



FACULTAD DE INGENIERÍA

DEPARTAMENTO DE ELECTROTECNIA

Tesis de grado:

**Plataforma de simulación de pacientes diabéticos para validación in silico de estrategias de control glucémico**

Autor: Nicolás Rosales

---

Directores:  
Hernán De Battista  
Fabricio Garelli



*Dedicado a  
mi familia y amigos*



## *Contemplar*

En el frágil seno de un verde asedio  
se alza la institución que clausura,  
cincela todo de su hermosura;  
inocencia viste, su fin, un tedio.

Bandadas migran hacia el gris cemento  
erigiendo un laberinto de rostros,  
olvidando el pretérito de otros:  
la poesía del negro firmamento

El asombro gradualmente perdido  
deja un foso en inquietos corazones  
que se llena de pérfidas razones,  
la trinidad, monedas y coronas,  
son grilletes que con tus pies adorna.  
Comienza la pesquisa del olvido.

Nicolás Rosales



# Agradecimientos

Gracias a los que me tuvieron paciencia, a los que me pasaron unos mates, a los que me distrajerón, a los que me hicieron trabajar. En particular a Hernán De Battista, Fabricio Garelli, Marina Costantini, Patricio Colmegna y Juan Jose Gagliardino por su tiempo y ayuda.





# Resumen

¿Por qué un trabajo final de ingeniería electrónica con fines medicinales? Porque no deja de maravillarme el funcionamiento de los organismos vivos, más aún uno tan complejo como el nuestro. En este trabajo pude contemplar la punta del iceberg de una pequeña sección del metabolismo del cuerpo humano. A esta inquietud se le agrega un importante compromiso social, la capacidad de cooperar en la ayuda de millones de personas con la enfermedad crónica diabetes mellitus. Quizá no se obtenga una solución inmediata, pero lograr una plataforma abierta para el desarrollo de tratamientos de lazo abierto y cerrado en pacientes con este padecer resulta satisfactorio. Pues en eso consiste este trabajo, desarrollar una plataforma capaz de simular una cohorte de pacientes virtuales para validar tratamientos en base a bombas de infusión subcutánea de insulina, en particular con controladores de lazo cerrado, in silico. Esto involucra el estudio de diferentes modelos matemáticos del sistema glucosa-insulina, de plataformas de desarrollo posibles, de estructuras de implementación, de la capacidad de realizar simulaciones de escenarios reales, y de obtener una herramienta práctica, versátil y útil.



# Índice general

<b>Agradecimientos</b>	<b>V</b>
<b>Resumen</b>	<b>VII</b>
<b>Lista de figuras</b>	<b>XIII</b>
<b>Lista de tablas</b>	<b>XIII</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Marco teórico del sistema endocrino pancreático . . . . .	1
1.1.1. Secreción de insulina . . . . .	1
1.1.2. Acción de la insulina . . . . .	1
1.1.3. Regulación . . . . .	2
1.2. Diabetes Mellitus . . . . .	3
1.2.1. Tratamientos . . . . .	3
1.3. Motivación . . . . .	3
1.4. Objetivos . . . . .	4
<b>2. Estudio de modelos y plataformas</b>	<b>5</b>
2.1. Modelo Sorensen . . . . .	5
2.2. Modelo minimal de Bergman . . . . .	9
2.3. Modelo Hovorka/Cambridge . . . . .	10
2.3.1. Acción de la insulina . . . . .	10
2.3.2. Glucosa . . . . .	11
2.3.3. Cinética y absorción de la insulina subcutánea . . . . .	11
2.3.4. Absorción del intestino . . . . .	12
2.3.5. Glucosa intersticial . . . . .	12
2.4. Modelo Dalla Man . . . . .	13
2.4.1. Glucosa . . . . .	13
2.4.2. Insulina . . . . .	14
2.4.3. Producción endógena de glucosa . . . . .	16
2.4.4. Tasa de aparición de la glucosa . . . . .	16
2.4.5. Utilización de la glucosa . . . . .	17
2.4.6. Secreción de insulina . . . . .	18
2.4.7. Cinética de la insulina subcutánea . . . . .	19
2.4.8. Excreción renal de glucosa . . . . .	19
2.5. Comparación de modelos . . . . .	20
2.6. Selección del modelo . . . . .	20
2.7. Plataformas de simulación . . . . .	23
2.7.1. Simulador UVa . . . . .	23

2.8. Selección de plataforma . . . . .	24
<b>3. Desarrollo de la plataforma de simulación</b>	<b>25</b>
3.1. Plataforma de simulación . . . . .	25
3.1.1. Bloques . . . . .	26
3.1.2. Escenarios . . . . .	27
3.1.3. Simulink . . . . .	27
3.1.4. Resultados de la simulación . . . . .	31
3.2. Simulación realista . . . . .	31
3.2.1. Comidas Mixtas . . . . .	31
3.2.2. Variación intra-paciente . . . . .	31
3.3. Programación . . . . .	32
3.3.1. Sujetos . . . . .	32
3.3.2. Carga del escenario . . . . .	33
3.3.3. Creación del vector de comidas . . . . .	34
3.3.4. Creación de comidas mixtas . . . . .	35
3.3.5. Creación de los vectores de suministro de insulina . . . . .	35
3.3.6. Creación de los vectores intravenosos . . . . .	35
3.3.7. Carga de Hardware . . . . .	36
3.3.8. Carga de parámetros del sujeto . . . . .	37
3.3.9. Creación de bolos óptimos según el paciente . . . . .	37
3.3.10. Condiciones iniciales . . . . .	38
3.3.11. Configuración de la simulación en Simulink . . . . .	38
3.3.12. Guardado . . . . .	38
3.3.13. Gráficos . . . . .	38
<b>4. Simulaciones y resultados</b>	<b>41</b>
4.1. Simulaciones . . . . .	41
4.1.1. Simulación N°1: Paciente adulto a lazo abierto . . . . .	41
4.1.2. Simulación N°2 . . . . .	44
4.1.3. Simulación N°3 . . . . .	45
4.1.4. Simulación N°4 . . . . .	46
4.2. Discusión de resultados . . . . .	48
<b>5. Conclusiones y trabajo futuro</b>	<b>51</b>
<b>A. Códigos de programación</b>	<b>53</b>
A.1. Script principal . . . . .	53
A.2. S-Function . . . . .	55
A.3. Sujetos . . . . .	57
A.4. Cargar escenario . . . . .	57
A.5. Creación de las comidas de escenario . . . . .	60
A.6. Creación de las comidas mixtas . . . . .	61
A.7. Creación de los bolos de escenario . . . . .	61
A.8. Creación de los suministros intravenosos de escenario . . . . .	61
A.9. Carga del hardware seleccionado . . . . .	62
A.10. Carga de los parámetros de los pacientes a simular . . . . .	63
A.11. Creación de bolo óptimo por paciente . . . . .	64
A.12. Cálculo de condiciones iniciales . . . . .	64
A.13. Configuración de la simulación . . . . .	64

A.14.Crear ruido . . . . .	65
A.15.Guardado de variables . . . . .	65
A.16.Gráficos . . . . .	66



# Índice de figuras

1.1. Secreción de la insulina . . . . .	2
2.1. Diagrama del sistema glucosa-insulina de Dalla Man. Las líneas sólidas representan los flujos de glucosa e insulina, y las líneas punteadas las señales de control. . . . .	14
2.2. Subsistema que representa a la glucosa . . . . .	15
2.3. Subsistema que representa a la insulina . . . . .	15
3.1. Sistema completo . . . . .	29
3.2. Subsistemas . . . . .	30
3.3. Donde se puede seleccionar pacientes a simular y configurar los parámetros deseados. . . . .	33
3.4. Ejemplo de escenario de simulación . . . . .	34
3.5. Ejemplo de suministro de comidas . . . . .	35
3.6. Menú de comidas mixtas . . . . .	36
3.7. Ejemplo de tasa de aparición de comidas por escenario y mixtas . . . . .	37
3.8. Comparación de señales con y sin hardware . . . . .	39
3.9. Selección de gráficos . . . . .	40
3.10. a) y b) Ejemplo de simulación de un paciente adulto. Se simularon tres comidas en un periodo de 24 horas, visualizándose la glucosa en plasma en mg/dl y las zonas clínicas en que se encontró el paciente en cada hora. c) y d) Mismo escenario sobre dos pacientes adultos. . . . .	40
4.1. Controlador empleado en el simulador . . . . .	42
4.2. Comparación de la tasa de aparición de glucosa del modelo de Dalla Man y de las comidas mixtas . . . . .	42
4.3. Caso 4.1.1a . . . . .	43
4.4. Caso 4.1.1b . . . . .	43
4.5. Caso 4.1.1c . . . . .	44
4.6. Caso 4.1.2a . . . . .	44
4.7. Caso 4.1.2b . . . . .	45
4.8. Caso 4.1.2c . . . . .	45
4.9. Caso 4.1.3a . . . . .	46
4.10. Caso 4.1.3b . . . . .	46
4.11. Caso 4.1.3c . . . . .	47
4.12. Caso 4.1.4a . . . . .	47
4.13. Caso 4.1.4b . . . . .	48
4.14. Caso 4.1.4c . . . . .	48





# Índice de cuadros

2.1. Tabla de comparación de modelos . . . . .	22
--	----

# Capítulo 1

## Introducción

En este capítulo se presentará un pequeño marco teórico del sistema glucosa-insulina del cuerpo humano, las consideraciones, motivaciones y objetivos para hacer este trabajo.

### 1.1. Marco teórico del sistema endocrino pancreático

Con fines de adentrarse en el marco teórico de la fisiología del sistema glucosa-insulina, se leyó acerca del sistema endocrino pancreático (ver [14]).

#### 1.1.1. Secreción de insulina

La insulina es producida y secretada por las células beta de los islotes de Langerhans del páncreas. La secreción basal de insulina depende del nivel de glucosa en plasma, no obstante en condiciones normales esta secreción puede cifrarse en 0,5 a 1  $\mu\text{U}/\text{hora}$  (U es una unidad de medida de la cantidad de una sustancia, basada en su actividad biológica, su equivalente en SI es 1  $\mu\text{U}/\text{ml} = 6,945 \text{ pmol}/\text{l}$ ). El nivel medio de insulina se cifra en 10  $\mu\text{U}/\text{ml}$ , nivel que baja en un 10% en condiciones de ayunas o durante el ejercicio prolongado, y que puede subir a 100  $\mu\text{U}/\text{ml}$  después de una comida normal. No obstante, estos valores son medidos en la circulación periférica; en la vena portal donde se vierte todo el contenido hormonal debe ser mucho mayor, ya que el hígado tiene la capacidad de metabolizar el 50% de insulina que recibe.

La secreción de insulina sube dentro de los 30 a 60 minutos después de una comida, aunque este incremento está sujeto a un ritmo circadiano de forma que, ante el suministro de aportes idénticos de calorías y nutrientes en distintos momentos del nictámero (i.e. el espacio de tiempo que comprende un día y una noche, es decir 24 horas), el mayor pico de secreción se obtiene por la mañana temprano.

Algo importante a tener en cuenta es que la secreción de insulina es oscilatoria, con un periodo de 3 a 6 minutos de máximos y mínimos que al parecer son importantes para evitar la regulación a la baja de los receptores de insulina por parte de las células objetivo y favorecer la extracción hepática de la hormona.

#### 1.1.2. Acción de la insulina

Exceptuando el cerebro, las gónadas, hematíes, túbulos renales, mucosa gastrointestinal y músculo esquelético en movimiento, el resto de los tejidos es sensible a la acción insulínica, aunque el efecto más potente se observe en el hígado, músculo esquelético en reposo y tejido adiposo, por ser éstos los tejidos destino del almacenamiento de glucosa, aminoácidos y ácidos grasos libres. Reduce los niveles de glucosa en sangre por el estímulo a la captación

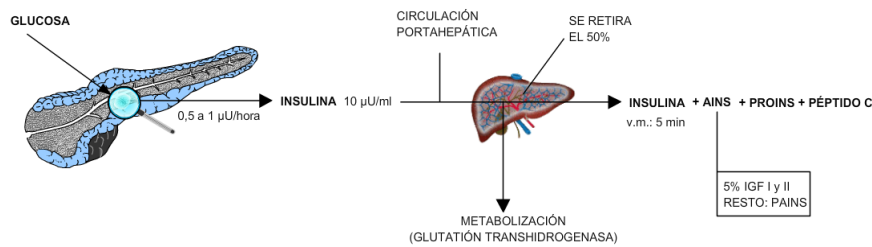


Figura 1.1: Secreción de la insulina

de glucosa que provoca sobre los tejidos, excepto en el hígado, donde además inhibe su producción y liberación. En los tejidos extra-hepáticos favorece el transporte de glucosa. En el músculo esquelético y cardíaco en reposo, deriva la glucosa hacia la formación de glucógeno y parte de la glucosa se utiliza como fuente para la síntesis de aminoácidos y proteínas.

### 1.1.3. Regulación

Son diversos los mecanismos reguladores que afectan a la síntesis y secreción de insulina, se clasifican en:

#### Metabólicos

Los reguladores metabólicos son los principales, siendo la glucosa el regulador fundamental, la cual actúa como estímulo directo sobre las células  $\beta$  del páncreas. La reacción de las células  $\beta$  a los niveles plasmáticos de glucosa es rapidísima. A valores de glucosa por debajo de 60 mg/dl la respuesta de insulina tiene un valor basal estable, pero a partir de este valor la respuesta de insulina crece de forma sigmoidea alcanzándose el 100 % a valores de glucosa de 300 mg/dl. Cuando la glucosa sube su nivel, rápidamente se produce un pico de secreción de insulina seguido de una caída (debido a la liberación de la insulina almacenada) y posteriormente, si sigue alta la glucosa, un nuevo pico mantenido de insulina debido a un estímulo directo de la síntesis y secreción de insulina (secreción bifásica de insulina).

#### Nerviosa

La estimulación parasimpática determina un estímulo en la síntesis y secreción, mientras que la estimulación simpática ejerce un efecto inhibitorio sobre la acción estimulante de la glucosa.

#### Hormonal

Las hormonas gastrointestinales secretina, gastrina y CCC tienen un efecto estimulador aunque a sus concentraciones fisiológicas parece que no ejercen un efecto importante. Sin embargo el GIP (péptido inhibitorio gástrico) y el GLP1 (péptido similar al glucagón 1) secretados por las neuronas entéricas, a sus concentraciones fisiológicas ejercen un potente efecto estimulador de la insulina, aunque su efecto requiere la presencia de glucosa. Además estos péptidos se liberan por estímulo de la glucosa, aminoácidos y ácidos grasos en el intestino delgado. Por este mecanismo se prepara el organismo para recibir una oleada de metabolitos que se derivan hacia los lugares de almacenamiento (hígado, músculo y tejido

adiposo). Explica además el hecho de que la glucosa administrada por vía oral determina una mayor respuesta insulínica que si se administra por vía intravenosa.

## 1.2. Diabetes Mellitus

La diabetes mellitus (DM) es un conjunto de trastornos metabólicos, que afecta a diferentes órganos y tejidos, y se caracteriza por un aumento de los niveles de glucosa en la sangre: hiperglucemia. Hay dos tipos de diabetes que conciernen a este trabajo: la diabetes mellitus tipo 1, en la que existe una destrucción total de las células  $\beta$  de los islotes de Largenhans, lo que conlleva una deficiencia absoluta de insulina; la diabetes mellitus tipo 2, generada como consecuencia de un defecto progresivo en la secreción de insulina, así como el antecedente de resistencia periférica a la misma. La causan varios trastornos, siendo el principal la baja producción de la hormona insulina, secretada por las células  $\beta$  de los Islotes de Langerhans del páncreas endocrino, o por su inadecuado uso por parte del cuerpo, que repercutirá en el metabolismo de los hidratos de carbono, lípidos y proteínas. La diabetes mellitus y su comorbilidad (enfermedades adicionales causadas como efecto adverso) constituyen una importante causa de preocupación en salud pública. Hay millones de diabéticos diagnosticados en el mundo, y el número de personas afectadas por la enfermedad ha venido creciendo sostenidamente en los últimos años, generando enormes costos sanitarios.

### 1.2.1. Tratamientos

El tratamiento de la diabetes mellitus tiene como objetivo mantener los niveles de glucosa en sangre dentro de la normo-glucemia, para minimizar las complicaciones y el riesgo de la enfermedad. Principalmente se basa en la dieta, el ejercicio físico y la medicación.

En pacientes con diabetes tipo I es necesaria la administración exógena de insulina ya que el páncreas es incapaz de producir esta hormona. La insulina se administra a través de inyecciones en la grasa existente debajo de la piel del brazo, ya que si se tomase por vía oral sería destruida en el aparato digestivo antes de pasar al flujo sanguíneo. Las necesidades de insulina varían en función de los alimentos que se ingieren y de la actividad física que se realiza. Las personas que siguen una dieta estable y una actividad física regular varían poco sus dosis de insulina. Sin embargo, cualquier cambio en la dieta habitual o la realización de algún deporte exigen modificaciones de las pautas de insulina. La insulina puede inyectarse a través de distintos dispositivos, en este caso particular, interesa el tratamiento por bombas de insulina. Éstas son dispositivos portátiles que administran insulina de acción rápida las 24 horas del día. De pequeño tamaño y no invasivas (a comparación con el suministro intravenoso), las bombas de insulina administran la insulina a través de un tubo pequeño (catéter) y una cánula (denominados equipo de infusión) que se implanta bajo la piel. La cantidad de insulina administrada se puede ajustar para satisfacer las necesidades de cada paciente.

## 1.3. Motivación

La diabetes es un problema demandante en la salud actual, tanto por su crecimiento en número de pacientes, como en los gastos económicos que produce.

El control automático de los niveles de glucosa en sangre en pacientes diabéticos ha sido un objetivo permanente desde la fabricación de los primeros monitores continuos de glucosa y bombas de insulina. Uno de los mayores desafíos para el mejoramiento de las

terapias actuales y la calidad de vida de los pacientes es automatizar la regulación de glucosa mediante un sistema de control que conecte de manera inteligente un monitor de glucosa con una bomba de insulina. Para el diseño de los algoritmos de control es necesario un modelo que describa la dinámica subyacente. De la misma manera, varios ensayos *in silico* deben ser efectuados para verificar la efectividad de este algoritmo antes de realizar pruebas clínicas en pacientes. Con un control automático de la glucosa en sangre para pacientes insulino dependientes se puede mejorar su calidad de vida y reducir las complicaciones asociadas con esta enfermedad

## 1.4. Objetivos

Siendo muy amplio el camino a recorrer, en esta tesis de grado se propone:

- Comparar los principales modelos matemáticos propuestos por diferentes entidades, analizando y presentando sus ventajas y desventajas.
- Seleccionar el modelo adecuado de la dinámica glucosa-insulina, de la dinámica subcutánea-intravenosa y de las dinámicas del monitor de glucosa y de la bomba de insulina.
- Analizar y seleccionar una plataforma adecuada para la implementación.
- Plantear un simulador sobre la base del ya desarrollado Simulador UVa, de forma didáctica y configurable para el usuario.
- Crear una simulación más realista agregando variaciones intra-paciente de la dinámica glucosa-insulina y un menú de comidas mixtas con distintas tasas de aparición de glucosa.
- Poder evaluar tratamientos de lazo abierto y cerrado en una cohorte de pacientes virtuales *in silico*.

## Capítulo 2

# Estudio de modelos y plataformas

La dinámica glucosa-insulina ha sido extensamente estudiada. Varios modelos matemáticos basados en ecuaciones diferenciales ordinarias (ODE) han sido utilizados para simulaciones y diseño de controladores.

Los pacientes virtuales realizados mediante modelos matemáticos de la absorción subcutánea de la insulina, la digestión y absorción de carbohidratos, y el control de la insulina sobre la utilización y producción endógena de la glucosa, entre otros, son usados como sustitutos de pacientes reales para la validación *in silico* de controladores. Su objetivo es realizar pruebas preclínicas de estrategias de control para los estudios del páncreas artificial.

