan. Clustering categorical data: an approach based on dynamical systems. *The VLDB Journal The International Journal on Very Large Data Bases*, 8(3):222–236, 2000. 3.2.7

[51] D.E. Goldstein et al. Tests of glycemia in diabetes. *Diabetes Care*, 7(Suppl 1):S91–93, 2004. 2.3.1

[52] D.A. Gough et al. Function of an implanted tissue glucose sensor for more than 1 year in animals. *Science Translational Medicine*, 2(42), 2010. 6.5.3, 6.10

[53] K.C. Gowda and E. Diday. Symbolic clustering using a new dissimilarity measure. *Pattern Recognition*, 24(6):567–578, 1991. 3.2.4

[54] S. Guha, R. Rastogi, and K. Shim. Rock: A robust clustering algorithm for categorical attributes. *Information Systems*, 25(5):345–366, 2000. 3.2.7

[55] S. Guha, R. Rastogi, and K. Shim. Cure: an efficient clustering algorithm for large databases. *Information Systems*, 26:35–58, 2001. 3.2.7

[56] E.E. Gustafson and W.C. Kessel. Fuzzy clustering with a fuzzy covariance matrix. *IEEE Conference on Decision and Control*, pages 761–766, 1978. 4.3

[57] A.C. Guyton and J.E. Hall. *Textbook of Medical Physiology*. Saunders, 2005. 2.1.2

[58] et al. H. Maciejewski. Analyzing solar power plant performance through data mining. *Journal of Solar Energy Engineering*, 130(4), 2008. 3.1.3

[59] R. Hanas. *Type I Diabetes. A Guide for Children, Adolescents, Young Adults, and Their Caregivers*. Marlowe and Company, 2005. 2.1.4

[60] N. Hansen. Invariance, self-adaptation and correlated mutation in evolution strategies. *Proceedings of Sixth International Conference on Parallel Problem Solving*, pages 355–364, 2000. 4.5.2, 4.6, 4.6

[61] J.A. Hartigan and W.A. Wong. A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C*, 28:100–108, 1979. 3.2.7, 3.3.1, 4.4

[62] R.J. Hathaway and J.C. Bezdek. Switching regression models and fuzzy clustering. *IEEE Transactions on fuzzy systems*, 1(3):195–204, 1993. 1.2, 3.2.7, 3.4, 3.4, 4.3

[63] Z. He, X. Xu, and S. Deng. Scalable algorithms for clustering large datasets with mixed type attributes. *International Journal of Intelligent Systems*, 20:1077–1089, 2005. 3.2.7, 3.3.3

[64] K.L. Helton, B.D. Ratner, and N. A. Wisniewski. Biomechanics of the sensor-tissue interface. effects of motion, pressure, and design on sensor performance and the foreign body response. part i: Theoretical framework. *Journal of Diabetes Science and Technology*, 5 (3):632–646, 2011. 2.3.2, 5.6, 6.2.1

[65] Y.J. Heo et al. Long-term in vivo glucose monitoring using fluorescent hydrogel fibers. *Proceedings of the National Academy of Sciences of the United States of America*, 108(33):13399–13403, 2011. 6.5.3, 6.10

[66] A. Hinneburg and D. Keim. An effcient approach to clustering in large multimedia databases with noise. *Proceedings of the 1998 International Conference on Knowledge Discovery and Data Mining*, pages 58–65, 1998. 3.2.7

[67] R. Hovorka. The future of continuous glucose monitoring: closed loop. *Current Diabetes Reviews*, 4(3):269–279, 2008. 2.2.3, 2.2.3

[68] R. Hovorka et al. Overnight closed loop insulin delivery (artificial pancreas) in adults with type 1 diabetes: crossover randomised controlled studies. *British Medical Journal (BMJ)*, doi:10.1136/bmj.d1855, 2011. 2.2.3, 2.2.3

[69] C.C. Hsu, C.L. Chen, and Y.W. Su. Hierarchical clustering of mixed data based on distance hierarchy. *Information Sciences*, 177(20):4474–4492, 2007. 3.3.3

[70] Z. Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3):283–304, 1998. 3.2.7, 3.3.2, 3.3.3

[71] Diabetes Research in Children Network (DirecNet). http://direcnet.jaeb.org. 2.3.3

[72] A.K. Jain and R.C. Dubes. *Algorithms for clustering data*. Prentice Hall, 1988. 3.2.3, 3.2.4

[73] A.K. Jain, M.N. Murty, and P.J. Flynn. Data clustering: a review. *ACM Computing Surveys (CSUR)*, 31(3):264–323, 1999. 3.2.6

