DESIGN AND ANALYSIS OF A CGM SENSOR GLUCOSE CONCENTRATION PREDICTION SYSTEM

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF APPLIED SCIENCES

OF

NEAR EAST UNIVERSITY

By

HANAN BADEEA AHMED

In Partial Fulfillment of the Requirements for

the Degree of Master of Science

in

Electrical and Electronic Engineering

NICOSIA 2013

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Hanan Badeea Ahmed

Signature:

Date:

ACKNOWLEDGEMENTS

Before any word, thanks to God Almighty for the blessings and giving me the health and strength to complete my thesis successfully. Hopefully, God always help and blessing me in the future. I would like to express my grateful and my sincere appreciation to my kind supervisor, Asst. Prof. Dr. Ali Serener, for his effort for support and guide to finish my study in courses and this thesis. I want to thank Dr. Özgür Özerdem for his supporting me in department and his help hand was giving to me in anything i need.

My deep thanks also to Near East University. Finally, I wish also to express my love and gratitude to my beloved husband for his emotional support and encouragement and to my sisters and my brother who's always pushing me forward throughout the duration of my studies. The Department of Electric and Electronic Engineering of the Near East University is the supporter for this research, I am grateful to all supporters.

ABSTRACT

Nowadays, continuous glucose monitoring (CGM) device has been the most useful tool for diabetes disease control, a diabetes patient is able to display his/her glucose concentration every minute for several days. This allows better controll of the prevention of occurrences of hypoglycemia and hyperglycemia, which can be excited by many factors, including insulin dosage, bodily activity, passionate tension, nervous tension and quantity of food consumed.

It is naturally desirable to avoid hypo/hyperglycemic cases before they occur and commercial devices exist that have an alarm to alert the patient for such cases. However, it is known that percentage of false alerts for those devices is still high and much is still needed to be done in order to improve that.

The purpose of this thesis is to design a blood glucose prediction system that can be used as part of a CGM device. With the help of a Kalman filter, glucose concentration is first reduced of its random noise component, and a neural network is then used for prediction of glucose up to two hours. Finally, this system is thoroughly tested for accuracy against various external factors. It is shown that such factors as patient's body weight, his/her exercise period and lifestyle may influence how well glucose concentration is predicted and therefore should be taken into account for early and accurate detection of hypo/hyperglycemic incidents.

Key words: Diabetes, CGM, Kalman Filter, Neural Network, Glucose Concentration

ÖZET

Günümüzde, sürekli glikoz izleme (CGM) cihazı diyabet hastalığının kontrolü için en yararlı bir araç olmuştur, bir diyabet hasta birkaç gün boyunca onun / glukoz konsantrasyonu her dakika takip edebiliyor. Bu insülin doz, fiziksel aktivite, duygusal stres ve gıda alımı da dahil olmak üzere birçok faktör tarafından heyecanlı olabilir hipoglisemi ve hiperglisemi, oluşunda önlenmesi daha iyi yönetimi sağlar.

Onlar ortaya çıkar ve ticari cihazlar bu tür olaylar için hasta uyarmak için bir alarm olduğunu var önce hipo / hiperglisemik olayları önlemek için doğal olarak arzu edilir. Ancak, bu cihazlar için sahte uyarılar yüzdesi hala yüksek ve çok hala geliştirmek için yapılması gereken olduğu bilinmektedir.

Bu tezin amacı, CGM cihazının bir parçası olarak kullanılabilecek bir kan şekeri tahmin sistemi tasarlamaktır. Bir Kalman filtresi yardımı ile, glukoz ilk olarak rastgele gürültü bileşeni azalır ve bir sinir ağı daha sonra iki saate kadar glukoz öngörülmesi için kullanılmaktadır. Son olarak, bu sistem iyice çeşitli dış etkenlere karşı hassasiyeti için test edilir. Bu hastanın vücut ağırlığı, onun / onu egzersiz süresi ve yaşam tarzı gibi faktörler glukoz konsantrasyonu tahmin ne kadar iyi etkileyebilir ve bu nedenle hipo / hiperglisemik bölüm erken ve doğru tespiti için dikkate alınması gerektiği gösterilmiştir.

Anahtar kelimeler: Diyabet, CGM, Kalman Filtresi, Yapay Sinir Ağları, Gliko Konsantrasyon

To my father and mother who are always pray for me

To my lovely friend and Guide, my husband (Mostafa) who always encourage me to keep going

To my lovely children (Ali, Nabaa), this for you

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Diabetes mellitus acts as the most crucial metabolic disease in the recent years, as a result of awareness lack of health. Glucose monitoring is an invention that will enhance the life of million patients. A patient can employs the readings of glucose monitoring to change any wrong activity contributed in glucose trend variations.

This research describes a system that shows the effect of a patient's body weight, his/her exercise period and lifestyle on glucose level prediction. The system uses a hybrid technique which comprises of a Kalman filter to initially take out noise from glucose concentration, and a back propagation neural network to predict new glucose concentration level up to two hours.

Chapter 2 is devoted to diabetes disease, and introduces continuous glucose monitoring (CGM) systems.

Chapter 3 is about CGM sensors. Chapter 4 describes some important factors which affect the measurement process performance of a CGM sensor.

Chapter 5 deals with the Kalman filter (KF) algorithm and how it can be tuned to denoise a signal.

Chapter 6 discusses the artificial neural networks (ANNs) and their terminologies, and how they can implement a prediction task.

Chapter 7explains the proposed network and gives details of its quantitative analysis.

Finally, chapter 8 summarizes the conclusions obtained from the suggested analysis and the recommendations for the future.

1.2 Literature Review

An inclusive search has been done by the Direct Net study group [1], which analyzed the enhancement in continuous glucose monitoring sensors accuracy backing to modifying the timing and number of the standardization points. The conclusion of study results leads to that the timing of the calibration points is even more imperative than the number. Gianni Marachetti et al [2] suggested an amended proportional integral derivative (PID) control approach for blood glucose controll and seriously evaluated in silica using physiologic style of Hovorka. M. Stemmann et al [3], his report get a consideration that a standardization model could be obtained involving original blood glucose readings and debasing noise to the readings of the non-invasive glucose monitoring (NIGM) sensor. Using the supposed procedure, the influence of the original readings and the noise on the sensor readings could be analyzed. Furthermore, they found that standardization models different among many patients gives imminent into the variability of the non-invasive sensor between various patients. Additionally, the calibration model to determine the dynamics of the sensor according to the fundamental blood glucose concentration used in their work and degrading noise could be investigated.

It is the initial instant that Kalman filter (KF) used to practice CGM information while Knobbe and Bukingham [4] presented their work; anywise the idea of this research was to remake blood glucose concentration, and not to diminish the noise CGM information. Most beneficial estimation with aid of KF has been anticipated by Palerm etal [5], the goal of this method is to predict the glucose trend and revealing hypoglycemia. KuureKinsy et al [6], employed the double - ratio Kalman filter for true time CGM device in order to get better CGM standardization as possible as. They utilized CGM sensor for prediction of glucose and its ratio -of-varying if an ordinary five minute sampling accounted of a noisy signal. The procedure gave an uncommon eight hour intervals essential glucose indicator samples the ability to the sensor gain and its varying ratio to be revised. This Kalman filter model accounts for ambiguity in both the CGM sensor and the essential glucose indicator. The research group tested this strategy on factitious and experimental concentrations, reinforcing its validity to straightforward one-point calibration. Facchinetti et al. [7], make an extensive challenge to denoise CGM sensor, their proposed method based on a Bayesian estimation designed and executed by kalman filter. The conclusion leads to the fact that a best possible filter, which satisfies the finest association between noise lessening and signal deformation, can be achieved by getting the filter design problem within a Bayesian background. Andrea Facchinetti et al [8] thought about a new online tactic to reducing noise of CGM signals using a Kalman filter, whose unidentified parameters are modified in a certain individual by a arbitrarily centered silky typical exploiting data of a burn-in period. They compared results with those calculated by a moving-average (MA) filtering approach with permanent parameters presently in use in probably all mercantile CGM devices. Conclusion show that the new kalman filter approach behaves much better than MA.

Mark B. Savage et al [9] formed an artificial nervous network to figure out the CGM sensor output plus the system parameters, and show a relationship between them and blood glucose levels, by emerging a noninvasive blood glucose determining mean, depend on employment of the visual viaduct in the near – infrared zone. A back propagation neural (BPN) network utilized for obtaining blood glucose in diabetic patients by V. Ashok et al [10] research. The analysis recorded a non- invasive recordings of CGM concentrations based on reflected laser ray from the index finger. During the process the index finger is cited in the laser beam transceiver element, the reflected visual wave is altered into its corresponding electrical wave and the resulted wave is processed by the nervous network which introduces the results in the form of BG concentration. Diabetes catalog employed for declared rapprochements and they concluded that back propagation nervous network carries out more perfectly. Scott M. Pappada et al [11] utilized NeuroSolutions program to build various neural network styles with variable predictive screens of 50, 75, 100, 120, 150, and 180 minutes. They trained the network using patient information groups ranging from 11-17 patients and calculated the patient information not incorporated in neural network structure. The calculation of mean absolute difference percent on the whole and at hypoglycemic and hyperglycemic cases is the aim of this tactic.

Facchinetti et al. [12] introduced again a novel technique for noise decline able to deal also with the personality varying of the signal to noise ratio(SNR).Their tactic depends on a Bayesian smoothing procedure that employs a statistically-based scale to get and continuously notify, filter parameters in real time. S.Shanthi and D.Kumar [13] took in their research the elimination of errors caused by various noise models in CGM device readings. They trains a feed forward neural network with Extended Kalman Filter (EKF) algorithm to negate the effects of white Gaussian, exponential and Laplace noise models in CGM time series. The nervous network elements renewed with respect to the signal to noise ratio of the entering signal. The plan is being tested in pretended data and twenty real patient’s data set. The validity of the proposed system is analyzed with root mean square error (RMSE) as metric and has been compared with preceding approximations in terms of time delay and smoothness relative gain (SRG). They concluded that hopeful results can deals with the usage of CGM signal auxiliary to systems such as hypoglycemic alert generation and input to artificial pancreas. C. Zecchin et al. [14] aimed in their work to build up a new short- period glucose prediction system using a neural network that, as well to all past CGM readings, also uses details on carbohydrates intakes quantitatively described through a physiological model. Results on simulated data quantitatively show that the new algorithm outperforms other published algorithms.

Panteleon and colleagues [15] advance the regulation of CGM with assist of a seventh order finite impulse response (FIR) filter by proposing that even if standardization with sensor current as the autonomous variable get a bias in the estimate of blood glucose, it is a more fitting regulation method as the decreasing of the mean absolute difference (MAD) between sensor present glucose reading and blood glucose had an initial anxiety. Keenan and associates [16] have studied the delays in CGMSGold and GuardianRT instruments, by a demonstration analysis of the data collection to determine a modern calibration algorithm utilized in the Paradigm Veo insulin pump.

An integral based fitting and filtering algorithm for a CGM data developed by Chase et al. [17], but it requires that insulin dosages should be identified. Their research compares two metabolic models in terms of the predictive power. When a extended prediction window of more than five hours employed, glucose sensor predictions attempt to be more accurate in the collection from New Zealand while the new model tries to predict better in the collection from Denmark. For both models, outlying prediction errors are subjugated by single patients, particularly type 1 diabetic patients. CGM sensor predicted blood glucose concentrations are generally higher compared to new predicted values. As expected, the root mean square (RMS) prediction error increases with prediction interval for both models and collections.

1.3 Aim of Thesis

In this study, the first goal is to remove noise associated with a CGM device using a Kalman filter, as it works well with non-linear applications. The second goal is to use a back propagation neural network to implement a prediction system for the filtered glucose concentration.

CHAPTER 2

DIABETES MELLITUS DISEASE

2.1 Diabetes Mellitus

In the present time, the foremost health problem in the world is Diabetes Mellitus. Newly developing countries suffering from the arising of this disease. All health organizations took on healthy lifestyle practices associated with the avoidance of diabetes specifically creating appreciation and balance diet, maintain ideal body weight and physical activities were encouraged. It is necessary also to remember people about the complications relating to diabetes, by offering a guide lines on the management of diabetes and by patient education. Diabetes is a disease in which the body stop make or not make enough the insulin hormone. Insulin is a hormone that is converts the blood glucose into energy needed for daily life. The effect of diabetes continues to be a dangerous problem in future, and this effect associated with both genetics and environmental factors. There are two major types of diabetes; type 1 and type 2 [18].

2.2 Type 1& Type 2 Diabetes

Diagnosis of Type 1 diabetes is customarily done in young adults and children. In type 1 diabetes, the body does not make insulin. Human body needs insulin to be able benefit from glucose. Glucose sugar is the essential fuel for the body cells, and insulin transfers the sugar from the blood into the cells. The most general shape of diabetes is type 2 diabetes. In this kind, either the body does not make enough insulin or the cells ignored the insulin. If the blood saturated with glucose instead of going into cells, this leads to two problems: right absent, cells may be very hungry for energy and over time, increasing in blood glucose concentrations may harm eyes, kidneys, nerves or heart.

When chronic hyperglycemia appears at the diabetic patient, increases the risk of micro vascular destruction, which causes retinopathy, nephropathy, and neuropathy. Hence, diabetes is the leading cause of blindness and visual weakness in adults in developed countries and is in charge for over one million lower limb amputations each year. A superior risk of macro vascular complications threats diabetic people, where they are two to four times more possibly to infect cardiovascular disease (CVD) than people without diabetes. Because of these complications, diabetes represents the fourth major cause of global death by disease. Actually, obesity, in particular, central obesity, physical inactivity, and unhealthy dietary habits speeds up the infection of type 2 diabetes. Whereas the patient reveals diabetes quickly, this would be of great to avoid consequences. In fact at least 50% and 80% in some countries, of all people with diabetes are unaware of their condition and will stay unaware until complications appear.

Recently clinical studies found that 80% of type 2 diabetes complications can be eliminated or delayed by premature appreciation in people at hazard of this disease, by varying their lifestyle and/or by curative methods. Smart data analysis, like continuous glucose monitoring sensor will help those people to enhance their conditions [19].

2.3 Continuous Glucose Monitoring System

Newly, the progress that happens in continuous glucose monitoring (CGM) devices awarded new opportunities to manage glycemia of diabetic patients. The minimally invasive nature of modern CGM devices offers a mean to compute and record a patient's current glycemic state as possible as every minute. a closed-loop artificial pancreas is mere a Continuous glucose sensors coupled with continuous insulin infusion pumps. The adjustments of insulin infusion rates don by closed-loop control algorithms automatically to sustain blood glucose at a desired concentration (e.g. 4-7 mg/dl). Despite of the high performance of developed control algorithms; they repeatedly require a time-consuming task of presenting an appropriate model for control. Consequently, it is desirable to promote modern models of techniques to be a basis for controller execution and design, and this leads to a correctly prediction of glucose level for long prediction windows to recompense for the lag time between: under skin glucose and blood glucose concentration. By other word, the prediction time based on the comparative delay between the CGM system readings and the blood glucose value [20].