A continuación se detallarán los principales modelos surgidos y usados para propósitos orientados al control (según lo estudiado en [1], [2], [3], [4], [5], [6], [7], [8] y [11]).

### 2.1. Modelo Sorensen

El modelo matemático de Sorensen representa el mecanismo fisiológico del metabolismo de la glucosa, dividiendo al cuerpo en seis compartimentos: (1) el cerebro, representando al sistema nervioso central; (2) el corazón y el pulmón, que representan los volúmenes de mezcla rápida del corazón, los pulmones y las arterias; (3) el intestino; (4) el hígado; (5) los riñones; y (6) la periferia, que incluye el músculo esquelético y tejido adiposo. Los subsistemas de glucosa, glucagón y la insulina son considerados por separado, interactuando a través de efectos metabólicos. Con el fin de obtener una representación matemática, un balance de masas es realizado en cada compartimento fisiológico. Como consecuencia, se obtienen 12 ecuaciones diferenciales no lineales ordinarias para la dinámica de la glucosa y del glucagón (tres asociadas a variables adimensionales), siete ecuaciones lineales para la insulina y 44 parámetros. La linealidad del modelo de la insulina es debido a que se considera la diabetes tipo I. Esto no solo induce linealidad, sino que desacopla la dinámica de la insulina de otras. Los valores de los parámetros se derivan de la literatura y por lo tanto, sólo representan un sujeto nominal promedio. Como todos los parámetros de este modelo son invariantes en el tiempo, el modelo no puede representar la variabilidad intra-paciente.

#### Variables del modelo

- $A$  [adimensional]: estado auxiliar
- $F$  [I,N,L/min, adimensional]: despeje fraccional

- $G$  [mg/dL]: concentración de glucosa
- $I$  [mU/L]: concentración de insulina
- $N$  [normalizado, adimensional]: concentración de glucagón
- $Q$  [L/min]: tasa de flujo de plasma vascular
- $q$  [dL/min]: tasa del flujo de la sangre vascular
- $T$  [min]: constante de tiempo de difusión transcápilar
- $V$  [L]: volumen
- $v$  [dL]: volumen
- $\Gamma$  [mg/min o mU/min]: fuente metabólica o tasa de caída

### índices

- |   |   |
|---|---|
| ▪ A: arteria hepática                   | ▪ LC: limpieza del hígado   |
| ▪ B: cerebro                            | ▪ N: glucagón   |
| ▪ BU: absorción del cerebro             | ▪ NHGP: efecto del glucagón sobre HGP   |
| ▪ C: espacio capilar                    | ▪ P: periferia (músculo/tejido adiposo)   |
| ▪ G: glucosa                            | ▪ PC: limpieza periférica   |
| ▪ H: corazón y pulmones                 | ▪ PGU: uso periférico de glucosa  |
| ▪ HGP: producción hepática de glucosa   | ▪ PIR: secreción de insulina pancreática  |
| ▪ HGU: absorción hepática de glucosa    | ▪ PNC: limpieza pancreática de glucagón   |
| ▪ I: insulina                           | ▪ PNR: secreción pancreática de glucagón  |
| ▪ IHGP: efecto de la insulina sobre HGP | ▪ RBCU: uso por los glóbulos rojos  |
| ▪ IHGU: efecto de la insulina sobre HGU | ▪ S: estómago/intestino   |
| ▪ IVI: infusión intravenosa de insulina | ▪ SIA: absorción insulínica hacia el flujo de sangre desde el depósito subcutáneo |
| ▪ K: riñón                              | ▪ SU: toma por el estómago/intestino  |
| ▪ KC: limpieza renal                    | ▪ T: tejido   |
| ▪ KE: excreción renal                   |   |
| ▪ L: hígado                             |   |

### Ecuaciones

#### Dinámica de la glucosa

$$\dot{G}_B^C = (G_H^C - G_B^C) \frac{q_B}{v_B^C} - (G_B^C - G_B^T) \frac{v_B^T}{T_B v_B^C} \quad (2.1)$$

$$\dot{G}_B^T = (G_B^C - G_B^T) \frac{1}{T_B} - \frac{\Gamma_{BU}}{v_B^T} \quad (2.2)$$

$$\dot{G}_H^C = (G_B^C q_B + G_L^C q_L + G_K^C q_K + G_P^C q_P - G_H^C q_H - \Gamma_{RBCU}) \frac{1}{v_H^C} \quad (2.3)$$

$$\dot{G}_S^C = (G_H^C - G_S^C) \frac{q_S}{v_S^C} + \frac{\Gamma_{meal}}{v_S^C} - \frac{\Gamma_{SU}}{v_S^C} \quad (2.4)$$

$$\dot{G}_L^C = (G_H^C q_A + G_S^C q_S - G_L^C q_L) \frac{1}{v_L^C} + \frac{\Gamma_{HGP}}{v_L^C} - \frac{\Gamma_{HGU}}{v_L^C} \quad (2.5)$$

$$\dot{G}_K^C = (G_H^C - G_K^C) \frac{q_K}{v_K^C} - \frac{\Gamma_{KE}}{v_K^C} \quad (2.6)$$

$$\dot{G}_P^C = (G_H^C - G_P^C) \frac{q_P}{v_P^C} + (G_P^T - G_P^C) \frac{v_P^T}{T_P^G v_P^C} \quad (2.7)$$

$$\dot{G}_P^T = (G_P^C - G_P^T) \frac{1}{T_P^G} - \frac{\Gamma_{PGU}}{v_P^T} \quad (2.8)$$

### Dinámica de la insulina

$$\dot{I}_B^C = (I_H^C - I_B^C) \frac{Q_B}{V_B^C} \quad (2.9)$$

$$\dot{I}_H^C = (I_B^C Q_B + I_L^C Q_L + I_K^C Q_K + I_P^C Q_P - I_H^C Q_H + \Gamma_{IVI}) \frac{1}{V_H^C} \quad (2.10)$$

$$\dot{I}_S^C = (I_H^C - I_S^C) \frac{Q_S}{V_S^C} \quad (2.11)$$

$$\dot{I}_L^C = (I_H^C Q_A + I_S^C Q_S - I_L^C Q_L) \frac{1}{V_L^C} + \frac{\Gamma_{PIR}}{V_L^C} - \frac{\Gamma_{LC}}{V_L^C} \quad (2.12)$$

$$\dot{I}_K^C = (I_H^C - I_K^C) \frac{Q_K}{V_K^C} - \frac{\Gamma_{KC}}{V_K^C} \quad (2.13)$$

$$\dot{I}_P^C = (I_H^C - I_P^C) \frac{Q_P}{V_P^C} + (I_P^T - I_P^C) \frac{V_P^T}{T_P^I V_P^C} \quad (2.14)$$

$$\dot{I}_P^T = (I_P^C - I_P^T) \frac{1}{T_P^I} + \frac{\Gamma_{SIA}}{V_P^T} - \frac{\Gamma_{PC}}{V_P^T} \quad (2.15)$$



## Glucagón

$$\dot{N} = (\Gamma_{PNR} - N) \frac{F_{PNC}}{V_N} \quad (2.16)$$

Ecuaciones auxiliares

$$\dot{A}_{IHGP} = \frac{1}{25} \left\{ 1,2088 - 1,138 \tanh \left[ 1,1669 \left( \frac{I_L^C}{21,43} - 0,8885 \right) \right] - A_{IHGP} \right\} \quad (2.17)$$

$$\dot{A}_{NHGP} = \frac{1}{65} \left[ \frac{2,7 \tanh(0,388N - 1)}{2} - A - NHGP \right] \quad (2.18)$$

$$\dot{A}_{IHGU} = \frac{1}{25} \left[ 2 \tanh \left( 0,549 \frac{I_L^C}{21,43} \right) - A_{IHGU} \right] \quad (2.19)$$

## Parámetros $\Gamma_i$

Los parámetros  $\Gamma_i$  que aparecen son los siguientes:  $\Gamma_{BU} = 70$ ,  $\Gamma_{RBCU} = 10$ ,  $\Gamma_{SU} = 20$ ,  $\Gamma_{PIR} = 0$ ,  $\Gamma_{LC} = FLC(I_H^C Q_A + I_S^C Q_S + \Gamma_{PIR})$  y

$$\Gamma_{HGU} = 20A_{IHGU} \left\{ 5,6648 + 5,6589 \tanh \left[ 2,4375 \left( \frac{G_L^C}{101} \right) \right] \right\} \quad (2.20)$$

$$\Gamma_{HGP} = 155A_{IHGP} [2,7 \tanh(0,388N) - A_{NHGP}] \times \left\{ 1,425 - 1,406 \tanh \left[ 0,1699 \left( \frac{G_L^C}{101} - 0,4969 \right) \right] \right\} \quad (2.21)$$

$$\Gamma_{PGU} = \frac{35G_P^T}{86,81} \left\{ 7,035 + 6,5163 \tanh \left[ 0,33827 \left( \frac{I_P^T}{5,304} - 5,82113 \right) \right] \right\} \quad (2.22)$$

$$\Gamma_{PNR} = \left\{ 1,3102 - 0,61016 \tanh \left[ 1,5071 \left( \frac{I_H^C}{15,15} - 0,46981 \right) \right] \right\} \times \left\{ 2,9285 - 2,095 \tanh \left[ 4,18 \left( \frac{G_H^C}{91,89} - 0,36191 \right) \right] \right\} \quad (2.23)$$

$$\Gamma_{PC} = \frac{I_P^T}{(1 - F_{PC}) / (Q_P F_{PC}) - T_P^I / V_P^T} \quad (2.24)$$

$$\Gamma_{KC} = F_{KC} I_H^C Q_K \quad (2.25)$$

$$\Gamma_{KE} = \begin{cases} 71 \{ 1 + \tanh[0,11(G_K^C - 460)] \} & \text{si } G_K^C < 460 \text{mg/dl} \\ 0,872 G_K^C - 330 & \text{si } G_K^C \geq 460 \text{mg/dl} \end{cases} \quad (2.26)$$

## 2.2. Modelo minimal de Bergman

Este modelo se basa en tres factores, que cumplen un rol importante en la disposición de la glucosa, estos son:

- Sensibilidad insulínica: la capacidad de la insulina de incrementar la disposición de glucosa en los músculos, hígado y tejido adiposo
- Efectividad de la glucosa: la habilidad de la glucosa para mejorar su propia disposición a niveles basales de insulina
- Respuesta pancreática: la habilidad de las células  $\beta$  de segregar insulina ante el estímulo de la glucosa

El modelo está basado en un IVGTT (Intravenous Glucose Tolerance Test), que consiste en administrar una dosis de glucosa (generalmente 0,3 gr de glucosa por kg de peso corporal) de forma intravenosa durante un periodo de 60 segundos a sujetos en ayunas desde la noche a la mañana, y luego se muestrean las concentraciones en plasma de glucosa e insulina durante un periodo de 180 minutos. Las dinámicas en el modelo son descriptas por dos componentes, donde los parámetros fueron estimados por separado dentro de cada uno.

### Variables

- $G(t)$  [mg/gl]: concentración de glucosa en plasma
- $I(t)$  [ $\mu$ U/ml]: concentración de insulina en plasma
- $X(t)$  [ $\text{min}^{-1}$ ]: efecto de la insulina en la desaparición de la glucosa

### Parámetros

- $G_b$  [mg/dl]: nivel basal de glucosa
- $I_b$  [ $\mu$ U/ml]: nivel basal de insulina
- $p_1$  [ $\text{min}^{-1}$ ]: tasa constante del consumo insulina-independiente de glucosa de los músculos, hígado y tejido adiposo
- $p_2$  [ $\text{min}^{-1}$ ]: tasa de disminución de la capacidad de captación de glucosa de los tejidos
- $p_3$  [ $\text{min}^{-2}(\mu\text{U/ml})^{-1}$ ]: incremento de la capacidad de consumo insulina-dependiente de glucosa en los tejidos por unidad de concentración por encima de  $I_b$
- $n$  [ $\text{min}^{-1}$ ]: tasa de descomposición de primer grado para la insulina en plasma
- $h$  [mg/dl]: valor umbral de glucosa encima del cual las células  $\beta$  segregar insulina
- $\gamma$  [ $\mu\text{U/ml min}^{-2}(\text{mg/dl})^{-1}$ ]: tasa de secreción de insulina después de la inyección de glucosa y por encima de la concentración  $h$
- $G_0$  [mg/dl]: concentración teórica de glucosa en plasma a tiempo 0
- $I_0$  [ $\mu\text{U/ml}$ ]: concentración teórica de insulina en plasma a tiempo 0

## Ecuaciones

$$\begin{aligned}
 \dot{G}(t) &= -p_1(G(t) - G_b(t) - X(t)G(t)) & G(0) &= G_0 \\
 \dot{X}(t) &= -p_2X(t) + p_3(I(t) - I_b) & X(0) &= 0 \\
 \dot{I}(t) &= -n(I(t) - I_b) + \gamma(G(t) - h)^+ & I(0) &= I_0
 \end{aligned} \tag{2.1}$$

Donde  $t=0$  es el tiempo de inyección de glucosa y  $+$  indica reflexión positiva

## 2.3. Modelo Hovorka/Cambridge

Este es un modelo compartimental de la cinética de la glucosa y la acción de la insulina, el cual incluye dos submodelos compartimentales de la cinética de la insulina y glucosa subcutánea, así como otro submodelo de dos compartimentos para representar la absorción de la glucosa en el tracto gastrointestinal. Una importante propiedad de este modelo es su capacidad para representar la variabilidad intra- e inter-paciente. La variabilidad inter-paciente es representada por una población de 18 sujetos virtuales con diabetes tipo I, donde los parámetros se obtuvieron a partir de estudios clínicos en pacientes con dicha enfermedad o de distribuciones de probabilidad poblacionales. La variabilidad intra-paciente se llevó a cabo mediante la superposición de oscilaciones sinusoidales en un subconjunto de parámetros del modelo. Un punto débil del paciente virtual de este modelo, es la simple representación de la absorción del intestino, además de que se podría mejorar significativamente la variabilidad intra-paciente. El modelo de simulación de Hovorka está siendo utilizado en el desarrollo de un prototipo de controlador de glucosa para niños y adolescentes con diabetes tipo I en un proyecto de la JDRF (Juvenile Diabetes Research Foundation).

### 2.3.1. Acción de la insulina

#### Variables

- $x_1(t)$ ,  $x_2(t)$  y  $x_3(t)$ : son el efecto remoto de la insulina sobre la distribución, disposición y producción endógena de la glucosa
- $I(t)$ : concentración de insulina en plasma

#### Parámetros

- $k_{ai}$ : tasa de activación
- $k_{bi}$ : tasa de desactivación
- $S_{IT} = k_{a1}/K_{b1}$ ,  $S_{ID} = k_{a2}/k_{b2}$  y  $S_{IE} = k_{a3}/k_{b3}$ : sensibilidades de la insulina para el transporte, distribución y la producción de glucosa

## Ecuaciones

$$\dot{x}_1(t) = -k_{a1}x_1(t) + S_{IT}k_{b1}I(t) \tag{2.1}$$

$$\dot{x}_2(t) = -k_{a2}x_2(t) + S_{ID}k_{b2}I(t) \tag{2.2}$$

$$\dot{x}_3(t) = -k_{a3}x_3(t) + S_{IE}k_{b3}I(t) \tag{2.3}$$

### 2.3.2. Glucosa

#### Variables

- $Q_1(t)$  y  $Q_2(t)$ : masas de glucosa en el compartimento accesible y no accesible
- $G(t)$ : concentración de glucosa en plasma
- $EGP$ : producción endógena de glucosa
- $F_{01}^C$ : flujo de glucosa no dependiente de la insulina
- $F_R$ : limpieza renal del glucosa

#### Parámetros

- $k_{12}$ : tasa de transferencia del compartimento accesible al inaccesible
- $V_G$ : representa la distribución de glucosa en el compartimento accesible
- $R_{thr}$ : umbral de glucosa para la limpieza renal

#### Ecuaciones

$$\dot{Q}_1(t) = - \left[ \frac{F_{01}^C}{V_G G(t)} + x_1(t) \right] Q_1(t) + k_{12} Q_2(t) - F_R + EGP(t) + U_G(t) \quad (2.4)$$

$$\dot{Q}_2(t) = x_1(t) Q_1(t) - [k_{12} + x_2(t)] Q_2(t) \quad (2.5)$$

$$G(t) = \frac{Q_1(t)}{V_G} \quad (2.6)$$

$$EGP(t) = \begin{cases} EGP_0[1 - x_3(t)] & EGP \geq 0 \\ 0 & EGP < 0 \end{cases} \quad (2.7)$$

$$F_{01}^C = \frac{F_{01}^S G}{G + 1} \quad (2.8)$$

$$F_R(t) = \begin{cases} R_{cl}(G - R_{thr})V_G & G \geq R_{thr} \\ 0 & G < R_{thr} \end{cases} \quad (2.9)$$

### 2.3.3. Cinética y absorción de la insulina subcutánea

#### Variables

- $S_1(t)$  y  $S_2$ : masa de insulina en el compartimento accesible y no accesible
- $u(t)$ : bolo administrado de insulina
- $I(t)$ : concentración de insulina en plasma

**Parámetros**

- $k_a$ : tasa de absorción de insulina
- $k_e$ : tasa de eliminación parcial del plasma
- $V_I$ : volumen de distribución de la insulina

**Ecuaciones**

$$\dot{S}_1(t) = u(t) - k_a S_1(t) \quad (2.10)$$

$$\dot{S}_2(t) = k_a S_1(t) - k_a S_2(t) \quad (2.11)$$

$$I(t) = \frac{K_a S_2(t)}{V_I} - k_e I(t) \quad (2.12)$$

**2.3.4. Absorción del intestino****Variables**

- $G_1(t)$  y  $G_2(t)$ : masa de glucosa en el compartimento accesible y no accesible
- $D(t)$ : cantidad de carbohidratos consumidos en el tiempo  $t$
- $U_G(t)$ : tasa de absorción del intestino
- $t_{max}$ : tiempo máximo de tasa de aparición de glucosa en el compartimento accesible

**Parámetros**

- $Bio$ : biodisponibilidad de los carbohidratos de la comida
- $U_{G\_ceil}$ : máximo flujo de glucosa desde el intestino derivado de  $U \sim [0, 02; 0, 035]$

**Ecuaciones**

$$\dot{G}_1(t) = -\frac{G_1(t)}{t_{max}} + Bio \cdot D(t) \quad G_1(0) = 0 \quad (2.13)$$

$$\dot{G}_2(t) = \frac{G_1(t)}{t_{max}} - \frac{G_2(t)}{t_{max}} \quad (2.14)$$

$$U_G = \frac{G_2(t)}{t_{max}} \quad (2.15)$$

$$t_{max} = \begin{cases} t_{max\_ceil} & U_G > U_{G\_ceil} \\ T_{max} & U_G \leq U_{G\_ceil} \end{cases} \quad (2.16)$$

**2.3.5. Glucosa intersticial****Variables**

- $C(t)$ : concentración de glucosa en el tejido subcutáneo

### Parámetros

- $k_{a\_int}$ : tasa de transferencia

### Ecuaciones

$$\dot{C}(t) = k_{a\_int}(G_C)(t) \quad (2.17)$$

## 2.4. Modelo Dalla Man

El grupo de Cobelli en Padova, Italia, desarrolló un modelo de simulación de comida del sistema glucosa-insulina utilizando datos tomados de 204 sujetos normales que se sometieron a un protocolo con trazadores de glucosa. Esto permitió calcular los flujos de glucosa y e insulina durante una comida. El modelo consta de dos subsistemas principales, descritos por dos compartimentos cada uno, siendo éstos los subsistemas glucosa e insulina. A éstos los acompañan varios modelos de procesos unitarios, identificados a partir de información promedio utilizando forcing function strategy y se estimaron 35 parámetros de un sujeto normal. La misma estrategia se aplicó a una base de datos más pequeña que contiene 14 sujetos con diabetes tipo II y se estimó el mismo número de parámetros que en un paciente sano. Se ha empleado este modelo para simular un día típico con tres comidas de un sujeto normal, teniéndose en cuenta la variación de la sensibilidad insulínica y la respuesta de las células  $\beta$ . Una de las mayores novedades de este modelo es una descripción más detallada del tránsito de la glucosa a través del tracto gastrointestinal, pero posee como debilidad el hecho que las variaciones diurnas de determinados parámetros no han sido modeladas. A pesar de que el modelo original fue identificado usando información de sujetos normales, se utiliza para simular diabéticos tipo I sustituyendo la secreción de insulina por un modelo de la cinética de la insulina subcutánea. Para considerar el mayor nivel de glucosa basal en la diabetes tipo I, se aumentó la producción endógena de glucosa. El simulador uVA desarrollado por la universidad a partir de este modelo fue aceptado por la FDA (Food and Drug Administration) como sustituto de los ensayos en animales para la experimentación pre-clínica de controladores en lazo cerrado.

