Universidad de Costa Rica Facultad de Ingeniería Escuela de Ingeniería Eléctrica

Implementación de un sistema para prueba de control de un páncreas artificial

Por:

David Ricardo Porras Cordero

Ciudad Universitaria "Rodrigo Facio", Costa Rica

Diciembre de 2021

Implementación de un sistema para prueba de control de un páncreas artificial

Por:

David Ricardo Porras Cordero

Sometida a la Escuela de Ingeniería Eléctrica de la Facultad de Ingeniería de la Universidad de Costa Rica como requisito parcial para optar por el grado de:

LICENCIATURA EN INGENIERÍA ELÉCTRICA Aprobada por el Tribunal:

> Ing. Aramis Perez Mora, Ph.D. Representante del Director de la Escuela

> > Ing. José David Rojas, Ph.D. Director, Comité asesor

Ing. Leonardo Marín Paniagua, Ph.D. Ing. Fabián Abarca Calderón, M.Sc. Miembro, Comité asesor Miembro, Comité asesor

> Diego Dumani Jarquin, Ph.D. Miembro, Tribunal

Dedicatoria

Dedicado a mi familia por su apoyo incondicional en cada etapa de mis estudios y al profesor José David Rojas por su colaboración durante la realización del proyecto.

Índice general

1.	\mathbf{Intr}	oducción	1
	1.1.	Antecedentes	1
	1.2.	Justificación	2
	1.3.	Planteamiento del Problema	3
	1.4.	Objetivos	4
		1.4.1. Objetivo General	4
		1.4.2. Objetivos Específicos	4
	1.5.	Alcance	5
	1.6.	Metodología	5
2.	Mar	rco Teórico	7
	2.1. Hardware in the Loop		
	2.2.	Sistema Glucosa-Insulina	8
	2.3.	Modelos del sistema glucosa-insulina	9
2.4. Modelo Mínimo de Bergman		Modelo Mínimo de Bergman	10
		Modelo de Hovorka	13
	2.6.	Simulador de de Dalla Man y Cobelli	16
		2.6.1. Modelo de Dinámica de Glucosa	17
		2.6.2. Modelo de Dinámica de Insulina	22
		2.6.3. Modelo de Absorción de Comida	24
		2.6.4. Adaptación del Simulador para Pacientes Tipo 1	26
	2.7.	Sensor de glucosa	28
	2.8.	Bomba de insulina	30
	2.9.	Algoritmos de Control	31

3.	. Implementación del Modelo 3		35	
	3.1.	Modelo del Sistema Glucosa-Insulina	35	
	3.2.	3.2. Plataforma de Simulación		
	3.3.	Efectos del Sensor de Glucosa y la Bomba de Insulina	36	
	3.4.	Implementación en MATLAB	37	
	3.5.	Controlador	42	
		3.5.1. Identificación del Sistema	43	
		3.5.2. Síntesis del Controlador	44	
	3.6.	Implementación en MyRIO	47	
		3.6.1. Implementación del Modelo de Cobelli y Dalla Man	47	
		3.6.2. Programa del MyRIO	52	
	3.7.	Controlador en Arduino	53	
	3.8.	Conexión del Sistema	55	
4.	\mathbf{Res}	ultados	57	
	4.1.	Control del PID en el Arduino	57	
	4.2.	Comparación entre la implementación del MyRIO y la simulación en		
		Simulink	59	
5.	Con	clusiones y recomendaciones	63	
	5.1.	Conclusiones	63	
	5.2.	Recomendaciones	64	
A	pénd	ices	70	
Α.	Cál	culo de Parámetros en Matlab	71	
в.	Cód	igo del Controlador	73	

Índice de figuras

1.1.	Concepto del Proyecto	4
2.1.	Hardware in the Loop ideal adaptado de [29]	8
2.2.	Diagrama del sistema Glucosa-Insulina (elaboración propia)	9
2.3.	Diagrama de la dinámica de la glucosa en el modelo mínimo de Bergman	
	adaptado de $[1]$	11
2.4.	Diagrama de la dinámica de la insulina en el modelo mínimo de Bergman	
	adaptado de $[1]$	12
2.5.	Diagrama del modelo de Hovorka adaptado de [15]	13
2.6.	Diagrama de las principales partes del simulador adaptado de $[9]$	16
2.7.	Diagrama del modelo de glucosa adaptado de [9]	17
2.8.	Diagrama del modelo de insulina $[9]$	22
2.9.	Diagrama del funcionamiento del sensor adaptado de [18]	29
2.10.	. Diagrama del funcionamiento de la bomba adaptado de $[17]$	31
2.11.	. Diagrama del control PID (elaboración propia)	32
2.12.	Diagrama del control MPC adaptado de [4]	33
3.1.	Implementación del subsistema de utilización de glucosa en Simulink.	37
3.2.	Implementación del subsistema de glucosa en Simulink	38
3.3.	Implementación del subsistema renal en Simulink	38
3.4.	Implementación del subsistema de insulina en Simulink	38
3.5.	Implementación de la aparición de insulina	39
3.6.	Implementación del subsistema de digestivo en Simulink	39
3.7.	Calculo del parámetro K_{empt}	40
3.8.	PID implementado para probar el modelo.	40

3.9.	Implementación del retraso del sensor en Simulink	41
3.10.	Resultado de referencia de la simulación del modelo, tomado de $\left[10\right]$	41
3.11.	Resultado de la Simulacion	42
3.12.	Cambio de la salida ante una entrada en forma de un escalón	43
3.13.	Cambio de la salida ante una entrada en forma de cadena de escalones	43
3.14.	Comparación entre tres controladores en control del valor deseado $\ . \ .$	45
3.15.	Comparación entre tres controladores en control regulatorio	46
3.16.	Comparación entre tres controladores en control regulatorio	46
3.17.	Diagrama del subsistema de la dinámica de la glucosa	48
3.18.	Diagrama del subsistema del sensor	48
3.19.	Diagrama para calculo de variable U_{id}	48
3.20.	Diagram del subsistema de la dinámica de la insulina	49
3.21.	Diagrama del subsistema de infusión de insulina	49
3.22.	Diagrama del subsistema digestivo.	50
3.23.	Diagrama para calculo de variable K_{empt}	50
3.24.	Diagrama para calculo de variables α y β	51
3.25.	Diagrama del modelo de Cobelli y DallaMan en LabView	51
3.26.	Comparación Simulink y LabView.	52
3.27.	Diagrama del programa principal en LabView para MyRio	52
3.28.	Diagrama de conexiones entre el Arduino y el MyRIO	56
3.29.	Imagen del sistema implementado	56
4.1.	Panel principal del programa del MyRIO	58
4.2.	Sistema ante una perturbación partiendo de $130mg/dL$ de glucosa inicial.	58
4.3.	Sistema ante dos perturbación partiendo de $180mg/dL$ de glucosa inicial.	59
4.4.	Tiempo en alcanzar el valor deseado partiendo de $180mg/dL$ de glucosa	
	inicial	60
4.5.	Sistema ante tres perturbaciones partiendo de 130 mg/dL de glucosa	
	inicial	61

Índice de cuadros

2.1.	Parámetros del subsistema de glucosa [9]	18
2.2.	Parámetros de la producción endógena [9]	20
2.3.	Parámetros de la utilización [9]	22
2.4.	Parámetros del expulsión renal [9]	22
2.5.	Parámetros del subsistema de insulina [9]	23
2.6.	Parámetros tasa de aparición de glucosa [9]	25
2.7.	Parámetros tasa de aparición de insulina [10]	26
3.1.	Parámetros de los controladores	44
3.2.	IAE en las pruebas realizadas	47

Resumen

El presente trabajo expone una implementación de una simulación del sistema que tiene que controlar un dispositivo de páncreas artificial, el cual es uno de los tratamientos para diabetes tipo 1 de mayor avance en los últimos años.

La diabetes es una enfermedad crónica provocada cuando el cuerpo no produce suficiente insulina para controlar los niveles de glucosa en la sangre, de los dos tipos que existen, la más grave es la llamada tipo 1, en la cual el cuerpo no produce insulina y depende totalmente de una fuente externa de esta.

El páncreas artificial consiste generalmente en una bomba de insulina que inyecta esta hormona en el plasma sanguíneo a una tasa indicada por un algoritmo de control que a su vez la determina mediante la medición de la glucosa en plasma con un sensor continuo de glucosa.

Para realizar la simulación se estudiaron modelos matemáticos que caracterizaran mediante ecuaciones diferenciales el sistema glucosa-insulina, y los efectos más generales sobre la medición de glucosa por parte del sensor y el ingreso de insulina por la bomba.

Al finalizar el trabajo se logró la implementación de un modelo que simulara de manera realista el sistema glucosa insulina y se tomaron en cuenta los efectos del sensor de glucosa y la bomba de insulina. Se implemento la simulación en un dispositivo de la empresa National Instruments llamado MyRIO que sirvió para simular el sistema en tiempo real y comunicarse mediante el protocolo de comunicación SPI con un Arduino que internamente tuvo programado un PID simple para probar el sistema en lazo cerrado.

Capítulo 1

Introducción

1.1. Antecedentes

El estudio de la diabetes data desde los antiguos egipcios, en manuscritos de alrededor del 1500 A.C. se describe una enfermedad caracterizada por grandes cantidades de orina. En el siglo primero de nuestra era el médico Areteo de Capadocia acuñó el término diabetes (del griego "sifón"). En el siglo XIX se empieza a determinar que el páncreas tiene que ver con la enfermedad. Posteriormente, en el siglo XX, se dieron los estudios que determinaron las causas de la diabetes, su relación con la dieta y su tratamiento [22].

Se han identificado dos tipos de diabetes, la diabetes tipo 1, que se da por una deficiencia a la hora de generar la insulina, y la diabetes tipo 2 que se produce cuando el cuerpo no produce suficiente insulina para la demanda de azúcar. La diabetes tipo 1 es incurable y muestra sus efectos desde la juventud. El otro tipo generalmente se da por condiciones de salud poco ideales como sobrepeso y se manifiesta usualmente cuando el paciente ya está en edad avanzada [3].

El tratamiento para la diabetes ha estado en un constante mejoramiento desde mediados del siglo XX. Actualmente se trata mediante inyecciones de insulina que se aplican periódicamente según la necesidad del paciente. La diabetes tipo 2 eventualmente puede llegar a desaparecer con una dieta rígida [3].

Dependiendo de la severidad y el tipo de diabetes que se padezca, en ocasiones se opta

por una bomba de insulina. La bomba de insulina por lo general solo se utiliza en pacientes del tipo 1; la tecnología que actualmente se usa en la gran mayoría de los casos es manual, el paciente o su cuidador debe accionar la bomba si se detecta una fluctuación en el nivel ideal de glucosa [3].

Desde la década de 1960 se ha intentado automatizar la bomba de glucosa, esto ha desembocado en un dispositivo llamado páncreas artificial, este lo que hace es monitorear los niveles de glucosa y suministrar insulina según sea necesario [20].

El páncreas artificial que actualmente se comercializa no es totalmente automatizado, el paciente debe suministrarle información como por ejemplo el momento en que se come y cuántas calorías se ingiere, o cuándo se realiza actividad física, ya que estas dos actividades generan fluctuaciones considerables en los niveles de glucosa. Esto lleva a que en estos momentos existan diversos esfuerzos por automatizar totalmente el páncreas artificial [20].

Debido a lo poco práctico de realizar pruebas clínicas cada vez que se quiere mejorar el algoritmo de control del dispositivo, estos se prueban utilizando un modelo del sistema describe la interacción entre la glucosa y la insulina [4].

Los modelos que describen este sistema se han trabajado desde los años 60, uno de los primeros y más utilizados es el modelo mínimo de Bergman ya que es uno de los más simples y al mismo tiempo conveniente para la síntesis de controladores [2].

Aunque el modelo de Bergman es funcional, en la actualidad se han trabajado modelos que realizan una descripción más realista de las interacciones bioquímicas que suceden en el consumo de glucosa. La mayoría de las investigaciones de controladores para páncreas artificial se han probado utilizando un modelo simulado in silico [12].

1.2. Justificación

De acuerdo con la Organización Mundial de la Salud, en el 2014 un estimado de 422 millones de adultos padecían diabetes, lo que supone un aumento del doble de pacientes con respecto a 1980 [28].

El aumento de pacientes está asociado al aumento de factores de riesgo como el sedentarismo. La mayor tasa de crecimiento se observa en países en vías de desarrollo. Se estima además, que la diabetes causó 1,5 millones de muertes en el 2014 y 2,2 millones más fueron causadas por niveles anormales de azúcar en la sangre [28].

Como no hay datos claros del tipo de diabetes en cada paciente, no se tiene el dato de cuántos pacientes padecen de qué tipo de diabetes, sin embargo, sí se sabe con certeza que la gran mayoría de pacientes tienen del tipo 2 [28].

El páncreas artificial es de los tratamientos para diabetes tipo 1 que más avance ha tenido en los últimos años, y al mismo tiempo presenta muchas oportunidades de mejora que se pueden explotar [20].

El proyecto propuesto a continuación presenta una oportunidad para el estudio de este tratamiento. Hacer una simulación del sistema glucosa-insulina, incluyendo los efectos del sensor de glucosa y la bomba de insulina, para probar un algoritmo de control, esto podría allanar el camino para una continuación en la búsqueda de mejores algoritmos de control para este sistema.

En la Escuela de Ingeniería Eléctrica ya se planteó una simulación de un páncreas artificial [25], sin embargo, esta se puede mejorar en aspectos como la comunicación entre el sensor y la bomba, no se logró simular el efecto que estos tienen en un sistema real; por otro lado, el modelo utilizado se puede modificar, ya que existen otros más complejos con los que se puede generar una simulación más realista.

Se espera además que el resultado del proyecto haga que el interés por el tema se incremente y se puedan realizar mejoras subsecuentes al mismo con proyectos derivados.