From the clinical point of view, a CGM system which has different chemical parameters is always necessary. The process of blood sampling in the intensive care units and analyzing in the laboratory takes long time and its be too slow in the critical cases. For that reason, the continuous monitoring of blood glucose can be used as an aid for the treatment of diabetes to improve the process of dose optimization in the beginning of the medical therapy. Patient can daily use a small wearable tool for the continuous glucose monitoring to decide about the required amount of insulin much more exactly and reduce the danger of hypoglycemic innings. As a future step the CGM system could be attached with an proper insulin delivery system to form a miniaturized artificial pancreas.

The CGM systems use a teeny sensor inserted subcutaneously to check glucose levels in tissue fluid. This sensor still in place for several days to a week and then should be changed. A transmitter sends information about glucose levels by means of radio waves from the sensor to a pager like wireless monitor. The patient must verify blood samples with a glucose meter to program the devices. Patients should corroborate glucose levels with a meter before making any change in treatment, because currently approved CGM devices are not as accurate and reliable as standard blood glucose meters, Figure 2.1 shows the complete system of glucose monitoring [19].

Fig. 2.1: Continuous glucose monitoring system [20]

CHAPTER 3

CONTINUOUSE GLUCOSE MONITORING SENSOR

3.1 Glucose Sensor

Several new continuous glucose monitoring systems have been introduced. Some are invasive, such as enzymatic sensors – which can be fully implanted or transcutaneous – or the transcutaneous micro dialysis technique. Others are non – (or almost none) – invasive, such as the iontophoretic or the optical techniques. Figure 3.1 shows the internal structure of glucose sensor [21].

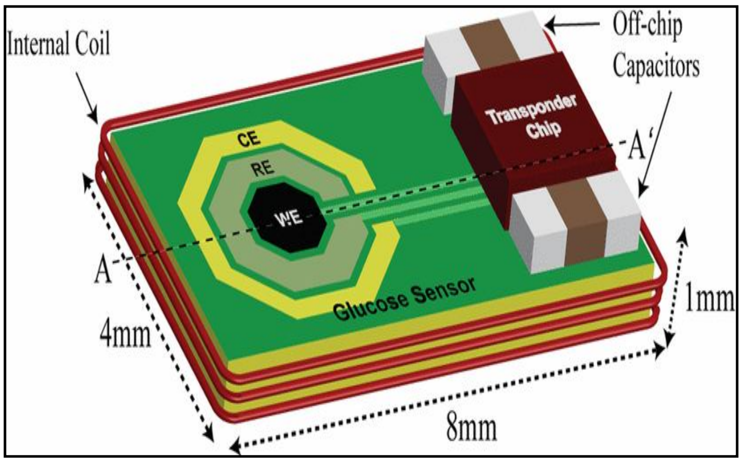

Fig.3.1: Internal construction of CGM sensor [22]

3.1.1 Fully Implanted Sensors

The biggest advantage of a fully implanted sensor that there is no material inserted through skin. Even if, there are many problems quiet remain, because the implant place is in a blood vessel (and this may be caused blood clotting) or in the under skin tissue. Although subcutaneous place is suitable because it is relatively easy to insert in, but there are many limitations about reliability and time of functioning of a sensor sited here. The future of the technique depends on these obstructions. For more convinced, most under skin ingrained glucose sensors are not able of monitoring glucose for more than few hours because of a foremost drift in electrical signal. The biological surroundings affected significantly in the difficulties faced with ingrained sensors because the separated sensors work correctly in laboratory. The recent study reports found that if an interfacing angiogenesis membrane added, to imitate the advance of capillaries around the sensor, this will briefed the early period of confused monitoring and enhanced sensor permanence. The meaning that it is possible to monitor glucose in subcutaneous tissue for long intervals of time: the mean lifetime of the sensors was > 100 days [21].

3.1.2 Transcutaneous Devices

The design of a through skin electro enzymatic glucose sensor began before 20 years ago. The sensor contains oxidize within envelop was placed at the top of a needle – like electrode. In recent times, an electrode like this has been developed commercially introduced by minima: mass production has offered a very low-cost glucose sensor, which can be rooted for about 3 days in subcutaneous tissue. The sensor's electrical signal drops specifically in the hours after implantation and drifts over the subsequent days, making calibration required several times a day with the patients over blood glucose. The monitor displays only the electrical current on its screen and the results can be interpreted only after all the recording has been transferred onto a computer. Therefore, in its present mode, the system is practical for a hyperglycemia holder secondarily interpreted by the physician, but new versions should make the blood concentration directly offered to the patient [21].

3.1.3. Non – Invasive Sensors

A new technique called Iontophoresis, in which a low – density electric current is passed through the skin between an anode and a cathode. Principally, the movement of sodium ions toward the cathode carrying the excited current. Other molecules that didn't charge such as glucose are transported by electro – osmosis. The quantity of extracted molecules at the cathode, calculated by a glucose oxides biosensor, is associated with blood glucose. After calibration, The GlucoWatch glucose oxides biosensor using finger stick blood glucose information and a three hours equilibration period, supply readings every twenty minutes. Adequate results from 40 – 400 mg/dl given by this watch, but causes some degree of local iteration and does not used throughout periods of increased sweating. The temperature and conductance of skin sensors eliminate these confusing factors, and about twenty per cent of all readings are passed over for these reasons.

Optical sensors use the opinion that fingertip represents the absorption pattern of near – infrared light. Highly structured mathematical models used to process the reflected light signal to filter out the interferences from biological molecules, tissue structures and the optical properties, and to enlarge any aspects of the signal that may illustrate some correlation with blood glucose. This technique had a difficulty that it is require for glucose selectivity, and investigate is needed to analysis the complex in vivo factors that affect the optical measurement of blood glucose [21].

3.2 Advantages of Glucose Monitoring

There are two major advantages identify the modern glucose sensors over old CGM techniques: they are fewer invasive and they permit monitoring of ambulatory patients. The most important benefit of CGM devices is that they supply information about the ambiguous variations of blood glucose level to patient every few minutes. The latest devices have a screen in which patient can see whether glucose levels are increasing or falling. There are some systems also have an alarm to let him know when his glucose reaches high or low levels. Others are able to display figures detecting glucose levels accumulated over an evident number of hours on its display screen. Resulted data on all devices can be downloaded to a computer for graphing and extra vital trend analysis. CGM device record patient blood glucose levels every few minutes so he/she can follow the direction of blood glucose varying. Depending on the trend – for example, whether the glucose is rising or falling – patient may decide to take action differently to the same number. It is possible to see trends in his/her glucose levels may inspire to him/her to change any wrong actions before glucose levels become awkward.

Actually CGM systems are very active instrument to detect early the occurrence of an “transitional " problem such as hypoglycemia. Data analysis translates patient response to the problem and may guide him/her to prevent the problem from happening again. The device can aid in checking blood glucose concentrations overnight, over a part of a day, or over several days to see the superior management view. Patient may be able to display trends on the monitor itself or he/she may need to download the information onto a computer, based on the CGM device being used, and during that there are many questions start taking shape in patient mind. The questions like: What classically happens after meals? Does it depend on the kind of foods eaten, time of day, and timing of insulin dose? When hypoglycemia does occur? What effect does exercise, school, work, or dining out have on glucose levels?, are more usual. The answers of all these questions offered by CGM device and the results from device must be attached with written records of his/her daily routines and assignments by diabetes care provider at the next appointment. So patient and device can decide what changes he/she may need to maintain blood glucose levels in the objective range [23].

3.3 Disadvantages of Glucose Sensor

Not always the CGM devices can be useful for diabetes patients in all cases. The reading of device must be always confirmed with test results of the care provider of patient. Patient must change the sensor every 3 or 4 days and monitors must change from 6 months to about 2 years, based on the manufacturer. For this reason traditional finger stick blood glucose measuring is still required, and is still considered necessary for device calibration and to verify hypo- or hyperglycemia before any corrective action. Time lag is what are researchers discussed continuously, it raised between 5 and 20 minutes registered by the multi types of CGM devices because the blood glucose reading is drawn from under skin fluid and does not give the actual blood glucose concentration that is measured in standard finger stick blood samples drawn from capillary blood. The keyword to remember the advantage of the CGM devices is trend. Lag time is trivial when blood glucose levels are relatively steady – and these appear clearly on the CGM monitor. On the other hand, if the CGM monitor shows that the blood glucose level has been dropping over a short phase of time, a finger stick test is recommended to check for hypoglycemia. [23].

3.4. Noise Associated with Glucose Sensor

In all applications of CGM sensors, precision of glucose readings are affected by the presence of various causes of error, allied to device physics, chemistry, and electronics. The modern experiments deal with the most general approach by comparing of CGM readings and original blood glucose (BG) samples collected at the same time by laboratory techniques. The process of measuring has many difficulties, because CGM measurements are collected in a location different from the blood, i.e., under skin, and, as well, originals BG are available at least around 30-min sampling. Whenever sensor exactitude is concerned, it is well be sure that CGM time series are also corrupted by a random noise component which complicates signal elucidation and, particularly, may decline the performance of hypo/hyperglycemic alert generation systems as well as that of the controllers entrenched within artificial pancreas algorithms. Though, categorization of sensor random noise is relatively unfamiliar. [24].

CHAPTER 4

FACTORS AFFECTING CGM MEASUREMENT PROCESS

4.1 Introduction

The CGM devices utilized electrochemical sensors have a property of negligible invasive put under skin. The CGM devices aid the diabetes people in recognizing that what happens with blood glucose and what cause its variation drift. Measurement of accuracy of CGM monitors is difficult for two initial reasons:

1. CGMs calculate blood glucose variations ultimately by measuring the concentration of under skin glucose and still calibrated using self-monitoring to converge to blood glucose.
2. CGM information point to an underlying process in time and consist of ordered-in-time highly inter dependent data points.

All factors have a physiological nature such as time lag, improper calibration, random noise, errors due to sensor physics, and chemistry affects the accuracy of CGM data. This damage the performance of CGM signals in hypoglycemic alert generation and control input to artificial pancreas. The standard reports of this technique have offers clear guidelines on how to use and present data in CGM devices. Different types of CGM devices are accessible nowadays. Sections below summarize the factors that affect on the measurement process of continuous glucose monitoring sensor [25].

4.1.1 Calibration

All online CGM systems that available in market need to a calibration. Calibration is the process of transformation of signal generated by glucose sensor at certain time which is just a very small current (nA), into estimation of glucose concentration. [26]. Glucose concentration in this process measured using on or many self measured blood glucose SMBG samples. Calibration need to assess the inspiration of the number, accuracy, and temporal position of the reference SMBG samples, as well as by the trend of glucose concentration at their pick uptimes. The position of drawing a blood sample by CGM devices is under patient skin, and thus they measure interstitial glucose (IG) instead of blood glucose (BG) concentration, therefore calibration is required here. In real conditions of patient’s life style, e.g., after taking food, IG and BG can be obviously different because of the existence of a BG-to-IG kinetics which has been described by a two- partition model [29],

$$\mathrm{IG}(t) = -\frac{1}{t}\mathrm{IG}(t) + \frac{g}{t}\mathrm{BG}(t) \tag{4.1}$$

where g represents the static gain of the BG-to-IG system (which considered equal to 1, i.e., in steady state, the concentration of glucose in both sites are equal) and t is a time constant (change from individual to other). Equation (4.1) represents a first order, linear, low-pass filter, and introduces a distortion and alleviation in amplitude and phase delay, which is readily evident in Figure 4.1 (top panel).

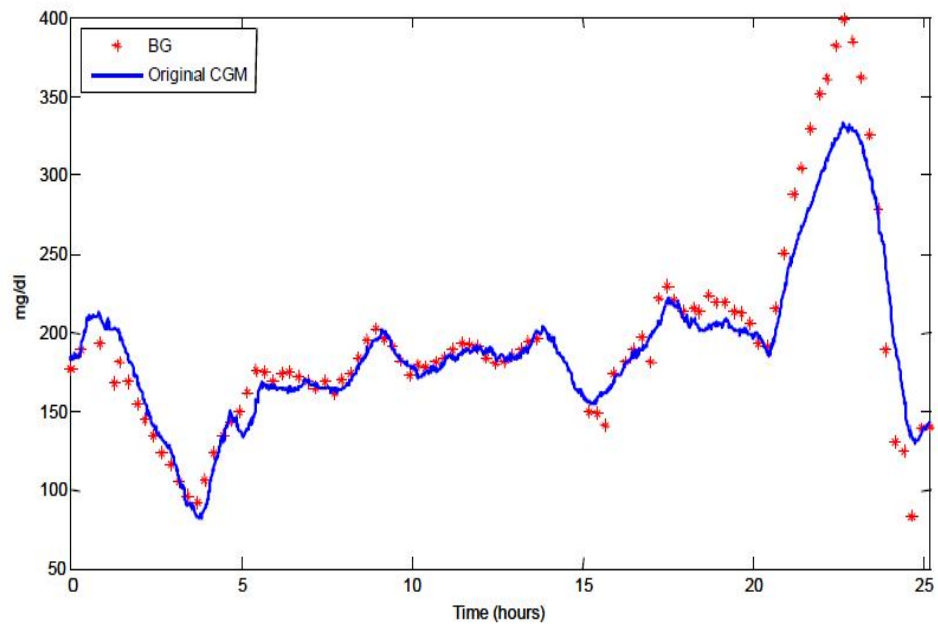

Fig. 4.1: Blood glucose levels (stars) vs. original CGM blood glucose levels (continuous line). [26]

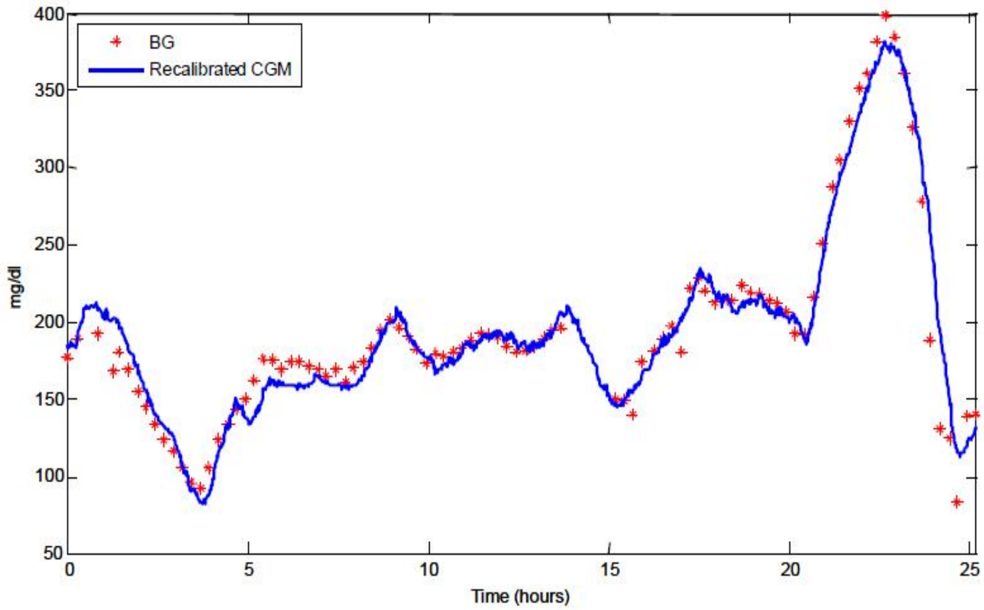

Fig. 4.1Blood glucose levels (stars) vs. CGM blood glucose levels (blue line) recalibrated by the method of King et al. [26].