### 2.4.1. Glucosa

Modelo de dos compartimentos (glucosa en plasma y en tejido adiposo) que describe la cinética de la glucosa.

#### VARIABLES

- $G_p$  [mg/kg]: masa de glucosa en plasma y en tejido de equilibrio rápido
- $G_t$  [mg/kg]: masa de glucosa en tejido de equilibrio lento
- $G$  [mg/dl]: concentración de glucosa en plasma
- $EGP$  [mg/kg/min]: producción endógena de glucosa (hígado)
- $Ra$  [mg/kg/min]: índice de aparición de glucosa en plasma
- $E$  [mg/kg/min]: excreción renal
- $U_{ii}$  [mg/kg/min]: utilización de glucosa independiente de la insulina

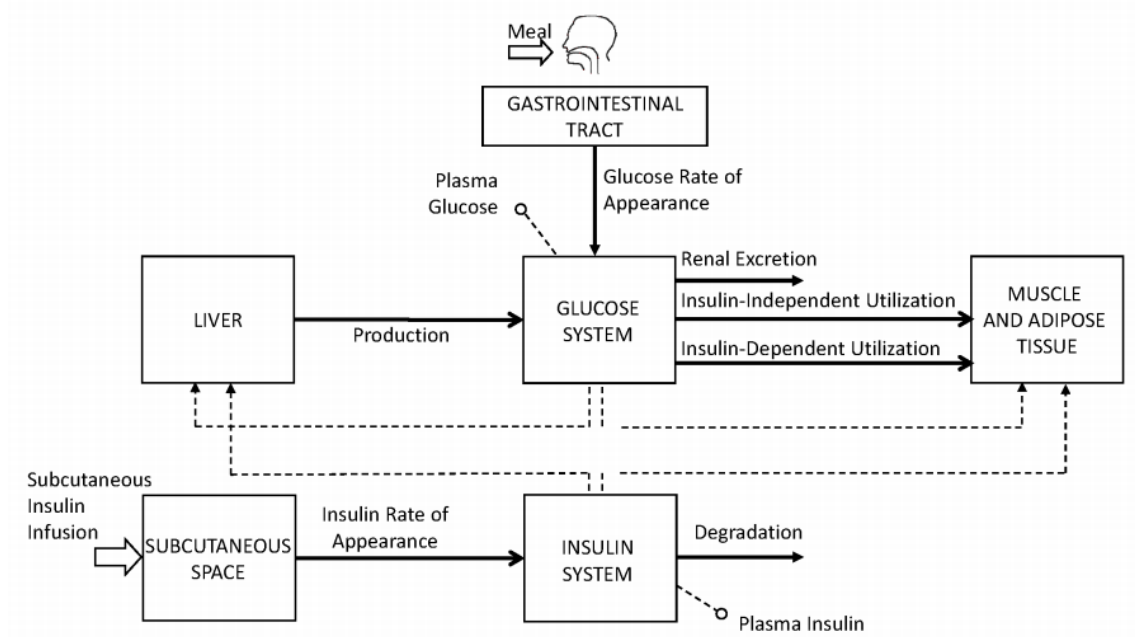


Figura 2.1: Diagrama del sistema glucosa-insulina de Dalla Man. Las líneas sólidas representan los flujos de glucosa e insulina, y las líneas punteadas las señales de control.

- $U_{id}$  [mg/kg/min]: utilización de glucosa dependiente de la insulina

(el sufijo b indica estado basal)

### Parámetros

- $V_g$  [dl/kg]: volumen de distribución de la glucosa
- $k_1, k_2$  [ $\text{min}^{-1}$ ]: parámetros de velocidad de la cinética de la glucosa

### Ecuaciones

$$\begin{cases} \dot{G}_p(t) = EGP(t) + Ra(t) - U_{ii}(t) - E(t) - k_1 \cdot G_p(t) + k_2 \cdot G_t(t) & G_p(0) = G_{pb} \\ \dot{G}_t(t) = -U_{id}(t) + k_1 \cdot G_p(t) - k_2 \cdot G_t(t) & G_t(0) = G_{tb} \\ G(t) = \frac{G_p}{V_G} & G(0) = G_b \end{cases} \quad (2.1)$$

#### 2.4.2. Insulina

Modelo de dos compartimentos (insulina en plasma y en hígado) que describe la cinética de la insulina.

#### Variabes

- $I$  [pmol/l]: concentración de insulina en plasma
- $I_p$  [pmol/kg]: masa de insulina en plasma

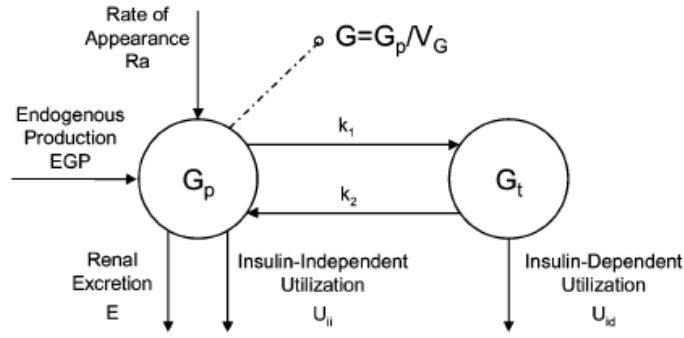


Figura 2.2: Subsistema que representa a la glucosa

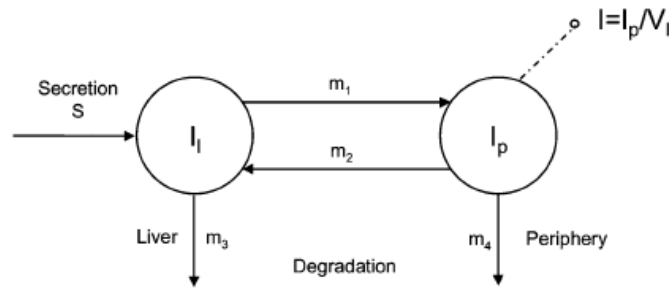


Figura 2.3: Subsistema que representa a la insulina

- $I_l$  [pmol/kg]: masa de insulina en hígado
- $S$  [pmol/kg/min]: secreción de insulina

### Parámetros

- $m_1, m_2, m_3$  [ $\text{min}^{-1}$ ]: parámetros de velocidad de la cinética de la insulina
- $m_4$  [ $\text{min}^{-1}$ ]: parámetro debido a la degradación periférica, asumida como lineal
- $V_I$  [l/kg]: distribución del volumen de insulina

### Ecuaciones

$$\begin{cases} \dot{I}_l(t) = -(m_1 + m_3(t)) \cdot I_l(t) + m_2 \cdot I_p(t) + S(t) & I_l(0) = I_{lb} \\ \dot{I}_p(t) = -(m_2 + m_4) \cdot I_p(t) + m_1 \cdot I_l(t) & I_p(0) = I_{pb} \\ I(t) = \frac{I_p}{V_I} & I(0) = I_b \end{cases} \quad (2.2)$$

La extracción hepática  $HE(t)$  se vincula con la secreción de insulina, y hay que tener en cuenta que el hígado es responsable de la absorción del 60% de la insulina

$$HE(t) = -m_5 \cdot S(t) + m_6 \quad HE(0) = HE_b \quad (2.3)$$

$$m_3(t) = \frac{HE(t) \cdot m_1}{1 - HE(t)} \quad HE(0) = HE_b \quad (2.4)$$



### 2.4.3. Producción endógena de glucosa

Glucosa que produce el hígado en función de la glucosa y la insulina en plasma

#### Variables

- $I_{po}$  [pmol/kg]: cantidad de insulina en la vena Porta (hígado)
- $I_d$  [pmol/l]: señal de la insulina retardada realizada con una cadena de dos compartimentos

#### Parámetros

- $k_{p1}$  [mg/Kg/min]: EGP extrapolada a glucosa e insulina cero
- $k_{p2}$  [ $\text{min}^{-1}$ ]: efectividad de la glucosa del hígado
- $k_{p3}$  [mg/Kg/min per pmol/liter]: parámetro que gobierna la amplitud de la acción de la insulina en el hígado
- $k_{p4}$  [mg/kg/min / (pmol/kg)]: parámetro que gobierna la amplitud de la acción de la insulina en la vena porta en el hígado
- $k_i$  [ $\text{min}^{-1}$ ]: parámetro de tasa que cuantifica retraso entre la señal de la insulina y la acción de la insulina

#### Ecuaciones

$$\begin{aligned} EGP(t) &= k_{p1} - k_{p2} \cdot G_p(t) - k_{p3} \cdot I_d(t) - k_{p4} \cdot I_{po}(t) \\ EGP(0) &= EGP_b \end{aligned} \quad (2.5)$$

$$\begin{cases} \dot{I}_1(t) = -k_i \cdot [I_1(t) - I(t)] & I_1(0) = I_b \\ \dot{I}_d(t) = -k_i \cdot [I_d(t) - I_1(t)] & I_d(0) = I_b \end{cases} \quad (2.6)$$

### 2.4.4. Tasa de aparición de la glucosa

Describe la transición de la glucosa en el estómago y el intestino, donde el estómago está representado por dos compartimentos (fase sólida  $Q_{sto1}$  y triturada  $Q_{sto2}$ ), mientras que un solo compartimento describe al intestino ( $Q_{gut}$ ).

#### Variables

- $Q_{sto}$  [mg]: cantidad de glucosa en el estómago (fase sólida,  $Q_{sto1}$  y líquida  $Q_{sto2}$ )
- $Q_{gut}$  [mg]: masa de glucosa en el intestino
- $Ra$  [mg/kg/min]: tasa de aparición de glucosa en plasma

### Parámetros

- $k_{gri}$  [ $\text{min}^{-1}$ ]: coeficiente de grado de “molido” que relaciona los primeros compartimientos
- $k_{empt}(Q_{sto})$  [ $\text{min}^{-1}$ ]: tasa de vacío gástrico hacia el intestino (función no lineal de  $Q_{sto}$ )
- $k_{abs}$  [ $\text{min}^{-1}$ ]: tasa de absorción intestinal
- $f$ : es la fracción de absorción intestinal que aparece en plasma
- $BW$  [kg]: peso del cuerpo (body weight)
- $D$  [mg]: cantidad de glucosa ingerida
- Resto de los parámetros arreglados según literatura o estimaciones

### Ecuaciones

$$\begin{cases} Q_{sto}(t) = Q_{sto1}(t) + Q_{sto2}(t) & Q_{sto}(0) = 0 \\ \dot{Q}_{sto1}(t) = -k_{gri} \cdot Q_{sto1}(t) + D \cdot d(t) & Q_{sto1}(0) = 0 \\ \dot{Q}_{sto2}(t) = -k_{empt}(Q_{sto}) \cdot Q_{sto2}(t) + k_{gri} \cdot Q_{sto1}(t) & Q_{sto2}(0) = 0 \\ \dot{Q}_{gut}(t) = -k_{abs} \cdot Q_{gut}(t) + k_{empt}(Q_{sto}) \cdot Q_{sto2}(t) & Q_{gut}(0) = 0 \\ Ra(t) = \frac{f \cdot k_{abs} \cdot Q_{gut}(t)}{BW} & Ra(0) = 0 \end{cases} \quad (2.7)$$

La tasa de vacío gástrico varía con el tiempo y depende de la cantidad total de glucosa en el estómago. En un comienzo es máximo ( $k_{max}$ ) y luego decrece con una tasa  $\alpha$  a su mínimo  $k_{min}$  para luego recuperar su máximo con tasa  $\beta$ .

$$k_{empt}(Q_{sto}) = k_{min} + \frac{k_{max} - k_{min}}{2} \{ \tanh(\alpha(Q_{sto}(t) - b \cdot D)) - \tanh(\beta(Q_{sto}(t) - d \cdot D)) + 2 \} \quad (2.8)$$

$$\alpha = \frac{5}{2 \cdot D \cdot (1-b)} \quad \beta = \frac{5}{2 \cdot D \cdot c}$$

#### 2.4.5. Utilización de la glucosa

A partir de la literatura se construyó la utilización de la glucosa por los tejidos del cuerpo. Se asumieron dos componentes, un compartimiento insulina-independiente, el cual es constante ( $F_{cns}$ ) y representa el uso del cerebro y de los eritrocitos (glóbulos rojos), y otro insulina-dependiente, y depende de forma no lineal de la glucosa en los tejidos.

### Variables

- $U_{ii}$  [mg/kg/min]: utilización de glucosa independiente de la insulina
- $U_{id}$  [mg/kg/min]: utilización de glucosa dependiente de la insulina
- $X(t)$  [pmol/L]: insulina en el fluido intersticial
- $F_{cns}$  [mg/kg/min]: glucosa consumida por el cerebro y los glóbulos rojos

**Parámetros**

- $V_{m0}$  [mg/kg/min]: parámetro de la ecuación Michaelis-Menten de utilización de glucosa ante acción nula de la insulina
- $K_{m0}$  [mg/kg]: parámetro de la ecuación Michaelis-Menten de utilización de glucosa ante acción nula de la insulina
- $V_{mx}$  [mg/kg/min per pmol/liter]: sensibilidad insulínica
- $p_{2U}$  [ $\text{min}^{-1}$ ]: tasa constante de la acción en la utilización de glucosa periférica

**Ecuaciones**

$$U_{ii}(t) = F_{cns} \quad (2.9)$$

$$U_{id} = \frac{V_m(X(t)) \cdot G_t(t)}{K_m(X(t)) + G_t(t)} \quad (2.10)$$

$$\begin{aligned} V_m(X(t)) &= V_{m0} + V_{mx} \cdot X(t) \\ K_m(X(t)) &= K_{m0} \end{aligned} \quad (2.11)$$

$$\dot{X}(t) = -p_{2U} \cdot X(t) + p_{2U}[I(t) - I_b] \quad X(0) = 0 \quad (2.12)$$

$$U(t) = U_{ii}(t) + U_{id}(t) \quad (2.13)$$

**2.4.6. Secreción de insulina**

Descripción de la secreción pancreática (en el caso de los pacientes diabéticos tipo I, este bloque se omite)

**Variables**

- $S(t)$  [pmol/kg/min]

**Parámetros**

- $\alpha$  [ $\text{min}^{-1}$ ]: retardo entre la señal de glucosa y la secreción de insulina
- $\beta$  [pmol/kg/min per mg/dl]: respuesta pancreática ante la glucosa
- $h$  [mg/dl]: nivel de umbral de glucosa en el cual las células  $\beta$  inician a producir nueva insulina (está establecido como  $G_b$  para garantizar el sistema en estado estacionario en condiciones basales)
- $K$  [pmol/kg per mg/dl]: respuesta pancreática a la tasa de cambio de la glucosa
- $\gamma$  [ $\text{min}^{-1}$ ]: tasa de transferencia constante entre la vena porta y el hígado

**Ecuaciones**

$$S(t) = \gamma \cdot I_{po}(t) \quad (2.14)$$

$$\dot{I}_{po}(t) = -\gamma \cdot I_{po}(t) + S_{po}(t) \quad I_{po}(0) = I_{pob} \quad (2.15)$$

$$S_{po}(t) = \begin{cases} Y(t) + K \cdot \dot{G}(t) + S_b & \text{si } \dot{G} > 0 \\ Y(t) + S_b & \text{si } \dot{G} \leq 0 \end{cases} \quad (2.16)$$

$$\dot{Y}(t) = \begin{cases} -\alpha \cdot [Y(t) - \beta \cdot (G(t) - h)] & \text{si } \beta \cdot (G(t) - h) \geq -S_b \\ -\alpha \cdot Y(t) - \alpha \cdot S_b & \text{si } \beta \cdot (G(t) - h) < -S_b \end{cases} \quad (2.17)$$

**2.4.7. Cinética de la insulina subcutánea**

Para simular un paciente diabético tipo I se sustituye el módulo de secreción por uno de infusión subcutánea de insulina.

**Variables**

- $I_{sc1}$  [pmol/kg]: cantidad de insulina no-monomérica en el espacio subcutáneo
- $I_{sc2}$  [pmol/kg]: cantidad de insulina monomérica en el espacio subcutáneo
- $IIR(t)$  [pmol/kg/min]: tasa de infusión exógena de insulina

**Parámetros**

- $k_d$  [ $\text{min}^{-1}$ ]: tasa de disociación de insulina
- $k_{a1}$  [ $\text{min}^{-1}$ ]: tasa de absorción de insulina no-monomérica
- $k_{a2}$  [ $\text{min}^{-1}$ ]: tasa de absorción de insulina monomérica

**Ecuaciones**

$$\begin{cases} \dot{I}_{sc1}(t) = -(k_d + k_{a1}) \cdot I_{sc1}(t) + IIR(t) & I_{sc1}(0) = I_{sc1ss} \\ \dot{I}_{sc2}(t) = k_d \cdot I_{sc1}(t) - k_{a2} \cdot I_{sc2}(t) & I_{sc2}(0) = I_{sc2ss} \end{cases} \quad (2.18)$$

$$R_i(t) = k_{a1} \cdot I_{sc1}(t) + k_{a2} \cdot I_{sc2}(t) \quad (2.19)$$

**2.4.8. Excreción renal de glucosa**

La extracción renal de glucosa por el riñón ocurre cuando la glucosa en plasma excede cierto umbral

**Variable**

- $E(t)$  [mg/kg/min]: Excreción de glucosa por los riñones

**Parámetros**

- $k_{e1}$  [ $\text{min}^{-1}$ ]: tasa de filtración glomerular
- $k_{e2}$  [mg/kg]: umbral de glucosa renal

## Ecuaciones

$$E(t) = \begin{cases} k_{e1} \cdot [G_p(t) - k_{e2}] & \text{si } G_p(t) > K_{e2} \\ 0 & \text{si } G_p(t) \geq K_{e2} \end{cases} \quad (2.20)$$

## 2.5. Comparación de modelos

La comparación de los modelos se presentará en términos de su comportamiento a lazo abierto, comparando ventajas y desventajas a la hora de la implementación (en base a [1], [2] y [3]).

El modelo de Sorensen fue el primero que presentó una dinámica compartimental completa que con la idea de un paciente promedio, podría ser representado paramétricamente. Éste permite una inmediata transformación de una persona normal a un paciente con DT1 eliminando el factor asociado a la secreción de insulina del páncreas  $\Gamma_{IR}$  y contempla el efecto del glucagón, a diferencia del resto de los modelos. Sin embargo, posee varias desventajas. Solo contempla insulina intravenosa, perdiéndose el gran retardo de la inyección de insulina de forma subcutánea, lo que resulta muy importante a tener en cuenta a la hora de realizar un control de lazo cerrado. Aunque la capacidad del modelo para predecir anormalidades metabólicas ha sido demostrada, se reconoce que un ajuste de parámetros es necesario, lo cual derivó, en estudios posteriores, a compensar la falta de variación inter-paciente variando ciertos parámetros fisiológicos. Sin embargo, éstos fueron sintetizados mediante el modelo a falta de datos de verdaderos pacientes.