[74] P.A. Jansson, J. Fowelin, U. Smith, and P. Lonnroth. Characterization by microdialysis of intracellular glucose level in subcutaneous tissue in humans. *American Journal of Physiology - Endocrinology and Metabolism*, 255(2 Pt 1):E218–220, 1988. 2.3.2

[75] E. Januzaj, H.P. Kriegel, and M. Pfeifle. Scalable density-based distributed clustering. *PKDD '04: Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 231–244, 2004. 3.2.7

[76] X. Jin, B. Huang, and D.S. Shook. Multiple model lpv approach to nonlinear process identification with em algorithm. *Journal of Process Control*, 21:182–193, 2011. 3.4

[77] T.A. Johansen and R. Murray-Smith. *The operating regime approach to nonlinear modelling and control*. Taylor & Francis, 1997. 1.2, 3.1.2, 3.1.3

[78] G. Karypis, E.H. Han, and V. Kumar. Chameleon: A hierarchical clustering algorithm using dynamic modeling. *IEEE Computer*, 32(8):68–75, 1999. 3.2.7

[79] D. Kasper et al. *Harrison's Principles of Internal Medicine*. McGraw-Hill, 2008. 2.1.4

[80] L. Kaufman and P.J. Rousseeuw. Finding groups in data. an introduction to cluster analysis. *Wiley Series in Probability and Mathematical Statistics. Applied Probability and Statistics.*, 1990. 3.2.7

[81] D.B. Keenan, R. Cartaya, and J.J. Mastrototaro. Accuracy of a new real-time continuous glucose monitoring algorithm. *Journal of Diabetes Science and Technology*, 4(1):11–118, 2010. 2.3.2

[82] J. Kennedy and R. Eberhart. Particle swarm optimization. *Proceedings IEEE International Conference on Neural Networks*, 4:1942–1948, 1995. 4.5.2

[83] E.Y. Kim, S.Y. Kim, D. Ashlock, and D. Nam. Multi-k: accurate classification of microarray subtypes using ensemble k-means clustering. *BMC Bioinformatics*, 10, 2009. 4.5

[84] C. King et al. Modeling of calibration effectiveness and blood-to-interstitial glucose dynamics as potential confounders of the accuracy of continuous glucose sensors during hyperinsulinemic clamp. *Journal of Diabetes Science and Technology*, 1(3):317–322, 2007. 2.3.3

[85] U. Klueh et al. Metabolic biofouling of glucose sensors in vivo: Role of tissue microhemorrhages. *Journal of Diabetes Science and Technology*, 5 (3):583–595, 2011. 1.1, 2.3.2, 6.2.1

[86] E.J. Knobbe and B. Buckingham. The extended kalman filter for continuous glucose monitoriing. *Diabetes Technology and Therapeutics*, 7(1):15–27, 2005. 1.1, 2.3.3, 2.3.3, 2.3.3, 4.2, 5.1.1, 6.10

[87] H.E Koschwanez and W.M. Reichert. In vitro, in vivo and post explanation testing of glucose-detecting biosensors: current methods and recommendations. *Biomaterials*, 28:3687–3703, 2007. 5.6, 5.7

[88] B. Kovatchev, S. Anderson, L. Heinemann, and W. Clarke. Comparison of the numerical and clinical accuracy of four continuous glucose monitors. *Diabetes Care*, 31(6):1160–1164, 2008. 2.3.1

[89] B.P. Kovatchev, M. Breton, C.D. Man, and C. Cobelli. In silico preclinical trials: a proof of concept in closed-loop control of type 1 diabetes. *J Diabetes Sci Technol.*, 3(1):44–55, 2009. 2.3.3, 6.8.2

[90] R. Krishnapuram and J.M. Keller. A possibilistic approach to clustering. *IEEE transactions on fuzzy systems*, 1:98–110, 1993. 4.3

[91] S. Kullback and R.A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22:49–86, 1951. 3.2.4

[92] M. Kuure-Kinsey, C.C. Palerm, and B.W. Bequette. A dual-rate kalman filter for continuous glucose monitoriing. *Proceedings of th 28th IEEE EMBS Annual International Conference.*, pages 63–66, 2006. 1.1, 2.3.2, 2.3.3, 2.3.3, 6.10

[93] Y.T. Leal et al. Real-time glucose estimation algorithm for continuous glucose monitoring using autoregressive models. *Journal of Diabetes Science and Technology*, 4(2)::391–402, 2010. 2.3.2, 2.3.3, 2.3.3, 5.1.1