This figure shows a comparison performed in a clinical study on a type 1 diabetic subject between a CGM blood glucose levels and blood glucose levels collected every 15 min and determined in laboratory. There are some of discrepancies are marked along the y-axis like those appears in the period starting at 18 hour until 25 hour, could not explicate because of BG-to-IG kinetics existence. Possibly, this difference is due to a change of behavior of the CGM sensor operation after its initial calibration. This make CGM profiles affected by calibration problems and have a crucial effects in several applications such as alert generation systems and artificial pancreas. For this reason, real time recalibration of CGM data is always required, and there is an intention to process sensor output (in mg/dl) by an algorithm can attached externally to the device in order to improve its precision. As a result of recalibration, the difference between BG and CGM samples should be more accurate depending on BG-to-IG kinetics only. A linear regression model thought for an off-line application for many recent recalibration procedure,, which is briefed by equation:

$$y = ax + b \tag{4.2}$$

where a and b are recalibration elements which are calculated by fitting them against a couple of BG and CGM pairs, i.e., y and x in Equation (4.2), correspondingly, collected at the same time. [27]

4.1.2 Filtering

The analysis of continuous glucose monitoring signal can be demonstrated with the equation,

$$y_k = u_k + v_k \tag{4.3}$$

where y_k is the measured CGM signal, u_k is the unidentified glucose value at time 'k' and v_k is the additive noise which due to measurement error. If the spectral characteristics of noise anticipated, low pass filtering can be used as the most natural nominee to denoise CGM signals. Since signal and noise spectra normally overlap, deduction of noise v_k will cause distortion in the true signal u_k and this a very vital problem with low pass filtering. Distortion and delay affecting the estimate of true signal. The CGM time series observed with different sampling rates, thus there is a need process it by filters with varied parameters and filter optimization made on order and weights cannot be directly reused from sensor to another. Furthermore, filter parameters should be tuned according to the SNR of the time series, whereas the SNR higher, the filtering be more flat. Exact tuning of filter parameters in an automatic manner is a thorny problem for the basic filters. So far the filtering approaches have been tested with a consideration of white Gaussian noise alone in CGM sensor data. Despite these marvelous works by various research groups, attainment of 100% accurate prediction is still an arduous task. This clarifies the need of more intelligent filtering algorithms. For reliable real time monitoring of blood glucose, the filtering algorithm should account for:

1. Short term errors due to motion artifacts.
2. Random Noise and other noise types.
3. Errors due to inadequate calibration.
4. Long term errors caused by performance decline of sensor, bio fouling, inflammatory complications etc.
5. Ambiguity in physiological elements [13].

4.1.3 Prediction

All ordinary on-line application of CGM sensors has the ability to reveal hypo/hyperglycemic cases. Some methods were proposed to generate alerts after the appearance of CGM sensors by few years in the market, in which the actual trend of the glucose concentration suggested that hypoglycemia was likely to take place within a short time. Projection methods are the other name for these techniques. The CGM sensors can be enhanced by generating hypo-/hyper-alerts manufactured on the bases oft ahead-of-time prediction of glucose concentration, which can be determined from past CGM data and appropriate time-series models. The different prediction windows surely affect on the process [27].

4.1.4 Alert Generation

The generation of alerts to match requirements and concepts is a very critical situation. All commercial systems that generate alerts comparing the actual glucose level and a pre-selected level. Nevertheless, the efficiency of these systems is divisive. High percentage of false identifies those alerts particularly, although many applications of CGM sensors allowing enough sensitivity. Besides, these systems cannot avoid the event, because they generate the alert when the event occurs while its need to be generated before the happening of event. To overcome this limitation, some devices now perform a trend analysis, pointing to variation direction and rate of glucose, in order to provide the patient with an early caution. However, to the best of our consciousness, no large scale studies have quantitatively renowned in rival -reviewed articles the benefit of this procedure. Because of inaccurate CGM data due to calibration problems and always uncertain, generating alerts accurately is difficult. The mathematical background behind the generation of alerts should therefore be set on more solid foundation by assuming, in addition to a trivial threshold comparison, the uncertainty of the data, which should be estimated in real-time in a statistical setting to evaluate, a suitable (SNR) [27].

CHAPTER 5

ON KALMAN FILTERING

5.1 What is Kalman Filter?

From the theoretical point of view Kalman filter is an recursively estimator for the linear-quadratic problem, which is the problem of estimating the immediate state of any linear dynamic system troubled by white noise by employing information linearly associated to the state and also corrupted by white noise. The result is an estimator statistically most favorable and provides solution for any quadratic function of estimation error. From practical side, it has very large importance in the field of statistical estimation theory and possibly the greatest invention in the twentieth century. It has become very necessary in estimation problems as the silicon necessary in the makeup of many electronic systems. Kalman filter has many abrupt applications all of them used for the control of complex dynamic systems like aircraft, ships, spacecraft and all continuous manufacturing processes. This filter make available to deduce the absent information from roundabout (and noisy) information, because in applications of control theory it is not at any time probable or required to determine all wanted variables. It's also used for predicting the anticipated future trajectories of dynamic systems which be vague to control, the examples are: flowing of rivers during flood, the paths of celestial bodies, or the prices of traded possessions [28].

The Kalman filter inspired its name from Rudolph E. Kalman who published in 1960 his famous paper clarifying a recursive answer about all discrete-data linear filtering problems. The researchers expanded when they dealing with the subject because of the great variety of applications in many fields from engineering to finance. All applications contain, in some way, stochastic estimation from noisy sensor measurements [29].

5.2 Why It's Called a Filter?

Generally, a filter is a physical device for removing unwanted fractions of mixtures. Firstly, a filter solved the problem of sorting out undesired components of gas-liquid-solid mixtures. In the field of crystal radios and vacuum tubes, the item was applied to analog circuits that “filter” electronic signals. There are different frequency components included in these signals, and these physical devices preferentially attenuate unwanted frequencies. This conception was extended to the isolation of “signals” from “noise,” both of which were characterized by their power spectral densities. In case of kalman filter it’s very unfamiliar that the term “filter” would apply to an estimator. Klomogrov and wiener employed this statistical specifications of their probability distributions in forming an optimal estimate of the signal, given the sum of the signal and noise.

The feature of Kalman filtering is explained in that it can used in the original ideal filtering of separation of the components of a mixture, and additionally it’s also solved the inversion problem, in which its possible to represent the determined variables as functions of the most interested variables. Essentially, it converse this functional rapport and predicts the autonomous variables as reversed functions of the dependent (measurable) variables. These variables of interest are also allowed to be dynamic that are only partially predictable [28].

5.3 The Mathematical Foundations

The essential subjects forming the mathematical basics for Kalman filtering theory are shown in figure 5.1. Despite this shows Kalman filtering as the top of pyramid, it is itself part of the basics of another punctuality control theory and a proper subset of statistical decision theory [28].

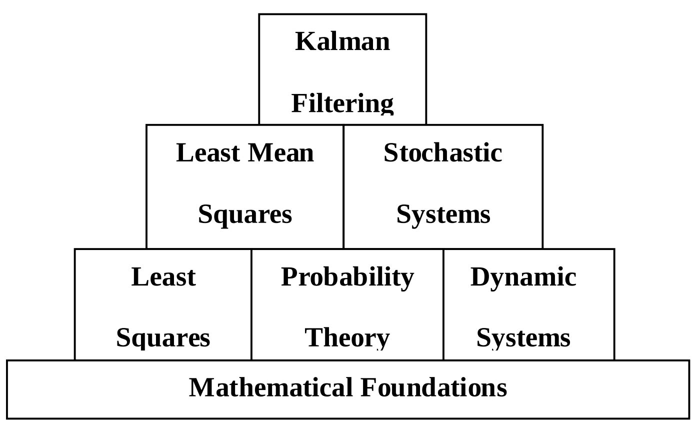

Fig.5.1: Basic Concepts in Kalman Filtering [28]

5.4 What It Is Used For?

The essential use of kalman filter is almost exclusively for two purposes: estimation and analysis of estimator's behavior, although its applications cover many fields. [28]. A whole characterization of the probability distribution uses estimation errors of The Kalman filter in evaluating the best filtering gains, and this probability distribution may be used in assessing its performance in term of "design parameters" of an estimation system, such as

- Kinds of sensors to be used,
- The various sensor types positions and directions according to the system to be estimated,
- Sensors primitive noise characteristics,
- The pre-filtering methods for soften sensor noise,
- The various sensor types data sampling rates, and
- The model simplification level to decrease implementation requirements.

5.5 Kalman Filter Algorithm

The description of steady-state Kalman filter can be briefed using the following equations:

Measurement update

$$\hat{x}[n|n] = \hat{x}[n|n-1] + M(y_v[n] - C\hat{x}[n|n-1]) \qquad (5.1)$$

Time update

$$\hat{x}[n+1|n] = A\hat{x}[n|n] + Bu[n] \qquad (5.2)$$

For these equations:

- $\hat{x}[n|n-1]$ is the predict of x[n] given past calculated value up to $y_v[n-1]$
- $\hat{x}[n|n]$ is the updated predict based on the last calculated value $y_v[n]$
- M is the innovation gain of Kalman filter
- u[n] is the original signal or input to be predicted
- $y_v[n]$ is the noisy signal which represents the last calculated value.

- A, B, and C are the matrices of mathematical model used.

Given the current predict $\hat{x}[n|n]$, the time update predicts the state value at the next sample (one-step-ahead predictor). The measurement update then adjusts this prediction based on the new value of $y_v[n + 1]$. The correction term is a function of the innovation, that is, the discrepancy,

$$y_v[n+1] - C\hat{x}[n+1|n]$$

between the measured and predicted values of $y[n + 1]$. The innovation gain M is chosen to minimize the steady-state covariance of the estimation error given the noise covariance

$$E\left(w[n]w[n]^T\right) = Q \qquad E\left(v[n]v[n]^T\right) = R \qquad N = E\left(w[n]v[n]^T\right) = 0$$

The time and measurement update equations can be attached into one state-space model (the Kalman filter).

$$\hat{x}[n+1|n] = A(I - MC)\hat{x}[n|n-1] + [B\ AM]\begin{bmatrix} u[n] \\ y_v[n] \end{bmatrix}$$
$$\hat{y}[n|n] = C(I - MC)\hat{x}[n|n-1] + CMy_v[n]$$

This filter generates an optimal estimate of $y[n]$. Note that the filter state is $\hat{x}[n|n-1]$ [30].

5.6 How to Tune the Kalman Filter

If Kalman Filter linked to the real system, then it must be tuned very well. The algorithm of this filter usually used two essential elements: process disorder (noise) auto-covariance Q and/or the measurement noise auto-covariance R. In real systems measurement noise mainly introduces noise into the estimates. If the value of Q is large, then stronger measurement-based updating of the state estimates because a large Q inspire to the Kalman Filter that there are large variations in the real state variables (keep in mind that the process noise influences on the state variables). Consequently, the larger Q the larger Kalman Gain K and the stronger updating of the estimates. The key tuning rule is as follows: Select as large Q as possible without the state estimates becoming too noisy [28].

5.7 The Impact of Kalman Filter on Technology

From all features involved in estimation and control problems, at least, this has to be considered the greatest achievement in estimation theory of the twentieth century. There are many achievements would not have been possible without it. It was one of the enabling technologies for the Space Age, in particular. Without it the precise and efficient navigation of spacecraft through the solar system could not have been done.

Kalman filtering has many standard uses have been employed in modern control systems, such as the tracking and navigation of all sorts of vehicles, and in predictive design of estimation and control systems. These technical activities were made possible by the introduction of the Kalman filter [28].

5.8 Advantages of Kalman Filter

- The Kalman filter is executable in the shape of a program run with a digital computer, this mean that it can replace analog circuitry for estimation and control. The implementation may slower, but it is capable of much higher accuracy than had been authentic with analog filters.
- The deterministic dynamics or the random processes that have stationary properties does not required with kalman filter, and many applications of importance involve no stationary stochastic processes.
- It is well-matched with the state-space formulation of optimal controllers for dynamic systems, and it was able to prove useful dual properties of estimation and control for these systems.
- The Kalman filter offers the required information for mathematically, statistically-based decision methods for revealing and refusing irregular measurements [28].

CHAPTER 6

ARTIFICIAL NEURAL NETWORKS

6.1 Introduction

Neural networks are extrapolated from biological nervous systems. Its contain simple units working in parallel. As in nature, the network determined its function mostly by the relations between units. The concept of work is the obligation of a neural network to perform a particular task by regulating the values of the relations (weights) between units. Frequently neural networks are adjusted, or trained, so that a particular input leads to a specific target output. Such a situation is shown in figure 6.1 below. In figure below, the network is regulated; by comparing the output and the target, until the network output corresponds the target. Typically many such input/target pairs are used to train a network [31].

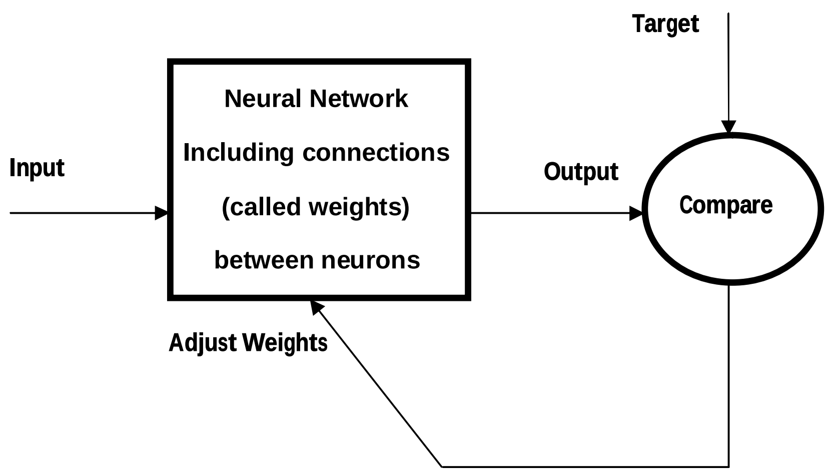

Fig. 6.1: Block diagram for neural network structure [31]

6.2 Definition of Artificial Neural Network

Nowadays, an artificial neural network would be very desirable. Although computing these days is actually advanced, there are specific functions that a program run for a general microprocessor is unable to perform; nevertheless a software execution of a neural network has many advantages and disadvantages [32].