El modelo de Cambridge está diseñado para contemplar el desarrollo de controladores de lazo cerrado para pacientes con DMTI, pero se han obtenido pacientes promedio con DMTII. Las predicciones basadas en una población generadas por el simulador desarrollado por el equipo de Cambridge, han sido validadas por estudios clínicos en sujetos jóvenes con DMTI en pruebas de noche a la mañana. Como problemática, posee que el modelo de absorción gastrointestinal de glucosa es simple.

El modelo de Dalla Man-Cobelli está también basado en compartimentos y representa al paciente promedio. En contraste con el modelo de Sorensen, resuelve el problema de la variabilidad inter-paciente mediante una gran cantidad de pacientes virtuales agregados al simulador desarrollado. Posee un detallado modelo de absorción de glucosa y modelos que representan el monitoreo de glucosa y la administración subcutánea de insulina, lo que implica simulaciones más realistas. Como principal atractivo, es que el simulador desarrollado por el grupo de Pavoda, el simulador UVa, ha sido aceptado por la FDA como sustituto de pruebas pre-clínicas en animales para controladores de lazo cerrado.

El modelo de Bergman posee como fortaleza parámetros variantes con el tiempo, pero está limitado por la representación simplista de la cinética de la glucosa y una sobreestimación de su efectividad. Se han desarrollado modelos más completos a partir del mismo agregando subsistemas desarrollados por otras entidades, como en el caso de Fabietti y Medtronic, los cuales se han obviado en este análisis.

En el cuadro 2.1 se plasma la comparación de las ventajas y desventajas de cada modelo estudiado.

## 2.6. Selección del modelo

Según [1], los principales componentes de un modelo de simulación para la validación efectiva de un controlador de glucosa se resumen en:

1. Un modelo de la cinética de la glucosa y de la acción de la insulina, con complejidad suficiente para representar adecuadamente dicha dinámica y simplicidad suficiente para que sus parámetros puedan ser identificados a partir de ensayos clínicos.
2. Un modelo de la cinética de la insulina en el espacio subcutáneo. Actualmente las bombas de insulina suministran la insulina de forma discreta en el espacio s.c., por lo que es necesario considerar el retraso temporal al transporte hacia el plasma.
3. Un modelo de la absorción de glucosa en el intestino capaz de representar distintos patrones de absorción de comidas mixtas.
4. Un modelo específico de los errores asociados a los sensores continuos de glucosa, que sea capaz de reproducir los retrasos temporales debidos a la lectura en el espacio s.c., las desviaciones propias del sistema y a la calibración, y el ruido aleatorio de los sensores.
5. Un modelo del efecto del ejercicio en la cinética de la glucosa. Aunque la mayor parte de los entornos de simulación actualmente disponibles no incluyen este componente, su inclusión representaría una aportación significativa.
6. Una población de pacientes virtuales lo suficientemente representativa de la variabilidad intra e inter-paciente de los parámetros metabólicos en la población general de sujetos con DMI. Ésta es una característica que desafortunadamente no incluyen todos los simuladores existentes y que en otros casos se debería mejorar significativamente.

Como conclusión tras la comparación de los diferentes modelos matemáticos, se optó por el modelo desarrollado por Dalla Man, debido a que cumple de manera adecuada muchas de las condiciones anteriormente mencionadas, teniendo varias ventajas respecto al resto (resaltadas en la tabla 2.1), y pudiéndose agregar variación intra-paciente y las comidas mixtas. Otra ventaja importante es el simulador UVa (ver 2.7.1) desarrollado en colaboración por la universidad de Virginia y de Padova, que ha sido aceptado por la FDA como reemplazo a pruebas en animales y las estrategias de control validadas en el simulador se pueden implementar en pruebas clínicas.

Modelo- Simulador	Ventajas	Desventajas
Sorensen	<ul style="list-style-type: none"> <li>▪ Transformación inmediata de un paciente normal a uno con DTI</li> <li>▪ Considera la dinámica del glucagón</li> </ul>	<ul style="list-style-type: none"> <li>▪ La administración de insulina es intravenosa</li> <li>▪ La variabilidad inter e intra-paciente no es considerada</li> </ul>
Bergman	<ul style="list-style-type: none"> <li>▪ Posee parámetros que varían con el tiempo</li> </ul>	<ul style="list-style-type: none"> <li>▪ Representación muy simplista de la cinética de la glucosa</li> <li>▪ Pequeña duración de la acción de la insulina</li> <li>▪ Sobre estimación de la efectividad de la glucosa</li> </ul>
Hovorka	<ul style="list-style-type: none"> <li>▪ Posee pacientes validados con estudios clínicos</li> <li>▪ Incluye variabilidad intra-paciente</li> <li>▪ Agrega modelos generales de monitoreo de glucosa y de administración subcutánea de insulina</li> </ul>	<ul style="list-style-type: none"> <li>▪ La dinámica del glucagón no es considerada</li> <li>▪ La absorción gastrointestinal de la glucosa es muy simple</li> </ul>
Dalla Man	<ul style="list-style-type: none"> <li>▪ Incluye variación inter-paciente</li> <li>▪ Posee gran cantidad de pacientes virtuales</li> <li>▪ Se puede obtener un paciente promedio con DTII</li> <li>▪ El modelo de absorción de glucosa es realista</li> <li>▪ Agrega modelos de marcas específicas de monitoreo de glucosa y de administración subcutánea de insulina</li> <li>▪ Incluye a la dinámica del glucagón</li> <li>▪ Incluye el efecto del ejercicio</li> <li>▪ Aceptado por la FDA</li> </ul>	<ul style="list-style-type: none"> <li>▪ No se considera la variabilidad intra-paciente</li> </ul>

## 2.7. Plataformas de simulación

Hoy en día se pueden encontrar diversas herramientas de software dentro de la ingeniería para el desarrollo matemático y para la simulación de diferentes sistemas. A pesar de esto, hay 2 que han sobresalido entre los usuarios, los cuales se los puede referir como los programas más habituales. Estos son el MATLAB y el LabVIEW. Se desea evitar un análisis exhaustivo de cada programa, debido a sus complejidad en cuanto a estructura, funcionamiento, diseño y opciones, pero se los comparará por algunas características que los diferencian y por el enfoque al cual apuntan.

MATLAB es un programa interactivo para computación numérica, orientado originalmente al trabajo con matrices y al fácil manejo de las librerías de cálculo LINPACK y EISPACK. El programa permite realizar de un modo rápido la resolución numérica de problemas en un tiempo mucho menor que si se quisiesen resolver estos mismos problemas con lenguajes de programación tradicionales. Éste dispone dentro de sí el programa Simulink, usado para el diseño, modelaje y simulación de sistemas de control y procesamiento de señales, permitiendo combinar el diseño por bloques y por el lenguaje de programación de MATLAB. También dispone de un amplio abanico de programas de apoyo especializados, denominados toolboxes, que extienden significativamente el número de funciones incorporadas en el programa principal. Estos toolboxes cubren en la actualidad prácticamente casi todas las áreas principales en el mundo de la ingeniería y la simulación, destacándose en este caso el SimBiology, el cual resulta conveniente para análisis de sistemas fisiológicos.

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) es una plataforma y entorno de desarrollo para diseñar sistemas, con un lenguaje de programación visual gráfico, siendo ésta una de sus principales características, ya que permite el uso del mismo al usuario con pocos conocimientos en lenguajes de programación. Recomendado para sistemas hardware y software de pruebas, control y diseño, simulado o real y embebido, pues acelera la productividad. Su orientación principal es la adquisición de datos de diversos hardwares y su procesamiento en vivo, permitiendo interactuar con diferentes interfaces y lenguajes.

### 2.7.1. Simulador UVa

El simulador se fundamenta en el modelo de Dalla Man ([6], fue desarrollado en colaboración entre la Universidad de Virginia, E.E.U.U, y la Universidad de Pavoda, Italia, e implementado en la plataforma Matlab y Simulink. Este simulador contiene una gran cantidad de pacientes in silico basados en información real individual, incluyendo una cohorte de 100 adultos, 100 adolescentes, y 100 niños, lo que refleja una variación inter-paciente importante. Posee incluido modelos de sensores de glucosa y de bombas de insulina. Los sensores incluyen los dispositivos Freestyle Navigator<sup>TM</sup>, Guardian RT y Dexcom STS<sup>TM</sup>, con posibilidad de modelar errores de calibración y ruido aleatorio. Las bombas presentan las limitaciones técnicas (bolo máximo, bolo mínimo, bolo incremental, etc), incluyéndose los dispositivos Omnipod Insulin Management System y Detec Cozmo. Sin embargo, el modelo UVa tiene algunas limitaciones prácticas. Por ejemplo, no considera variaciones intra-paciente, el efecto de comidas mixtas, ni aspectos tecnológicos como las dinámicas del monitor y de la bomba, ruido y derivas en la medición, retardo subcutáneo-intravenoso en el monitor y la bomba, etc. Más adelante se tendrán en cuenta estos aspectos a la hora de montar un nuevo simulador.



## 2.8. Selección de plataforma

La plataforma que se encontró adecuada para implementar el simulador fue MATLAB, en particular debido a que el simulador UVa ha sido desarrollado en esta plataforma, lo que es una gran ventaja como punto de partida. Su capacidad de combinar lenguaje de programación con la herramienta Simulink (orientada a la simulación y control de sistemas) resulta potente a la hora de desarrollar estrategias de control, y así validar los controladores in silico. Se podría haber considerado utilizar el toolbox SimBiology del MATLAB, pero éste no está orientado a realizar sistemas de control y no es práctico para representar un sistema con ecuaciones diferenciales, por el contrario, evita la representación con el uso de las mismas, usando relaciones en las diferentes reacciones del sistema (como por ejemplo la relación de masas).

## Capítulo 3

# Desarrollo de la plataforma de simulación

### 3.1. Plataforma de simulación

Antes de comenzar con la programación del simulador, se investigó y estudio la estructura del simulador UVa, cuya versión disponible por la Facultad de Ingeniería de la UNLP tiene la posibilidad de acceder a los archivos con los códigos de programación, y a una cantidad limitada de pacientes virtuales. En este simulador no resulta sencillo aplicar cambios en el régimen de simulación, pacientes, entorno, parámetros, etc. Solo permite ajustar un número limitado de parámetros accesibles por la interfaz gráfica de usuario. Estas limitaciones motivan el interés en desarrollar un simulador propio a los fines de futura investigación. A partir de un extenso análisis de cada función que integra el programa, se obtuvo un panorama del marco de simulación, conformado por elementos como

- Configuración de la simulación que realiza Simulink
- Creación y administración de bolos de insulina y comidas al paciente
- Implementación del modelo del paciente
- Parámetros configurables
- Carga de los pacientes
- Gráficos de las variables de importancia a la hora de verificar el tratamiento
- Modelo de la bomba de insulina y sensor de glucosa
- Aplicación del controlador

Una vez comprendido el proceder y los aspectos que tiene en cuenta el simulador UVa, a la hora de diseñar la estructura del programa se tuvieron los siguientes criterios en cuenta:

- Estructura didáctica
- Programación en bloques específicos
- Generación de funciones determinadas para cada bloque, con objetivos específicos

- Desarrollo intuitivo para el usuario de las acciones de cada bloque y el proceder de cada función, para una posible manipulación
- Fácil modificación de los parámetros tanto de simulación como del sistema
- Capacidad de simular con diferentes valores de paso de simulación fijo
- Una interfaz gráfica intuitiva y ágil de simulación que no limite la versatilidad del simulador
- Variedad de muestras gráficas de los resultados, con posibilidad de generar nuevos gráficos sin necesidad de volver a correr la simulación y sencillez de agregar gráficos por el usuario

### 3.1.1. Bloques

Los bloques de la estructura del programa que dan el aspecto didáctico, intuitivo y estructurado que se consideraron apropiados fueron:

Bloque de inicialización: se crean las estructuras a utilizar y se les asigna un espacio de memoria para mayor eficiencia a la hora de correr el programa.

Bloque de configuración de simulación: se utiliza una interfaz gráfica (figura 3.3) donde el usuario podrá determinar:

- Pacientes a simular
- Elección de un escenario diseñado previamente a partir de un archivo ASCII
- Paso de simulación
- Tiempo de simulación
- Elección de hardware (bomba de insulina y sensor de glucosa)
- Optar por agregar comidas mixtas
- Optar por incluir variación sinusoidal circadiana intra-paciente y ajustar sus periodo, amplitud y fase
- Determinar los rangos de hipoglucemia e hiperglucemia
- Seleccionar tipo de tratamiento (lazo abierto, cerrado o híbrido)

Bloque de creación del entorno: a partir de las condiciones definidas en el bloque anterior y en el escenario, se crean los vectores de las comidas, bolos de insulina manuales y óptimos, insulina basal, insulina y glucosa intravenosa, variación intra-paciente, comidas mixtas, que se les suministrarán al paciente a lo largo de la simulación. Los vectores creados poseen el valor de la variable para cada paso de simulación.

Bloque de simulación: dentro de este bloque, se crea un bucle cuya repetición depende de la cantidad de pacientes a simular. Se cargan los datos del primer sujeto y se lo simula en la plataforma de Simulink. Luego se guardan los resultados y variables obtenidos y se repite para el siguiente sujeto, hasta completar todos los pacientes seleccionados anteriormente.

Bloque de resultados: Terminadas las simulaciones, resulta muy importante guardar los datos obtenidos de forma práctica y accesible, tanto para un análisis inmediato como para

un futuro análisis más profundo. Para el análisis inmediato se ofrece una serie de diferentes gráficos y una tabla con valores estadísticos. A su vez, todos los vectores creados, los estados y salidas del sistema, los parámetros de configuración de la simulación, se guardan para un posible análisis más profundo.

### 3.1.2. Escenarios

Para la creación del entorno de simulación, se partió de los escenarios que utiliza el simulador UVa, además que esto agrega la fácil comparación y verificación entre los resultados de las simulaciones de ambos simuladores. Éstos consisten en la determinación por el usuario de las condiciones a las que se someterá al/a los paciente/s a simular. En los mismos se definirán:

- Horario de cada comida
- Cantidad de CHO de cada comida
- Duración de las comidas
- Tiempo en que se suministrarán las dosis de insulina
- Cantidad de insulina de cada bolo a suministrar
- Tiempo y cantidad de suministros intravenosos al paciente de glucosa y/o insulina
- Capacidad de suministro de bolos óptimos en función de la sensibilidad insulínica de cada paciente y el anuncio de comidas

Éstos tienen un formato de texto ASCII de fácil edición por el usuario (figura 3.4), pudiéndose crear distintos tipos de escenario.

### 3.1.3. Simulink

A la hora de plasmar el modelo de estados del paciente dentro del Simulink, en una primera instancia se decidió por describirlo en composición de bloques integradores, constantes, realimentaciones, etc, pero surgieron complicaciones debido a lo tedioso en que se convertía la tarea, falta de claridad de comprensión del modelo en una primera aproximación, dificultad de configurar valores iniciales, entre otros, por lo que se optó por el bloque propio del programa llamado S-Function.

El bloque S-Function consiste en una serie de entradas, estados, y salidas, donde las últimas son función del tiempo de simulación, los estados del sistema y las entradas, utilizado para simular sistemas continuos, discretos o híbridos. El modelo de la S-Function de Simulink procede en pasos. En primer lugar viene la fase de inicialización, donde el simulador incorpora los bloques de bibliotecas al modelo, propaga los valores de las señales, los tipos de data, tiempos de muestreo, evalúa los bloques de parámetros, determina el orden de ejecución de los bloques y asigna la memoria. Luego, el simulador entra en un bucle de simulación, donde cada paso a través del bucle se lo refiere como a un paso de simulación. Durante cada paso, se ejecuta cada bloque del modelo en el orden determinado en la inicialización. Para cada bloque, se llama a las funciones que calculan los bloques de estados, derivadas y salidas de la muestra de tiempo actual.

Los estados que se definieron dentro del bloque fueron los del modelo de Dalla Man (ver `refmodeldalla`):

1. Glucosa en el estómago fase sólida [mg]
2. Glucosa en el estómago fase líquida [mg]
3. Glucosa en el intestino [mg]
4. Glucosa en plasma y en tejidos de rápido equilibrio [mg/Kg]
5. Glucosa en tejidos de equilibrio lento [mg/Kg]
6. Insulina en plasma [pmol/Kg]
7. Insulina en líquido intersticial [pmol/L]
8. Insulina retardada [pmol/L]
9. Acción de la insulina en la producción de glucosa [pmol/L]
10. Insulina en el hígado [pmol/L]
11. Insulina no monomérica en el espacio subcutáneo [pmol/Kg]
12. Insulina monomérica en el espacio subcutáneo [pmol/Kg]
13. Glucosa intersticial [mg/Kg]

A partir del bloque S-Function (figura 3.2 (a)), que representa al paciente con el modelo de Dalla Man, se definieron las siguientes entradas:

- CHO sólido consumido [mg]
- Insulina subcutánea [pmol/Kg]
- Glucosa en el estómago e ingerida [mg]
- Insulina intravenosa [pmol/Kg]
- Glucosa intravenosa (dextrose) [mg/Kg]
- Tasa de aparición de comidas mixtas [mg/kg/min]

Y las siguientes salidas:

- Glucosa intersticial
- Insulina en plasma
- Producción de glucosa
- Tasa de aparición de la glucosa
- Utilización de la glucosa
- Tasa de aparición de la insulina
- Glucosa en plasma
- Los estados del sistema