1.3. Planteamiento del Problema

El problema por solucionar consiste en poder simular de manera realista la manera en que trabajaría un páncreas artificial conectado a un paciente, para esto se debe contar con un sistema que internamente, mediante ecuaciones diferenciales, modele la interacción glucosa-insulina en una persona, así como la interacción de la bomba de insulina y el sensor de glucosa, de manera que funcione como el diagrama de la figura 1.1.

Para lograr esto, se debe realizar una revisión de los modelos realizados mediante

Introducci'on



Figura 1.1: Concepto del Proyecto

pruebas clínicas, definir qué parámetros se utilizarán para seleccionar el mejor modelo e implementar la solución del modelo seleccionado.

Con el modelo ya implementado, se realizará una síntesis de un controlador que haga las funciones como lo haría un páncreas artificial, para hacer pruebas de la validez de la simulación.

1.4. Objetivos

1.4.1. Objetivo General

Implementar un sistema de prueba de control que simule de manera realista el sistema que debe controlar un páncreas artificial.

1.4.2. Objetivos Específicos

- Investigar modelos matemáticos que se hayan propuesto para simular el sistema glucosa insulina.
- Analizar los modelos encontrados y determinar cuál representa el sistema de manera más realista y que al mismo tiempo sea factible de implementar.

- Implementar el modelo glucosa-insulina en un medio computacional de manera que se pueda generar la simulación tipo "hardware in the loop".
- Averiguar la forma en la que la comunicación entre la bomba de insulina y el sensor de glucosa se da en el sistema de páncreas artificial y tomarlo en cuenta en la simulación.
- Probar que la simulación implementada tenga resultados acordes con la respuesta del modelo seleccionado y asegurar una correcta reproducción del sistema.
- Sintetizar un controlador básico para probar el funcionamiento del sistema simulado.

1.5. Alcance

El enfoque del proyecto está orientado al estudio e implementación de un modelo del sistema glucosa-insulina, se implementará un modelo ya desarrollado para generar una simulación. Se realizará además un controlador para el sistema, con el objetivo de probar el mismo en un lazo cerrado.

1.6. Metodología

Se plantea una metodología con una primera parte basada en revisión bibliográfica de artículos científicos, libros y tesis, para ahondar en los modelos matemáticos que describen el sistema, posteriormente, en una segunda etapa se tiene la implementación del modelo en la plataforma computacional y la realización de pruebas, siguiendo los siguientes puntos:

- Revisión y lectura de bibliografía sobre los modelos que describen el sistema determinando sus respectivas ventajas y desventajas.
- Programación del modelo escogido en MATLAB para verificar su comportamiento mediante simulaciones.
- Estudio de las plataformas disponibles para implementar el sistema y revisar sus características para escoger la más conveniente.
- Implementación del modelo escogido en la plataforma que se consideró mejor.

- Síntesis de controlador capaz de regular el sistema.
- Prueba del sistema en lazo cerrado para determinar su correcto funcionamiento.

Capítulo 2

Marco Teórico

2.1. Hardware in the Loop

El concepto de "Hardware in the Loop" (HiL) consiste en simular un mecanismo mediante una representación matemática fiable de un sistema específico, que trabaje en tiempo real sobre un simulador digital, los datos obtenidos de esta simulación son convertidos en las señales necesarias para probar algún dispositivo de hardware que se quiera evaluar [6].

La principal ventaja que presenta esta técnica es que se pueden realizar pruebas a un sistema de control sin intervenirlo en la realidad, ya sea porque podría resultar peligroso o, simplemente porque no se cuenta con el sistema real en el momento para probarlo, en otras palabras, se puede probar un sistema antes de implementarlo físicamente.

Las ventajas del HiL se pueden resumir en los siguientes tres puntos [6]:

- Incrementa la seguridad
- Aumenta la calidad
- Se reduce el tiempo de implementación.

El diagrama ideal de una configuración HiL se puede ver en la figura 2.1, donde el controlador, el actuador y el sensor son dispositivos de hardware a probar en el modelo del sistema. La comunicación entre los aparatos puede ser analógica o digital [29].



Figura 2.1: Hardware in the Loop ideal adaptado de [29]

En un principio el HiL se utilizó en el campo aeronáutico, en simuladores para pilotos. En su etapa más temprana las simulaciones se hacían en sistemas analógicos, posteriormente se hicieron digitales y luego en tiempo real. En la actualidad, manuales de control de instituciones como la NASA y la ECSS indican esta técnica como la última etapa de verificación y validación de un sistema [29].

Esta técnica se utiliza actualmente en múltiples industrias y es el tema de cada vez más investigaciones, como ejemplo, en [27] se utiliza para probar sistemas de control en tractores de agricultura, por otro lado en [24] se hace una evaluación de un sistema para probar ejes de vehículos.

En ambos casos el procedimiento es similar a lo ya mencionado, como se busca probar un hardware que es parte de un sistema más grande, primero se busca crear un modelo que represente de forma virtual el sistema físico sobre el cual se quiere hacer la prueba, teniendo en cuenta que las entradas y salidas sean iguales a la realidad para que el dispositivo de control se pueda probar en la simulación como si fuera el sistema real.

2.2. Sistema Glucosa-Insulina

Se puede decir que el páncreas funciona como un sistema de control lazo cerrado biológico. La cantidad normal de glucosa en la sangre se encuentra entre 70mg/dL y 110mg/dL, si la cantidad de glucosa detectada en la sangre es más alta, se induce una reacción de las llamadas células β , que son las que generan insulina, la cual estimula



Figura 2.2: Diagrama del sistema Glucosa-Insulina (elaboración propia)

el consumo de glucosa por las células, si más bien la cantidad de glucosa es baja, las llamadas células α liberan la hormona glucagón, que estimula al hígado para liberar glucosa en la sangre [14].

En la figura 2.2 se tiene una representación básica del sistema, donde, partiendo de un estado con niveles normales de glucosa, si se empieza a consumir alimento, subirá el nivel glucosa, provocando una señal dirigida al páncreas que estimulará las células β para que produzcan insulina y se promueva la el consume de glucosa. De manera análoga, si se hace actividad física se producirá glucagón para producir glucosa porque los niveles de esta empezarían a bajar.

2.3. Modelos del sistema glucosa-insulina

Existen dos tipos de enfoques para elaborar modelos del sistema en cuestión. Los modelos fisiológicos, que describen las interacciones entre la glucosa y la insulina con ecuaciones diferenciales y reglas basadas en el entendimiento de la dinámica entre ambas sustancias, la transferencia de masa y las reacciones químicas. Por otro lado modelos basados en datos usan técnicas de modelado empírico basándose en mediciones de varias entradas ante cambios en la glucosa o la insulina [4].

Los modelos fisiológicos usualmente dividen el cuerpo en diferentes compartimentos donde se representa la distribución de la glucosa y la insulina entre órganos y tejidos. Las ecuaciones diferenciales se deducen a partir de reglas de balance de masa. La mayoría de estos modelos fisiológicos son demasiado complejos para utilizar en algoritmos de control, pero son de utilidad en simuladores [4].

Las pruebas realizadas en pacientes para obtener las relaciones de transferencia de masa y los parámetros de los modelos son variadas y dependen de la información que se quiera obtener. La más popular es la prueba de tolerancia a la glucosa intravenosa (IVGTT). Se hace la prueba intravenosa porque es más sencilla para deducir un modelo, ya que durante esta prueba el sistema es perturbado con una entrada conocida [8].

En la prueba IVGTT a los sujetos se les administra una dosis de glucosa conocida y las concentraciones de insulina y glucosa en el plasma sanguíneo son medidas constantemente hasta que el efecto de esa glucosa desaparezca [4]. Existen diversas variaciones que se le pueden realizar al experimento, una de las más útiles es marcar la glucosa con un isótopo radioactivo para trazar con precisión el recorrido que hace a través de los tejidos.

Otro tipo de prueba es la de tolerancia a la glucosa oral, esta es más fácil de realizar, pero resulta más complejo deducir un modelo porque la entrada del sistema, que en este caso es la tasa de aparición de glucosa en el plasma (R_a) , presenta dinámicas más complicadas de estimar [8].

Aparte de la categorización de acuerdo con los exámenes realizados para obtener los modelos, estos también se pueden clasificar de acuerdo a su complejidad en modelos mínimos y máximos. Un modelo mínimo contempla solo una pequeña parte de los procesos fisiológicos de la persona, y solo se dividen en un pequeño número de compartimentos. Los modelos máximos por lo general dividen el cuerpo en una gran cantidad de compartimentos y se componen de una gran número de ecuaciones y parámetros [8].

2.4. Modelo Mínimo de Bergman

El modelo mínimo de Bergman se basa en la prueba IVGTT. Para modelar el sistema este se dividió en dos subsistemas uno para caracterizar el comportamiento de la glucosa



Modelo Mínimo de Desaparición de Glucosa

Figura 2.3: Diagrama de la dinámica de la glucosa en el modelo mínimo de Bergman adaptado de [1]

y otro para el de la insulina. El primer subsistema es el segmento de la glucosa, este lo gobierna la insulina en plasma y de manera análoga, el otro subsistema es el segmento de la insulina y lo regula la glucosa en plasma [1].

El diagrama del modelo del segmento de la glucosa se puede ver en la figura 2.3. La acción de la insulina sobre la glucosa tiene su efecto en el compartimento remoto de insulina. El efecto de la insulina hace que la glucosa desaparezca a una tasa específica, su comportamiento viene dado por las siguientes ecuaciones:

$$\frac{dG(t)}{dt} = -(p_1 + X(t))G(t) + p_1G_b$$
$$\frac{dX(t)}{dt} = -p_2X(t) + p_3(I(t) - Ib)$$

donde:

• G(t) es la concentración de glucosa en plasma.

- G_b es la concentración de glucosa en plasma en estado base.
- X(t) es proporcional a la insulina en el compartimento remoto.
- $\bullet \ p_1$ es el coeficiente de intercambio de glucosa independiente de la insulina.
- p_2 y p_3 son parámetros relacionados con el índice de sensibilidad de insulina mediante la relación $S_1 = \frac{-P_3}{P_2}$

Modelo Mínimo Dinámica de Insulina



Figura 2.4: Diagrama de la dinámica de la insulina en el modelo mínimo de Bergman adaptado de [1]

El segmento de la insulina se representa en la figura 2.4 y su comportamiento lo describe:

$$\frac{dI(t)}{dt} = -nI(t) + \gamma(G(t) - h)t \tag{2.1}$$

donde:

- I(t) es la concentración de insulina en plasma.
- $n (\min^{-1})$ es la constante de tiempo de la desaparición de la insulina.
- γ es una constante de proporción.
- *h* es el nivel máximo de glucosa.

La exactitud del modelo mínimo ha sido objeto de debate en varias investigaciones a lo largo del tiempo desde que se propuso en los años 80. Se objeta, por ejemplo, que se sobrestima el efecto de la glucosa sobre el consumo de la misma y se subestima la contribución de la insulina [26]. Por otra parte, el modelo mínimo no admite equilibrio y la concentración de insulina incrementa sin límite en algunos casos [11] [4].



Figura 2.5: Diagrama del modelo de Hovorka adaptado de [15]

Los problemas y limitaciones identificados en el modelo mínimo han dado paso a múltiples modificaciones del mismo [5] [19], dando como resultado que este siga vigente como uno de los modelos más populares por su estructura simple y la cantidad pequeña de parámetros [4].

2.5. Modelo de Hovorka

Así como el objetivo de Bergman y compañía era representar los resultados de una prueba IVGTT, lo que intentan Hovorka con sus colegas es dividir en su modelo la distribución, la eliminación y la producción endógena de glucosa partiendo de los resultados de una prueba IVGTT marcada con glucosa enriquecida [16]. De los modelos que se obtuvieron de las pruebas realizadas, a continuación se presenta uno que modela el sistema regulador de glucosa, mediante tres subsistemas, representando las dinámicas de la glucosa, la insulina y la acción de la insulina[15]. El comportamiento del modelo se puede representar mediante la figura 2.5.

El subsistema que representa la dinámica de la glucosa viene dado por el sistema de ecuaciones [15]:

$$\begin{cases} \dot{Q}_{1}(t) = -\left[\frac{F_{01}^{c}}{V_{G}G(t)} + x_{1}(t)\right]Q_{1}(t) + k_{12}Q_{2}(t) - F_{R} + U_{G}(t) + \text{EGP}_{0}\left[1 - x_{3}(t)\right]\\ \dot{Q}_{2}(t) = x_{1}(t)Q_{1}(t) - \left[k_{12} + x_{2}(t)\right]Q_{2}(t) \end{cases}$$

$$(2.2)$$

con G(t) dado por:

$$G(t) = \frac{Q_1(t)}{V_G}$$
(2.3)

donde:

 Q_1 (mmol) es la masa de glucosa en compartimentos accesibles (donde se toma la medida).

 Q_2 (mmol) es la masa de glucosa en compartimentos no accesibles.

 $k_{12} (\min^{-1})$ representa la tasa de transferencia entre compartimentos accesibles y no accesibles.

 V_G (L) es el volumen de distribución del compartimento accesible.

G(t) (mmol/L) es la concentración de glucosa medible.

EGP (mmol/min) es la producción de glucosa endógena.