Advantages

- A neural network can achieve functions that a normal program cannot.
- If any unit of the neural network fails, neural network can go on without any problem as parallel nature.
- This network learns and does not need to be reprogrammed.
- It can be implemented in any application and without any problem.

Disadvantages

- The training is necessary for neural network to operate.
- A neural network structure is unlike the microprocessors structure, therefore needs to be emulated.
- High processing time required for large neural networks.

6.3 Artificial Neural Networks: Terminology

Processing Unit: The artificial neural network (ANN) looked a very easy model if it's compared with the biological neural network. It involves interlinked processing units. The processing unit has a broad frame contains a summing part go ahead by an output part. The summing part extradites N number of input values, multiplied each value by a weight, and counts a weighted sum. The result from summing part is called the activation value. A signal from the activation value produced by the output part. The weight's sign for each input measures if the input is excitatory (positive weight) or inhibitory (negative weight). The input could be discrete or continuous data values, and by same the output also could be discrete or continuous. The input and output could also be deterministic or random or vague.

Interconnections: Several processing units are interlinked in an artificial neural network with respect to some manner to accomplish a pattern recognition task. For this reason, the inputs to a processing unit may come from the outputs of other processing units, and/or from outside resources. Each unit gives its output to some units including it. The strength of the connection between the units affected on the amount of the output of one unit received by another unit, and it is translated in the weight value associated with the connecting link. If there are N units in a given ANN, then at any instant of time each unit will have a single activation value and a single output value. The activation state of the network at that instant defined by the group of the N activations values of the same network. By the same manner, the group of the N output values of the network defines the output state of the network at that instant. Depending on the discrete or continuous nature of the activation and output values, the state of the network can be described by a discrete or continuous point in an N-dimensional space.

Operation: In this stage, each unit of an ANN receives inputs from other linked units and/or from an outer resource. At a given instant of time the weighted sum of the inputs is evaluated. The actual output from the output function unit determined by the activation value, i.e, the output state of the unit. Sequentially, The activation and output states of other units determined by the output values and other external inputs. The activation values of all units determined by the activation dynamics and the activation state of a network as a function of time. The activation dynamics also determines the dynamics of output state of the network. The activation states group defines the activation state space of the network. The trajectory of the path of the states in the state space of the network determined by the activation dynamics. For a given network, defined by the units and their interlinking with suitable weights, the activation states determine the short term memory function of the network.

Update: During the implementation, there are several choices accessible for both activation and synaptic dynamics. In particular, the updating of the output states of all units could be performed at the same time. In this case, the activations values of all units are counted at the same time. The new output state of the network is derived from the activation values. On the other hand, in an asynchronous update, unit is updated sequentially, receiving the present output state of the network each time. For each unit the activation value determines the output state either deterministically or stochastically. In reality, the activation dynamics including the update is much more complicated in a biological neural network than the simple models mentioned above. The ANN models along with the equations prevailing the activation and synaptic dynamics are designed according to the pattern recognition task to be carried out [32].

6.4 Learning Rules

From all that mentioned above, weights are adapted by learning rules. The learning rules verify how “experiences” of a network make use of their influence on its future behavior.

Essentially, three types of learning rules are found: supervised, reinforcement, and non-supervised or unsupervised.

6.4.1 Supervised learning

The idiom supervised has two meaning in a very common and narrow technical sense. In the narrow technical sense supervised denotes that if for a certain input the analogous output is known, the network is to learn the charting from inputs to outputs. For all supervised learning enforcements, the real output must be identified and available to the learning algorithm. The job of the network is to know how the map is drawn. The amount of the error that the network produces at the output layer controlling the varying in weights values, the larger error will largely change the weights. The divergence between the output that the network produces (the real output) and the accurate output value (the required output), represents this error. This is why this method called error-correction learning. There are some examples for supervised method such as the perceptron learning rule, the delta rule, and the famous one is back propagation. Back-propagation is very forceful and there are many types of it. The energy for applications is giant, especially because such networks can be used as a common approximates. Such learning algorithms are used in the circumstance of feed forward networks. Back-propagation requires a multi-layer network. Many different areas should be a wide environment for these networks, whenever a problem can be transformed into one of classification. A foremost example is the recognition of handwritten zip codes which can be practical to automatically cataloging mail in a post office. The word supervised has also a non-technical use. In a non-technical sense it means that the learning, for example with children, is done with the present of supervision of a teacher who supplies them with some guidance. The word used here is very vague and hard to translate into concrete neural network algorithms [32].

6.4.2 Reinforcement learning

Reinforcement learning described by the following: when the teacher merely notifies a student whether his/her answer is right or not and leaves the task of knowing why this answer is right or wrong to the student. The credit assignment or blame assignment problem defined as the trouble of attributing the error to the right cause. It is fundamental to many learning theories. For the neural network literature there is also a more technical meaning of the term (reinforcement learning). It is used to appoint learning where a particular behavior is to be reinforced. For example, the robot receives a positive support signal if the result was good, no support or a negative strengthening signal if it was bad. If the robot has controlled to raise up an object, has found its way through a table, or if it has managed to shoot the ball into the goal, it will get a positive reinforcement. Reinforcement learning is not attached to neural networks: there are many reinforcement learning algorithms in the field of machine learning in general. [32]

6.4.3 Unsupervised learning

Unsupervised learning has two categories of learning rules: Hebbian learning and competitive learning. Hebbian learning establishes connections where, if two nodes are active at the same time (or within some time window) the connection between them is strengthened. Hebbian learning has become well-liked because, though it is not very vigorous as a learning mechanism, it requires only local information and it is reasonable biologically. Hebbian learning is very much associated to point -time- dependant elasticity, where the change of the synaptic force depends on the precise timing of the pre-synaptic and post-synaptic activity of the neuron. Hebbian learning is not used in industrial applications. Competitive learning, in particular Kohonen networks is used to locate clusters in information sets. Kohonen networks also have a certain biological plausibility. In addition, they have many industrial usages [32]

6.5 Back Propagation Algorithm

There is a large number of Neural Network types have been discovered over the years. In fact, because Neural Nets are so broadly revised by Computer Scientists, Electronic Engineers, Biologists and Psychologists, they are have many distinctive names. They are called Artificial Neural Networks (ANNs), Connectionism or Connectionist Models, Multi-layer Perceptrons (MLPs) and Parallel Distributed Processing (PDP). On the other hand, despite all the different terms and different types, there are a small group of “classic” networks which are commonly used and on which many others are based. These are: Back Propagation, Hopfield Networks, Competitive Networks and networks using Spiky Neurons. There are many variations even on these topics [33].

6.6 Implementation of Back Propagation Algorithm

The most ideal Neural Net is Back Propagation network. Actually, Back Propagation is the keeping fit or educating algorithm rather than the network itself. The network used is generally of the simple type shown in figure 6.2. These are called Feed-Forward Networks or irregularly Multi-Layer Perceptrons (MLPs).

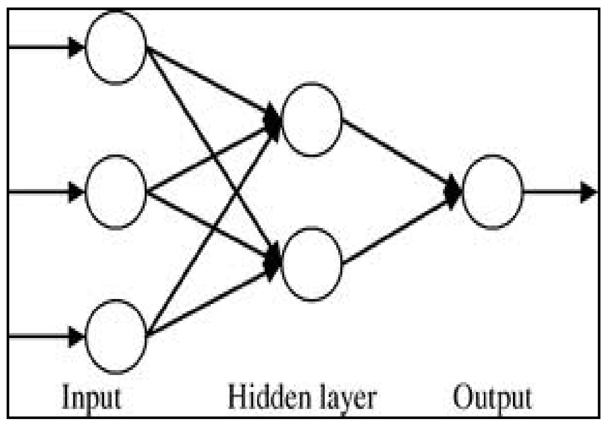

Fig. 6.2:Multi-layer back propagation neural network [34]

The learning of a Back Propagation network done by example. The algorithm takes the examples of the desired task from the network to do and it changes the network's weights so that, when training is finished, it will give the required output for a particular input. It is perfect for simple Pattern Recognition and Mapping Tasks. As said before, to train the network there is a need to give it examples of the desired object (called the Target) for a particular input as shown in Figure 6.3.

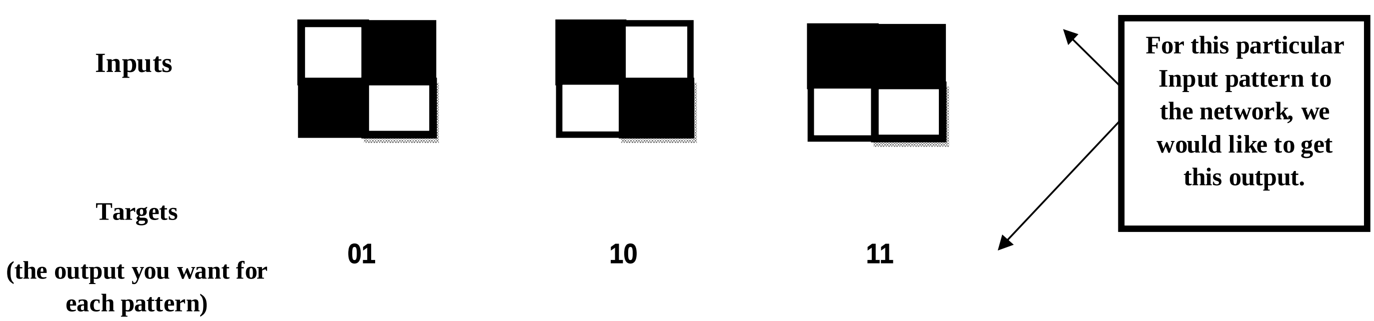

Fig. 6.3: Back propagation training Set [35]

If first prototype assumed to the network, we want the output to be 0 1 as shown in figure 6.3, (a black pixel is noted by 1 and a white by 0 as in the previous examples). The input and its corresponding target are called a Training Pair.

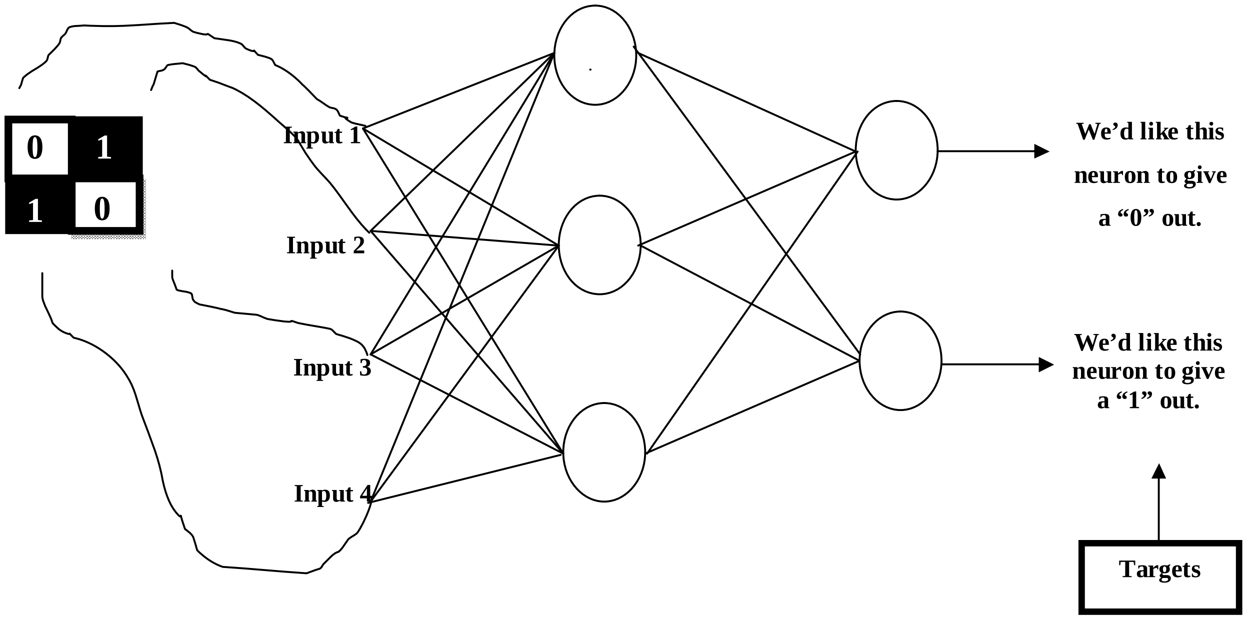

Fig. 6.4: Applying a training pair to a network [35]

From the first time that the network is trained, it will offer the desired output for any of the input prototypes. The network is first excited by setting up all its weights to be small random numbers (between –1 and +1). Next, the input pattern will be available and the output is got (this is called the forward pass). The calculation gives an output which is totally different than what is desired (the target), since all the weights are random. After that the error of each neuron can be calculated, which is essentially: Target – Actual Output. This error is then used mathematically to vary the weights in a manner that the error will be very small. In other words, the output of each neuron will converge to its target (this part is called the reverse pass). The process is recurring again and again until the error is being minimal [35].

6.7. The Activation Function

The input to the neuron is calculated as the weighted sum given by equation (6.1),

$$n = \sum_{1}^{r} Q_i W_i \tag{6.1}$$

In Figure 6.5, F is the activation function, which has a sigmoid form.

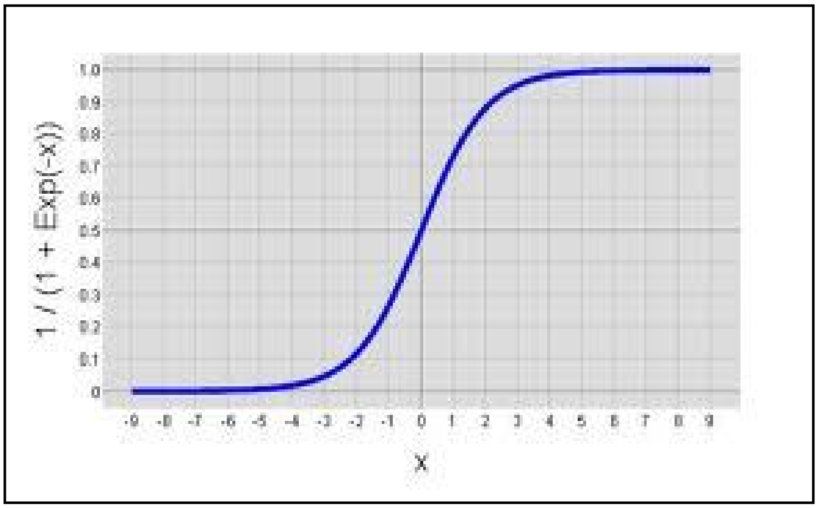

Fig. 6.5: Sigmoid function [37]

The ease of the sigmoid function derivative explains its familiarity and use as an activation function in training algorithms [36]. Figure 6.6 shows an artificial neuron. With a sigmoid activation function, the output of the neuron is given by equation (6.2) and equation (6.3),

$$Out = F(n) \tag{6.2}$$

$$F(n) = \frac{1}{(1 + e^{-n})} \tag{6.3}$$

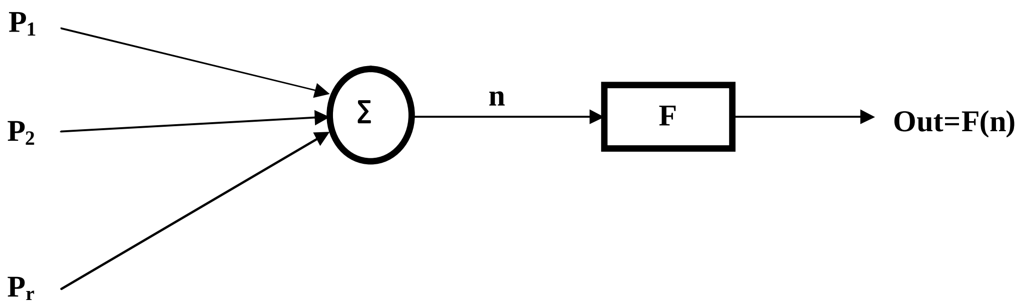

Fig. 6.6: Artificial neuron [36]

The derivative of the sigmoid function can be obtained as follows equation (6.4):

$$(df(n)/dn) = out * (1 - out) = F(n) * (1 - F(n)) \qquad (6.4)$$

6.8 Feed Forward Calculation

It is very necessary to modify the previous input data to training. The input data values within the input layer must be extent from 0 to 1. The feed forward computations have many stages can be described according to the layers. The indexes i, h and j are used for input, hidden and output respectively [36].