[94] H. Lee, B.A. Buckingham, D.M. Wilson, and B.W. Bequette. A closed-loop artificial pancreas using model predictive control and a sliding meal size estimator. *Journal of Diabetes Science and Technology*, 3(5):1082–90, 2009. 2.2.3, 2.2.3

[95] L.M. Lesperance, A. Spektor, and K.J. McLeod. Calibration of the continuous glucose monitoring system for transient glucose monitoring. *Diabetes Technology and Therapeutics*, 9 (2):183–190, 2007. 6.2.1

[96] M. Lovera and F. Previdi. Identification of linear models for the dynamics of a photodetector. *Control Engineering Practice*, 8 (11):1149–1158, 2000. 3.1.3

[97] E. Mamdani. Application of fuzzy logic to approximate reasoning using linguisitic systems. *Fuzzy Set and Systems*, 26:1182–1191, 1977. 3.1.3

[98] S.D. Müller, N. Hansen, and P. Koumoutsakos. Increassing the serial and parallel performance of the cma-es with large populations. *Proceedings of Seventh International Conference on Parallel Problem Solving from Nature*, pages 422–431, 2002. 4.6

[99] T.P. Monsod et al. Do sensor glucose levels accurately predict plasma glucose concentrations during hypoglycemia and hyperinsulinemia? *Diabetes Care*, 25(5):889–893, 2002. 2.3.2

[100] G.J.L. Naus. Gain scheduling. robust design and automated tuning of automotive controllers. http://www.mate.tue.nl/mate/pdfs/11022.pdf. 3.1.3

[101] P.C. Nayak and K.P. Sudheer. Fuzzy model identification based on cluster estimation for reservoir inflow forecast. *Hydrological processes*, 22:827–841, 2008. 3.4

[102] J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer, 1999. 4.5.1, 6.4, 6.5.2

[103] S.E. Noujaim, D. Horwitz, M. Sharma, and J. Marhoul. Accuracy requirements for a hypoglycemia detector: an analytical model to evaluate the effects of bias, precision, and the rate of glucose change. *Journal of Diabetes Science and Technology*, 1(5):652–668, 2007. 2.3.3

[104] Y. Onuki et al. A review of the biocompatibility of implantable devices: current challenges to overcome foreign body response. *Journal of Diabetes Science and Technology*, 2:1003–1015, 2008. 5.1.1, 5.6, 5.7

[105] S.M. Pappada, B.D. Cameron, and P.M. Rosman. Development of a neural network for prediction of glucose concentration in type 1 diabetes patients. *Journal of Diabetes Science and Technology*, 2(5):792–801, 2008. 2.3.3

[106] W. Pedrycz. An identification algorithm in fuzzy relational systems. *Fuzzy Set and Systems*, 13:153–167, 1984. 3.1.3

[107] F. Previdi. *Identification and control with local linear models*. PhD thesis, Politecnico di Milano, 1998. 4.2.1

[108] P. Rapp, M. Mesch, H. Giessen, and C. Tarín. Regression methods for ophthalmic glucose sensing using metamaterials. *Journal of Electrical and Computer Engineering*, 2011. 2.2.3

[109] K. Rebrin, G.M. Steil, W.P. van Antwerp, and J.J. Mastrototaro. Subcutaneous glucose predicts plasma glucose independent of insulin: implications for continuous monitoring. *American Journal of Physiology - Endocrinology and Metabolism*, 277(3 Pt 1):E561–571., 1999. 2.3.2

[110] AP@home European research project. http://www.apathome.eu/. 2.2.3, 2.2.3

[111] E.A. Rosei and D. Rizzoni. Small artery remodelling in diabetes. *Journal of Cellular and Molecular Medicine*, 14(5):1030–1036, 2010. 3

[112] P. Rossetti, J. Bondia, J. Vehi, and C.G. Fanelli. Estimating plasma glucose from interstitial glucose: The issue of calibration algorithms in commercial continuous glucose monitoring devices. *Sensors*, 10(12):10936–10952, 2010. 1.1, 2.3.1, 2.3.2, 2.3.3, 6.2.1

[113] P. Rossetti, F. Porcellati, C.G. Fanelli, and G.B. Bolli. Evaluation of the accuracy of microdialysis-based glucose sensor during insulin-induced hypoglycemia, its recovery, and post-hypoglycemic hyperglycemia in humans. *Diabetes Technology and Therapeutics*, 8(3):326–337, 2006. 5.4