La variable F_{01}^c (mmol/min) es el flujo de la glucosa no dependiente de la insulina, viene dado por la siguiente ecuación:

$$F_{01}^{c} = \begin{cases} F_{01} & \text{si } G \ge 4,5\text{mmolL}^{-1} \\ F_{01}G/4,5 \end{cases}$$
(2.4)

Por otro lado, F_R (mmol) es la expulsión renal de glucosa, es mayor a cero cuando se alcanza el límite de G = 9 mmol/L, como se ve en:

$$F_R = \begin{cases} 0.003(G-9)V_G & \text{si } G \ge 9 \text{mmolL}^{-1} \\ 0 \end{cases}$$
(2.5)

La tasa de absorción de glucosa ingerida por parte del intestino es representada por una cadena de dos compartimentos con la misma tasa de transferencia, esto se modela en (2.6), donde, $t_{max,G}$ (min) es el tiempo de aparición máxima de glucosa en el compartimento accesible; D_G (mmol) es la cantidad de carbohidratos digeridos y A_G es la bio-disponibilidad de los carbohidratos.

$$U_G(t) = \frac{D_G A_G t e^{-t/t_{max},0}}{t_{max,G}^2}$$
(2.6)

La absorción de insulina viene dado por (2.7) y (2.8) donde S_1 y S_2 son dos compartimentos en cadena que representan la absorción de insulina de rápida acción administrada de manera subcutánea; u(t) representa la administración de insulina y $t_{\text{máx},1}$ (min) es el tiempo a la máxima absorción de insulina. La aparición de insulina en plasma viene dada por $U_I = S_2/t_{\text{máx},I}$.

$$\dot{S}_{1}(t) = u(t) - \frac{S_{1}(t)}{t_{\text{máx},1}}$$
(2.7)

$$\dot{S}_{2}(t) = \frac{S_{1}(t)}{t_{\text{máx,I}}} - \frac{S_{2}(t)}{t_{\text{máx,I}}}$$
(2.8)

La concentración de insulina en plasma viene descrita por (2.9) donde $k_e (\min^{-1})$ es la tasa de eliminación y V_l (L) es el volumen de distribución.

$$\dot{I}(t) = \frac{U_I(t)}{V_I} - k_e I(t)$$
(2.9)

La acción de la insulina se modela con tres compartimentos representando tres efectos de la insulina remota sobre la glucosa, primero sobre la distribución, el asentamiento y la producción. Estos compartimentos se modelan en (2.10).

$$\begin{cases} \dot{x_1}(t) = -k_{a1}x_1(t) + k_{b1}I(t) \\ \dot{x_2}(t) = -k_{a2}x_2(t) + k_{b2}I(t) \\ \dot{x_3}(t) = -k_{a3}x_3(t) + k_{b3}I(t) \end{cases}$$
(2.10)

donde:

- $x_1 (\min^{-1}), x_2 (\min^{-1}) y x_3 (\min^{-1})$ representan la acción de la insulina sobre la glucosa.
- k_{ai}, i = 1,..., 3 (min⁻¹) y k_{bi}, i = 1,..., 3 (min⁻²por mU/L) representan tasas de activación.

2.6. Simulador de de Dalla Man y Cobelli



Figura 2.6: Diagrama de las principales partes del simulador adaptado de [9]

Este modelo asume que los subsistemas de glucosa e insulina están ligados por el efecto que tiene la insulina sobre la utilización y la producción endógena de glucosa [21].

Para la identificación del modelo se utilizó la estrategia de forzar la función a los datos promedio, que en este caso provienen de una base de datos de 204 pacientes con diversas características físicas [9].

El modelo y los parámetros obtenidos están pensados para personas sin diabetes o con tipo 2, pero con algunas modificaciones, puede describir también un paciente con tipo 1 [10].

La descripción de las partes del modelo, representadas en la figura 2.6, se detalla a continuación.

2.6.1. Modelo de Dinámica de Glucosa



Figura 2.7: Diagrama del modelo de glucosa adaptado de [9]

La cinética de la glucosa viene dada por un modelo de dos compartimentos descritos en

$$\begin{cases} \dot{G}_{p}(t) = EGP(t) + Ra(t) - U_{ii}(t) - E(t) - k_{1} \cdot G_{p}(t) + k_{2} \cdot G_{t}(t) & G_{p}(0) = G_{pb} \\ \dot{G}_{t}(t) = -U_{id}(t) + k_{1} \cdot G_{p}(t) - k_{2} \cdot G_{t}(t) & G_{t}(0) = G_{tb} \\ G(t) = \frac{G_{p}}{V_{G}} & G(0) = G_{b} \\ (2.11) \end{cases}$$

donde:

 $G_p (mg/kg)$ es la masa de glucosa en plasma y en tejidos de rápida absorción. $G_t (mg/kg)$ representa la glucosa presente en tejidos de lenta absorción. G (mg/dL) es la concentración de glucosa en el plasma. EGP (mg/kg/min) es la producción endogena de glucosa.

 $R_a (mg/kg/min)$ es la tasa de aparición de glucosa en el plasma.

E(mg/kg/min) es la tasa de expulsión renal.

 U_{ii} (mg/kg/min) es la utilización de glucosa independiente de la insulina.

 $U_{id} (mg/kg/min)$ es la utilización de glucosa dependiente de la insulina.

 $V_G (dl/kg)$ es el volumen de distribución de glucosa.

 k_1 y k_2 (min⁻¹) son parámetros de proporción.

este sistema toma en cuenta la glucosa presente el tejidos de rápida y lenta absorción y la utilización de la misma. Un diagrama de la relación entre los compartimentos y las señales asociadas se puede ver en la figura 2.7. Se considera que la utilización de la glucosa puede darse en tejidos que no ocupan insulina, como el cerebro, y los que necesitan de insulina para su procesamiento [9].

Parametro	Valor (Normal)
$V_G (dl/kg)$	1,88
$k_1 \ (min^{-1})$	0,065
$k_2 (min^{-1})$	0,079

Cuadro 2.1: Parámetros del subsistema de glucosa [9] Parámetro | Valor (Normal)

Los parámetros de (2.11) vienen dados en el cuadro 2.1, se tomaron en cuenta los valores en una persona sin diabetes (normal) porque posteriormente son los que se utilizaran en el paciente con diabetes tipo 1 [10]; lo mismo se aplica para los parámetros de las demás partes del modelo.

La tasa de aparición de glucosa y la expulsión renal son variables que están ligadas al subsistema que describe el consumo de glucosa por vía oral y la absorción por parte del intestino.

Producción Endógena de Glucosa

La producción endógena de glucosa sucede en el hígado y tiene relación con la cantidad de insulina y glucosa en la sangre. En el estado base, la producción endógena de glucosa

es proporcional a la tasa de desaparición de esta, que a su vez es la suma de la utilización por el organismo (U_b) y la expulsión renal base (E_b) [9], como se muestra en:

$$EGP_b = U_b + E_b. \tag{2.12}$$

En

$$EGP(t) = k_{p1} - k_{p2} \cdot G_p(t) - k_{p3} \cdot I_d(t) - k_{p4} \cdot I_{po}(t)$$

$$EGP(0) = EGP_b$$
(2.13)

donde:

 $G_p (mg/Kg)$ es la masa de glucosa en plasma y en tejidos de rápida absorción.

 k_{p1} (mg/Kg/min) es la producción endógena de glucosa extrapolada a niveles nulos de glucosa e insulina.

 $k_{p2} (min^{-1})$ es la efectividad de la glucosa del hígado.

 $k_{p3} (mg/Kg/minporpmol/l)$ es un parámetro que defina la amplitud de la acción de la insulina en el hígado.

 k_{p4} (mg/Kg/minporpmol/Kg) es un parámetro que defina la amplitud de la acción de la insulina porta en el hígado.

 $I_d \ (pmol/l)$ es la cantidad de insulina.

 $I_{po} (pmol/kg)$ es la cantidad de insulina en la vena porta hepática.

se muestra la relación de la producción endógena de glucosa con la glucosa en plasma, la insulina de la vena porta hepática (I_{po}) y además una señal de insulina (I_d) [9].

En

$$\begin{cases} \dot{I}_1(t) = -k_i \cdot [I_1(t) - I(t)] & I_1(0) = I_b \\ \dot{I}_d(t) = -k_i \cdot [I_d(t) - I_1(t)] & I_d(0) = I_b \end{cases}$$
(2.14)

se describe el comportamiento de la señal I_d , aquí, $k_i(min^{-1})$ representa el delay entre la señal de insulina y la acción de de esta. Por otra parte,

valor (Normar)
2,70
$0,\!0021$
0,009
0,0618
0,0079

Cuadro 2.2: Parámetros de la producción endógena [9] Parámetro | Valor (Normal)

$$k_{p1} = EGP_b + k_{p2} \cdot G_{pb} + k_{p3} \cdot I_b + k_{p4} \cdot I_{pob}$$
(2.15)

representa el comportamiento de las variables en estado base. En el cuadro 2.2 están los parámetros para (2.13), (2.14) y (2.15).

Utilización de la Glucosa

El modelo asume que la utilización de la glucosa ocurre en tejidos tanto de manera independiente como dependiente de la insulina. La utilización en el cerebro y en eritrocitos ocurre de manera constante e independiente de la insulina [9], este valor es $F_{cns}(mg/Kg/min)$ en

$$U_{ii}(t) = F_{cns}.$$
(2.16)

La utilización dependiente de insulina describe el comportamiento en los compartimentos remotos y depende de la glucosa en tejidos [9], de acuerdo con

$$U_{id}(t) = \frac{V_m(X(t)) \cdot G(t)}{K_m(X(t)) + G(t)}.$$
(2.17)

Las variables $V_m(X(t))$ y $K_m(X(t))$ en (2.17) dependen linealmente de una insulina remota $X(t) \ (pmol/L)$ [7], como se puede ver en:

$$V_m(X(t)) = V_{m0} + V_{mx} \cdot X(t)$$
(2.18)

$$K_m(X(t)) = K_{m0} + K_{mx} \cdot X(t)$$
(2.19)

La variable X(t) depende de la insulina en plasma (I(t)) y la tasa p_{2U} (min^{-1}) que relaciona la acción de la insulina con la utilización de glucosa, el comportamiento de esta variable se describe como

$$\dot{X}(t) = -p_{2U} \cdot X(t) + p_{2U}[I(t) - I_b] \quad X(0) = 0$$
(2.20)

La utilización total de la glucosa comprende la suma del uso dependiente e independiente de la insulina de acuerdo con

$$U(t) = U_{id}(t) + U_{ii}(t)$$
(2.21)

En estado base, se puede calcular la glucosa presente en tejidos de rápido equilibrio mediante (2.22). Por otro lado, los estados base de la utilización de glucosa y la variable V_m se describen en (2.23) y (2.24) respectivamente. Los parámetros para las ecuaciones se encuentran en el cuadro 2.3.

$$G_{tb} = \frac{F_{cns} - EGP_b + k_1 \cdot G_{pb}}{k_2}$$
(2.22)

$$U_b = EGP_b = F_{cns} + \frac{V_{m0} \cdot G_{tb}}{K_{m0} + G_{tb}}$$
(2.23)

$$V_{m0} = \frac{(EGP_b - F_{cns}) \cdot (K_{m0} + G_{tb})}{G_{tb}}$$
(2.24)

Expulsión Renal

La eliminación de glucosa por medio de los riñones sucede si la glucosa en plasma sobrepasa un límite determinado, provisto por el parámetro k_{e2} (mg/Kg) en:

Parametro	valor (Normal)
F_{cns}	1
V_{m0}	2,50
V_{mx}	$0,\!047$
K_{m0}	$225,\!59$
K_{mx}	0
p_{2U}	$0,\!0331$

Cuadro 2.3: Parámetros de la utilización [9] Parámetro | Valor (Normal)

Cuadro 2.4: Parámetros del expulsión renal [9] Parámetro | Valor (Normal)

i arametro	(iteliai)
k_{e1}	$0,\!0005$
k_{e2}	339

$$E(t) = \begin{cases} k_{e1} \cdot [G_p(t) - k_{e2}] & \text{si } G_p(t) > k_{e2} \\ 0 & \text{si } G_p(t) \le k_{e2} \end{cases}$$
(2.25)

La expulsión tiene una relación lineal con la glucosa del plasma como se ve en la ecuación donde también k_{e1} representa la tasa de filtración en los riñones. Los parámetros de este subsistema se encuentran en el cuadro 2.4.

2.6.2. Modelo de Dinámica de Insulina



Figura 2.8: Diagrama del modelo de insulina [9]

El modelo de insulina viene dado por un modelo de dos compartimentos descrito por

$$\begin{cases} \dot{I}_{l}(t) = -(m_{1} + m_{3}(t)) \cdot I_{l}(t) + m_{2}I_{p}(t) + S(t) & I_{l}(0) = I_{lb} \\ \dot{I}_{p}(t) = -(m_{2} + m_{4}) \cdot I_{p}(t) + m_{1} \cdot I_{l}(t) & I_{p}(0) = I_{pb} \\ I(t) = \frac{I_{p}}{V_{I}} & I(0) = I_{b} \end{cases}$$

$$(2.26)$$

donde:

 $I_p \; (pmol/kg)$ es la masa de insulina en plasma.

 $I_l (pmol/kg)$ representa la insulina presente en el hígado.

 $I \ (pmol/L)$ es la concentración de insulina en el plasma.

 $S \ (pmol/kg/min)$ es la secreción de insulina.

 $V_{I}\left(L/kg\right)$ es el volumen de distribución de insulina.

 $m_1, m_2, y m_4 (min^{-1})$ son parámetros de proporción.

y en forma de diagrama en la figura 2.8. Los parámetros de las ecuaciones viene dados en el cuadro 2.5.

Valor (Normal)
$0,\!05$
0,190
$0,\!484$
0,194
0,0304
0,6471
0,6

Cuadro 2.5: Parámetros del subsistema de insulina [9]

La extracción hepática (HE(t)) de la insulina está relacionada con la secreción de esta, como se ve en

$$HE(t) = -m_5 S(t) + m_6 \quad HE(0) = HE_b \tag{2.27}$$
y a su vez define el comportamiento de la variable $m_3(t)$ que representa la tasa de desaparición de la insulina en el hígado.

$$m_3(t) = \frac{HE(t) \cdot m_1}{1 - HE(t)}$$
(2.28)

$$m_3(0) = \frac{HE_b \cdot m_1}{1 - HE_b} \tag{2.29}$$

2.6.3. Modelo de Absorción de Comida

Se asume que la glucosa en este caso llega por vía oral. Los efectos que le suceden a la glucosa ingerida por vía oral se describen en [7]:

$$\begin{array}{ll}
Q_{sto}(t) = Q_{sto1}(t) + Q_{sto2}(t) & Q_{sto}(0) = 0 \\
\dot{Q}_{sto1}(t) = -k_{gri} \cdot Q_{sto1}(t) + D \cdot \delta(t) & Q_{sto1}(0) = 0 \\
\dot{Q}_{sto2}(t) = -k_{empt}(Q_{sto}) \cdot Q_{sto2}(t) + k_{gri} \cdot Q_{sto1}(t) & Q_{sto2}(0) = 0 \\
\dot{Q}_{gut} = -k_{abs} \cdot Q_{gut}(t) + k_{empt}(Q_{sto}) \cdot Q_{sto2}(t) & Q_{gut}(0) = 0 \\
Ra(t) = \frac{f \cdot k_{abs} \cdot Q_{gut}(t)}{BW} & Ra(0) = 0
\end{array}$$
(2.30)

donde:

 Q_{sto} (mg): Cantidad de glucosa en el estómago.