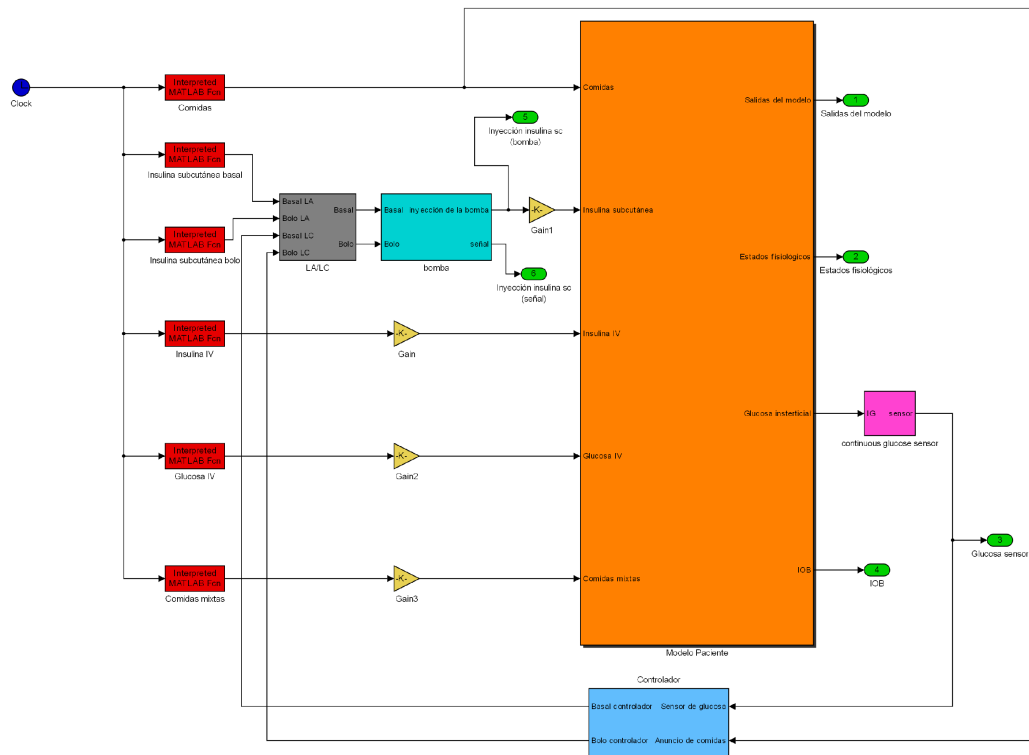


Figura 3.1: Sistema completo

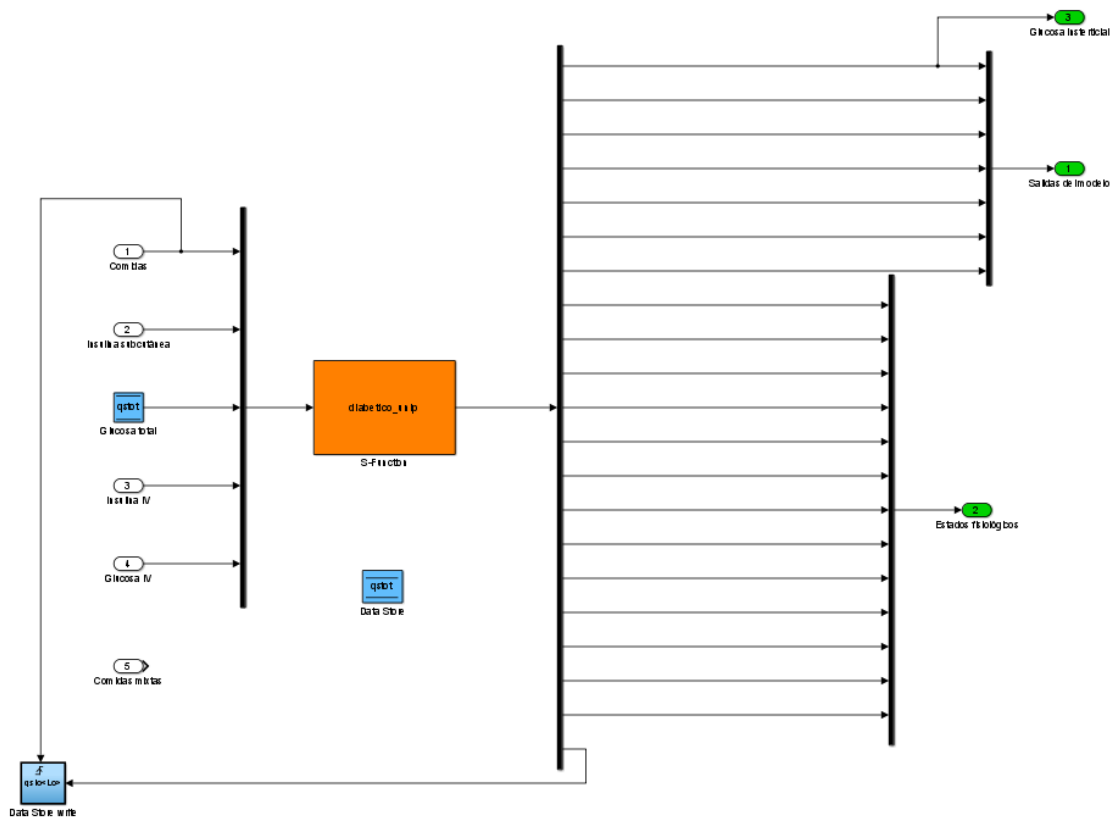
Para proporcionar las entradas al sistema, se utilizan bloques que llaman a los vectores creados en el Matlab por el programa, donde a la salida de los mismos tienen el valor del vector para cada instante de tiempo de simulación.

Al paciente lo acompañan el modelo de la bomba de insulina, el modelo del sensor continuo de glucosa y el controlador de lazo cerrado.

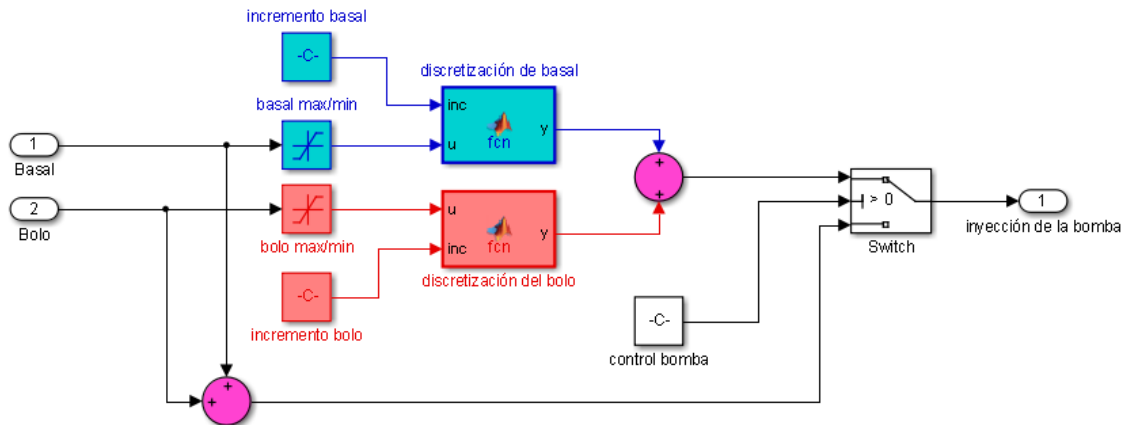
El modelo de la bomba incluye sus limitaciones técnicas principales: discretización, valores máximos y mínimos que puede aportar (figura 3.2 (b)).

El modelo de sensor continuo de glucosa concierne a: valor mínimo discriminable, valor máximo, tasa de muestreo, un reconstructor de orden cero, offset, ganancia, retardo y ruido. El ruido del sensor no es blanco, es modelado por un promedio móvil autoregresivo  $e_n = 0,7 * (e_{n-1} + v_n)$ , donde  $v_n \phi(0, 1)$  es independiente e idénticamente distribuido (no depende de errores previos y la distribución de probabilidad es independiente del tiempo). El error del sensor es calculado usando la transformación de Johnson a partir de los parámetros del sensor  $\xi$ ,  $\lambda$ ,  $\gamma$  y  $\delta$ , siendo ésta determinada por  $\epsilon_n = \xi + \lambda \sinh(\frac{\epsilon_n - \gamma}{\delta})$ .

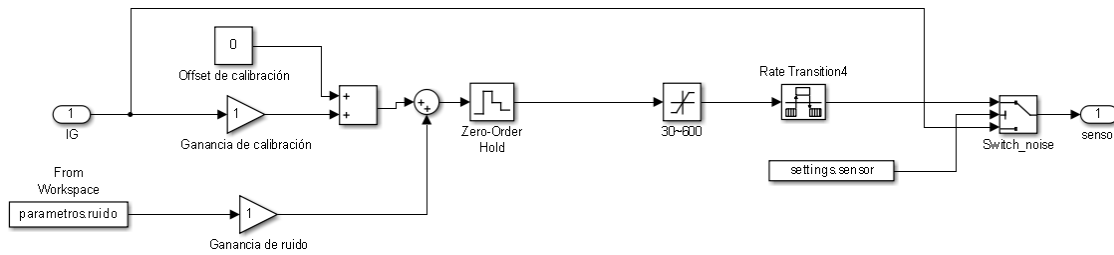
Se previó un subsistema para el diseño de un futuro controlador, y la posible adaptación de un tratamiento de lazo cerrado. Al mismo se le adjuntaron como entradas la glucosa subcutánea y el anuncio de comidas, siendo las salidas la insulina basal y los bolos propiamente dichos. Este subsistema controlador es totalmente modificable por el usuario (figura 3.1).



(a) Subsistema Paciente representado por la S-Function



(b) Subsistema bomba de insulina



(c) Subsistema sensor de glucosa intersticial

Figura 3.2: Subsistemas

### 3.1.4. Resultados de la simulación

Los resultados obtenidos deben ser presentados de forma apropiada al usuario, facilitando el análisis del ensayo realizado y la obtención de conclusiones. Finalizada la simulación, la información colectada se guarda en un archivo propio del matlab de formato *.mat* conteniendo todos los parámetros y vectores utilizados en la simulación para cada paciente. Esto permite verificar después de realizada la simulación aspectos como el escenario utilizado, los pacientes simulados, los parámetros y vectores creados y/o modificados por el programa. En un archivo tabla formato *.xls* se guardan datos utilizados para el análisis cuantitativo de la simulación, guardándose para cada paciente el valor medio, máximo y mínimo de la glucosa en plasma, el tiempo en hiperglucemia e hipoglucemia, y el número de hipoglucemias. Respecto a los resultados gráficos, se podrán seleccionar gráficos individuales de todas las salidas definidas en la S-Function y de las zonas clínicas transitadas, pero también gráficas poblacionales como la glucosa promedio, desviación estándar, valores máximos y mínimos, y un gráfico CVGA utilizado típicamente en estos ensayos (coloca el rendimiento del algoritmo de control en algún área en particular según un periodo de observación. Cada punto representa los valores mínimos/máximos obtenidos en la excursión de glucosa).

## 3.2. Simulación realista

En pos de someter a situaciones más realistas a los sujetos, se agregó al simulador un menú de comidas mixtas y variación intra-paciente, no presentes en el simulador Uva.

### 3.2.1. Comidas Mixtas

Los estudios *in silico* usados con propósitos para el páncreas artificial exigen una implementación de condiciones complejas, reflejando las demandas reales al algoritmo de control. Diferentes modelos del sistema glucosa-insulina fueron desarrollados para explicar las complicadas dinámicas implicadas. Sin embargo algunos procesos, como la absorción de carbohidratos o la acción de la insulina, son alterados en la vida día a día por perturbaciones externas, las cuales no fueron modeladas hasta ahora, teniendo peso en el desempeño de los algoritmos de control. En el caso de la absorción de carbohidratos, en el modelo de Dalla Man la aparición de glucosa no tiene en cuenta la composición de la comida. El grupo MICELab desarrolló una biblioteca de comidas mixtas, conteniendo diferentes perfiles de la tasa de aparición de glucosa a partir de varias comidas compuestas, usando el método de identificación bayesiano. Se implementó esa biblioteca a la plataforma como una opción adicional, permitiendo incluir variabilidad de comidas de diferente tamaño y composición. Cuando una comida es seleccionada para la simulación, el modelo de absorción de carbohidratos es reemplazado por el perfil de dicha comida.

### 3.2.2. Variación intra-paciente

Los pacientes virtuales del modelo de Dalla Man no consideran ninguna variación intra-paciente. En otros modelos, como en el Hovorka, se incluye una variación sinusoidal a la sensibilidad insulínica, pero no se ha detallado ni aplicado profundamente las variaciones en pacientes virtuales. La inclusión de estas variaciones permite una simulación más realista. En el modelo de Dalla Man el parámetro  $V_{mx}$  representa la sensibilidad insulínica, por lo que se le agregó una variación sinusoidal de amplitud, periodo y fase ajustable por el



usuario. Lo mismo se aplicó a la dinámica de la insulina subcutánea, sobre los parámetros  $k_{a1}$ ,  $k_{a2}$  y  $k_d$ .

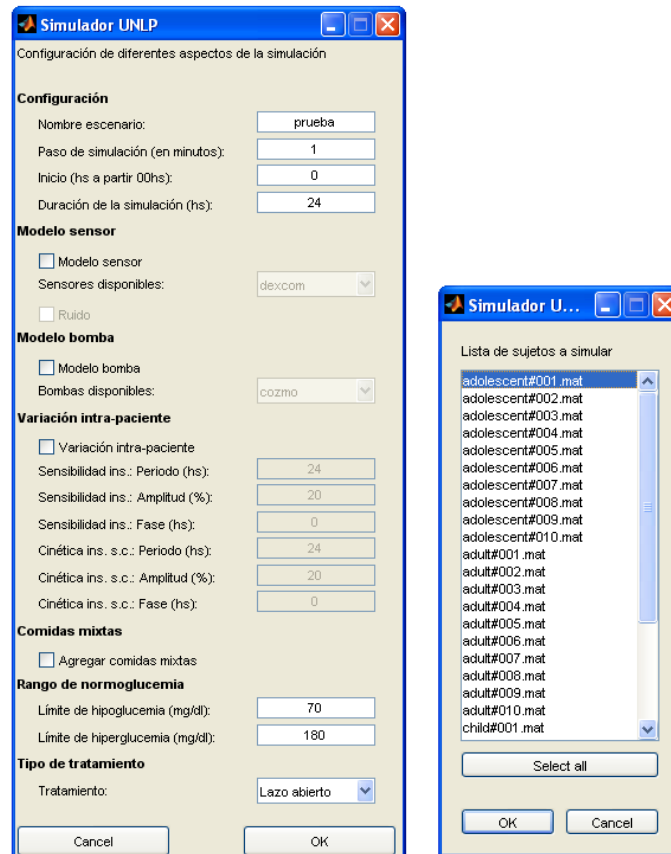
### 3.3. Programación

El simulador consiste en un programa principal que llama a diferentes funciones para ir creando los aspectos de simulación y los parámetros a utilizar. A continuación se describen las funciones por orden de uso. Cada una es fácil de reemplazar y/o modificar, ofreciendo una gran versatilidad a la hora de manipular al simulador. En el apéndice se detallan los códigos de cada función. La interfaz gráfica principal (ver 3.3 a) fue implementada a partir de la función *settingsdlg*, obtenida en [12], donde el usuario puede ingresar y determinar lo siguiente:

- Nombre del escenario: nombre del archivo ASCII que contiene el escenario
- Paso de simulación: paso de simulación fijo de simulink, con unidades en minutos
- Inicio: comienzo de la simulación en horas a apartir de las 0hs
- Duración de la simulación: cantidad de horas que dura la simulación
- Modelo sensor: activar o desactivar el modelo del sensor. Si se activa, se habilita la elección del modelo y la posibilidad de agregar ruido.
- Modelo bomba: activar o desactivar el modelo de la bomba. Si se activa, se habilita la elección del modelo.
- Variación intra-paciente: activar o desactivar la variación intra-paciente. Si se activa, se habilita a modificar amplitud (variación porcentual del parámetro), periodo y fase de los parámetros involucrados con la sensibilidad insulínica y con la dinámica de la insulina subcutánea.
- Agregar comidas mixtas: si se activa, se le permite al usuario agregar al escenario seleccionado las comidas mixtas que desee y sus respectivos horarios.
- Rango de normoglucemia: permite al usuario determinar los rangos de glucosa en sangre hipo-glucemia e hiper-glucemia, para el análisis de resultados y gráficos de salida del simulador.
- Tipo de tratamiento: permite seleccionar entre lazo abierto, lazo cerrado o híbrido.

#### 3.3.1. Sujetos

Crea un vector de cadenas que contiene los nombres de los archivos de extensión *.mat* guardados en la carpeta *Sujetos*. En esta carpeta se sitúan los archivos que posean parámetros de un paciente en particular para el modelo de Dalla Man. Luego, mediante una interfaz gráfica, el usuario selecciona los sujetos que desea simular de la lista completa de sujetos disponibles otorgada por el vector. La función devuelve un arreglo de cadenas con los nombres de los pacientes seleccionados.



(a) Parámetros configurables

(b) Lista de pacientes disponibles

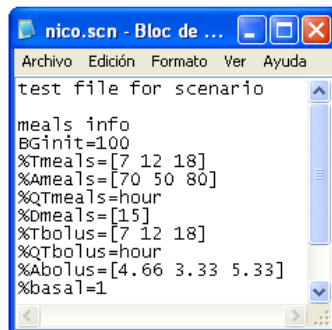
Figura 3.3: Donde se puede seleccionar pacientes a simular y configurar los parámetros deseados.

### 3.3.2. Carga del escenario

Se lee un archivo ASCII creado por el usuario, donde el mismo debe ser de terminación `.scn`. Este archivo contiene diversos parámetros para la simulación. Se reconocen las líneas cuyo primer elemento sea el símbolo `%`, y luego compara la cadena que sigue con las reconocidas por el programa, para así asignar los valores otorgados por el usuario. En caso de que se hayan omitido, se inicializan en valores default. Este script fue levemente modificado de la versión UVa, para que los escenarios creados sean compatibles para ambos simuladores. Los parámetros reconocidos son:

- **BGinit:** permite al usuario establecer la glucosa inicial a cualquier valor positivo. Si no es especificado, la simulación empieza con el valor de ayuno de cada sujeto. La tasa inicial de cambio de la glucosa siempre es 0.
- **Tmeals:** un vector con los horarios de las comidas
- **Ameals:** un vector con la cantidad de CHO de las comidas en g
- **Dmeals:** un vector con la duración de las comidas (si no está presente, se supone 15 min). Se puede usar un solo número (todas las comidas de una misma duración) o un vector de largo Ameals, especificando la duración de cada comida.

- QTmeals: unidad de Tmeals (min, hour o day)
- Tbolus: horarios de los bolos a lazo abierto
- QTbolus: unidad de Tbolus (min, hour o day)
- Abolus: cantidad de insulina del bolo a lazo abierto en U
- basal: la inyección de insulina basal a lazo abierto
- TIVINS: un vector con los horarios de las inyecciones intra-venosas de insulina en minutos
- AIVINS: un vector de la tasa de inyección de insulina IV (U por minuto)
- TIVD: un vector con los horarios de las inyecciones IV de dextrose
- AIVD: un vector con la tasa de inyección IV de dextrose (gramos de CHO por minuto)
- LAoptimo: si se coloca **on**, se calculan los bolos óptimos de lazo abierto para cada sujeto
- Sensibilidad: se define otra relación I:CHO que reemplaza la óptima de cada paciente
- deltat: tiempo de diferencia entre la comida y la inyección del bolo óptimo de lazo abierto. Si es un número positivo es adelanto, si es negativo atraso respecto al horario de cada comida



```

nico.scn - Bloc de ...
Archivo Edición Formato Ver Ayuda
test file for scenario
meals info
BGinit=100
%Tmeals=[7 12 18]
%Ameals=[70 50 80]
%QTmeals=hour
%Dmeals=[15]
%Tbolus=[7 12 18]
%QTbolus=hour
%Abolus=[4.66 3.33 5.33]
%basal=1

```

Figura 3.4: Ejemplo de escenario de simulación

### 3.3.3. Creación del vector de comidas

A partir de los vectores  $Tmeals$ ,  $Ameals$  y  $Dmeals$  obtenidos del escenario, y el tiempo y paso de simulación determinados por el usuario, se crea un vector que contiene para cada instante de simulación la cantidad de CHO que se le suministra al paciente. El vector tiene un largo igual al cociente entre el tiempo de simulación y el paso de simulación. La cantidad total de CHO determinada por  $Ameals$  de cada comida se distribuye de forma uniforme en un intervalo determinado por  $Dmeals$  a partir del tiempo de suministro definido por  $Tmeals$ . En caso que la duración de la comida sea menor al paso indicado, se suministra toda la comida en un instante único. En la figura 3.5 se ilustra el vector comidas en el tiempo, a partir de  $Ameals=[60 40 70]$ ,  $Tmeals=[7 13 19]$  y  $Dmeals=[15 10 20]$ .

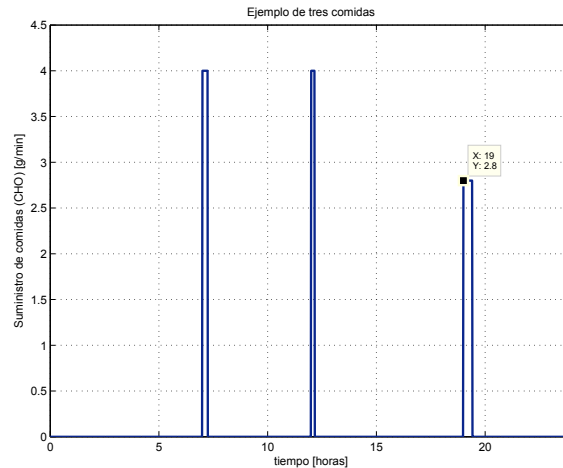


Figura 3.5: Ejemplo de suministro de comidas

### 3.3.4. Creación de comidas mixtas

Los vectores con los valores de las diferentes tasas de aparición de cada comida se encuentran en el archivo *RaPoint.mat*. A partir del número de la comida seleccionada de la lista otorgada por *Lista de comidas.mat*, se obtiene el vector correspondiente a esa comida. Luego se solicita el horario de la comida a partir de las 0hs.