6.8.1 Input Layer (i)

Figure 6.7 shows a neuron in the input layer. The output of each input layer neuron is exactly equal to the modified input.

$$\text{input layer output} = O_i = I_i \qquad (6.5)$$

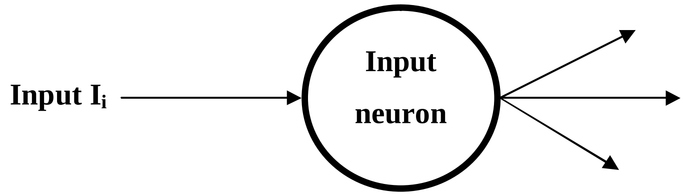

Fig. 6.7:An input layer neuron [36]

6.8.2 Hidden Layer (h)

Figure 6.8 illustrates a neuron in the hidden layer. The hidden layer neuron received a signal is equal to the sum of all the outputs of the input layer neurons multiplied by their associated connection weights, as in equation (6.6),

$$\text{hidden layer input} = I_h = \sum_i W_{hi} O_i \qquad (6.6)$$

The sigmoid function used to calculate each output of a hidden neuron. This is described in equation (6.7),

$$\text{hidden layer output} = O_h = \frac{1}{1 + \exp(-I_h)} \qquad (6.7)$$

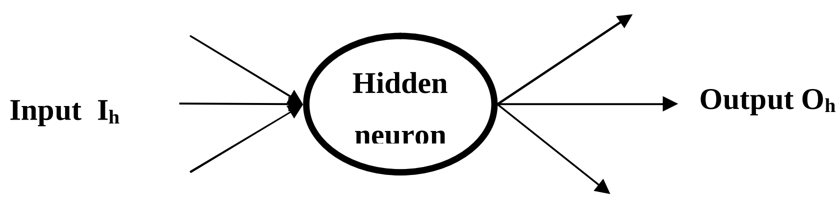

Fig. 6.8:A hidden layer neuron [36]

6.8.3 Output Layer (j)

A neuron in the output layer can be demonstrated in figure 6.9. A neuron in the output layer received signal is equal to the sum of all the outputs of the hidden layer neurons multiplied by their related connection weights plus the bias weights at each neuron, as in equation (6.8),

$$\text{output layer input} = I_j = \sum_h W_{jh} O_h \qquad (6.8)$$

Again the sigmoid function used to determines each output of an output neuron in the output layer. [36] This is described in equation (5.9),

$$\text{output layer output} = O_j = \frac{1}{1+\exp(-I_j)} \tag{6.9}$$

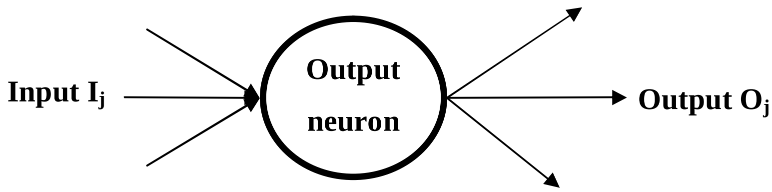

Fig. 6.9: An output layer neuron [36]

6.9 Error Back Propagation Calculation

Exclusively, during the training of the neural network, error back propagation calculations are made. The crucial elements in these calculations are the learning rate, error signal, weight adjustment, and momentum factor [38].

6.9.1 Signal Error

The feed forward output state calculation is joined with backward error propagation and weight adjustment calculations, throughout the network training, representing the network's learning. It's very important to the concept of training that a neural network can designate the network error. Error term affected strongly by the difference between the output values an output neuron is assumed to have, called the target value T_j, and the value it actually has as a result of the feed forward calculations, O_j. The error term represents an index of how well a network is training on a particular training set [38]. Equation (6.10) demonstrates the definitions for the error. The subscript p denotes what the value is for a given pattern.

$$E_P = \sum_{j=1}^{r_j} (T_{Pj}-O_{pj})^2 \tag{6.10}$$

Reduction of this error over all training patterns represents the goal of the training process. It's concluded that the output of a neuron in the output layer is a function of its input, or $O_j = f(I_j)$. The first derivative of this function, $f'(I_j)$ is a central element in error back propagation. For output layer neurons, the error signal is a quantity represented by Δj which is defined in equation (6.11),

$$\Delta j = f'(I_j)\,(T_j - O_j) = (T_j - O_j)O_j\,(1 - O_j) \qquad (6.11)$$

This error value is propagated back and suitable weight regulated. This is done by accumulating the Δ's for each neuron for the entire training set, add them, and propagating back the error based on the huge total Δ. This is called bunch (epoch) training [39].

6.9.2 Learning Rate and Momentum Factor

The learning ability of the neural network impacted with two essential parameters. The first parameter is the learning coefficient η which identifies the learning force of a neural network, and the second is the momentum factor α which identifies the speed at which the neural network learns. Those parameters can be attuned to a certain value in order to prevent the neural network from local energy minima occurrence. Both values can range between 0 and 1 [40]. Each initial value has a weight, thus a random initialization is usually performed. Weight adjustment is achieved in stages, starting at the end of the feed forward phase, and going backward to the inputs of the hidden layer [41].

6.9.3 Output Layer Weights Update

The weights that feed the output layer (W_{jh}) are updated using equation (6.12). This also involves the bias weights at the output layer neurons. Anyway, in order to stay away from the risk of the neural network falling in local minima, the momentum term can be added as in equation (6.13),

$$W_{jh}\,(new) = W_{jh}\,(old) + \eta\,\Delta_j O_h \qquad (6.12)$$

$$W_{jh}\,(new) = W_{jh}\,(old) + \eta\,\Delta_j O_h + \alpha\,[\delta W_{jh}\,(old)] \qquad (6.13)$$

where δW_{jh} (old) stands for the previous weight change [41].

6.9.4 Hidden Layer Weights Update

The identification of error term for an output layer is illustrated in equation (6.11). For the hidden layer, it is not easy to identify the error term. However, the error term for a hidden neuron is described as in equation (6.14) and, subsequently, in equation (6.15) [41].

$$\Delta_h = f'(I_h) \tag{6.14}$$

$$\Delta_h \eta = O_h (1 - O_h) \tag{6.15}$$

The weight adjustments for the linking feeding the hidden layer from the input layer are now found in a similar way to those feeding the output layer [41]. These regulations are calculated using equation (6.17),

$$W_{hi}\,(new) = W_{hi}\,(old) + \eta\, \Delta_h O_i + \alpha\, [\delta W_{hi}\,(old)] \tag{6.17}$$

The bias weights at the hidden layer neurons are updated, by the same manner, using equation (6.17).

6.10 Prediction Using Neural Network

Predicting is expecting about something that will occur, often based on information from past and from present state. There are many solutions for problem of prediction every day with various degrees of success. For example weather, harvest, energy consumption, earthquakes, and a lot of other substance needs to be predicted. In the era of technology predictable parameters of a system can be often articulated and evaluated using equations, prediction is then simply the evaluation or solution of such equations. In spite of this, practically we stand facing problems where such a description would be too difficult or not possible at all. In addition, the computation solution by this method could be very complicated, and sometimes the matter solved after the event to be predicted happened. All various approximations are available here, for example regression of the dependency of the predicted variable on other events that is then extrapolated to the future. Finding such approximation can be also difficult. This approach generally means creating the model of the predicted event.

Various levels of success satisfied with neural networks prediction. The advantage of those networks involves automatic learning of dependencies only from determined data without any need to add extra data. Using the historical data, the neural network could be trained with the hope that it will realize hidden dependencies and that it will be able to use them for predicting into future. Neural network not just a black box that is able to learn something. It is possible to predict various types of data, however in the rest of thesis focusing on predicting of time series. Time series shows the development of a value in time. The value can be influenced by also other factors than just time. Time series represents discrete history of a value and from a continuous function it can be obtained using sampling [42].

6.11 Time Series

A sequence of vectors, $x(t)$, $t = 0,1,\dots$, where t represents beyond time can characterized the time series is. It considered here only sequences of scalars for simplicity, although the methods assumed specify easily to vector series. From the theoretical side, x may be a value which varies continuously with, any physical quantity. In practice, x will be dividing into specimens to result a series of distinct data points, equally spaced in time for any given physical system. The ratio at which specimens are taken dictates the maximum determination of the model; nevertheless, it is not truth usually that the model with the highest resolution has the best predictive power; therefore superior results may be obtained by employing only every nth point in the series [42].

Analysis in neural networks has focused on forecasting future developments of the time series from values of x up to the present time. Officially this can be stated as: for any function $f: \Re N \rightarrow \Re$ such as to obtain an estimate of x at time t + d, from the N time steps back from time t, so that:

$$x(t + d) = f\,(x(t), x(t-1), K, x(t - N + 1))$$

$$x(t + d) = f\,(y(t))$$

where y(t) is the N-array vector of delayed x values. Usually d will be one, so that f will be forecasting the next value of x.

6.12 Neural Network Predictors

The encouragement of the function f using any feed forward function approaching neural network framework, such as, a standard multi layer preceptons MLP, an radial basis function RBF architecture, or a Cascade correlation model, is the essential neural network method of carrying out time series prediction [8], using a set of N-clusters as inputs and a single output as the target value of the network. The sliding window technique is the name of this method as the N-cluster input slides over the full training set. Figure 6.10 shows the basic structure.

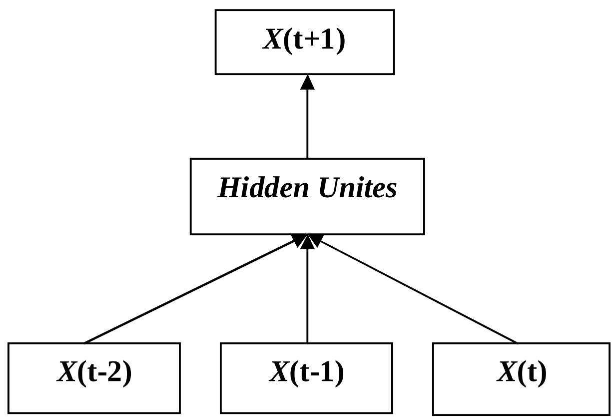

Fig.6.10 The essential method of carrying out time series prediction using a sliding window of, in this example, three time steps [42]

As noted in this technique it can be seen as an extension of auto-regressive time series modeling, in which the function f is supposed to be a linear mixture of a permanent number of previous series values. Such a bound does not apply with the non-linear neural network approach as such networks are general function approximators [42].

CHAPTER 7

DESIGN AND ANALYSIS OF GLUCOSE CONCENTRATION PREDICTION SYSTEM

7.1 The Aim

The aim of this thesis is to design a system for glucose concentration prediction of diabetic patients and analyze the performance of it against different factors. Data is taken from GlucoSim software [43] (figure 7.1), which simulates a continuous glucose monitoring (CGM) system, and is fed to a new network comprising of a Kalman filter and an artificial neural network (figure 7.2). In this approach, Kalman filter is used to denoise the CGM sensor data, and artificial neural network model acts as a predictor. Using the back propagation algorithm and two predictive windows, neural network predicts glucose values up to two hours. This helps avoid hypo/hyperglycemia, which can lead to serious complications.

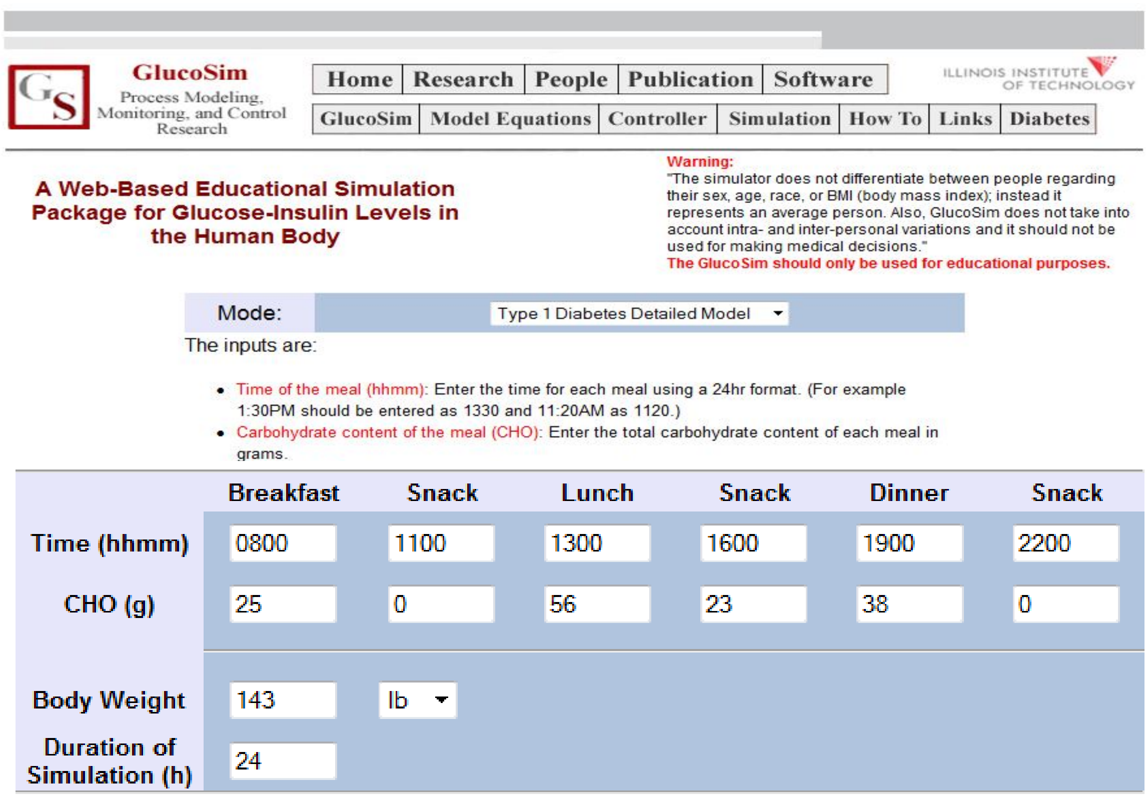

Fig 7.1:GlucoSim software [43]