 Q_{sto1} (mg): Cantidad de glucosa en el estómago (sólida).

 Q_{sto2} (mg): Cantidad de glucosa en el estómago (líquida).

 Q_{gut} (mg): Cantidad de glucosa en intestino

 k_{gri} (min⁻¹): tasa de procesamiento.

 k_{empt} (min^{-1}) : tasa de vaciado gástrico.

 k_{abs} (min⁻¹): tasa de absorción intestinal.

 $R_a (mg/kg/min)$ tasa de aparición de glucosa en plasma.

f: fracción de la absorción intestinal que aparece en plasma.

BW(Kg): Peso corporal.

Parámetro	Valor (Normal)
k_{max}	0,0558
k_{min}	0,0080
k_{abs}	0,057
k_{gri}	0,0558
f	0,90
a	0,00013
b	0,82
c	0,00236
d	0,010

Cuadro 2.6: Parámetros tasa de aparición de glucos
a $\left[9\right]$

D(mg): Cantidad de glucosa ingerida.

Los parámetros de las ecuaciones se encuentran en el cuadro 2.6.

La tasa de vaciado gástrico es una función de la cantidad de glucosa en el estómago y viene dada por la variable k_{empt} . En

$$k_{empt} \left(\mathbf{Q}_{sto} \right) = k_{min} + \frac{k_{max} - k_{min}}{2} \left\{ \tanh \left[\alpha \left(Q_{sto} - b\bar{D}(t) \right) \right] - \tanh \left[\beta \left(Q_{sto} - a\bar{D}(t) \right) \right] + 2 \right\}$$

$$(2.31)$$

se presenta manera de calcular la tasa k_{empt} en función de la cantidad de glucosa en el estómago (Q_{sto}) . En (2.32), (2.33) y (2.34) se muestra como se calculan los parámetros necesarios para el calculo de la tasa de vaciado gástrico.

$$\alpha = \frac{5}{2\bar{D}(t)(1-b)}$$
(2.32)

$$\beta = \frac{5}{2\bar{D}(t)a} \tag{2.33}$$

$$\bar{D}(t) = Q_{sto} + \int_t^{t_f} d(\tau) d\tau$$
(2.34)

Cuadro 2.7: Parámetros tasa de aparición de insulina [10]

Parámetro	Valor
k_d	$0,\!0164$
k_{a1}	$0,\!0018$
k_{a2}	$0,\!0182$

2.6.4. Adaptación del Simulador para Pacientes Tipo 1

La secreción de insulina $S \ (pmol/kg/min)$ está presente cuando se considera un paciente sin diabetes, o diabetes tipo 2. Para una persona con diabetes tipo 1 se debe sustituir esta variable por un modelo que caracterice la aparición de insulina externa, por una bomba de insulina, en el caso del páncreas artificial.

Los autores del modelo proponen utilizar el siguiente modelo para la aparición de insulina[10]:

$$\begin{cases} \dot{I}_{sc1}(t) = -(k_d + k_{a1}) \cdot I_{sc1}(t) + IIR(t) & I_{sc1}(0) = I_{sc1ss} \\ \dot{I}_{sc2}(t) = k_d \cdot I_{sc1}(t) - k_{a2} \cdot I_{sc2}(t) & I_{sc2}(0) = I_{sc2ss} \end{cases}$$
(2.35)

$$R_i(t) = k_{a1} \cdot I_{sc1}(t) + k_{a2} \cdot I_{sc2}(t)$$
(2.36)

donde:

 I_{sc1} (pmol/kg): Insulina no monomérica en el espacio subcutáneo.

 I_{sc2} (pmol/kg): Insulina monomérica en el espacio subcutáneo.

IIR (pmol/kg/min): Tasa de infusión de insulina externa.

 $k_d \ (min^{-1})$: tasa de disociación de la insulina.

 k_{a1} (min⁻¹): tasa de absorción de insulina no monomérica

 k_{a2} (min⁻¹): tasa de absorción de insulina monomérica

 $R_i (pmol/kg/min)$: Tasa de aparición de insulina.

Las ecuaciones de la dinámica de la glucosa se ven afectadas por la modificación anterior, pasan de lo definido en (2.26) a las siguientes ecuaciones [13]:

$$\begin{cases} \dot{I}_{l}(t) = -(m_{1} + m_{3}(t)) \cdot I_{l}(t) + m_{2}I_{p}(t) & I_{l}(0) = I_{lb} \\ \dot{I}_{p}(t) = -(m_{2} + m_{4}) \cdot I_{p}(t) + m_{1} \cdot I_{l}(t) + R_{i}(t) & I_{p}(0) = I_{pb} \\ I(t) = \frac{I_{p}}{V_{I}} & I(0) = I_{b} \end{cases}$$

$$(2.37)$$

Además, las ecuaciones que definen los estados base cambian, la insulina en el hígado (I_{lb}) pasa a ser proporcional a la insulina en plasma directamente como se define en:

$$I_{lb} = I_{pb} \frac{m_2}{m_1 + m_3}.$$
 (2.38)

De igual manera, la definición del parámetro k_{p1} cambia de (2.15) a lo que se tiene en:

$$k_{p1} = EGP_b + k_{p2} \cdot G_{pb} + k_{p3} \cdot I_b.$$
(2.39)

La tasa de infusión de insulina externa dependerá de si se cuenta con un sistema de lazo cerrado o abierto; para describirla se puede utilizar

$$IIR(t) = \sum_{j=1}^{N_{meal}} D_j^{ins} \cdot \delta(t - t_j) + IIR_b$$
(2.40)

donde:

- N_{meal} es la cantidad de comidas.
- D_{j}^{ins} es la dosis de insulina externa inyectada de manera subcutánea.
- t_j es el tiempo en que se dio la comida.
- IIR_b es la tasa base de infusión de insulina externa.

en el caso de un sistema a lazo abierto.

Por otro lado, la insulina en plasma $(I_p b)$ termina dependiendo de la tasa de infusión de insulina, para un sistema de lazo cerrado esta pueda cambiar, pero para un sistema en lazo abierto se rige por:

$$I_{pb} = \frac{IIR_b}{m_2 + m_4 - \frac{m_1m_2}{m_1 + m_3}}.$$
(2.41)

En estado estacionario las cantidades insulina en el espacio subcutáneo vienen dadas por

$$I_{sc1ss} = \begin{cases} \frac{I_{pb}}{k_d + k_{a1}} \cdot \left(m_2 + m_4 - \frac{m_1 \cdot m_2}{m_1 + m_3} \right) & \text{en lazo cerrado} \\ \frac{IIR_b}{k_d + k_{a1}} & \text{en lazo abierto} \end{cases}$$
(2.42)

$$I_{sc2ss} = \frac{k_d}{k_{a2}} \cdot I_{sc1ss} \tag{2.43}$$

donde I_{sc1ss} es la insulina no monomérica en estado estacionario y I_{sc2ss} es la insulina monomérica en estado estacionario.

En el caso de tener un sistema a lazo cerrado, la tasa de infusión de insulina pasaría a depender del algoritmo de control que se utilice.

2.7. Sensor de glucosa

Existen varios sensores para medir el nivel de glucosa en el cuerpo, en principio, se podría medir en cualquier parte del cuerpo, el reto que se presenta es el tiempo de espera desde que el nivel de glucosa cambió, hasta que se vea afectada la parte del cuerpo desde donde se está midiendo, esto depende de lo que dure el transporte a través de la sangre y de los fluidos entre los tejidos [30].

Al tiempo de que le toma la glucosa llegar al lugar cercano al sensor, se le añade la dinámica del sensor, dependiendo de la tecnología presenta diferente tiempo de retardo. Este retardo se debe a membranas protectoras alrededor del sensor y reacciones electroquímicas que se deben llevar a cabo para obtener la medición [30].

Un tipo de sensor es el subcutáneo, que se puede adquirir comercialmente, este presenta un retardo de al menos 6 minutos. Este sensor hace su medición con reacciones electro-químicas o enzimáticas para determinar, a partir del fluido del tejido subcutáneo, el nivel de glucosa en la sangre [30]. Estos sensores subcutáneos tienen diferentes formas y tamaños, pueden ser agujas milimétricas de 3mm de largo o arreglos de agujas microscópicas. [30].



Figura 2.9: Diagrama del funcionamiento del sensor adaptado de [18].

A la hora de simular, el sensor se puede modelar de distintas maneras según la complejidad que se requiera, una de las maneras más simples es considerándolo como un retraso en la señal de glucosa [10]:

$$\dot{G}_s = \frac{1}{T_s}G(t) - \frac{1}{T_s}G_s(t)$$
(2.44)

La figura 2.9 representa otro ejemplo de como se pude modelar el sensor, en (2.45) se describe matemáticamente su funcionamiento [18].

$$\frac{dy_m(t)}{dt} = -\left(k_3 + k_2\right)y_m(t) + k_1 \frac{V_b}{V_{isf}}y_{IV}(t)$$
(2.45)

donde:

- $y_m(t) (mg/dL)$ es la concentración de glucosa medida por el sensor.
- $y_{IV}(t) \ (mg/dL)$ es la concentración de glucosa en la sangre.
- V_b es el volumen de sangre.
- V_{isf} es el volumen de fluido intersticial.
- $k_1, k_2 \neq k_3$ son tasas de difusión definidas en la figura 2.9

La ecuación (2.45) se puede representar a manera de una función de transferencia de

primer orden aplicando la transformada de Laplace, transformándose en la siguiente ecuación:

$$G_S(s) = \frac{Y_m(s)}{Y_{IV}(s)} = \frac{K}{\tau_s s + 1}$$
(2.46)

Los parámetros de (2.46) vienen definidos como $\tau_s = 1/(k_2 + k_3)$ (min) y $K = k_1 V_b / ((k_3 + k_2) V_{isf})$. Aplicando esta conversión, los parámetros que se deben averiguar son la ganancia (K) y la constante de tiempo (τ_s) [18].

En el caso de la ganancia, se puede decir que es 1 porque la idea del sensor es reflejar la cantidad de glucosa medida sin amplificarla. La constante de tiempo se puede variar desde 1 min hasta 30 min dependiendo del sensor que se quiera modelar, siendo para sensores subcutáneos un tiempo promedio de $12,4 \pm 3,6$ min [18]. Otros factores que afectan la medición de glucosa es la calibración del sensor y el ruido de la señal.

2.8. Bomba de insulina

La bomba de insulina consiste en una reserva de esta hormona, para poder ser suministrada automáticamente cuando sea necesario. Generalmente la bomba, al igual que el sensor, se puede operar desde el tejido subcutáneo para evitar los inconvenientes que tendría si se instalara internamente, aunque se tiene también esa posibilidad [30].

Uno de los grandes retos del páncreas artificial es la bomba de insulina. Se debe tomar en cuenta que su tamaño no sea demasiado inconveniente para el paciente, que las recargas de insulina no tengan que ser tan frecuentes y que se puedan realizar con relativa facilidad, y que además se cuente con una batería lo suficientemente grande para que el funcionamiento no se vea interrumpido frecuentemente [3].

Además, está el reto de que la bomba, a diferencia de una inyección de insulina directa al sistema circulatorio, depende de los fluidos del tejido subcutáneo para poder transportar la insulina hasta la sangre, generando un retardo en la llegada de esta cuando puede ser necesaria [30].

Debido al constante avance en la microelectrónica existen diferentes versiones de la bomba de insulina, siendo sus elementos básicos: una aguja (o arreglo de agujas), una



Figura 2.10: Diagrama del funcionamiento de la bomba adaptado de [17]

micro bomba y la reserva de insulina [32]. La bomba de insulina se controla generalmente mediante una señal PWM [17].

Un posible diagrama para la bomba se puede ver en la figura 2.10, el funcionamiento de esta bomba en particular depende del motor paso a paso que mediante el tornillo de avance empuja el contenido de la jeringa, la insulina se inyecta a través de un canal flexible que lleva a la aguja [17].

2.9. Algoritmos de Control

El algoritmo de control que maneja estos sistemas debe lidiar con los retardos generados por el sensor y la bomba de insulina, así como las dinámicas internas del cuerpo, incluido los picos y valles en los niveles de glucosa provocados por las actividades humanas ya sea el ejercicio o la ingesta de comida [3].

El efecto más peligroso para el paciente es caer en un estado de hipoglucemia provocado por una sobredosis de insulina, por lo que a estos controladores se les está implementando generalmente métodos de seguridad como un predictor de glucosa baja [3].

Generalmente, estos algoritmos realizan su accionar basados en modelos del comportamiento del cuerpo, el algoritmo que actualmente es el más prometedor es el control predictivo por modelo (MPC), el cual se basa en un modelo conocido y mediciones para determinar su acción. Por otro lado, se han hecho pruebas con controles basados en redes neuronales y lógica difusa [3].

El control PID, a pesar de que se le han notado falencias a la hora de controlar periodos de sueño y una relativa lentitud en la reacción en la respuesta ante la ingesta de alimentos, sigue siendo uno de los más estudiados debido a su amplia presencia en la industria y a su relativa facilidad de sintonización [3].