Como los vectores de las tasas de aparición de glucosa fueron muestreados cada 5 minutos durante un tiempo de 7 horas, se realiza una interpolación entre los puntos para así obtener una muestra por paso de simulación. Luego se interpola linealmente el último valor del vector hacia cero, a partir de la derivada tomada desde los últimos dos puntos.

Finalmente, a partir del horario de la comida, se genera la tasa de aparición de glucosa de la comida mixta seleccionada. En un vector se guardan los valores de CHO de cada comida y el horario correspondiente para el futuro cálculo de bolos óptimos.

Se pueden agregar tantas comidas como se desee, devolviéndose un vector con la tasa de aparición completa de todas las comidas mixtas, el vector con las cantidades de CHO y otro con los horarios.

Luego en la S-Function, se sumará de forma lineal a la salida del sistema correspondiente a la tasa de aparición de glucosa el vector de las comidas mixtas.

### 3.3.5. Creación de los vectores de suministro de insulina

Se crea un arreglo de dos vectores, uno correspondiente a los bolos y otro a la dosis basal de insulina de lazo abierto. El vector de suministro de bolos de insulina subcutánea para el paciente se crea de forma análoga al vector de comidas a partir de los vectores *Tbolus* y *Abolus*. Dentro de la función se puede ajustar la duración del suministro de bolos, siendo por defecto 1 minuto. El vector de la dosis basal de insulina se crea a partir de una distribución uniforme para cada paso de simulación del valor de insulina basal por hora, resultando una constante en toda la simulación.

### 3.3.6. Creación de los vectores intravenosos

Se crea un arreglo con los vectores de suministro de forma intravenosa de glucosa e insulina. En los instantes determinados por los vectores *TIVD* y *TIVINS*, el vector contiene los valores descriptos por *AIVD* y *AININS* respectivamente.

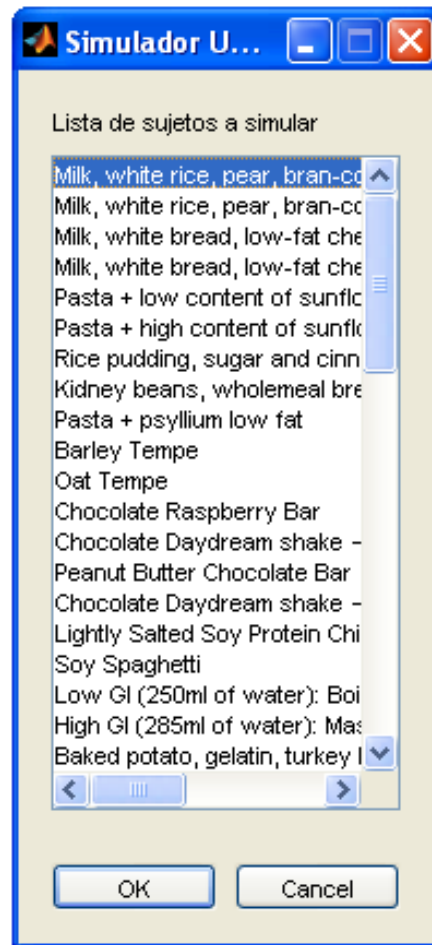


Figura 3.6: Menú de comidas mixtas

### 3.3.7. Carga de Hardware

En caso de especificar el uso de una bomba o de sensor de glucosa, se carga el modelo a partir de un archivo ASCII de forma similar a como se carga el escenario. Se reconocen los siguientes parámetros para la bomba

- minbolus: valor mínimo de bolo que puede suministrar la bomba
- maxbolus: valor máximo de bolo que puede suministrar la bomba
- incbolus: incremento discreto para suministro de bolo
- minbasal: valor mínimo de insulina basal que puede suministrar la bomba
- maxbasal: valor máximo de insulina basal que puede suministrar la bomba
- incbasal: incremento discreto para suministro basales

y los siguientes para el sensor

- PACF: parámetro para el cálculo de ruido
- type: parámetro para el cálculo de ruido

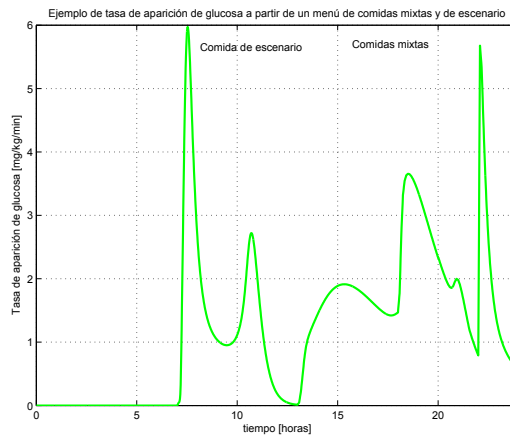


Figura 3.7: Ejemplo de tasa de aparición de comidas por escenario y mixtas

- $\gamma$ : parámetro para el cálculo de ruido
- $\lambda$ : parámetro para el cálculo de ruido
- $\delta$ : parámetro para el cálculo de ruido
- $\xi$ : parámetro para el cálculo de ruido
- $\text{sampling}$ : tiempo de muestreo en minutos
- $\text{min}$ : valor mínimo que discrimina
- $\text{max}$ : valor máximo que mide

En la figura 3.8 se puede apreciar un ejemplo la diferencia entre la señal de glucosa proveniente de la S-Function y la salida del sensor, y la diferencia entre la señal de control de inyección de insulina proveniente de un controlador proporcional-derivativo (PD) y el suministro que proporciona la bomba.

### 3.3.8. Carga de parámetros del sujeto

A partir del vector que contiene el nombre de los sujetos seleccionados a simular, dentro de un bucle determinado por el número de pacientes, se carga en cada instancia los parámetros del modelo correspondientes a cada sujeto desde los archivos de extensión *.mat* alojados en la carpeta *Sujetos*. Éstos se guardan en la estructura *parámetros*.

### 3.3.9. Creación de bolos óptimos según el paciente

En caso que se haya colocado en el escenario *LAoptimo=on*, para cada paciente, se cargan los datos de bolo óptimo contenidos en el archivo *OB.mat*, que poseen la relación I:CHO (por ejemplo  $OB=15$  corresponde a 1U por cada 15g de CHO). Mediante esta relación se calculan los bolos para cada comida y se determina el tiempo de suministro a partir del vector que contiene los horarios de cada comida. Antes, se agregan las cantidades de CHO y los horarios de comidas mixtas, si las hubiere. Se puede configurar la variable *deltat* para determinar cuánto tiempo antes o después de la comida se desea suministrar los bolos. El bolo óptimo es calculado optimizando un día típico de 24 horas con tres comidas de 60g de CHO y un tentempié de 10g a las 23 horas. Cada comida es asociada a

un bolo computado como gramos de CHO/OB. El criterio usado es el rango min/max de glucosa en el espacio de riesgo. La tasa basal que mantiene el nivel de glucosa en ayuno está dada dentro de los parámetros del sujeto. El código es análogo a la creación de bolos explicado anteriormente, salvo que se utilizan los datos de bolo y tasa basal óptimos.

### 3.3.10. Condiciones iniciales

En caso de determinar un nivel específico de glucosa en sangre inicial en *BGinit*, a partir de éste se recalculan los valores iniciales del resto de los estados del sistema del modelo de Dalla Man.

### 3.3.11. Configuración de la simulación en Simulink

Una vez creados los vectores de suministro de comidas, bolos de insulina, insulina basal, glucosa e insulina intravenosa, y se cargaron y configuraron los parámetros necesarios para la simulación, se configura la simulación que realizará Simulink, especificándose tipo de simulación, el tamaño de paso, tipo de paso, tipo de algoritmo numérico, comienzo y duración de la simulación, etc.

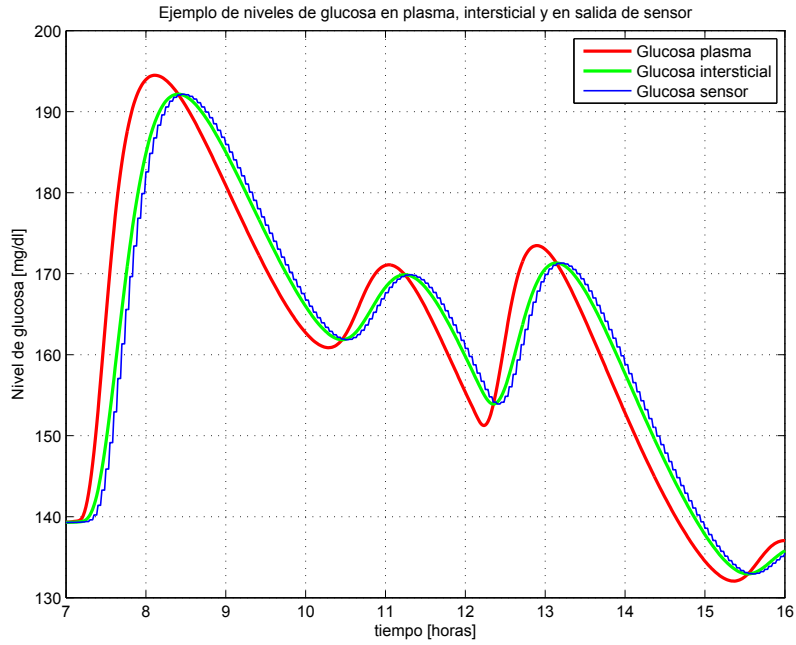
### 3.3.12. Guardado

Luego de cada simulación, se guarda en un archivo de extensión *.mat*, en la carpeta *Sim\_data*, la estructura *data* que contiene para cada paciente simulado los parámetros del sujeto, los parámetros de simulación de Simulink y las salidas de la simulación (estados y salidas del sistema). Se crea un archivo de extensión *.xls* con los valores medios, máximos y mínimos de glucosa, y el tiempo en hipo-glucemia e hiper-glucemia que tuvo cada paciente.

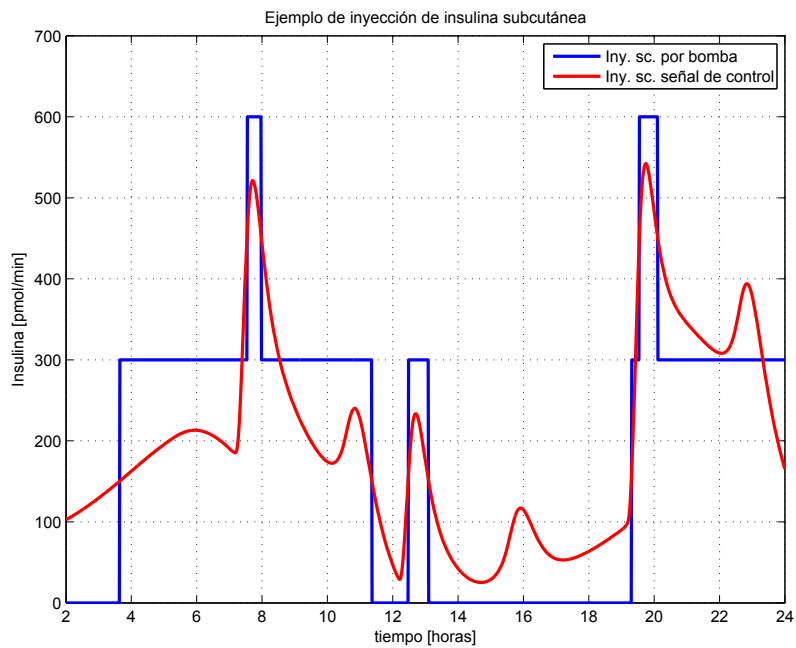
### 3.3.13. Gráficos

Esta función produce a partir del archivo que contiene la estructura *data* una selección de gráficos de salida. Esta función puede ser corrida independientemente de *simulador\_unlp*. El usuario puede seleccionar mediante una interfaz gráfica los siguientes:

- Glucosa en plasma
- Glucosa intersticial
- Glucosa intersticial (salida del sensor)
- Inyección de bolos subcutáneos de insulina (señal)
- Inyección de bolos subcutáneos de insulina (salida de la bomba)
- IOB
- Tasa de producción y utilización de la glucosa, y tasa de aparición de glucosa e insulina
- Glucosa media
- CVGA



(a) Niveles de glucosa



(b) Inyección subcutánea de insulina

Figura 3.8: Comparación de señales con y sin hardware



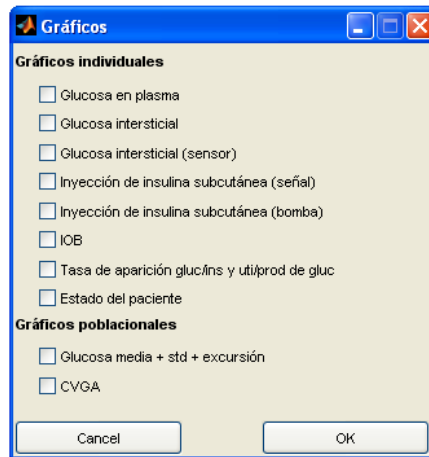
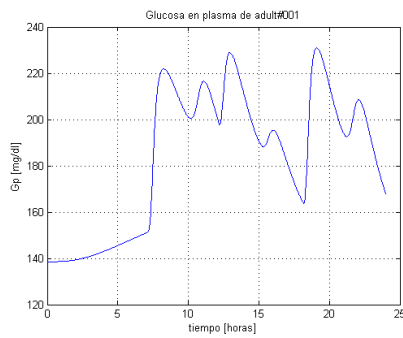
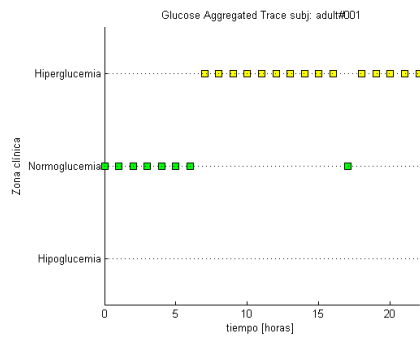


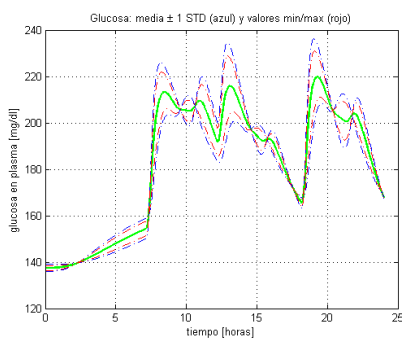
Figura 3.9: Selección de gráficos



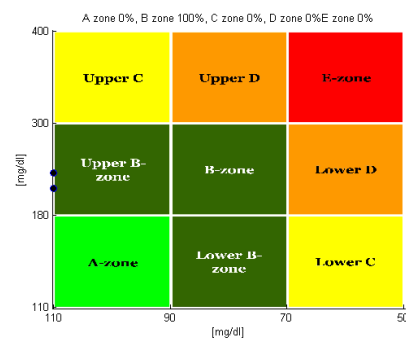
(a) Concentración de la glucosa en plasma



(b) Zona clínica del paciente por hora



(c) Glucosa media  $\pm$  1 DS de dos pacientes adultos bajo el mismo régimen de simulación



(d) CVGA

Figura 3.10: a) y b) Ejemplo de simulación de un paciente adulto. Se simularon tres comidas en un periodo de 24 horas, visualizándose la glucosa en plasma en mg/dl y las zonas clínicas en que se encontró el paciente en cada hora. c) y d) Mismo escenario sobre dos pacientes adultos.

## Capítulo 4

# Simulaciones y resultados

### 4.1. Simulaciones

Se creó un escenario de prueba con el fin de demostrar el potencial del simulador para crear diferentes condiciones de simulación. El mismo consta de tres comidas, una a las 7hs (horarios respecto a las 0hs) de 60g de CHO, otra a las 12hs de 40g de CHO y una última a las 19hs de 70g de CHO, de una duración de 15 minutos cada una. Cada simulación tuvo una duración de 24hs (comenzando a partir de las 0hs). En los casos que se suministraron comidas mixtas, se reemplazaron las comidas del escenario por un menú compuesto de *Standar Breakfast 3* (50g) a las 7hs, *Pasta + high content of sunflower oil* (75g) a las 12hs y *Boiled rice, corn, turkey breast* (50g) a las 19hs. En la figura 4.2 se puede observar la tasa de aparición de glucosa de las comidas del escenario por la absorción del modelo de Dalla Man, y la tasa de aparición de las comidas mixtas, de los ensayos de MICELab. En el caso del control a lazo abierto, los bolos de se suministraron en el mismo horario de cada comida correspondiente, mientras que en el caso del control a lazo cerrado, se implementó un controlador PD sintonizado a partir de la literatura ([10]). La ecuación general del PD es

$$u(t) = K_p e(t) + K_p \tau_D \frac{de(t)}{dt}$$

donde  $u(t)$  es la tasa de infusión de insulina, y  $e(t)$  es la señal de error. La sintonización nominal depende de la dosis diaria total de insulina del sujeto ( $I_{TDD}$  en U). La ganancia del controlador es calculada como

$$K_p = \frac{60}{90} \frac{I_{TDD}}{1500} \quad (4.1)$$

siendo su unidad U/h per mg/dL, convirtiéndose a pmol/min per mg/dL por un factor de 100 pmol/U per h/min. La constante de tiempo derivativa  $\tau_D$  tiene un valor de 90 minutos, para la respuesta al crecimiento de los niveles de glucosa. El termino integral se obvió ya que corresponde al ajuste de los niveles basales.

#### 4.1.1. Simulación N°1: Paciente adulto a lazo abierto

Se simula al paciente adult#001 a lazo abierto, suministrando los bolos óptimos para cada comida según la relación I:CHO del paciente, y la insulina basal correspondiente al nivel de ayuno, ambos valores obtenidos a partir de la información clínica que se brinda del paciente.