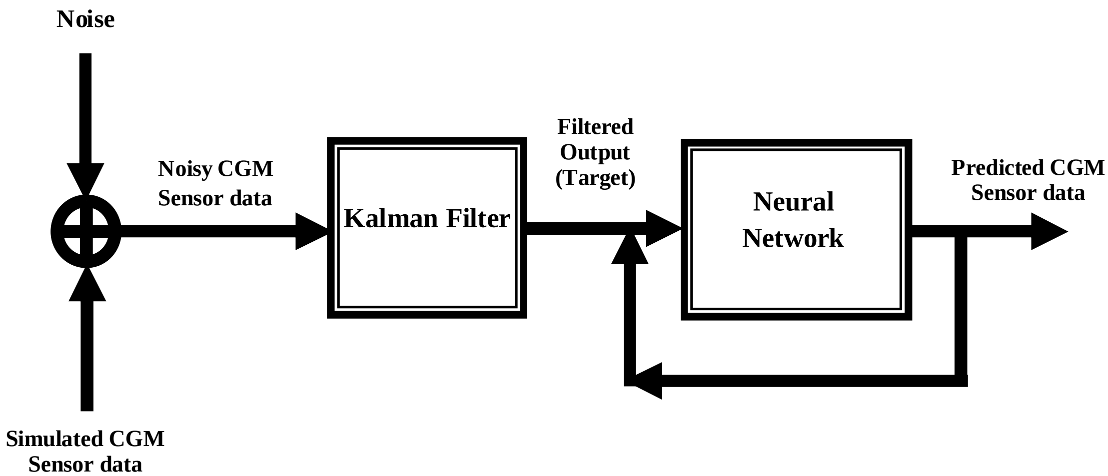

Fig. 7.2: Block diagram of suggested system

7.2. Denoising of CGM Sensor Data Using a Kalman Filter

Errors can be vital in several practical enforcements, as an example, in both open- and closed-loop control algorithms if the numerical properties of CGM sensor are known. Regrettably, layout the accuracy of CGM sensors is very difficult for both empirical and methodological causes [24]. There are many noise models will be associated with the time series of this device. It must be noted that the accuracy of CGM data affected by several sources of error. In particular, defective calibration can causes an error component usually. Other resources of error are belonged to the sensor chemistry, physics, and electronics. As a final point, the random noise component is also corrupting the CGM signal, which overcomes the original signal at high frequency. In this thesis, we will deal with the reduction of this last component. To enhance the quality of the signal and diminish the random noise component of the error, it's better to use the digital filtering techniques. In more strict terms, if following equation is considered,

$$y(t) = u(t) + v(t) \tag{7.1}$$

where $y(t)$ is the glucose concentration measured at time t, $u(t)$ is the original, mysterious, glucose concentration, and $v(t)$ is the random noise affecting it, which is assumed to be additive. The purpose of filtering is to get back $u(t)$ from $y(t)$. If the predicted spectral specifications of noise known, for example, noise is white, then (causal) low-pass filtering represents the most natural entrant to separate signal from noise in online applications. Since signal and noise spectra normally overlap, so there is a major problem in low-pass filtering in difficulty to eradicate the random noise v(t) from the determined signal y(t) without distorting the true signal u(t).In particular, delay is the result of distortion and affecting the estimated u(t) with respect to the original u(t); the more the filtering, the larger the delay. For this reason, many version of CGM data could be ineffective in practice especially for the generation of timely hypo alerts, because all having a constantly delayed, even if less noisy. The clinically point of view is thus the establishment of a conciliation between the regularity of estimated u(t) and its delay with respect to the true u(t).This thesis therefore suggests noise removal using a Kalman filter [8].

Figure 7.3 shows a patient's 12 hour glucose profile simulated by GlucoSim software. A zero-mean white Gaussian noise sequence has then been added to this reference time-series and resulting signal is filtered using a Kalman filter. Figures 7.4-7.6 show the glucose concentration of this patient after it has been filtered by a Kalman filter having three different sets of variances (Q=1 and R=1, Q=3.5 and R=2.5, Q=5 and R=5). Results show that the filter can be tuned by adjusting the parameters Q and R.

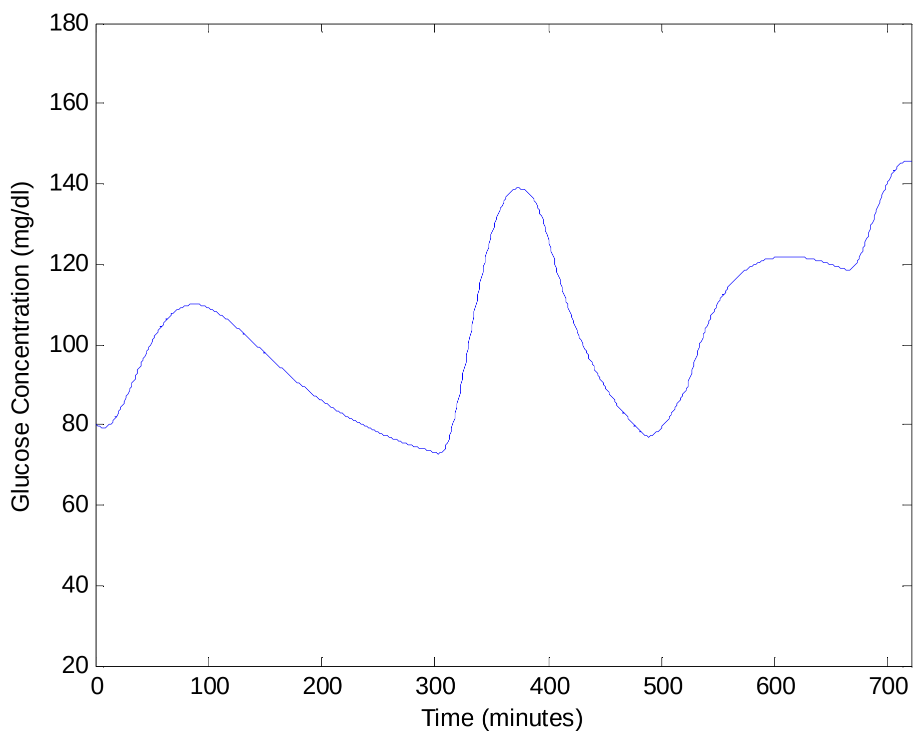

Fig. 7.3: Simulated CGM time-series

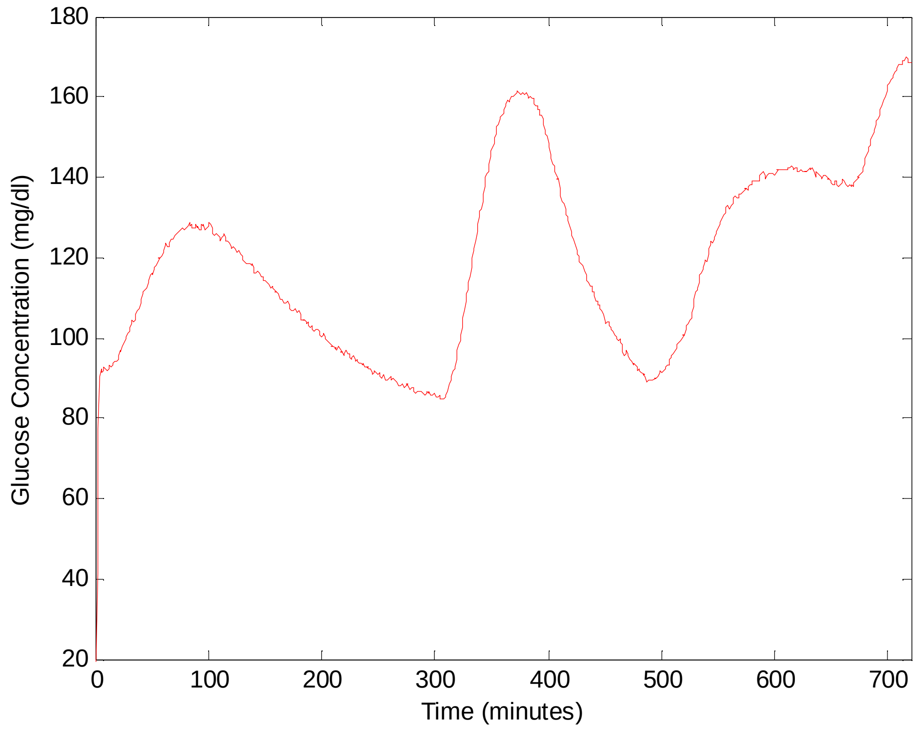

Fig. 7.4: Filtered CGM time-series (Q=1 and R=1)

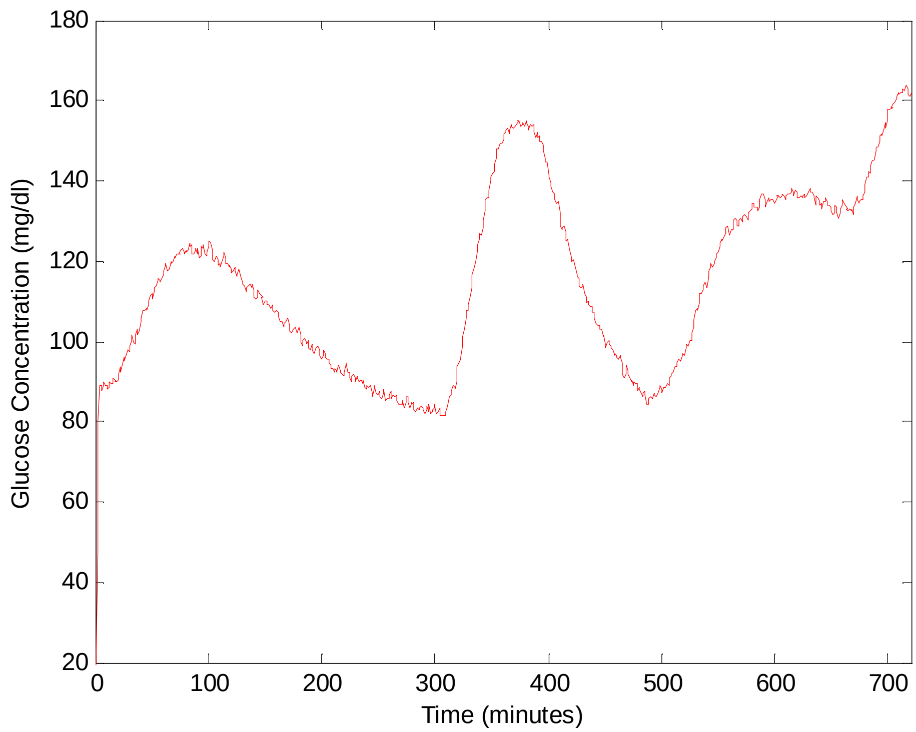

Fig. 7.5: Filtered CGM time-series (Q=3.5 and R=2.5)

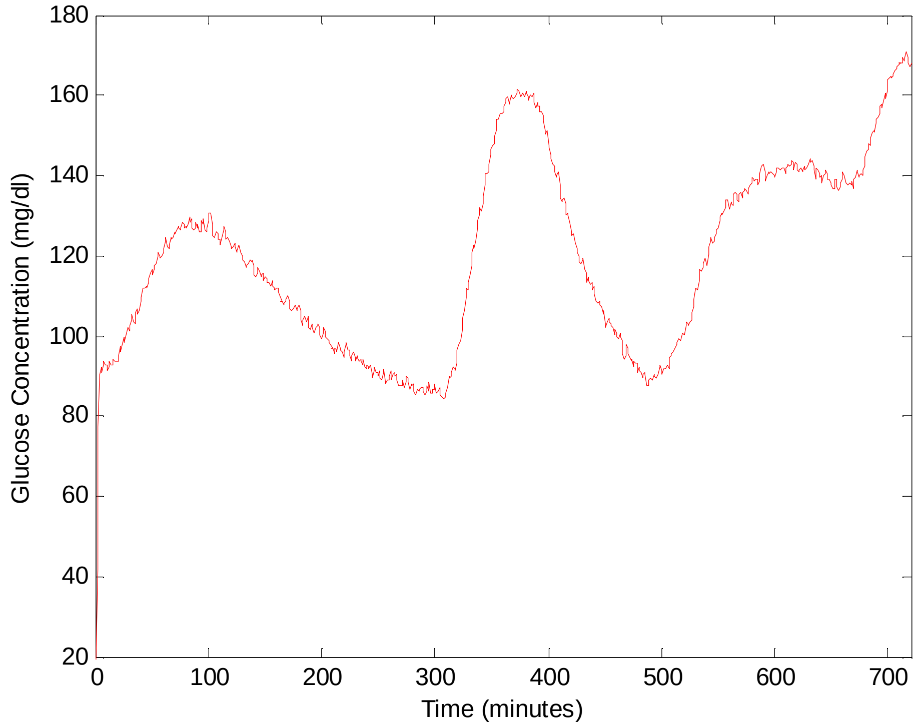

Fig. 7.6: Filtered CGM time-series (Q=5 and R=5)

Table 7.1 shows signal-to-noise ratio (SNR) values in decibels of five patients' noisy glucose concentrations before and after they have been filtered. Results are tabulated for three different sets of filter Q and R values. As expected, the SNR values have increased after filtering.

Table 7.1 Signal-to-Noise ratio (SNR) of five noisy CGM time-series before and after Kalman filtering

Patient No.	**Q=1, R=1**		**Q=3.5, R=2.5**		**Q=5, R=5**	
	SNR_{before}	**SNR_{after}**	**SNR_{before}**	**SNR_{after}**	**SNR_{before}**	**SNR_{after}**
1	3.50 dB	15.31 dB	2.83 dB	17.72 dB	3.51 dB	15.32 dB
2	3.53 dB	15.50 dB	2.81 dB	18.14 dB	3.51 dB	15.53 dB
3	3.54 dB	15.61 dB	2.82 dB	18.33 dB	3.52 dB	15.64 dB
4	3.51 dB	15.22 dB	2.80 dB	17.60 dB	3.54 dB	15.23 dB
5	3.52 dB	15.71 dB	2.81 dB	18.41 dB	3.53 dB	15.74 dB

7.3 Prediction of Glucose Concentration Using a Neural Network

Time series forecasters used by neural networks have been broadly used as: most often these are feed-forward networks which utilize a sliding window over the input sequence. Typical examples of this approach are market predictions, meteorological and network traffic forecasting [42].