Figura 2.11: Diagrama del control PID (elaboración propia)

El algoritmo de control PID consiste en la suma de tres partes, recibe como entrada el error (e(t)) que resulta de comparar la señal de salida del proceso que se esta controlando con el valor deseado de esta. La salida del controlador u(t) viene dada por (2.47)[4]. La configuración básica de este control se puede ver en la figura 2.11.

$$u(t) = K_P e(t) + K_I \int_0^t e(t) dt + K_D \frac{de(t)}{dt}$$
(2.47)

donde:

- K_P es la ganancia proporcional.
- K_I es la ganancia de la parte integral.
- K_D es la ganancia de la parte derivativa.

El cálculo de las ganancias se puede hacer de varias maneras y depende del proceso a controlar y el tipo de salida que se quiera lograr. También se puede prescindir de la ganancia de la parte integral o derivativa, o ambas si fuera necesario. Cada sumando de la ecuación tiene su efecto sobre la señal medida: la ganancia proporcional genera un efecto inmediato del controlador respecto a la señal de error, la parte integradora suma el valor del error en el tiempo y fuerza el error en estado estacionario a cero; finalmente la parte derivativa responde a la dirección y magnitud del cambio en la salida [4].

Por los efectos de cada parte sobre la salida se debe considerar si la ganancia proporcional o derivativa se hacen muy grandes pueden causar inestabilidad en el sistema [4].

La acción de control en la configuración MPC se calcula mediante una optimización en tiempo real. El funcionamiento de este tipo de control se basa en cuatro componentes principales [4]:



Figura 2.12: Diagrama del control MPC adaptado de [4]

- Un modelo para predecir futuras salidas del proceso.
- Una función objetivo para optimizar el comportamiento del sistema.
- Un algoritmo de optimización.
- Restricciones para las entradas, salidas y sus tasas de cambio.

En la figura 2.12 se tiene un diagrama de la estructura de un control MPC, en este se puede observar que el controlador funciona con tres entradas: la realimentación del proceso que el controlador comparará con la salida del modelo, el valor deseado y la medida de la perturbación. El funcionamiento lo realiza mediante dos horizontes de predicción, uno para predecir los valores futuros de la salida del proceso y otro para predecir futuras configuraciones sobre la entrada del proceso [4].

Otros métodos de control también se han estudiado para la operación de un páncreas artificial, entre ellos el control adaptativo y el control basado en lógica difusa [4].

Capítulo 3

Implementación del Modelo

3.1. Modelo del Sistema Glucosa-Insulina

En la sección anterior se explicaron tres de los modelos con más referencias para modelar el sistema glucosa-insulina; el modelo de Bergman, siendo el más antiguo de los tres, es al mismo tiempo el más estudiado y el que más se usa de base para otros modelos sin embargo, en su forma original presenta suposiciones que lo alejan de la realidad [31].

El modelo de Hovorka, diseñado específicamente para pacientes de tipo 1 y pensado para implementarse como base para un controlador MPC que sirviera dentro de un páncreas artificial es considerado una buena aproximación del sistema [31].

Si se comparan el modelo de Hovorka con el de Dalla Man se tiene que este último presenta más parámetros y ecuaciones obtenidas directamente de datos clínicos [31], además, ha sido objeto de modificaciones para incluir el glucagón, que es la otra hormona involucrada en el proceso lo cual da una posibilidad de expansión en el futuro [8].

Aunque el modelo de Dalla Man primero fue descrito a partir de pruebas con pacientes sin diabetes o con tipo 2, las modificaciones hechas posteriormente para diabetes tipo 1 sirvieron de base para el primer simulador aprobado por la FDA para reemplazar a los animales en las pruebas de nuevos controladores para diabetes [8]. Considerando lo anterior se escogió el modelo de Dalla Man para simular el sistema glucosa-insulina.

3.2. Plataforma de Simulación

Para realizar la simulación del sistema comprendido por el sensor, la bomba, y la dinámica glucosa-insulina, fue seleccionado el dispositivo MyRIO de National Instrumets, esto para aprovechar sus características de capacidad de simulación en tiempo real, así como sus entradas y salidas integradas, y su capacidad de conectar una memoria externa.

Para cerrar el lazo de control y completar las pruebas se hizo uso de una plataforma de desarrollo basada en el Arduino UNO. Esta ofrece la capacidad suficiente para implementar un controlador simple basado en PID que funcione a tiempo real, así como entradas analógicas y digitales para su uso si fueran necesarias.

Dentro de los detalles que se tomaron en cuenta la hora de simular fue el tiempo de muestreo, el modelo seleccionado tiene sus constantes de tiempo y de transferencia medidas en minutos, por lo que cada paso que avance la simulación, se contará como un minuto en el sistema real sin embargo, para facilidad, se hizo que el MyRIO resolviera el sistema de ecuaciones diferenciales que representa el sistema con una velocidad de un paso cada 500 ms, este tiempo fue seleccionada porque a esta velocidad funcionaba tanto para el MyRIO como para el Arduino.

3.3. Efectos del Sensor de Glucosa y la Bomba de Insulina

Considerando lo descrito en el Marco Teórico, los efectos valorados para modelar el sensor de glucosa son el retraso que ocasiona sobre la señal de glucosa y el tiempo de muestreo. No se consideraron efectos de mala calibración o mal uso del sensor.

Para modelar el retraso de la señal se puede utilizar (2.46), por otro lado, el tiempo de muestreo, que para la mayoría de sensores va de tres a cinco minutos, se logró implementar en el controlador, haciendo que el tiempo de muestreo de este fuera de 300ms que equivaldrían a 3min el sistema real.

En la sección de la bomba de insulina se tiene el efecto de la transferencia de insulina en la zona subcutánea, el cual viene modelado en (2.35), y como lo describe la ecuación, se trata de un retraso entre la señal de insulina al llegar a la zona subcutánea y el efecto de esta en el sistema.

3.4. Implementación en MATLAB

Para implementar el modelo, primero se calcularon los parámetros iniciales necesarios. Para esto se tomaron como referencia las constantes de los cuadros del 2.1 al 2.6; así como las ecuaciones de la 2.12 a la 2.43. Estas ecuaciones brindaron información para calcular parámetros y condiciones iniciales, se implementaron en el código de Matlab del apéndice A para posteriormente usar los datos en Simulink.

Las ecuaciones restantes se simularon usando Simulink, empezando por (2.11), cuya implementación se ve en la figura 3.2. El objetivo de este diagrama es calcular la señal de salida de glucosa, la cual va a depender de los valores de la producción endógena de glucosa (EGP), la tasa de aparición de glucosa por la comida (R_a) , la utilización de glucosa tanto independiente (U_{ii}) como dependiente (U_{id}) de la insulina; y la extracción renal (E).

Para complementar el diagrama de glucosa, la utilización independiente de insulina se considero como un valor constante de 1, valor sugerido en la referencia de donde se obtuvo el modelo. Por otra parte, las funciones de la utilización de glucosa dependiente de la insulina U_{id} y extracción renal dadas por (2.17) y (2.25), se implementaron en los diagramas 3.1 y 3.3.

La utilización dependiente de la insulina será además dependiente del valor inicial de insulina basal, que para este caso se consideró que fuera cero, y de la tasa p2u. Por otro lado la extracción renal depende de si se supera la barrera en el nivel de glucosa dada por k_{e2} , como se ve en la figura 3.3



Figura 3.1: Implementación del subsistema de utilización de glucosa en Simulink.



Figura 3.2: Implementación del subsistema de glucosa en Simulink.



Figura 3.3: Implementación del subsistema renal en Simulink.

El efecto de la insulina, descrito por (2.26), se implementó en el diagrama de la figura 3.4. La entrada de este subsistema es la tasa de aparición de insulina R_i , la cual depende de la tasa de infusión externa de insulina IIR que varia dependiendo de si el sistema está en lazo abierto o cerrado, como se explico en la sección (2.6.4).



Figura 3.4: Implementación del subsistema de insulina en Simulink.

La variable R_i modela el retraso de la llegada de insulina al plasma provocado por el tejido subcutáneo en el cual se inyecta, esta se caracteriza mediante (2.36), la cual se implementó en el diagrama 3.5.



Figura 3.5: Implementación de la aparición de insulina.

El efecto del sistema digestivo se implementó en el diagrama de la figura 3.6, el cual fue hecho basado en (2.30) del modelo de absorción de comida, así mismo se realizó el diagrama de la figura 3.7 para calcular la variable K_{empt} tomando como referencia (2.31). La entrada del sistema digestivo se compone de una cantidad de glucosa ingerida (D) multiplicada por una función impulso, esta función impulso se modelo como un vector de ceros que alcance un valor de uno en un tiempo de muestra específico; también como entrada se tiene el inverso del valor del peso del paciente a modelar.



Figura 3.6: Implementación del subsistema de digestivo en Simulink.



Figura 3.7: Calculo del parámetro K_{empt} .

Se optó por probar el sistema en lazo cerrado debido a que es el modo en que va a operar la simulación final, para ello, la tasa de infusión externa de insulina se manejó mediante un controlador PID. Para las primeras pruebas se utilizaron los mismos parámetros de ganancia, tiempo integral y tiempo derivativo que el controlador de dos grados de libertad utilizado en [10], este se implementó en el diagrama de la figura 3.8.



Figura 3.8: PID implementado para probar el modelo.

Para comprobar que el sistema estuviera correctamente implementado se tomó como referencia la simulación realizada en [10]. Se realizó el lazo cerrado con el PID de la figura 3.8 y un bloque representara el sensor basado en (2.44), implementado en el diagrama de la figura 3.9.



Figura 3.9: Implementación del retraso del sensor en Simulink.

Se simuló en primer lugar tomando en consideración la figura 3.10 tomada de [10]. Se replicaron las mismas circunstancias de peso y esquema de alimentación, y posteriormente se obtuvieron los resultados, estos pueden ver en la figura 3.11.

Para obtener la figura 3.11, se ingresó un peso del paciente de 78 Kg, se simuló por 1400 min para obtener la simulación de un día completo y se ingresaron tres comidas, la primera de 35 g de glucosa a los 480 min, la segunda de 70 g a los 720 min, y la última también de 70 g a los 1200 min.

Comparando los gráficos de 3.11 con los de referencia dados en 3.10, se puede observar una diferencia en la magnitud de la tasa de aparición de glucosa. Esta diferencia provocada por la manera en que se simula la función impulso tiene su efecto ampliando las magnitudes de las demás señales de interés. A pesar de la diferencia mencionada, las otras señales se comportaron de manera similar por lo que se concluyó que el modelo estaba bien implementado en MATLAB.



Figura 3.10: Resultado de referencia de la simulación del modelo, tomado de [10]



Figura 3.11: Resultado de la Simulacion

3.5. Controlador

Se programó un controlador de referencia, como el utilizado en [10], sin embargo, se desarrolló otro controlador con el objetivo de lograr un mejor rendimiento. Para esto primero se decidió obtener un modelo de primer o segundo orden del sistema para aplicar una regla de sintonización.

A pesar de que se reconocen deficiencias en el control PID, este fue el algoritmo de control que se optó por aplicar debido a que, como se mencionó en el Marco Teórico, sigue siendo utilizado en una considerable parte de la industria y además es de relativa facilidad a la hora de sintonización. Por otra parte, el objetivo del proyecto está orientado a implementar y probar el modelo, no así a un estudio profundo para obtener el algoritmo de control más eficiente.

3.5.1. Identificación del Sistema

Para la identificación se contó con la herramienta de MATLAB llamada "System Identification Toolbox". Como primer paso se identificó el punto de operación en un nivel de glucosa de 120mg/dL y una tasa de infusión de insulina de $0.5pmol/(Kg \cdot min.$ Posteriormente se hicieron dos pruebas que se pueden ver en las figuras 3.12 y 3.13.

Las pruebas se hicieron con el objetivo de que el algoritmo de identificación de sistemas de MATLAB, pudiera obtener una función de transferencia que se ajustara a los datos. Se buscó principalmente que la dinámica a la hora de hacer un cambio en la entrada fuera muy parecida entre el modelo encontrado por el algoritmo y el sistema simulado.



Figura 3.12: Cambio de la salida ante una entrada en forma de un escalón



Figura 3.13: Cambio de la salida ante una entrada en forma de cadena de escalones

Tomando como referencia estas pruebas, se utilizó la opción de buscar un modelo de proceso, se dejó la elección de algoritmo de estimación en automático y se fijó el número máximo de iteraciones en 100; con estas opciones se obtuvo el siguiente modelo:

$$G(s) = \frac{-133,95e^{-30}}{(1+318,43s)(1+164,78s)}$$
(3.1)

En la misma herramienta de MATLAB se probó el modelo (3.1) y se vio que comparando con los datos de la figura 3.12, el modelo tenía una coincidencia de 99% y de 79% con los de la figura 3.13.

3.5.2. Síntesis del Controlador

Con el modelo escogido e identificado, se probaron múltiples reglas de sintonización [23], estás se seleccionaron considerando que fueran controladores PID enfocados en control regulatorio, ya que el objetivo del páncreas artificial es repeler las perturbaciones causadas en el nivel de glucosa debido a la ingesta de esta.

De las reglas de sintonización probadas, la que mejor controló el sistema fue una propuesta por Huang y Chao [23] sin embargo, también se probó la herramienta de sintonización de PID de Matlab y se encontró otro controlador.

Los parámetros de los controladores que se probarón se encuentran en el cuadro 3.1, en este cuadro, el controlador llamado "Huang" fue el obtenido por la regla de sintonización, el llamado "MATLAB" fue el obtenido mediante la herramienta de Matlab y el llamado "GIM" (del inglés: glucose insulin model) fue el controlador de referencia obtenido de [10].

<u>Cuadro 3.1: Parámetros de los controladores</u>						
Parámetros	GIM	MATLAB	Huang			
Ganancia	-0.032	-0.0304	-0.013			
Tiempo integral	450	424	271.06			
Tiempo derivativo	66	95.9	157.93			

La figura 3.14 compara los tres controladores ante un cambio en el valor basal de glucosa, ya que se inicia con una glucosa base de 180 mg/dL, pero se tiene un valor

Controlador

deseado de 120 mg/dL. Como se puede ver en la figura, todos los controladores inician con un pico en la acción del controlador por la gran diferencia entre el valor inicial de glucosa y el valor deseado. Para los tres controladores se impuso la restricción de un máximo de $10 pmol/(Kg \cdot min)$ para considerar márgenes de acción realistas.