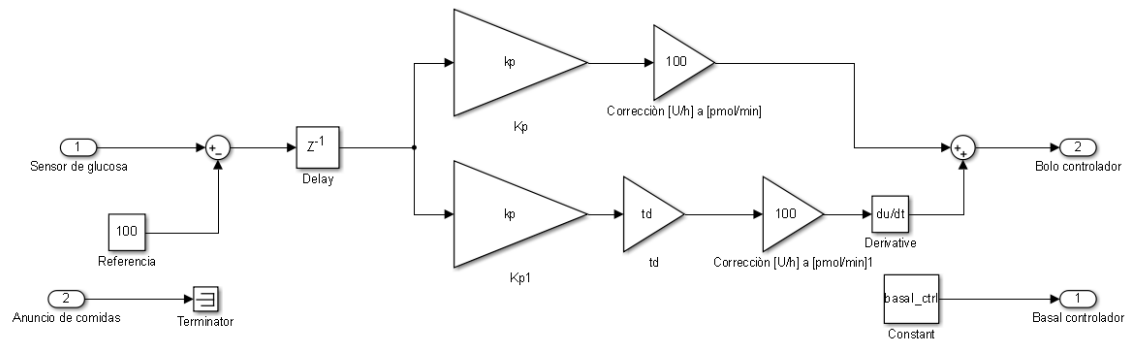


Figura 4.1: Controlador empleado en el simulador

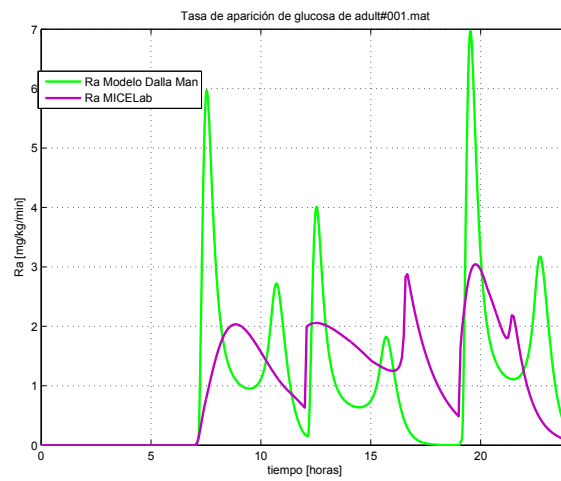
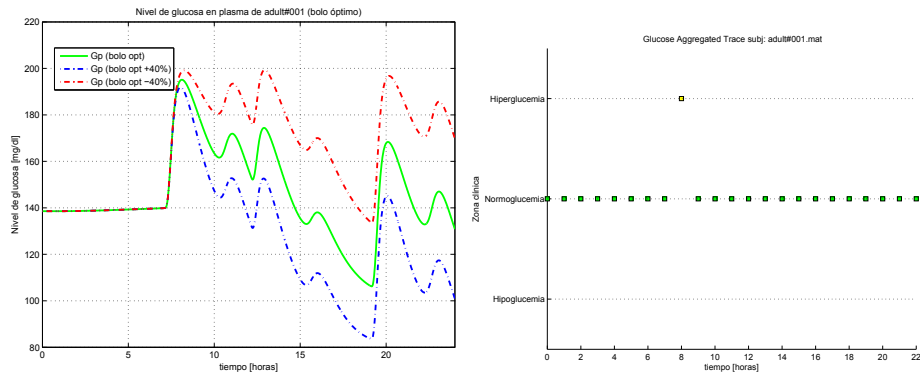


Figura 4.2: Comparación de la tasa de aparición de glucosa del modelo de Dalla Man y de las comidas mixtas

### Caso a

Se realizaron 3 simulaciones con suministro de bolo óptimo sin variación intra-paciente y sin comidas mixtas, una con los bolos exactos para cada comida, una con bolos 40% mayores y otra con bolos 40% menores, estimando error de cálculo de bolo por el usuario.



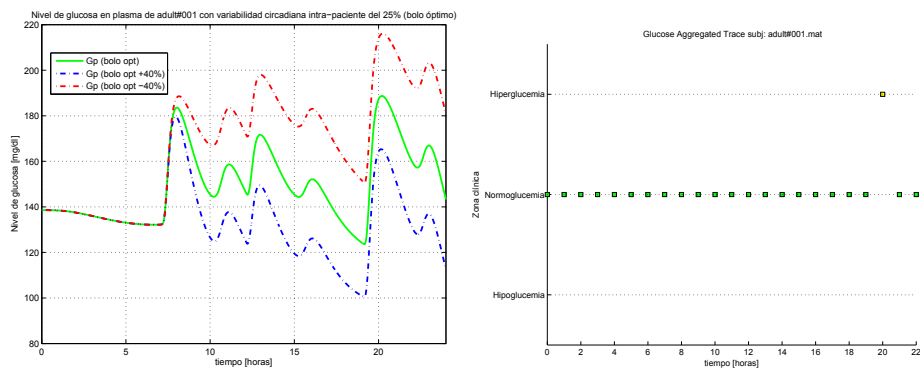
(a) Glucosa en plasma con bolo óptimo y (b) Estado del paciente con bolo óptimo con error de bolos

Figura 4.3: Caso 4.1.1a

Se observa que la relación para el bolo óptimo es conservadora, alejando al paciente de la hipo-glucemia, incluso suministrando un bolo 40% mayor.

### Caso b

Se realizaron 3 simulaciones con suministro de bolo óptimo con una variación intra-paciente circadiana de la sensibilidad insulínica y de la cinética de la insulina subcutánea del 25%, y sin comidas mixtas, una con los bolos exactos para cada comida, una con bolos 40% mayores y otra con bolos 40% menores, estimando error de cálculo de bolo por el usuario.



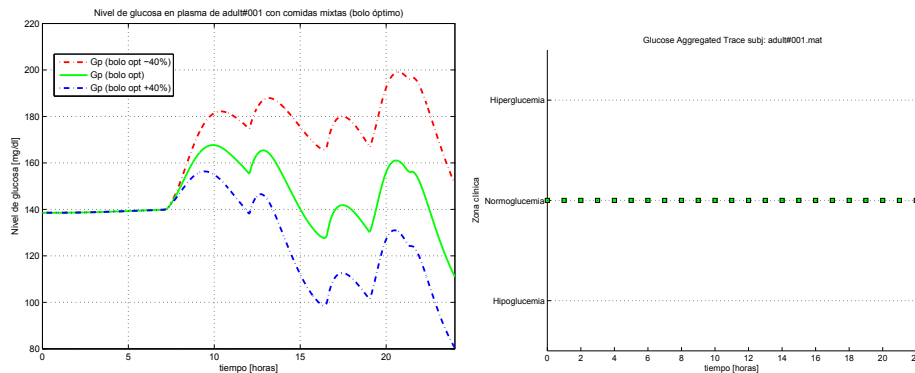
(a) Glucosa en plasma con bolo óptimo y (b) Estado del paciente con bolo óptimo con error de bolos

Figura 4.4: Caso 4.1.1b

### Caso c

Se realizaron 3 simulaciones con suministro de bolo óptimo sin variación intra-paciente y con comidas mixtas (con el menú detallado anteriormente), una con los bolos exactos

para cada comida, una con bolos 40% mayores y otra con bolos 40% menores, estimando error de calculo de bolo por el usuario.



(a) Glucosa en plasma con bolo óptimo y (b) Estado del paciente con bolo óptimo con error de bolos

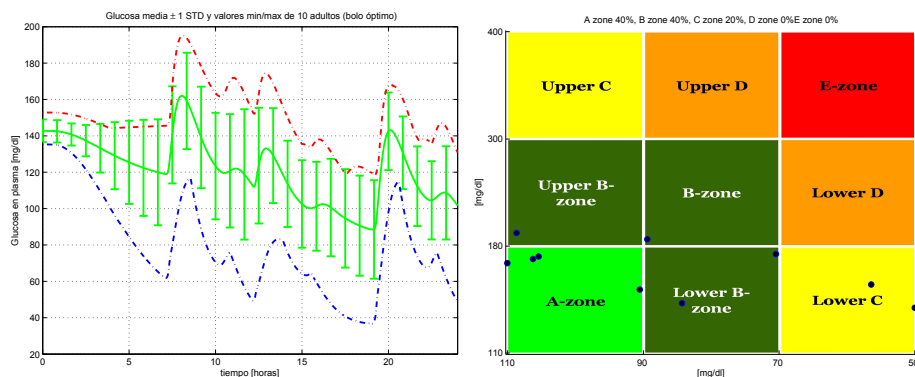
Figura 4.5: Caso 4.1.1c

### 4.1.2. Simulación N°2

Se simula una cohorte de 10 pacientes adultos a lazo abierto, suministrando los bolos óptimos para cada comida según la relación I:CHO del paciente, y la insulina basal correspondiente al nivel de ayuno, ambos valores obtenidos a partir de la información clínica que se brinda del paciente.

#### Caso a

Se suministra el bolo óptimo exacto para cada comida sin variación intra-paciente y sin comidas mixtas.



(a) Concentración de glucosa en plasma de 10 adultos con bolo óptimo (b) CVGA de 10 adultos con bolo óptimo

Figura 4.6: Caso 4.1.2a

#### Caso b

Se suministra el bolo óptimo exacto para cada comida con una variación intra-paciente circadiana de la sensibilidad insulínica y de la cinética de la insulina subcutánea del 25%,

y sin comidas mixtas.

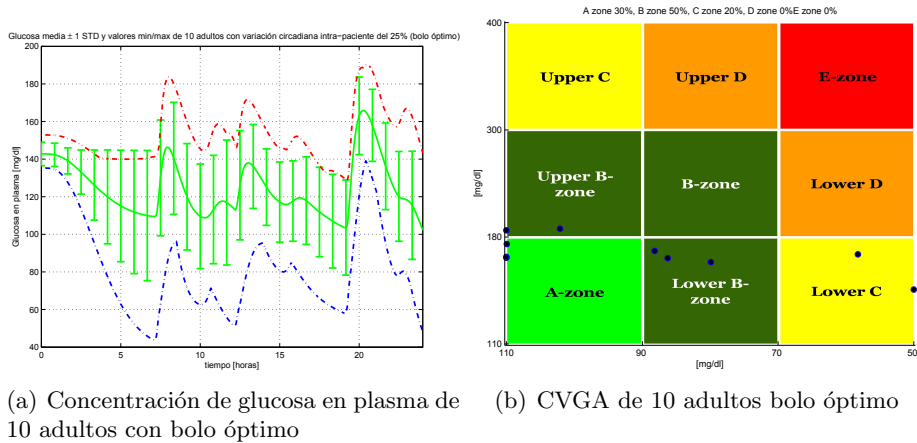


Figura 4.7: Caso 4.1.2b

**Caso c**

Se suministra el bolo óptimo exacto para cada comida sin variación intra-paciente y con comidas mixtas.

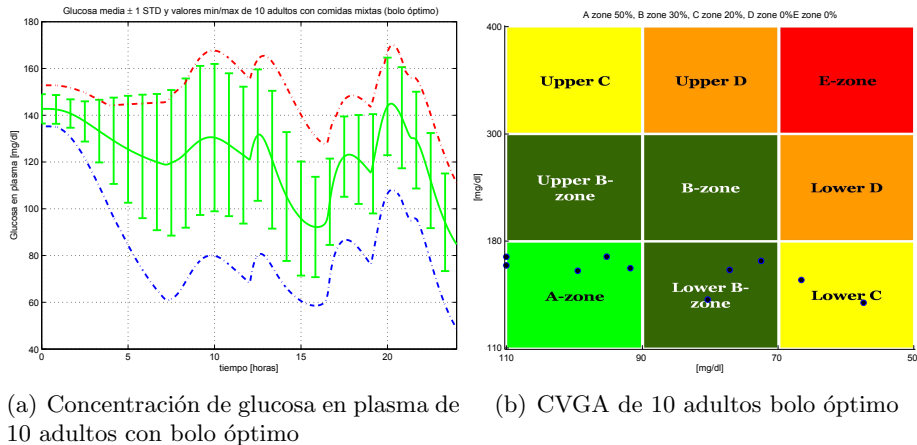


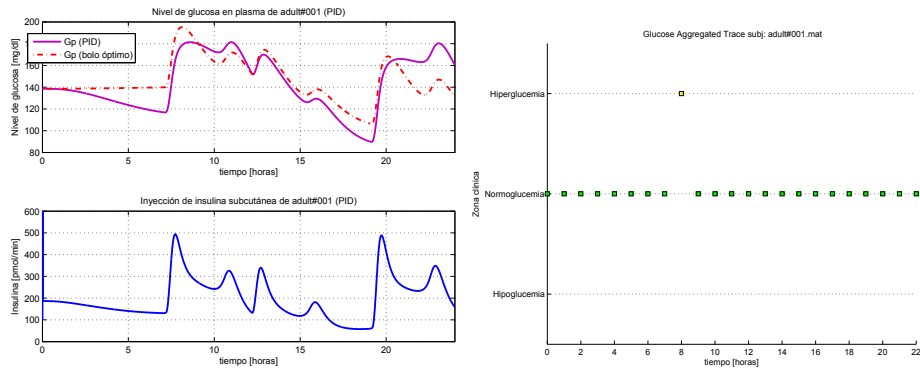
Figura 4.8: Caso 4.1.2c

**4.1.3. Simulación N°3**

Se simula al paciente adult#001 a lazo cerrado, suministrando los bolos por el controlador y la insulina basal corresponde a 1U/h. Se utilizó el  $I_{TDD}$  del paciente para sintonizar el controlador. La referencia fue ajustada a 100mg/dL, un valor promedio de normo-glucemia de adultos.

**Caso a**

El controlador suministra los bolos al paciente sin variación intra-paciente y sin comidas mixtas.



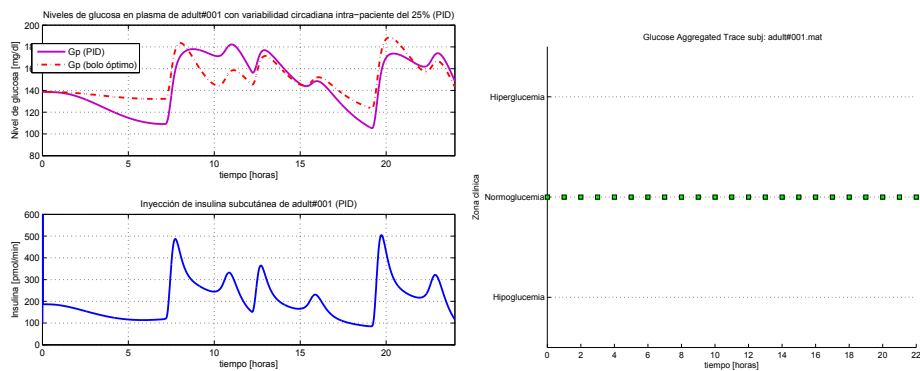
(a) Comparación de glucosa en plasma con controlador y con bolo óptimo (arriba); y inyección de insulina subcutánea por el controlador (abajo)

(b) Estado del paciente con PID

Figura 4.9: Caso 4.1.3a

### Caso b

El controlador suministra los bolos al paciente con una variación intra-paciente circadiana de la sensibilidad insulínica y de la cinética de la insulina subcutánea del 25%, y sin comidas mixtas.



(a) Comparación de glucosa en plasma con controlador y con bolo óptimo (arriba); y inyección de insulina subcutánea por el controlador (abajo)

(b) Estado del paciente con PID

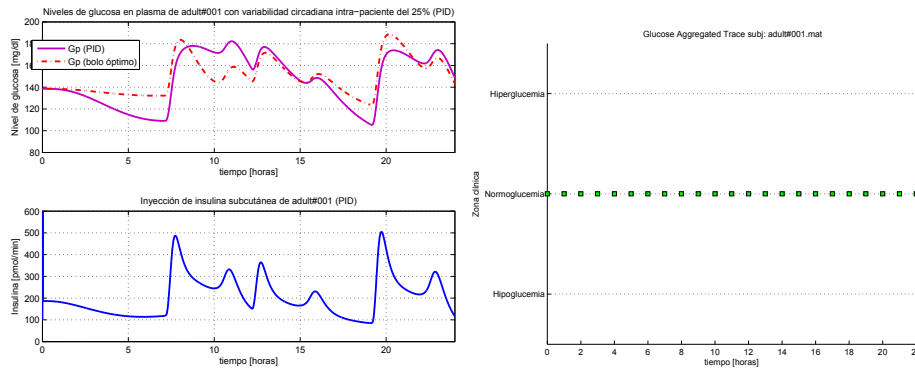
Figura 4.10: Caso 4.1.3b

### Caso c

El controlador suministra los bolos al paciente sin variación intra-paciente y con comidas mixtas.

#### 4.1.4. Simulación N°4

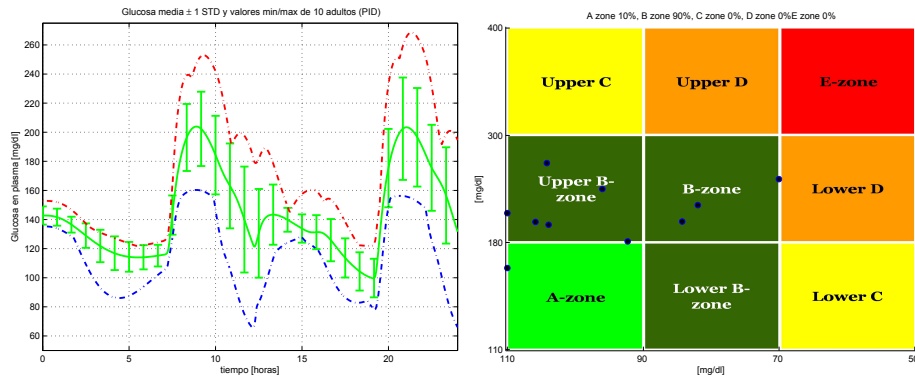
Se simula una cohorte de 10 pacientes adultos a lazo cerrado, suministrando los bolos por el controlador, y la insulina basal corresponde a 1U/h. Se utilizó el  $ITDD$  respectivo



(a) Comparación de glucosa en plasma con controlador y con bolo óptimo (arriba); y inyección de insulina subcutánea por el controlador (abajo)

(b) Estado del paciente con PID

Figura 4.11: Caso 4.1.3c



(a) Concentración de glucosa en plasma de 10 adultos con PID

(b) CVGA de 10 adultos PID

Figura 4.12: Caso 4.1.4a

de cada paciente para sintonizar el controlador durante la simulación. La referencia fue ajustada a 100mg/dL, un valor promedio de normo-glucemia de adultos.

**Caso a**

El controlador suministra los bolos a los pacientes sin variación intra-paciente y sin comidas mixtas.

**Caso b**

El controlador suministra los bolos a los pacientes con una variación intra-paciente circadiana de la sensibilidad insulínica y de la cinética de la insulina subcutánea del 25 %, y sin comidas mixtas.

**Caso c**

El controlador suministra los bolos a los pacientes sin variación intra-paciente y con comidas mixtas.



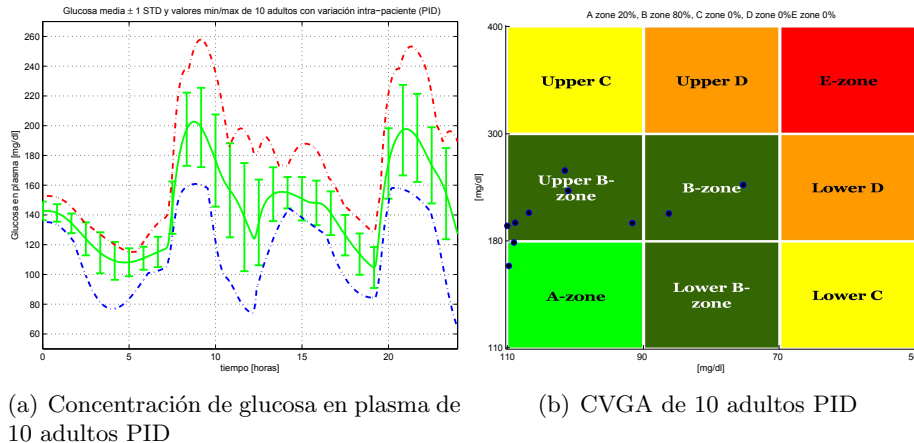


Figura 4.13: Caso 4.1.4b

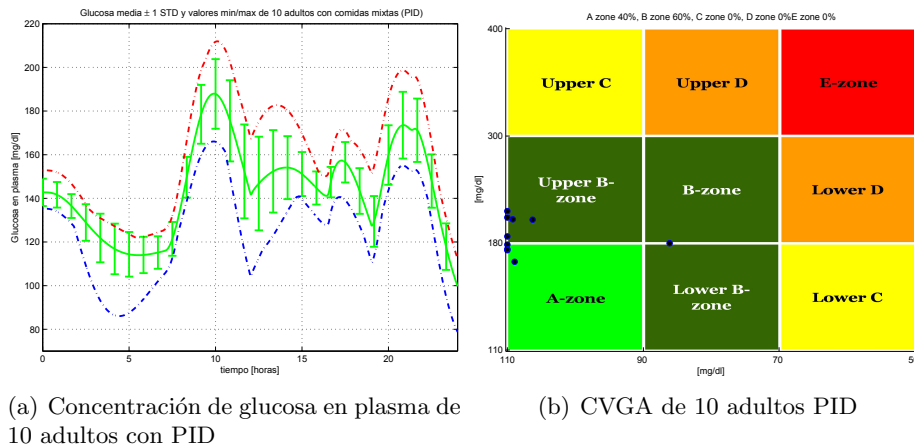


Figura 4.14: Caso 4.1.4c

## 4.2. Discusión de resultados

A partir de la simulación de 4.1.1, se puede observar cómo afecta la variación intra-paciente al control de lazo abierto por bolos óptimos, viéndose en el caso a) un estado de hiper-glucemia a la hora 8, que no se presenta en el caso b) debido a que el paciente posee una sensibilidad mayor a la dosis de la mañana, y hacia el final del día la sensibilidad disminuye, surgiendo un estado de hiper-glucemia a la hora 20.