The neural network developed in this thesis was time-lagged feed-forward neural network. It is categorized as multilayer perceptrons that have memory units to store previous values of data within the network. The network was trained using a method known as the back propagation of errors called Back Propagation Algorithm (BPA). The neural network was configured to stop training after 1000 epochs or if the mean squared error was less than 0.1 [11].

Figure 7.7 shows the internal structure of proposed neural network. In this network, the input is the filtered CGM data. Input layer contains input neurons equal to 720 glucose levels for each patient, hidden layer contains 10 neurons, and output layer represents the predicted CGM data that also includes 720 new glucose levels after prediction.

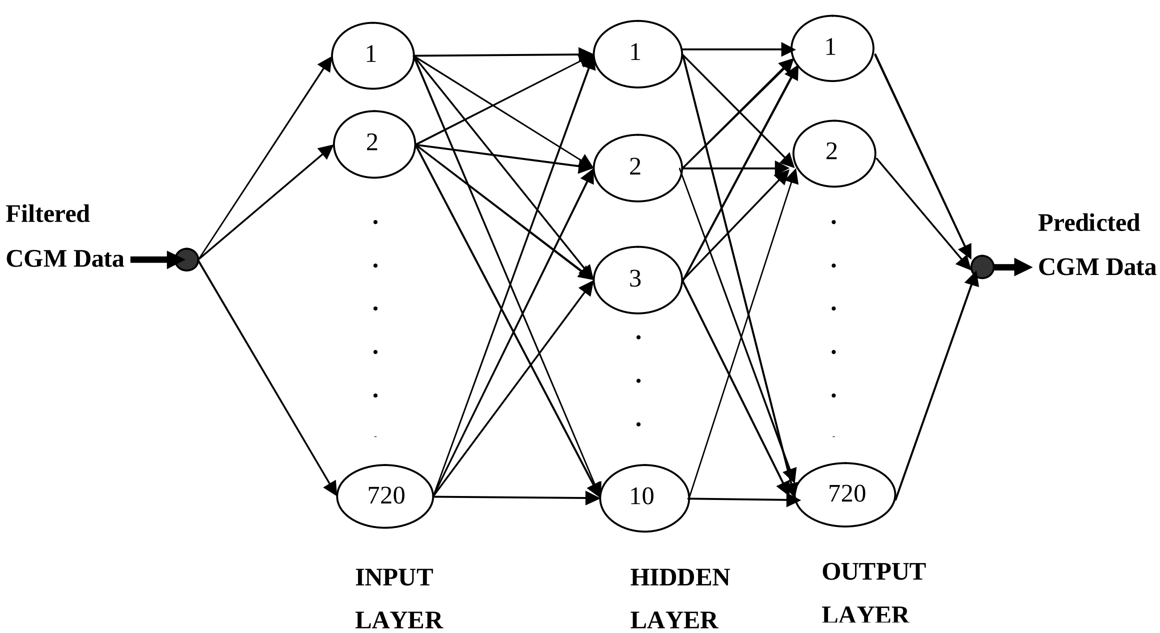

Fig. 7.7: Structure of neural network predictor

The model of neural network used here was developed with predictive windows equal to 60 and 120 minutes, which is very important for diabetes patients, specifically after meals and insulin dosages. Each glucose value was collected every minute; therefore during a 60 minute predictive window, the neural network was configured to predict 60 CGM values. During the process the dataset is divided to three groups: 70% of data is used for training, 15% for validation and 15% for testing the neural network. Figure 7.8 shows predicted data for CGM time-series of figure 7.3 (prediction length is 60 minutes).

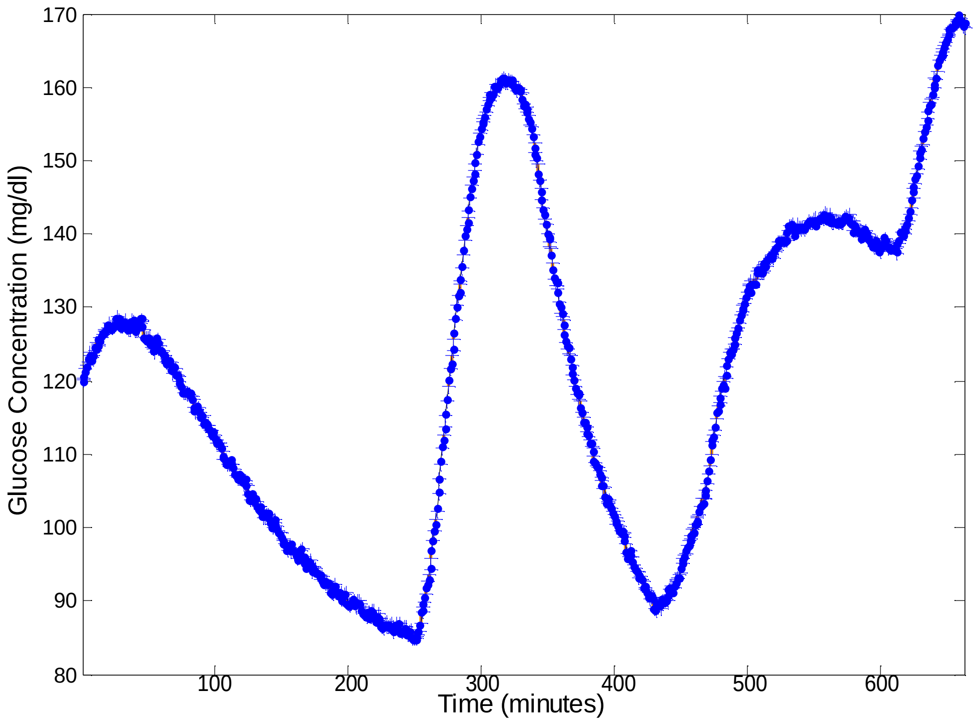

Fig. 7.8: Predicted CGM time-series

7.4 Quantitative Analyses

This section presents the quantitative analyses of the proposed system. Data used here are 25 sets of simulated blood glucose concentrations for 25 patients with various weights (25, 35, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 110, 115, 120, 125, 130, 135,140, 145, 150, 155, 160kilograms).For all analyses, simulation period is equal to 720 minutes (or 12 hours) and Kalman filter Q and R values are equal to 1.

7.4.1 Methods of Performance Analysis

Five methods are implemented to analyze the accuracy and validity of the developed system. The aim of all methods is the valuation of the mean absolute difference percent (MAD%) of the neural networks predictive abilities. Before, absolute difference percent (AD%) of each patient must be determined. Equation 7.2 is utilized for calculating the absolute difference percent (AD%) between each neural network's predicted value and the corresponding actual CGM value. In the analysis of this thesis a low MAD% values desired [11].

$$AD\%(t)=\frac{\left|NNet_{predict}(t)-CGM_{actual}(t)\right|}{CGM_{actual}(t)}\times 100\,\% \tag{7.2}$$

where AD%(t) is the calculated AD% at time t, $NNet_{predict}(t)$ is the predicted neural network glucose value at time t, $CGM_{actual}(t)$ is the actual CGM data point at time t, and N is the number of data points. Equation 7.3 is used to calculate the MAD%, which is defined as the mean of all obtained AD% values.

$$MAD\%=\frac{\sum_{i=1}^{N} AD\%(t)}{N} \tag{7.3}$$

7.4.2 Effect of Denoising

This method compares predicted CGM glucose concentrations of the suggested system with those by a system that contains neural network only. For each system, the neural network is first trained using entire dataset of 25 patients and then MAD% value of each patient is determined. Table 7.2 shows the average MAD% values for entire 25 patients. It can be seen that the proposed system has lower average MAD% values for both prediction windows. This emphasizes the importance of having a filter in such a system.

Table 7.2 Effect of denoising

System	Prediction Window of60 min.	Prediction Window of120 min.
	MAD_{avg}%	MAD_{avg}%
NN with KF	29.10	33.08
NN	55.19	58.78

7.4.3 Variation of Training Set and Prediction Window Length

This method involves analysis of the suggested system with variable training set and prediction window lengths. In this analysis, training sets using 10 to 24 patients are used for the neural network with predictive windows of 60 and 120 minutes. Performance of neural network is evaluated using diabetes data for patients who are not included in the training data. Average MAD% values of these patients are tabulated in table 7.3.

Table 7.3 Effects of varying training set and predictive window length

No. of Patients in a Training Set	Prediction Window of60 min.	Prediction Window of120 min.
	MAD_{avg}%	MAD_{avg}%
10	24.02	40.14
12	25.10	37.43
14	27.86	38.09
16	27.95	38.31
18	27.17	36.60
20	26.88	35.65
23	27.50	34.45
24	27.42	33.86

It is observed that average MAD% values of patients are relatively constant for different training sets when prediction window is smaller (the fact that increasing the training data increases the neural network performance has not been observed here). However, as the prediction window is increased, there is an increase in all average MAD% values, i.e.

accuracy of the system decreases, but higher number of patients in a training set seems to have increased the accuracy a bit.

7.4.4. Effect of Body Weight

It is known that weight can influence diabetes and diabetes can influence weight. Hence, it becomes important to control body weight fluctuations for people with diabetes. This analysis aims to check if the thesis system is suitable for blood glucose concentration prediction of various groups of patients with different body weights. Glucose concentration data of 25 patients are first used to train the neural network and then four groups of patients are used to check the system's accuracy: Group 1 contains 25 patients of average weight of 96.60 kilograms; Group 2 represents 8 patients with an average weight of 50.63 kilograms (i.e. light weight patients); Group 3 contains 9 patients with an average weight of 96.67 kilograms (i.e. medium weight patients) and Group 4 contains 8 patients with an average weight of 142.50 kilograms (i.e. heavy weight patients). Table 7.4 lists the average MAD% results of this analysis.

Table 7.4 Effect of patients' weight in a training Set

Patients Group	**Prediction Window of60 min.**	**Prediction Window of120 min.**
	MAD_{avg}%	**MAD_{avg}%**
Group 1	29.10	33.08
Group 2	25.52	39.16
Group 3	24.31	36.70
Group 4	23.89	30.08

It can be seen from the results of groups 2 to 4 that when prediction window is smaller, patients' weights almost have no effect on the accuracy of the system. However, increasing the number of patients tested has decreased the accuracy (as in group 1). This proves that the system may not be tested on high number of patients after it has been trained. On the other hand, the accuracy has decreased, as expected, for the higher prediction window. No solid conclusion can be drawn from the results here as far as the effect of body weight is concerned.

7.4.5 Effect of Exercise

Exercise is a key to life time management of diabetes. This method analyzes the impact of exercise on glucose concentration prediction. Patients in group 3 of the previous analysis (section 7.4.4) are now exercised for periods of 30, 60, and 90 minutes. The neural network is trained with the dataset of 25 patients and analysis results of patients in group 3 for 60 minute prediction window are given in table 7.5.

Table 7.5 Effect of exercise

Patient Group	Duration of Exercise	MAD_{avg}%
Group 3	30 min.	24.04
	60 min.	38.68
	90 min.	42.45

It is observed that increasing the exercise period dramatically decreases the accuracy of this system due to a temporary sharp decrease of patients' blood glucose concentrations during exercising.

7.4.6. Effects of Lifestyle of a Patient

Hypoglycemia can suddenly occur in people using insulin if too little food is eaten, if a meal is delayed or in the case of too much exercise. Hyperglycemia can occur when too much food is eaten or not enough insulin is taken. Therefore, how a patient lives his/her life is often crucial in keeping diabetes under control. It is hence the intent of this analysis to see the effects of a combination of factors such as meal intake, exercise and insulin injection on the accuracy of the performance of the suggested system. The neural network is, again, trained with the dataset of 25 patients and is tested on a single patient. Table 7.6 shows the results.

Table 7.6 Effects of a patient's lifestyle

Action	**MAD$_{avg}$%**
Increase Meal Intake	26.23
Exercise and Increase Meal Intake	44.42
Increase Insulin Dosage	25.91
Exercise and Increase Insulin Dosage	16.05

Average MAD% values are consistent here, except when the patient has exercised as well as consumed more food. In that case, the accuracy has decreased due possibly to sudden hypo/hyperglycemic events.

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

Continuous Glucose Monitoring (CGM) is very much necessary for avoidance of diabetic complications. Perfect filtering of various types of noise distributions in CGM data enables it to be employed for further processing like hypo/hyperglycemic alert generation and as control input to closed loop artificial pancreas. Conventional filtering methods are not adequate to chase the variations of physiological signal neglecting the noise effects. The proposed work comprising of an intelligent artificial neural network and a Kalman filter algorithm has been proved to be successful in denoising the CGM signal with simulated data sets.

Results prove that signal to noise ratio (SNR) of the filtered CGM signal changes with Kalman filter covariances Q and R, and therefore their values must be tuned well to give the best results. Furthermore, prediction results show that there is an increase in MAD% values whenever there is an increase in prediction window length. This indicates that it is better for patients to use small prediction windows during the measurement process to get accurate prediction results and avoid the two dangerous blood glucose levels, hyperglycemia and hypoglycemia. Results further show that the following cases should be avoided as they decrease the prediction accuracy: Testing the system (i) after extended periods of exercise, and (ii) after an excessive exercise when it is combined with increased food consumption.

8.2 Future Work

This work can be improved to also cope with SNR variations from individual to individual and from sensor to sensor, and by using other values of Kalman filter covariances Q and R in prediction analysis. For more accurate results it is better to use real diabetes data which would be more sensitive to all types of analyses.

REFERENCES

[1] Diabetes Research In Children Network (Direcnet) Study Group; Buckingham, B.A.; Kollman, C.; Beck, R.; Kalajian, A.; Fiallo-Scharer, R.; Tansey, M.J.; Fox, L.A.; Wilson, D.M.; Weinzimer, S.A.; Ruedy, K.J.; Tamborlane, W.V. Evaluation of factors affecting CGMS calibration. *Diabetes Technol. Ther*. 2006, 8, 318-325.

[2] Gianni Marchetti et al. An Improved PID Switching Control Strategy For Type 1 Diabetes. IEEE Transactions in Biomedical Engineering, March 2008, Vol.55, No.3

[3] M. Stemman et al. Sensor Calibration Models for a Non-Invasive Blood Glucose Measurement Sensor. 32nd Annual International Conference of the IEEE EMBS Buenos Aires, Argentina, August 31- September 4, 2010.

[4] Knobbe, E.J.; Buckingham, B. The extended Kalman filter for continuous glucose monitoring. *Diabetes Technol. Ther*. 2005, 7, 15-27.

[5] Palerm, C.C.; Willis, J.P.; Desemone, J.; Bequette, B.W. Hypoglycemia prediction and detection using optimal estimation. *Diabetes Technol. Ther*. 2005, 7, 3-14.

[6] Kuure-Kinsey, M.; Palerm, C.C.; Bequette, B.W. A dual-rate Kalman filter for continuous glucose monitoring. *In Proceedings of the IEEE Engineering in Medicine and Biology Society*, New York, NY, USA, August 2006; pp. 63-66.