Por otra parte al analizar el control regulatorio se hicieron dos pruebas cuyo resultado se ve en las figuras 3.15 y 3.16, la primera se hizo para analizar la reacción del sistema ante una perturbación y la segunda para emular el esquema de alimentación de las pruebas de referencia de la figura 3.10.

En la prueba de la figura 3.15, se simuló una cantidad de glucosa de 4,5 g, ingerida a los 720 min, esto para mostrar como el sistema tiende al inicio a aumentar el nivel de glucosa debido a la producción endógena de glucosa, para luego estabilizarse gracias a la acción del controlador en los deseados 120 mg/dL. Posteriormente al minuto de la perturbación se ve que el sistema sube y luego baja del nivel deseado, todos los controladores superan de manera similar el valor de 120 mg/dL sin embargo, el controlador hecho con la herramienta de MATLAB para sintonización es el que más rápido se recupera de un nivel bajo de azúcar, la cual es la situación más critica a controlar.



Figura 3.14: Comparación entre tres controladores en control del valor deseado



Figura 3.15: Comparación entre tres controladores en control regulatorio

Por otro lado, en la prueba de la figura 3.16, se simuló una cantidad de glucosa de 4,5g, ingerida a los $480 \min$, de 7 g, ingerida a los $720 \min$ y de 7 g, ingerida a los $1200 \min$; esto para mostrar como se comporta el sistema ante un esquema de alimentación de tres comidas en 24 h, al igual que la primera prueba, al inicio tienda a aumentar el nivel de glucosa debido a la producción endógena de glucosa, para luego estabilizarse gracias a la acción del controlador en los deseados 120 mg/dL.

Posteriormente al minuto de la perturbación se ve que el sistema sube y luego baja del nivel deseado, y al igual que la prueba anterior, todos los controladores superan de manera similar el valor de 120 mg/dL sin embargo, el controlador hecho con la herramienta de MATLAB para sintonización es el que más rápido se recupera de un nivel bajo de azúcar, la cual es la situación más critica a controlar.



Figura 3.16: Comparación entre tres controladores en control regulatorio

Para las tres pruebas realizadas se hizo el calculo del IAE, los resultados se muestran en el cuadro 3.2, en él se ve que el mejor control regulatorio se obtiene con el controlador sintetizado en la herramienta de MATLAB, razón por la cual es el que se utilizó posteriormente en la implementación de este en el Arduino.

Ouauro 5.2. TAE en las pruebas realizadas					
\mathbf{Prueba}	GIM	MATLAB	Huang		
Cambio en valor deseado	14870	16380	25320		
Una perturbación	49540	47350	64110		
Tres perturbaciones	73690	69350	85520		

Cuadro 3.2: IAE en las pruebas realizadas

3.6. Implementación en MyRIO

3.6.1. Implementación del Modelo de Cobelli y Dalla Man

Para la implementación en LabView se utilizaron bloques del llamado "Control & Simulation Toolbox", con esta herramienta, los diagramas entre LabView y Simulink se volvieron similares. En la figura 3.17 se tiene el modelo homólogo de la figura 3.2 al que también se le añadió la parte del diagram de la figura 3.3, para simular el subsistema de glucosa y el de la expulsión renal juntos.

A diferencia de MATLAB, donde se hizo un script para calcular las variables que luego se cargaron en Simulink, en LabView las constantes y ganancias se escribieron directamente en los diagramas, y las variables como el peso, la glucosa inicial y la glucosa externa ingerida fueron implementadas como elementos que se pudieran modificar en el panel principal del programa. En el caso del diagrama de glucosa, no hay variables que se puedan modificar.

El efecto del sensor en la medición de la glucosa se implementó en el diagrama de la figura 3.18. Al igual que en el caso de MATLAB, el sensor se simula como un retardo en la señal G, cuyo tiempo de retardo dependerá del valor T_s que puede ser variado en el panel principal de LabView.



Figura 3.17: Diagrama del subsistema de la dinámica de la glucosa.



Figura 3.18: Diagrama del subsistema del sensor.

El subsistema que modela a la utilización de glucosa dependiente de la insulina se implementó con el diagrama de la figura 3.19 que a su vez corresponde al diagrama de Simulink de la figura 3.1. Este diagrama irá conectado posteriormente al de glucosa para calcular el nivel de glucosa en plasma, los parámetros y las ganancias son constantes, así que se incluyeron en los bloques del diagrama, no se considero que fuera necesario modificarlos.



Figura 3.19: Diagrama para calculo de variable U_{id}

Por otra parte, el subsistema que modela la dinámica de la insulina se implementó en el diagrama de la figura 3.20. La entrada de este diagrama es la tasa de aparición de insulina externa R_i , que proviene de la infusión de insulina externa. Se consideró que la insulina al inicio de la simulación es cero, por lo que el valor inicial en los bloques de integración se fijó en cero sin opción a modificación en el panel principal del programa.

El subsistema que modela la dinámica de infusión de la insulina exógena en el diagrama de la figura 3.21. Este diagrama es el que recibirá la señal del control en la variable de tasa de infusión de insulina (IIR) y actuará como el retraso que hay entre la llegada de la insulina al espacio subcutáneo y la entrada de esta al torrente sanguíneo, tal y como se hizo para Simulink en el diagrama 3.5.



Figura 3.20: Diagram del subsistema de la dinámica de la insulina.



Figura 3.21: Diagrama del subsistema de infusión de insulina.

La dinámica del sistema digestivo, que genera la señal de aparición de glucosa, se implementó primeramente con el diagrama 3.22, para el correcto funcionamiento de este subsistema se implementó ademá el diagrama 3.23. El sistema digestivo se implementó para recibir tres señales que se pueden modificar. La señal que indica la cantidad de glucosa ingerida (D), la misma señal multiplicada por un impulso que indica el momento en que se ingirió esa señal de glucosa (D * d) y la información del peso del sujeto (BW).



Figura 3.22: Diagrama del subsistema digestivo.

La variable K_{empt} en 3.23, necesaria para el subsistema digestivo, se calculó mediante el diagrama 3.23. La entrada del calculo de K_{empt} fue la variable de la glucosa a ingerir (D), con esta información se calcularon las constantes α y β en 3.24, que luego se usaron para la salida del diagrama K_{empt} .



Figura 3.23: Diagrama para calculo de variable K_{empt} .



Figura 3.24: Diagrama para calculo de variables α y β .

Los subsistemas anteriores se juntaron en una sola simulación que implementara el modelo de Cobelli y Dalla Man, esta implementación se hizo en el diagrama de la figura 3.25.



Figura 3.25: Diagrama del modelo de Cobelli y DallaMan en LabView.

Para poder comprobar que el desarrollo del modelo en LabView fuera el correcto, se hizo una simulación de este con los mismos parámetros que la simulación hecha en Simulink vista en la figura 3.11, excepto que para hacer la prueba solo se hizo la simulación de tal manera que se partiera de un valor inicial de glucosa de 180mg/dL y se solicitara al PID mantenerlo en 120mg/dL.

Los datos se extrajeron de LabView y se graficaron en Matlab donde se compararon con los resultados de Simulink, esta comparación se ve en la figura 3.26. Para realizar esta figura, se simuló corriendo LabView en una PC, los resultados se exportaron a MATLAB y se extrajeron los datos para hacer la comparación con Simulink, como se ve en la figura, los datos son muy similares entres si, habiendo visualmente ninguna diferencia entre ellos. Esta comparación sirvió de base para concluir que la implementación del modelo en LabView estaba realizada correctamente.



Figura 3.26: Comparación Simulink y LabView.

3.6.2. Programa del MyRIO

El diagrama del modelo de Cobelli y Dalla Man se puso a prueba en LabView y luego se introdujo en un programa para el MyRIO que incluyera además la comunicación con el controlador que iba a hacerse en el Arduino. El diagrama completo del programa del MyRIO se tiene en la figura 3.27; en este programa se implementó la comunicación por medio del protocolo SPI, así como una salida digital que le indicara al Arduino el modo del controlador; también se implementó en el botón del MyRIO la señal impulso que indica la ingesta de comida.



Figura 3.27: Diagrama del programa principal en LabView para MyRio.

Se optó por la comunicación SPI porque en la mayoría de proyectos investigados relacionados con páncreas artificial, la comunicación entre sensor y controlador se da mediante un protocolo serial sin embargo, no hay un consenso entre cuál protocolo usar, y en este caso, el que mejor realizó su función fue el SPI.

En la figura 3.27 también se puede ver las operaciones hechas para transferir y recibir datos mediante SPI, que fueron básicamente convertir el dato de glucosa en un número entero y luego enviarlo como dos Bytes, para luego convertirlo en el Arduino en el número real correspondiente. Así como los bloques utilizados para transferir los datos a un documento de texto localizado en la memoria externa que fue colocada en el puerto USB del MyRIO.

Por otra parte, cada subsistema se configuró para que las ecuaciones diferenciales fueran resueltas mediante el algoritmo de Runge-Kutta, y que el paso de resolución sea de 1 s, lo que se traduciría a pasos de $1 \min$ en el sistema real. Para lograr una simulación continua, los subsistemas de simulación se colocaron dentro de un ciclo "while", el cual se ejecuta cada 100 ms.

3.7. Controlador en Arduino

Las ecuaciones de PID implementadas fueron las de dos grados de libertad, aunque se usaron los parámetros obtenidos de la herramienta de Matlab para un PID ideal, para evitar el golpe derivativo, además se limito la salida del controlador a $10pmol/(Kg \cdot min)$ para estar entre los límites de una bomba real, las ecuaciones utilizadas fueron las siguientes [25]:

$$\begin{cases}
P(k) = k_p(R - Y) \\
I(k) = I(k - 1) + \frac{k_p T_s}{T_i} (R(k - 1) - Y(k - 1)) \\
D(k) = \frac{a T_d D(k - 1)}{(a T_d + T_s)} - \frac{k_p T_d}{(a T_d + T_s)} (Y(k) - Y(k - 1))
\end{cases}$$
(3.2)

donde:

- P(k) (pmol/kg/min) Parte proporcional del PID.
- R (mg/kg) Valor deseado de glucosa en plasma.
- Y(k) (mg/dL) Medición de gluclosa en plasma.
- $k_p \ pmol/kg/min \ per \ mg/dl$ Ganancia proporcional.
- I(k) pmol/kg/min Parte integral del PID
- $T_i min$ Tiempo integral
- D(k) pmol/kg/min Parte derivativa del PID
- $T_d \min$ Tiempo derivativo
- a Filtro de la parte derivativa.
- $T_s min$ Tiempo de muestreo.
- I(k-1) pmol/kg/min Parte integral en el tiempo anterior.
- D(k-1) pmol/kg/min Parte derivativa en el tiempo anterior.

Las partes proporcional, integral y derivativa de (3.2) finalmente se sumaron para obtener la tasa de infusión de insulina como se ve en la siguiente ecuación:

$$IIR = P + I + D \tag{3.3}$$

Para lograr implementar el control se añadió el filtro de la parte derivativa a y se utilizaron los parámetros obtenidos en la sintonización con la herramienta de MATLAB.

A manera de comprobación se ejecutaron las siguientes instrucciones en MATLAB para obtener los parámetros del controlador, en estas, el archivo "PID_tool.mat" contiene los parámetros obtenidos con la herramienta de sintonización en un modelo tipo "zpk" de MATLAB llamado con el nombre "C", se puede ver que los parámetros obtenidos de la conversión son prácticamente los mismos que los de la tabla 3.1, debido a que para la conversión se utilizó un tiempo de muestreo de $T_s = 1$.

El único valor que no está incluido en la tabla 3.1 es el del filtro a, el cual es representado por su inverso N en el código siguiente:

>> Controlador = load ('PID_tool.mat'); >> Controlador_Discreto = c2d(C,1, 'tustin');>> Controlador_Discreto = pidstd (Controlador_Discreto) Controlador_Discreto = $Kp * (1 + \frac{1}{Ti} * \frac{Ts}{z-1} + Td * \frac{1}{(Td/N)+Ts/(z-1)})$ with Kp = -0.0304, Ti = 425, Td = 95.8, N = 60.1, Ts = 1

Para poder implementar el controlador en tiempo discreto en el Arduino se recurrió a la librería TaskScheduler, esta librería permite realizar varias tareas a una frecuencia regular cada una. Se aprovecharon las características de esta librería para ejecutar el algoritmo de PID cada 300 ms para emular un tiempo de muestreo de 3 min.

Aparte del controlador, en el código del Arduino también se implementó la comunicación SPI, esto se hizo mediante el uso de la librería "SPI", mediante la cuál se puede configurar la comunicación entre un maestro y varios esclavos, en este caso se configuro el Arduino como un esclavo ya que el maestro que le solicitará información será el MyRIO.

Para poder transferir los datos del controlador al MyRIO, se transformó la salida del controlador a un byte y se multiplicó por 20 para tener una resolución aproximada de 0,05, de nuevo, para ajustarse un valor utilizable en una bomba real.

3.8. Conexión del Sistema

Para conectar el sistema se utilizaron los canales SPI del MyRIO, así como del Arduino. Una de las ventajas proporcionadas por el MyRIO es que tiene incluida una resistencia de Pull-Up necesaria para establecer una comunicación SPI, por lo que no fue necesario incluir un resistor externo a la conexión.

Por otra parte se utilizó una salida digital del MyRIO para indicarle al Arduino si el controlador debía estar en modo manual o automático. Se debió tomar en cuenta que el MyRIO trabaja sus entradas y salidas lógicas a una tensión de 3,3 V mientras que el Arduino trabaja a 5 V sin embargo, en el Arduino aún se toma como positivo una señal de 3,3 V, y como esa señal digital sale del MyRIO hacia el Arduino, no se tuvo problema al implementar.