[114] M. Ryoke, Y. Nakamori, and K. Suzuki. Adaptive fuzzy clustering and fuzzy prediction models. In *Fuzzy Systems, 1995. International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium., Proceedings of 1995 IEEE International Conference on*, volume 4, 1995. 3.2.7, 3.4, 3.4, 4.4

[115] M.A. Sainz, P. Herrero, J. Vehi, and J. Armengol. Continuous minimax optimization using modal intervals. *Journal of Computation and Applied Mathematics*, 339(1):18–30, 2007. 4.5

[116] G. Sheikholeslami, S. Chatterjee, and A. Zhang. Wavecluster: a wavelet-based clustering approach for spatial data in very large databases. *The VLDB Journal*, 8(3):289–304, 2000. 3.2.7

[117] Nobel Information Site. http://nobelprize.org. 2.2.1

[118] F. Sternberg et al. Does fall in tissue glucose precede fall in blood glucose? *Diabetologia*, 39(5):609–612, 1996. 2.3.2

[119] R. Storn. Differential evolution research – trends and open questions. *Studies in Computational Intelligence*, 143:1–31, 2008. 4.5.2

[120] R. Storn and K. Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11:341–359, 1997. 4.5.2

[121] M. Sugeno and T. Yasukawa. A fuzzy-logic based approach to qualitative modelling. *Transactions on Fuzzy Systems*, 1:7–31, 1993. 4.7.1

[122] W.H. Swann. *Direct search methods: Numerical Methods For Unconstrained Optimization.* Academic Press, New York, 1972. 4.5.1

[123] T. Takagi and M. Sugeno. Fuzzy identification of systems and its application to modeling and control. *IEEE Transactions on Systems, Man and Cybernetics*, 15(1):116–132, 1985. 3.1.3

[124] W.V. Tamborlane et al. Continuous glucose monitoring and intensive treatment of type 1 diabetes. *New English Journal of Medicine*, 359(14):1464–1476, 2008. 2.3.1

[125] H. Tanaka, S. Uejima, and K. Asai. Linear regression analysis with fuzzy model. *IEEE Transactions on Systems, Man and Cybernetics*, 12(6):903–907, 1982. 3.1.3

[126] The Expert Committee on the Diagnosis and Classication of Diabetes Mellitus. Report of the expert committee on the diagnosis and classication of diabetes mellitus. *Diabetes Care*, 25(1):S5–20, 2005. 2.1.3

[127] V. Thome-Duret et al. Use of a subcutaneous glucose sensor to detect decreases in glucose concentration prior to observation in blood. *Analytical Chemistry*, 68(21):3822–3826, 1996. 2.3.2

[128] M. Wang, N. Li, and S. Li. Local modeling approach for spatially distributed system based on interval type-2 t-s fuzzy sets. *Industrial and Engineering Chemistry Research*, 49:4352–4359, 2010. 3.4

[129] W. Wang, J. Yang, and R. Muntz. Sting: A statistical information grid approach to spatial data mining. *Proceedings of the 23rd International Conference on Very Large Data Bases*, pages 186–195, 1997. 3.2.7

[130] Diabetes Net Web. http://www.diabetesnet.com/. 1.1, 2.2.3

[131] Medtronic web site. http://www.medtronic.com/. 2.6

[132] S.A. Weinzimer et al. Fully automated closed-loop insulin delivery versus semiautomated hybrid control in pediatric patients with type 1 diabetes using an artificial pancreas. *Diabetes Care*, 31(5):934–939, 2008. 2.2.3

[133] I.M. Wentholt et al. Comparison of a needle-type and a microdialysis continuous glucose monitor in type 1 diabetic patients. *Diabetes Care*, 28(12):2871–2876, 2005. 2.3.1

[134] G. Williams and J.C. Pickup. *Handbook of Diabetes*. Blackwell Publishing, 2004. 1.1, 2.1, 2.1.3, 2.1.4, 2.2.1, 2.2.2

[135] B.J. Winer, D.R. Brown, and K.M. Michels. *Statistical Principles in Experimental Design (3rd ed)*. New York, McGraw Hill, 1991 p. 497–582. 5.3.1

[136] L. Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man and Cybernetics*, 1:28–44, 1973. 3.1.3

[137] T. Zhang, R. Ramakrishnan, and M. Livny. Birch: An efficient data clustering method for large databases. *Proc. SIGmod*, 6:103–114, 1996. 3.2.7

[138] H.J. Zisser. Accuracy of the seven continuous glucose monitoring system: comparison with frequently sampled venous glucose measurements. *Journal of Diabetes Science and Technology*, 3(5):1146–1154, 2009. 2.3.1