[7] Facchinetti, A.; Sparacino, G.; Cobelli, C. An online self-tunable method to denoise CGM sensor data. *IEEE Trans. Biomed. Eng*. 2010, 57, 634-641.

[8] Andera Facchinettie, Giovanni Sparacino, and Claudio Cobelli. An Online Self-Tunable Method to Denoise CGM Sensor Data. IEEE Transactions on Biomedical Engineering, March 2010, Vol.57, No.3.

[9] Mark B. Savage et al. Development of a Non-Invasive Blood Glucose Monitor: Application of Artificial Neural Networks for Signal Processing. Worcester Polytechic Institute, Biomedical Engineering Department, Worcester, USA, 2000.

[10] V. Ashok et al. Determination of Blood Glucose Concentration by Back Propagation Neural Networks. *Indian Journal of Science and Technology*, India, August 2012, Vol.3, No.8.

[11] Scott M. Pappada, B.S, Brent D. Cameron et al, “Development of a Neural Network for Prediction of Glucose Concentration in Type 1 Diabetes Patients”, *Diabetes Science Technology*, September 2008, Vol.2, Issue 5.

[12] Andera Facchinetti et al. Online Denoising Method to Handel Intraindividual Variability of Signal–to–Noise Ratio in Continuous Glucose Monitoring. IEEE Transactions on Biomedical Engineering, September 2011, Vol.58, No.9.

[13] S. Shanthi and D. Kumar. Neural Network Based Filer for Continuous Monitoring: Online Tuning with Extended Kalman Filter Algorithm. Wseas Transactions on Information Science and Applications, India, July 2012, Issue 7, Vol.9.

[14] C. Zecchin et al. A New Neural Network Approach for Short-Term Glucose Prediction Using Continuous Glucose Monitoring Time-Series and Meal Information. 33rd Annual International Conference of IEEE EMBS, Boston, Massachusetts, USA, August 30-September 3, 2011.

[15] Panteleon, A.E.; Rebrin, K.; Steil, G.M. The role of the independent variable to glucose sensor calibration. Diabetes Technol. Ther. 2003, 5, 401-410.

[16] Keenan DB, Mastrototaro JJ, Voskanyan G, Steil GM. Delays in minimally invasive Continuous glucose monitoring devices: a review of current technology. *J Diabetes Science Technology*, September 2009; Vol.3, No.5, pp.1207–1214.

[17] Chase, J.G.; Hann, C.E.; Jackson, M.; Lin, J.; Lotz, T.; Wong, X.W.; Shaw, G.M. Integral-based filtering of continuous glucose sensor measurements for glycaemic control in critical care. *Computer Methods Programs Biomed*. 2006, 82, 238-247.

[18] Bt jaafar, S.F., and Ali, D.M., Diabetes Mellitus Forecast Using Artificial Neural Network (ANN), Faculty of Electrical Engineering, Univrsiti Teknologi Mara, 40450 Shah Alam.Asian Conference on Sensors and the International Conference on New Techniques in Paramedical and Biomedical Research Procedures, 5-7, September, 2005, Kuala lumpur, Malaysia.

[19]Department of Health and Human Services, National Institutes of Health, “ Continuous glucose monitoring”, *National Diabetes Information Clearing house*,October 2008, U.S., NIH Publication No. 09–4551.

[20] Yale NEWS, “ For Type1 Diabetics, Pump Therapy With Continuous Glucose Monitoring Is Better Than Daily Injections”, 29 June, 2010.

[21] Jean-Jacques Robert, “Continuous Monitoring of Blood Glucose”, *Hormone Research*, 2002, Vol. 57 (Suppl1):81-84, Hopital Necker-Enfants Malades, Paris, France.

[22] Ahmadi, M.M and Jullien. G.A. A Wireless-Implantable Micro system for Continuous Blood Glucose Monitoring. IEEE Transactions on Biomedical Circuits and Systems, June, 2009, Vol.3, Issue: 3.

[23] Cleveland Clinic, “Continuous Glucose Monitoring for Diabetes”, *http://my.clevelandclinic.org/devices/glucose_monitoring/hic_continuous_glucose_monitoring.aspx [6*, [Accessed on 20 Feb. 2013].

[24] University of Lowa, Lowa Research Online, “Data Acquisition Unit for Low-Noise Continuous Glucose Monitoring”, *http://ir.uiowa.edu/cgi/viewcontent.cgi?article=3214&context=etd&sei-redir=1&referer=http%3A%2F%2Fwww.google.com%2Furl%3Fsa%3Dt%26rct%3Dj%26q%3Dglucose%2Bmonitoring%2Bnoise%26source%3Dweb%26cd%3D2%26ved%3D0CDsQFjAB%26url%3Dhttp%253A%252F%252Fir.uiowa.edu%252Fcgi%252Fviewcontent.cgi%253Farticle%253D3214%2526context%253Detd%26ei%3DTdcOUfbbJsf4sgarrYCQDQ%26usg%3DAFQjCNHCeosMTtfaKz0lg_TfP2ZH73YCtg#search=%22glucose%20monitoring%20noise%22*, [Accessed on 12 Feb. 2013].

[25]Jay S.Skyler ,”Continuous GlucoseMonitoring : An Overview of itsDevelopment”, Diabetes Technology &Therapeutics, vol. 11, sup.1,2009.

[26] B.Kovatchev, L.A.Gonder-Frederick, D.J.Cox, W.L.Clarke, “Evaluating the Accuracy of Continuous Glucose Monitoring Sensors: continuous glucose-error grid analysis illustrated byTheraSenseFreestyle Navigator data”,Diabetes Care, 2004 Aug;27(8); 1922-1928.

[27] Giovanni Sparacino, Anderwa Fachinetti, and Claudio Cobelli, "Smart" Continuous Glucose Monitoring Sensors: On-Line Signal Processing Issues, Department of Information Technology Engineering, University of Padova, *Sensors*, 2010.

[28] Mohinder S. Grewal, Angus P. Andrews, "Kalman Filtering: Theory and Practice Using MATLAB", John Wiely & Sons, Second Edition, 2001, ISBNs: 0-471-39254-5 (Hardback); 0-471-26638-8 (Electronic).

[29] F. Haugen, "Kalman Filter", Chapter 8,*Kompendium for Kyb. 2 ved*, Høgskolen I, Oslo

[30] MathWorks, Documentation Center, "Kalman Filtering",

http://www.mathworks.com/help/control/ug/kalman-filtering.html, [Accessed on 10 Apr. 2013].

[31] Haward Demuth and Mark Beal. Neural Network Toolbox For Use with MATLAB; Neural Networks Users Guide, *The Mathworks Inc*., 1992-2002, Version 4

[32] Rolf Pfeifer, Dana Damian, and Rudolf Fuchslin, "Neural Networks", University of Zurich, 2010.

[33] Neuro ai, Artificial Intelligence Technologies, "Introduction to Back Propagation Algorithm", *http://www.learnartificialneuralnetworks.com/#Intro*, [Accessed on 30 Mar. 2013].

[34] A. Lin Cao, Qing Jun Zhu, Sheng Tao Zhang, Bao Rong Hou, (2010) "BP neural network predictive model for stray current density of a buried metallic pipeline", Anti-Corrosion Methods and Materials, Emerald journal ,Vol. 57 Iss: 5, pp.234 – 237

[35]The Back Propagation Algorithm, Chapter 3, *http://www4.rgu.ac.uk/files/chapter3%20-%20bp.pdf*, [Accessed on 2 Mar. 2013]

[36] *http://www.andrew.cmu.edu/user/jterlesk/robotics/writing/writing.html*

[37] Colin Fahey, "Neural Network with Learning by Backward Error Propagation", http://www.colinfahey.com/neural_network_with_back_propagation_learning/, [Accessed on 10 Apr. 2013].

[38] Neural Networks Home, "Solutions of Neural Networks Problems",

http://www.cs.nps.navy.mil/people/faculty/rowe/alvesthesis.html, [Accessed on 18 Mar. 2013].

[39] University of Wisconsin-Madison, Department of Computer Science, “Using Neural Networks in Solving the Engineering Problems”,

http://www.cs.wisc.edu/~dyer/cs540/hw/hw3/hw3.html, [Accessed on 23 Mar. 2013].

[40] “Neural Networks Exampls”, *http://users.rowan.edu/~shreek/spring00/ann/lab3.html*, [Accessed on 23 Mar. 2013].

[41] Adnan Khashman, “Back Propagation Learning Algorithm in Neural Networks”, Handout for MSC. Courses EE508/COM507, Faculty of Engineering, Near East University, Nicosia.

[42] R.J. Frank, N. Davey, and S.P. Hunt, “Time Series Prediction and Neural Networks”, Department of computer science, University of Hertfordshire, Hatfield, UK.

[43] Meriyan Oruklu, Baris U. Agar, F. Ceylan Erzen and Ali Cinar. “A Web-Based Educational Simulation Package for Glucose-InsulinLevels in the Human Body”, Department of Chemical and Biological Engineering, Illinois Institute of Technology

http://simulator.iit.edu/web/glucosim/index.html, [Accessed on 20 Feb. 2013].

APPENDIX

Source Code of the System within the Thesis

Software: MATLAB R2012a

```
A = [1.1269   -0.4940    0.1129;
    1.0000         0         0;
         0    1.0000         0];


B = [-0.3832; 0.5919; 0.5191];


C = [-1 0 0];


D = 0;
u=pat2(:,2)
t = [1:720]';
n = length(t)
randn('seed',0);
Q=1;
R=1;
w = sqrt(Q)*randn(n,1);
v = sqrt(R)*randn(n,1);
sys = ss(A,B,C,D,-1);
noise=u+w;
[kalmf,L,P,M] = kalman(sys,Q,R);



 y = lsim(kalmf(1,:),noise);   % w = process noise  %y = lsim(sys,noise);
```

```
 yv = y + v;          % v = meas. noise
 P=B*Q*B';         % Initial error covariance
 x=zeros(3,1);     % Initial condition on the state
 ye = zeros(length(t),1);
 ycov = zeros(length(t),1);
 errcov = zeros(length(t),1);

 for i=1:length(t)
 % Measurement update
 Mn = P*C'/(C*P*C'+R);
 x = x + Mn*(yv(i)-C*x);  % x[n|n]
 P = (eye(3)-Mn*C)*P;     % P[n|n]
 ye(i) = C*x;
 errcov(i) = C*P*C';

% Time update
x = A*x + B*u(i);        % x[n+1|n]
P = A*P*A' + B*Q*B';     % P[n+1|n]
end
plot(t,u);
figure(2)
plot(yv);
xlabel('Time in Minuets')
ylabel('Glucose Levels')
axis([0 720 20 180])
hold;
figure(3)
plot(ye,'r');
xlabel('Time in Minuts')
ylabel('Glucose Concentration (mg/dl)')
```

```
axis([0 720 20 180])
hold;
figure(4)
plot(y,'g');
%plot the original signal over top the
snr_before = mean( u .^ 2 ) / mean( noise .^ 2 );
snr_before_db = 10 * log10( snr_before ) % in dB
R = u - ye;
snr_after = mean( ye .^ 2 ) / mean( R .^ 2 );
snr_after_db = 10 * log10( snr_after )
 % Solve an Auto regression Time-Series Problem with a NAR Neural Network
% Script generated by NTSTOOL
% This script assumes this variable is defined:
%close_data - feedback time series.

targetSeries = tonndata(ye,false,false);

% Create a Nonlinear Autoregressive Network
feedbackDelays = 1:4:60;
hiddenLayerSize = 10;
net = narnet(feedbackDelays,hiddenLayerSize);

% Choose Feedback Pre/Post-Processing Functions
% Settings for feedback input are automatically applied to feedback output
% For a list of all processing functions type: help nnprocess
net.inputs{1}.processFcns = {'removeconstantrows','mapminmax'};

% Prepare the Data for Training and Simulation
% The function PREPARETS prepares timeseries data for a particular network,
```

```
% shifting time by the minimum amount to fill input states and layer
states.

% Using PREPARETS allows you to keep your original time series data
unchanged, while

% easily customizing it for networks with differing numbers of delays, with

% open loop or closed loop feedback modes.

[inputs,inputStates,layerStates,targets] =
preparets(net,{},{},targetSeries);


% Setup Division of Data for Training, Validation, Testing

% For a list of all data division functions type: help nndivide

net.divideFcn = 'dividerand';  % Divide data randomly

net.divideMode = 'time';  % Divide up every value

net.divideParam.trainRatio = 70/100;

net.divideParam.valRatio = 15/100;

net.divideParam.testRatio = 15/100;


% Choose a Training Function

% For a list of all training functions type: help nntrain

net.trainFcn = 'trainlm';  % Levenberg-Marquardt


% Choose a Performance Function

% For a list of all performance functions type: help nnperformance

net.performFcn = 'mse';  % Mean squared error


% Choose Plot Functions

% For a list of all plot functions type: help nnplot

net.plotFcns = {'plotperform','plottrainstate','plotresponse', ...

  'ploterrcorr', 'plotinerrcorr'};
```

```
% Train the Network
[net,tr] = train(net,inputs,targets,inputStates,layerStates);

% Test the Network
outputs = net(inputs,inputStates,layerStates);
errors = gsubtract(targets,outputs);
performance = perform(net,targets,outputs)

% Recalculate Training, Validation and Test Performance
trainTargets = gmultiply(targets,tr.trainMask);
valTargets = gmultiply(targets,tr.valMask);
testTargets = gmultiply(targets,tr.testMask);
trainPerformance = perform(net,trainTargets,outputs)
valPerformance = perform(net,valTargets,outputs)
testPerformance = perform(net,testTargets,outputs)

% View the Network
view(net)

% Plots
% Uncomment these lines to enable various plots.
figure, plotperform(tr)
figure, plottrainstate(tr)
figure, plotresponse(targets,outputs)

% Closed Loop Network
% Use this network to do multi-step prediction.
% The function CLOSELOOP replaces the feedback input with a direct
```

```
% connection from the output layer.

netc = closeloop(net);

[xc,xic,aic,tc] = preparets(netc,{},{},targetSeries);

yc = netc(xc,xic,aic);

perfc = perform(net,tc,yc)



% Early Prediction Network

% For some applications it helps to get the prediction a timestep early.

% The original network returns predicted y(t+1) at the same time it is
given y(t+1).

% For some applications such as decision making, it would help to have
predicted

% y(t+1) once y(t) is available, but before the actual y(t+1) occurs.

% The network can be made to return its output a timestep early by removing
one delay

% so that its minimal tap delay is now 0 instead of 1.  The new network
returns the

% same outputs as the original network, but outputs are shifted left one
timestep.

nets = removedelay(net);

[xs,xis,ais,ts] = preparets(nets,{},{},targetSeries);

ys = nets(xs,xis,ais);

closedLoopPerformance = perform(net,tc,yc)

for t=1:660

m=cell2mat(yc)

AD=abs(m(t)-u(t))./u(t)

MAD=mean(AD);

end
```