El diagrama con las conexiones hechas se tiene en la figura 3.28, en este diagrama se observa un costado del MyRIO, las conexiones se hicieron en el módulo A de pines del MyRIO. En la imagen 3.29 se ve como queda conectado el MyRIO con el Arduino para realizar la simulación del sistema.



Figura 3.28: Diagrama de conexiones entre el Arduino y el MyRIO.



Figura 3.29: Imagen del sistema implementado.

Capítulo 4

Resultados

4.1. Control del PID en el Arduino

Luego de implementar el modelo en el MyRIO y el controlador en el Arduino, se procedió a probar ambos dispositivos, en la figura 4.1 se tiene el panel principal de LabView desde donde se observan los datos del MyRIO y la entrada de insulina proveniente del controlador; además de observar las gráficas en tiempo real, se puede variar el modo del controlador de manual a automático y modificar el valor de la glucosa que se ingiere y el peso del sujeto de prueba, así como el valor de salida del controlador en modo manual.

En esta primera prueba, el sistema ya tenía 120 s corriendo así que so se observa el pico inicial del controlador, como sí se observó en las gráficas de Simulink, lo que sí se puede observar es la manera en que la tasa de infusión de insulina reacciona ante un pico de aparición glucosa ingerida, y al mismo tiempo, como se ven afectadas las variables de insulina y glucosa en plasma.

Los datos del MyRIO se extrajeron mediante el uso de una memoria externa USB para poderlos comparar con los resultados de Simulink, la herramienta utilizada para hacer los gráficos fue Matlab. En la figura 4.2, se tiene la respuesta del sistema y del controlador ante una perturbación, comenzando con un valor inicial de glucosa de 130 mg/dL y un valor deseado de 120 mg/dL, para una persona de 78 Kg que consume 45 g de glucosa alrededor del minuto 1200. Por otro lado, en la figura 4.3 se muestran los



Figura 4.1: Panel principal del programa del MyRIO.

resultados de la misma persona, pero esta vez considerando un valor inicial de glucosa de 180 mg/dL y dos perturbaciones, una de 45 g alrededor del minuto 3200 y otra de 70 g de glucosa a los 4800 min.

Los experimentos realizados, cuyos resultados obtenidos en las figuras 4.2 y 4.3, se hicieron con la intención de mostrar el comportamiento del sistema para observar la diferencia que se da con respecto al valor deseado y el tiempo de estabilización luego de una perturbación.



Figura 4.2: Sistema ante una perturbación partiendo de 130 mg/dL de glucosa inicial.



Figura 4.3: Sistema ante dos perturbación partiendo de 180 mg/dL de glucosa inicial.

Con respecto a la acción del controlador, se podría obtener una respuesta que tuviera menos fluctuación si se considerara un mayor resolución en la señal de insulina sin embargo, como se menciono anteriormente, se escogió una resolución de $\pm 0,05$ para acercar la simulación a una bomba de insulina real. En estos casos no se tuvo el pico en la magnitud de la acción del controlador al inicio de la simulación porque esta se inicio en modo manual, con un valor de $0,5pmol/(Kg \cdot min)$, luego de inmediatamente se cambió a automático y se siguió la simulación.

4.2. Comparación entre la implementación del My-RIO y la simulación en Simulink

Para poder hacer una comparación se graficaron los datos extraídos del MyRIO en la misma gráfica que los datos de la simulación de Simulink. Para hacer esto, se hizo primero la prueba en el MyRIO y posteriormente, el vector de datos que representa la tasa de aparición de glucosa (R_a) se ingresó en Simulink para realizar la simulación,
considerando también los mismos datos de peso y valor inicial de glucosa.

La primera comparación se puede ver en la figura 4.4, donde se probó un cambio del valor inicial de 180 mg/dL al valor deseado de 120 mg/dL. En este caso no hay datos de aparición de glucosa, por lo que solo se puede observar el tiempo que tarda en alcanzar el valor deseado, siendo este tiempo de aproximadamente 1500 min; este tiempo se consideró muy grande, pero considerando los niveles de glucosa nunca cayeron a niveles de hipoglucemia, a menos de 80 mg/dL, se considera que se hace un control adecuado en esta prueba.

Por otra parte, el objetivo principal de esta primera prueba era comprobar su similitud con la simulación de Simulink, para verificar que tanto la simulación en MATLAB, como la de MyRIO tuvieran las mismas reacciones.



Figura 4.4: Tiempo en alcanzar el valor deseado partiendo de 180 mg/dL de glucosa inicial.

La segunda comparación se realizó con perturbaciones, estas se hicieron para emular las tres comidas, tal como se hizo para la figura 3.11, primero una ingesta de 4,5 gde glucosa aproximadamente a los 480 min, una de 7 g a los 720 min; para luego finalizar con una de 7 g a los 1200 para una simulación de un día con tres comidas. Esta comparación se puede ver en la figura 4.5, y se puede notar la similitud entre las señales de Simulink y del MyRIO.

Además de la similitud de los resultados del MyRIO y Simulink, se puede ver que la simulación también es comparable con la referencia de la figura 3.10, con excepciones en el pico inicial de la acción de control y los tiempos de respuesta, esto porque se utilizo un controlador un poco más agresivo para mejorar dichos tiempos sin embargo,

son similares en la forma de la señal y en que ambos evitan en la mayoría del tiempo entrar en estados de hipoglucemia o hipoglucemia, ya que a solo en breves momentos se superó la marca de 180 mg/dL (en verde) y nunca se bajo de 80 mg/dL (en rojo). Para tratar de mejorar el pico inicial se varió el filtro derivativo, sin embargo, como no se obtuvo diferencia significativa con el cambio, se optó por dejarlo en un valor de a = 0,1.



Figura 4.5: Sistema ante tres perturbaciones partiendo de 130 mg/dL de glucosa inicial.

Con las comparaciones de las figuras 4.4 y 4.5 se obtuvo lo esperado ya que la simulación y la implementación del MyRIO con el Arduino como controlador se comportan de manera muy similar, con esto se concluyó que el modelo estaba correctamente implementado en el MyRIO y el controlador en el Arduino.

Capítulo 5

Conclusiones y recomendaciones

5.1. Conclusiones

Con la realización del presente proyecto se pudo analizar diferentes modelos del sistema glucosa-insulina, cada uno de ellos toma un enfoque diferente para modelar el sistema, y generalmente se hacen suposiciones de acuerdo a si se modela a partir de una base de datos de un estudio clínico o a partir de reglas físicas. Para la simulación realizada se utilizó un modelo basado en datos experimentales y de una complejidad suficiente para considerarse realista, aún así se estudiaron otros modelos aceptables que no se utilizaron para dar prioridad a la complejidad del modelo escogido.

Se comprendió que la utilidad de los modelos depende en gran parte a la utilización que se le fuera a dar, como el objetivo del proyecto fue crear un sistema para probar algoritmos de control que eventualmente funcionaran en un páncreas artificial, se logró encontrar y utilizar un modelo que considerara a la glucosa ingerida de manera oral, pasando por el sistema digestivo; y además que la insulina ingresara mediante el espacio subcutáneo, con sus retrasos respectivos.

El sensor y la bomba de insulina tienen diferentes efectos sobre la medición de glucosa y la inyección de insulina, dependiendo de la tecnología utilizada sin embargo, se pudieron simular los efectos más generales que son: el retraso de medición y el tiempo de muestreo en el caso del sensor, y el retraso en la llegada de insulina en el caso de la bomba.

Los algoritmos de control utilizados en este tipo de sistemas son diversos y están en

constante evolución, se utilizó PID porque todavía es tomado en cuenta en estudios de este tipo de simulaciones, pero se concluyó de acuerdo con la investigación bibliográfica hecha que no es el más eficiente para controlar el sistema, lo cuál fue comprobado a la hora de realizar la simulación y observar tiempos de reacción considerablemente largos, aún así el controlador logró mantener el sistema fuera de límites peligrosos. Como el objetivo del proyecto fue la implementación de la simulación y no encontrar el control más optimo, se considera que se logró el objetivo.

El dispositivo de National Instruments MyRIO resultó ser una herramienta eficiente para simular el sistema en tiempo real, con la facilidad de poder programarse desde LabView, y con sus entradas y salidas integradas. Aunque no se utilizaron todas las utilidades que presenta el dispositivo, para el presente proyecto fue de gran utilidad sin embargo, tiene la desventaja de ocupar una licencia de LabView para aprovechar sus facilidades.

Por otra parte el dispositivo tipo Arduino utilizado, a pesar ser menos potente a nivel de procesador y no tener tantas características tiene la ventaja de que no requiere ninguna licencia para ser utilizado, y sus capacidades fueron suficientes para manejar el control PID implementado.

5.2. Recomendaciones

A pesar de que se logró llevar a cabo el sistema planteado al principio del presente proyecto, la idea principal fue desde el comienzo implementar un sistema que sirviera de base para proyectos futuros, en caso de que se continúe con el tema, se tiene las siguientes recomendaciones para ampliar y mejorar el sistema:

- Ampliar el modelo para incluir el glucagón como segunda hormona de control de los niveles de insulina en el plasma sanguíneo. Se tiene conocimiento de que el modelo utilizado para el presente proyecto ha sido modificado para admitir el glucagón sin embargo, no se incluyó en la simulación implementada, esto sería una mejora para volverlo más realista y poder probar algoritmos de control que funciones con las dos hormonas como salidas.
- El algoritmo de control utilizado para probar la implementación fue el PID, el cual se consideró lo suficientemente apropiado para evaluar la validez de la simulación

sin embargo, la idea del sistema es que se puedan probar diferentes esquemas de control; en este apartado se recomienda probar otras estrategias de control más avanzadas para lograr mejores rendimientos a la hora de controlar el sistema. Se puede probar el control MPC, el cuál se ha visto que tiene mejor rendimiento a la hora de controlar este sistema o un control más avanzado.

- Para analizar los efectos de la bomba y del sensor, se consideraron los efectos más generales de estos sobre la insulina y la medición de glucosa respectiva, además se usaron parámetros que generalizaban ambos elementos, se podrían contemplar los efectos para sensores y bombas específicos disponibles comercialmente, aplicando un estudio más a fondo del funcionamiento de ambos elementos, se podría conseguir una simulación más real.
- Por la manera en que se planteó la simulación, las ecuaciones que representan el sistema glucosa-insulina, el sensor y la bomba se encuentran en un solo dispositivo (MyRIO), se puede añadir más realismo a la simulación si se separaran estos elementos y se explorara como es la comunicación entre ellos en la realidad, y como esto puede afectar en las interacciones entre los tres elementos.
- Por su conveniencia y disponibilidad se programó el sistema en el dispositivo My-RIO mediante el uso de LabView, se puede considerar trasladar la programación a una plataforma de código abierto para mayor disponibilidad de uso.

Bibliografía

- Richard N Bergman, Lawrence S Phillips y Claudio Cobelli. «Physiologic evaluation of factors controlling glucose tolerance in man. Measurement of insulin sensitivity and β-cell glucose sensitivity from the response to intravenous glucose». En: Journal of Clinical Investigation 68.6 (1981), págs. 1456-1467. ISSN: 00219738. DOI: 10.1172/JCI110398.
- Fernando D. Bianchi y col. «Invalidation and low-order model set for artificial pancreas robust control design». En: *Journal of Process Control* 76 (abr. de 2019), págs. 133-140. ISSN: 09591524. DOI: 10.1016/j.jprocont.2019.02.004.
- Graham Brooker. «The artificial pancreas». En: Handbook of Biomechatronics. Elsevier, 2018, págs. 405-456. ISBN: 9780128125397. DOI: 10.1016/B978-0-12-812539-7.00015-5. URL: https://linkinghub.elsevier.com/ retrieve/pii/B9780128125397000155.
- [4] Ali Cinar y Kamuran Turksoy. Advances in Artificial Pancreas Systems: Adaptive and Multivariable Predictive Control. Springer Nature, 2018. ISBN: 9783319722443. URL: http://www.springer.com/series/10280.
- C. Cobelli y col. «An integrated mathematical model of the dynamics of blood glucose and its hormonal control». En: *Mathematical Biosciences* 58.1 (1982), págs. 27-60. ISSN: 0025-5564. DOI: https://doi.org/10.1016/0025-5564(82)90050-5. URL: http://www.sciencedirect.com/science/article/pii/0025556482900505.
- [6] Marco Cupelli y col. «Hardware In the Loop Implementation and Challenges».
 En: Modern Control of DC-Based Power Systems. Elsevier, 2018, págs. 249-259.
 DOI: 10.1016/b978-0-12-813220-3.00007-7.
- [7] Chiara Dalla Man, Michael Camilleri y Claudio Cobelli. «A system model of oral glucose absorption: Validation on gold standard data». En: *IEEE Transactions*

on Biomedical Engineering 53.12 (2006), págs. 2472-2478. ISSN: 00189294. DOI: 10.1109/TBME.2006.883792.

- [8] Chiara Dalla Man y Claudio Cobelli. «Glucose Modelling». En: Modeling Methodology for Physiology and Medicine: Second Edition (2013), págs. 355-379. DOI: 10.1016/B978-0-12-411557-6.00016-1.
- [9] Chiara Dalla Man, Robert A. Rizza y Claudio Cobelli. «Meal simulation model of the glucose-insulin system». En: *IEEE Transactions on Biomedical Engineering* 54.10 (oct. de 2007), págs. 1740-1749. ISSN: 00189294. DOI: 10.1109/TBME. 2007.893506.
- [10] Chiara Dalla Man y col. «GIM, simulation software of meal glucose Insulin model». En: Journal of Diabetes Science and Technology 1.3 (2007), págs. 323-330.
 ISSN: 19322968. DOI: 10.1177/193229680700100303. URL: www.journalofdst. org.
- [11] Andrea De Gaetano y Ovide Arino. «Mathematical modelling of the intravenous glucose tolerance test». En: Journal of Mathematical Biology 40.2 (2000), págs. 136-168. ISSN: 1432-1416. DOI: 10.1007/s002850050007. URL: https: //doi.org/10.1007/s002850050007.
- [12] Giuseppe De Nicolao y col. «Modeling and control of diabetes: Towards the Artificial Pancreas». En: *IFAC Proceedings Volumes (IFAC-PapersOnline)*. Vol. 44.
 1 PART 1. IFAC Secretariat, 2011, págs. 7092-7101. ISBN: 9783902661937. DOI: 10.3182/20110828-6-IT-1002.03036.
- [13] Qiang Fang, Lei Yu y Peng Li. «A new insulin-glucose metabolic model of type 1 diabetes mellitus: An in silico study». En: Computer Methods and Programs in Biomedicine 120.1 (jun. de 2015), págs. 16-26. ISSN: 18727565. DOI: 10.1016/j.cmpb.2015.03.009. URL: https://www.sciencedirect.com/science/article/abs/pii/S0169260715000759?via%7B%5C%%7D3Dihub.
- [14] Esben Friis-Jensen. «Modeling and Simulation of Glucose-Insulin Metabolism». Tesis doct. Technical University of Denmark, 2007. ISBN: 9783527311439. DOI: 10.1002/9783527610600.ch3. URL: www.imm.dtu.dk.
- [15] Roman Hovorka y col. «Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes». En: *Physiological Measurement* 25.4 (2004), págs. 905-920. ISSN: 09673334. DOI: 10.1088/0967-3334/25/4/010.

- [16] Roman Hovorka y col. «Partitioning glucose distribution/transport, disposal, and endogenous production during IVGTT». En: American Journal of Physiology
 - Endocrinology and Metabolism 282.5 45-5 (2002). ISSN: 01931849. DOI: 10. 1152/ajpendo.00304.2001.
- [17] Ruiqiang Hu y Chengwei Li. «The design of an intelligent insulin pump». En: Proceedings of 2015 4th International Conference on Computer Science and Net- work Technology, ICCSNT 2015. Institute of Electrical y Electronics Engineers Inc., jun. de 2016, págs. 736-739. ISBN: 9781467381727. DOI: 10.1109/ICCSNT. 2015.7490848.
- [18] Lauren M. Huyett y col. «Glucose Sensor Dynamics and the Artificial Pancreas: The Impact of Lag on Sensor Measurement and Controller Performance». En: *IEEE Control Systems* 38.1 (feb. de 2018), págs. 30-46. ISSN: 1066033X. DOI: 10.1109/MCS.2017.2766322.
- [19] Moustafa M.A. Ibrahim y col. «The integrated glucose insulin minimal model: An improved version». En: European Journal of Pharmaceutical Sciences 134 (jun. de 2019), págs. 7-19. ISSN: 18790720. DOI: 10.1016/j.ejps.2019.04.010.
- [20] Boris Kovatchev. A Century of Diabetes Technology: Signals, Models, and Artificial Pancreas Control. Jul. de 2019. DOI: 10.1016/j.tem.2019.04.008.
- Boris P Kovatchev y col. «In silico preclinical trials: A proof of concept in closed-loop control of type 1 diabetes». En: Journal of Diabetes Science and Technology. Vol. 3. 1. 2009, págs. 44-55. DOI: 10.1177/193229680900300106. URL: www.journalofdst.org.
- [22] Ritu Lakhtakia. «The History of Diabetes Mellitus». En: SQU Mediacl Journal 13.3 (ene. de 2013), págs. 367-370. ISSN: 00281298. DOI: 10.1007/BF01831214.
- [23] Aidan O'Dwyer. HANDBOOK OF PI AND PID CONTROLLER TUNING RU-LES. Imperial College Press, 2009.
- [24] S. Olma y col. «Model-based Method for the Accuracy Analysis of Hardware-inthe-Loop Test Rigs for Mechatronic Vehicle Axles». En: *Procedia Technology* 26 (2016), págs. 105-112. ISSN: 22120173. DOI: 10.1016/j.protcy.2016.08. 015.
- [25] Luisa Quesada. Diseño e implementación de un sistema hardware-in-the-loop para la regulación glucosa-insulina de pacientes con diabetes mellitus tipo 1. 2018.
- [26] M J Quon y col. «Non-insulin-mediated glucose disappearance in subjects with IDDM. Discordance between experimental results and minimal model analysis.»

eng. En: *Diabetes* 43.7 (jul. de 1994), págs. 890-896. ISSN: 0012-1797 (Print). DOI: 10.2337/diab.43.7.890.

- [27] Satyam Raikwar y col. «Hardware-in-the-Loop test automation of embedded systems for agricultural tractors». En: *Measurement: Journal of the International Measurement Confederation* 133 (feb. de 2019), págs. 271-280. ISSN: 02632241. DOI: 10.1016/j.measurement.2018.10.014.
- [28] Organización Mundial de la Salud. Global Report on Diabetes. Online. 2017.
- [29] Pouria Sarhadi y Samereh Yousefpour. «State of the art: hardware in the loop modeling and simulation with its applications in design, development and implementation of system and control software». En: International Journal of Dynamics and Control 3.4 (dic. de 2015), págs. 470-479. ISSN: 21952698. DOI: 10. 1007/s40435-014-0108-3.
- [30] Øyvind Stavdahl y col. «The Artificial Pancreas: A Dynamic Challenge». En: IFAC-PapersOnLine 49.7 (2016), págs. 765-772. ISSN: 24058963. DOI: 10.1016/ j.ifacol.2016.07.280.
- [31] Malgorzata E. Wilinska y Roman Hovorka. Simulation models for in silico testing of closed-loop glucose controllers in type 1 diabetes. Dic. de 2008. DOI: 10.1016/j.ddmod.2009.07.005.
- [32] Zhi Xu y col. «An integrated intelligent insulin pump». En: 2006 7th International Conference on Electronics Packaging Technology, ICEPT '06. IEEE Computer Society, 2006. ISBN: 142440620X. DOI: 10.1109/ICEPT.2006.359761.

Apéndice A

Cálculo de Parámetros en Matlab

```
G_b=185; %(mg/dL) Concentracion de glucosa en plama
VG=1.88; %(dL/Kg) Distribucion de volumen de glucosa
Gp_b=G_b*VG; %(mg/Kg) Masa de glucosa en plasma
I_b=40.95; %(pmol/L) Concentracion de insulina en plasma
IIR_b=1; %(pmol/Kg/min) Tasa base de infusion de insulina
  exoqena
EGP_b=2.4; %(mg/kg/min) Produccion endogena de glucosa
k1=0.065; %(1/min) Parametro de proporcion
k2=0.079; %(1/min) Parametro de proporcion
V1=0.05; %(L/Kg) Distribucion de volumen de insulina
kp2=0.0021; %(1/min) Efectividad de glucosa en el higado
kp3=0.009; %(mg/Kg/min por pmol/L)Amplitud de la accion de
  insulina en el higado
ki=0.0079; %(1/min) Delay entre la senal de la insulina y la
  accion de esta
Fcns=1; %(mg/Kg/min) Utilizacion de glucosa en cerebro y
  eritrocitos
Vm0=2.50; %(mg/Kg/min)
Vmx=0.047; %(mg/Kg/min por pmol/L)
km0=225.59; %(mg/Kg)
p2u=0.0331; %(1/min)
```

```
K=2.30; %(pmol/Kg por mg/dL)
alfa=0.05; %(1/min)
beta=0.11; %(pmol/Kg/min por mg/dL)
gama=0.5; %(1/min)
ke1=0.0005; %(1/min)
ke2=339; %(mg/Kg)
kmax=0.0558; %(1/min)
kmin=0.0080; %(1/min)
kabs=0.057; %(1/min)
kgri=0.0558; %(1/min)
f=0.9;
a=0.00013; %(1/mg)
b=0.82;
c=0.00236; %(1/mg)
d=0.01;
m1=0.190; %(1/min)
m2=0.484; %(1/min)
m4=0.194; %(1/min)
m5=0.0304; %(min*Kg/pmol)
m6=0.6471;
HEb=0.6;
kp1=EGP_b+kp2*Gp_b+kp3*I_b; %(mg/Kg/min)
Gt_b=(Fcns-EGP_b+(k1*Gp_b))/k2;
m3=m6*m1/(1-m6);
%m3=HEb*m1/(1−HEb);
Ip_b=IIR_b/((m2+m4)-((m1+m2)/(m1+m3)));
Il_b=Ip_b*(m2/(m1+m3));
kd=0.0164; %(1/min)
ka1=0.0018; %(1/min)
ka2=0.0182; %(1/min)
Isc1_ss=IIR_b/(kd+ka1); %(pmol/Kg)
Isc2_ss=Isc1_ss*(kd/ka2); %pmol/Kg
```

72

Apéndice B

Código del Controlador

```
#include "Arduino.h"
#include <SPI.h> //Carga libreria SPI
#include <math.h> //Carga libreria math
#include <TaskScheduler.h> //Se carga la libreria del
  Scheduler
Scheduler RealTimeCore; //Esto crea un objeto del tipo
  Scheduler
byte Buf [4]; //Buffer para recibir los datas desde el MyRio
volatile byte posicion, respuesta; //Posicion del buffer y
  dato a enviar de respuesta (salida del PID)
volatile boolean fin; //Boolean para procesar el vector de
  informacion de entrada.
int centenas, unidades, ref; //Variables para guardar datos
  recibidos.
double glucosa;
                         //Variable para guardar el dato
  real de glucosa
double Ts =3;
                               // TIEMPO DE MUESTREO
```

//Parametros del controlador
double a = 0.1; //Filtro derivativo

```
double kp = -0.0304; //Ganancia Proporcional
double Ti = 424; //Tiempo Integral
double Td = 95.9; //Tiempo Derivativo
//Parametros PID
double P, I, D = 0.0;
//Entrada a la planta
double U = 0.0;
double U manual =0.0;
//Valores anteriores
double IO, RO, DO = 0.0;
double Y0=0.0;
int modo, s_modo = 0;
//Valor deseado y salida
double R;
double Y=0.0;
int out;
// Se definen los prototipos de funciones
void controlFun();
void comunicacionFun();
// Se crean las tareas:
Task control(300, TASK_FOREVER, &controlFun, &RealTimeCore);
  //Tarea de control
//Tarea de comuncacion, actualizada mas rapido que la de
  control
Task comunicacion(50, TASK_FOREVER, &comunicacionFun, &
  RealTimeCore);
```

// Definicion de las funciones

```
void comunicacionFun() {
  //Enviar la salida al registro SPDR
   SPDR=out;
}
void controlFun() {
  //Lectura modo manual o automatico
  s_modo = digitalRead(2);
  if (s_modo == HIGH) {
    modo = 1; //Manual
  }
  else if (s_modo == LOW) {
    modo = 0; //Automatico
  }
  switch (modo) {
    case 1:
    // Control automatico
    R = (double) ref;
    //Serial.print (R);
    //Serial.print (' ');
    //Lectura de salida de la planta
    Y = glucosa;
   // Serial.print (Y);
   // Serial.print (' ');
    //Parte proporcional del PID
    P = kp \star (R-Y);
    //Parte integral del PID
    I = I0 + kp*Ts/Ti*(R0-Y0);
        if (I>10){
    I = 10;
    }
    if (U<0){
    I = 0;
```

```
}
 //Parte derivativa del PID
 D = a * Td/(a * Td+Ts) * D0 - kp * Td/(a * Td+Ts) * (Y-Y0);
 //Senal de control
 U = (P+I+D);
 //Limites de la senal
 if (U>10){
 U=10; I = I0; D = D0;
 }
 if (U<0){
 U=0; I = I0; D = D0;
 }
 //Conversion de la senal a Byte
 out=(byte)(round(U*20));
// Serial.println (U);
 //Tiempos pasados
 IO = I; RO = R; YO = Y; DO = D; //tiempo0=tiempo;
 break;
 case 0:
 //Control manual
 R = (double) ref;
// Serial.print (R);
// Serial.print (' ');
 //Lectura de salida de la planta
 Y = glucosa;
//Cambio de manial a automatico
 I = U; // Integral inicial diferente de 0
 YO = Y; // Salida actualizada
```

```
out=(byte) (round(U_manual*20));
    Serial.println (U);
    break;
}
}
void setup() {
  SPDR=0;
  // Activa el pin de comunicacion en modo esclavo para SPI
  pinMode(MISO, OUTPUT);
  // Configura en SPI en modo esclavo
  SPCR |= _BV(SPE);
  // Variables para procesar la informacion recibida
  posicion = 0;
  fin = false;
  // Interrupcion para establecer la comunicacion
  SPI.attachInterrupt();
  pinMode(2, INPUT); //Lectura modo del controlador
 // Serial.begin(9600); //se inicia la comunicacion serial a
    9600 bauds
  RealTimeCore.init();//Inicializa el scheduler
  RealTimeCore.addTask(control); //Se agrega la tarea 01 al
     scheduler
  RealTimeCore.addTask(comunicacion); //Se agrega la tarea 01
      al scheduler
  control.enable(); //Tarea a ejecutar
  comunicacion.enable(); //Tarea a ejecutar
}
```

```
// Rutina de interrupcion SPI
ISR (SPI_STC_vect)
{
byte c = SPDR; // Guarda el byte del registro SPI
int tamano Buf = sizeof Buf;
  // Anade el byte al buffer de informacion
  if (posicion < tamano_Buf)</pre>
    {
   Buf [posicion] = c;
    posicion=posicion+1;
   //Si se alcanza el fin del buffer, se detiene la rutina
    if (posicion == tamano_Buf)
      fin = true;
    }
}
void loop() {
  // Cuando se recibe la senal de que la rutina de
    comunicacion termino,
  // se procesa la informacion de glucosa y el valor de
    referencia
  // actual.
    if (fin)
    {
    unidades=Buf[0];
    centenas=Buf[1];
    glucosa = (double)centenas*10+(double)unidades/10;
    ref=Buf[2];
    U_manual=(double)Buf[3]/100;
    //Se reinicia el buffer de informacion
    Buf [posicion] = 0;
```

```
posicion = 0;
   //Se deja en el Byte de registro SPI
   SPDR=respuesta;
   fin = false;
   }
RealTimeCore.execute();
}
```