Respecto a las comidas mixtas, en el caso c) el paciente se mantiene en normo-glucemia en parte debido a que las cantidades de CHO ingeridas son menores en comparación con el menú del escenario, salvo en el almuerzo. Pero a su vez, la tasa de aparición de glucosa es más lenta que en el modelo de Dalla Man, por lo que se metaboliza de mejor manera.

Cuando se aplica el PD en el lazo cerrado en 4.1.3, el nivel de glucosa del paciente se mantiene cercano al provocado por el suministro de bolos óptimos, lo cual es una ventaja muy importante, porque el suministro de bolos no depende del cálculo del paciente. En el caso a) se presenta un estado de hiper-glucemia a la misma hora que en el caso a) de 4.1.1, pero al agregar variación intra-paciente, el control a lazo cerrado responde de mejor manera que en la simulación de 4.1.1, manteniendo al sujeto en normo-glucemia constante, sin superar el nivel de 180mg/dL. Se observa que para las comidas mixtas, el controlador PD responde de forma adecuada, sin superar los 180mg/dL y manteniendo el estado de

normo-glucemia.

A partir de las simulaciones poblacionales, se percibe una respuesta similar para los tres casos en 4.1.4, mientras que en la simulación de lazo abierto de 4.1.2, hay una mayor variación en la respuesta de los pacientes. La dispersión en cada caso de 4.1.2 es mayor al caso correspondiente producido en 4.1.4, percibiéndose una notable mejora en el control de lazo cerrado cuando hay variación intra-paciente o comidas mixtas, o sea, cuando el escenario se vuelve más realista.



## Capítulo 5

# Conclusiones y trabajo futuro

La contribución principal del trabajo es una plataforma confiable y versátil para la evaluación in silico de sistemas de control de glucosa como paso previo a su evaluación en ensayos clínicos. Su presentación logra su cometido de uso intuitivo, tanto como la interfaz gráfica, como el desarrollo de la programación, y la muestra de resultados es útil para el análisis inmediato. Tiene una gran facilidad en agregar tanto bloques de simulación como nuevas funciones o modificaciones al modelo, lo que le da un gran potencial a seguir desarrollándolo y empleándolo en el futuro. Respecto a las virtudes en relación al simulador Uva, se encuentran:

- Diseño intuitivo en el desarrollo de la programación
- Interfaz gráfica amigable
- Facilidad para agregar bloques de simulación, agregar nuevas funciones, o modificar el modelo
- Posee escenarios con comidas mixtas
- Posee variación intra-paciente

Queda camino por recorrer respecto a las prestaciones del simulador, en especial en la construcción de escenarios aún mas realista. En una siguiente etapa se podría agregar:

- un modelo de la dinámica del glucagón
- una mejor variación intra-paciente y agregar una variación no circadiana
- una cohorte mayor de pacientes virtuales
- un modelo los efectos del ejercicio
- modelos de sensor y bomba más realistas
- posibilidad de suministro de diferentes tipos de bolo

Aún así, se puede comenzar la evaluación de algoritmos de control de lazo cerrado desarrollados en el Departamento de de Electrotecnia de la Facultad de Ingeniería de la Universidad de La Plata, colaborando en una problemática muy presente hoy en día en la población mundial.

Como comentario final, se desea que a partir de este trabajo se pueda, en un futuro próximo, mejorar la calidad de vida de personas con Diabetes Mellitus tipo 1, a partir de confiables algoritmos de control de lazo cerrado, omitiendo así el error humano de cálculo de bolos y/o el olvido.



# Apéndice A

## Códigos de programación

### A.1. Script principal

```
1
2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3 %Inicializacin
4 addpath(genpath(cd))
5 clear
6
7 %Carga de los sujetos a simular
8
9 [sujeto, s] = sujetos(); %el arreglo de cadenas sujeto contiene los sujetos seleccionados
10
11 if s==0 %en caso de cancelar la seleccin se cierra el programa
12     clear
13     return;
14 else
15
16 %Creacin de las estructuras de salida
17 data(length(sujeto)) = struct('paciente',[], 'glucosa',[], 't',[], 'salidas',[], 'estados',[], '
18     parametros',[]);
19 resultados(length(sujeto)) = struct('paciente',[], 'debajo',[], 'encima',[], 'normo',[], 'hipo',[], '
20     hiper',[], 'gmedia',[], 'gmax',[], 'gmin',[], 'numhipos',0);
21 hardware = struct('pump_bolus_min',0, 'pump_bolus_max',1000, 'pump_bolus_inc',0.05, 'pump_basal_min
22     ',0, 'pump_basal_max',1000, 'pump_basal_inc',0.05, 'pump_sampling',1, ...
23     'sensor_sampling',1, 'sensor_max',1000, 'sensor_min',0);
24 parametros = struct('ti',[], 'tf',[], 'paso',[], ...
25     'Tcomida',[], 'comida',[], 'durcomida',[], 'Tcomix',[], 'Acomix',[], 'Tbolos',[], 'bolos',[], 'basal'
26     ',[], 'Tivins',[], 'ivins',[], 'Tivd',[], 'ivd',[], ...
27     'comidas',[], 'insulina',[], 'iv',[], ...
28     'BW',[], 'EGPb',[], 'Gb',[], 'Ib',[], 'kabs',[], 'kmax',[], 'kmin',[], 'b',[], 'f',[], 'c',[], 'Vg',[], 'Vi
29     ',[], 'Gpb',[], 'Ipb',[], 'Vmx',[], ...
30     'Km0',[], 'k1',[], 'k2',[], 'Fcns',[], 'Gtb',[], 'Vm0',[], 'Rdb',[], 'PCRB',[], 'p2u',[], 'ml',[], 'm30'
31     ',[], 'm4',[], 'm2',[], 'm5',[], 'CL',[], ...
32     'HEb',[], 'Ilb',[], 'ki',[], 'kp1',[], 'kp2',[], 'kp3',[], 'ke1',[], 'ke2',[], 'ksc',[], 'kd',[], 'ka1'
33     ',[], 'ka2',[], 'dosekempt',[], 'x0',[], ...
34     'u2ss',[], 'ueq',[], 'xeq',[], 'isc1ss',[], 'isc2ss',[], ...
35     'rango',[], 'variacion',[], 'periodovmx',[], 'periodoisc',[]);
36
37 %Ventana de carga de datos
38 [settings, boton] = settingsdlg(... %esta funcin
39     fue desarrollada por Rody Oldenhuis
40     'Description', 'Configuracin de diferentes aspectos de la simulacin', ...
41     'title', 'Simulador UNLP', ...
42     'separator', 'Configuracin', ...
43     {'Nombre escenario'; 'escenario'}, 'prueba', ...
44     {'Paso de simulacin (en minutos)'; 'paso'}, 1, ...
45     {'Inicio (hs a partir 00hs)'; 'ti'}, 0, ...
46     {'Duracin de la simulacin (hs)'; 'tf'}, 24, ...
47     'separator', 'Modelo sensor', ...
48     {'Modelo sensor'; 'sensor'}, [false, true], ...
49     {'Sensores disponibles'; 'nombresensor'}, {'dexcom', 'guardianRT', 'navigator'}, ...
50     {'Ruido'; 'ruido'}, false, ...
51     'separator', 'Modelo bomba', ...
52     {'Modelo bomba'; 'bomba'}, [false, true], ...
53     {'Bombas disponibles'; 'nombrebomba'}, {'cozmo', 'insulet'}, ...
54     'separator', 'Variacin intra-paciente', ...
55     {'Variacin intra-paciente'; 'variacion'}, [false, true], ...
56     {'Sensibilidad ins.: Periodo (hs)'; 'periodovmx'}, 24, ...
57     {'Sensibilidad ins.: Amplitud (%); 'ampvmx'}, 20, ...
58     {'Sensibilidad ins.: Fase (hs)'; 'fasevmx'}, 0, ...
59     {'Cintica ins. s.c.: Periodo (hs)'; 'periodoisc'}, 24, ...
60     {'Cintica ins. s.c.: Amplitud (%); 'ampisc'}, 20, ...
61     {'Cintica ins. s.c.: Fase (hs)'; 'faseisc'}, 0, ...
62     'separator', 'Comidas mixtas', ...
63     {'Agregar comidas mixtas'; 'mix'}, false, ...
64     'separator', 'Rango de normoglucemia', ...
65     {'Lmite de hipoglucemia (mg/dl)'; 'rango1'}, 70, ...
66     {'Lmite de hiperglucemia (mg/dl)'; 'rango2'}, 180, ...
```

```

59     'separator' , 'Tipo de tratamiento' , ...
60     {'Tratamiento'; 'lazo'}, {'Lazo abierto', 'Lazo cerrado', 'Lazo hbrido'});
61
62     if ~strcmp(boton, 'OK')                %en caso de cancelar se sale del programa
63         clear
64         return;
65     end
66
67     escenario = char(settings.escenario);    %nombre del escenario especificado
68     escenario = [escenario '.scn'];
69     esc = cellstr(1s('Escenarios'));
70     if ~max(strcmp(escenario, esc))
71         error('Escenario erroneo');
72         clear
73         return;
74     end
75     parametros.ti = settings.ti*60;        %tiempo de inicio de la simulacin en minutos a
76         partir de las 0hs
77     parametros.tf = settings.tf*60;        %duracin de la simulacin
78     parametros.paso = settings.paso;      %paso de simulacin en minutos
79     t=linspace(parametros.ti, parametros.tf, parametros.tf/parametros.paso+1); %creacin del vector de
80         tiempo de simulacin
81     nombre.bomba = char(settings.nombrebomba); %bomba seleccionada
82     nombre_sensor = char(settings.nombresensor);
83     parametros.rango = [settings.rangol settings.rango2]; %rango a observar de estados clinicos
84     parametros.variacion = settings.variacion;
85     parametros.periodovmx = settings.periodovmx*60;
86     parametros.periodoisc = settings.periodoisc*60;
87     parametros.ampisc = settings.ampisc/100;
88     parametros.ampvmx = settings.ampvmx/100;
89     parametros.fasevmx = settings.fasevmx;
90     parametros.faseisc = settings.faseisc;
91     parametros.ra_comidas_mixtas = zeros(1, size(t, 2));
92     switch settings.lazo
93     case 'Lazo abierto'
94         lazo=1;lazoh=-1;
95     case 'Lazo cerrado'
96         lazo=-1;lazoh=-1;
97     case 'Lazo hbrido'
98         lazo=1;lazoh=1;
99     end
100     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
101     %Carga del escenario
102     [BGini, parametros.Tcomida, parametros.comida, parametros.durcomida, parametros.Tbolos, parametros.
103         bolos, parametros.basal, parametros.Tivd, parametros.ivd, parametros.Tivins, parametros.ivins,
104         opcion, sensibilidad, deltat]=cargar_escenario(escenario, parametros);
105     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
106     %Creacin del vector de comidas
107     if settings.mix
108         [parametros.ra_comidas_mixtas parametros.Tcomix parametros.Acomix] = creacion_comidas_mixtas
109         (parametros.ra_comidas_mixtas, t, parametros.paso);
110     end
111     parametros.comidas = creacion_comidas(parametros.Tcomida, parametros.comida, parametros.durcomida,
112         t, parametros.paso);
113     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
114     %Creacin del vector de bolos
115     if ~strcmp(opcion, 'on')
116         parametros.insulina = creacion_bolos(parametros.Tbolos, parametros.bolos, sensibilidad,
117         parametros.basal, t, parametros.paso);
118     end
119     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
120     %Creacin del vector de suministros intravenosos
121     parametros.iv=creacion_IV(parametros.Tivd, parametros.ivd, parametros.Tivins, parametros.ivins, t,
122         parametros.paso);
123     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
124     %Carga del hardware
125     hardware = cargar_hardware(nombre_bomba, nombre_sensor, hardware, settings.bomba, settings.sensor);
126     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
127     %Ventana de barra de progreso
128     h = waitbar(0, 'Simulando pacientes, por favor espere...');
129     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
130     for v=1:length(sujeto)
131         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
132         %Carga de parmetros de sujeto
133         parametros = cargar_sujeto(char(sujeto(v)), parametros);
134         parametros.x0 = condiciones_iniciales(parametros, BGini);
135         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
136         %Creacin del vector de bolos ptimos
137         if strcmp(opcion, 'on')
138             parametros.insulina = creacion_bolos_optimo(parametros.Tcomida, parametros.comida,
139                 sensibilidad, t, parametros.paso, deltat, sujeto, v, parametros, settings.mix, parametros.Tcomix
140                 , parametros.Acomix);
141         end
142     end

```

```

144 end
145 %parametros.insulina(1,:) = parametros.insulina(1,:)*.6;      %error de bolo
146
147 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
148 %Simulacin
149 crear_ruido;
150 control;
151 paramNameValStruct = config_sim();
152 simOut = sim('plataforma_de_prueba_unlp', paramNameValStruct);
153
154 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
155 %Guardado de variables de salida de la simulacin
156
157 salidas = simOut.get('salidas');
158 estados = simOut.get('estados');
159 ts = simOut.get('tout');
160
161 waitbar(v/length(sujeto));
162 if v==length(sujeto)
163     close(h);
164 end
165
166 [data, resultados, filename]=guardar(data, resultados, v, sujeto, parametros, salidas, estados,
167     escenario, ts);
168
169 end
170 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
171 %Graficos
172
173 graficos(filename);
174 clearvars -except data
175
176 end

```

## A.2. S-Function

```

1 function [sys,x0,str,ts] = diabetico_nico(t,x,u,flag,parametros)
2 % Attenzione: x(1)=stomach1, qsto1(t) (mg)
3 %             x(2)=stomach2, qsto2(t) (mg)
4 %             x(3)=intestine qgut(t) (mg)
5 %             x(4)=palsma glucose + insulin-independent tissues, Gp(t) (mg/kg)
6 %             x(5)=insulin-dependent tissues, Gt(t) (mg/kg)
7 %             x(6)=plasma insulin Ip(t) (pmol/kg)
8 %             x(7)=insulin action, X(t) (pmol/L)
9 %             x(8)=delay compartment for insulin action on glucose production, Il(t) (pmol/L)
10 %            x(9)=insulin action on glucose production, Id(t) (pmol/L)
11 %            x(10)=insulin in the liver, Il(t) (pmol/kg)
12 %            x(11)=1st compartment of subcutaneous insulin, Isc1(t) (pmol/kg)
13 %            x(12)=2nd compartment of subcutaneous insulin, Isc2(t) (pmol/kg)
14 %            x(13)=subcutaneous glucose, (mg/kg)
15
16 switch flag,
17
18     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
19     % Initialization %
20     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
21     case 0,
22         [sys,x0,str,ts]=mdlInitializeSizes(parametros);
23
24         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
25         % Derivatives %
26         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
27     case 1,
28         sys=mdlDerivatives(t,x,u,parametros);
29
30         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
31         % Output %
32         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
33     case 3,
34         sys = mdlOutputs(t,x,u,parametros);
35
36         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
37         % Unhandled flags %
38         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
39     case { 2, 4, 9 },
40         sys = [];
41
42         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
43         % Unexpected flags %
44         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
45     otherwise
46         error(['Unhandled flag = ',num2str(flag)]);
47
48 end
49
50 %end sfundscl
51
52 %
53 %

```



```

54 % mdlInitializeSizes
55 % Return the sizes, initial conditions, and sample times for the S-function.
56 %
57 %
58 function [sys,x0,str,ts]=mdlInitializeSizes(parameters)
59 sizes = simsizes;
60
61 sizes.NumContStates = 13;
62 sizes.NumDiscStates = 0;
63 sizes.NumOutputs = 22;
64 sizes.NumInputs = 6;
65 sizes.DirFeedthrough = 1;
66 sizes.NumSampleTimes = 1;
67
68
69
70 sys = simsizes(sizes);
71 x0=parameters.x0;
72 str = [];
73 ts = [0 0];
74
75 % end mdlInitializeSizes
76
77 %
78 %
79 % mdlUpdate
80 % Handle discrete state updates, sample time hits, and major time step
81 % requirements.
82 %
83 %
84 function sys=mdlDerivatives(t,x,u,parameters)
85
86 %Variacin intrapaciente
87 if parameters.variacion
88     parameters.Vmx = parameters.Vmx*(1+parameters.ampvmx*sin(2*pi/parameters.periodovmx*(t+
89         parameters.fasevmx*60)));
90     parameters.kal = parameters.kal*(1+parameters.ampisc*sin(2*pi/parameters.periodoisc*(t+
91         parameters.faseisc*60)));
92     parameters.ka2 = parameters.ka2*(1+parameters.ampisc*sin(2*pi/parameters.periodoisc*(t+
93         parameters.faseisc*60)));
94     parameters.kd = parameters.kd*(1+parameters.ampisc*sin(2*pi/parameters.periodoisc*(t+
95         parameters.faseisc*60)));
96 end
97
98 % ABSORPTION
99 %glucose in the stomach
100 qsto=x(1)+x(2);
101
102 parameters.dosekempt=u(3)+parameters.comida(find(parameters.Tcomida<=t,1,'last'));
103
104 %Stomach solid
105 sys(1)=-parameters.kmax*x(1)+u(1);
106
107 if parameters.dosekempt>0
108     alfa=5/2/(1-parameters.b)/parameters.dosekempt;
109     beta=5/2/parameters.c/parameters.dosekempt;
110     kempt=parameters.kmin+(parameters.kmax-parameters.kmin)/2*(tanh(alfa*(qsto-parameters.b*
111         parameters.dosekempt))-tanh(beta*(qsto-parameters.c*parameters.dosekempt))+2);
112 else
113     kempt=parameters.kmax;
114 end;
115 %tomach liquid
116 sys(2)=parameters.kmax*x(1)-x(2)*kempt;
117 % sys(2)=(x(2)>=0)*sys(2);
118 %intestine
119 sys(3)=kempt*x(2)-parameters.kabs*x(3);
120 % sys(3)=(x(3)>=0)*sys(3);
121
122 %Rate of appearance
123 Rat=parameters.f*parameters.kabs*x(3)/parameters.BW+u(6);
124 %Glucose Production
125 EGPT=parameters.kp1-parameters.kp2*x(4)-parameters.kp3*x(9);
126 %Glucose Utilization
127 Uuit=parameters.Fcns;
128
129 % RENAL EXCRETION ;
130 if x(4)>parameters.ke2
131     Et=parameters.kel*(x(4)-parameters.ke2);
132 else Et=0;
133 end
134
135 % GLUCOSE KINETICS;
136 sys(4)=max(EGPT,0)+Rat-Uuit-Et-parameters.k1*x(4)+parameters.k2*x(5)+u(5); %u(5)= glucosa IV
137 sys(4)=(x(4)>=0)*sys(4);
138
139 Vmt=parameters.Vm0+parameters.Vmx*x(7);
140 Kmt=parameters.Km0; %
141 Uidt=Vmt*x(5)/(Kmt+x(5));
142 sys(5)=Uidt+parameters.k1*x(4)-parameters.k2*x(5);
143 sys(5)=(x(5)>=0)*sys(5);

```

```

144
145 % INSULIN KINETICS
146 sys(6)=-(parametros.m2+parametros.m4)*x(6)+parametros.m1*x(10)+parametros.kal*x(11)+parametros.
      ka2*x(12)+u(4); %u4=Insulina IV
147 It=x(6)/parametros.Vi;
148 sys(6)=(x(6)>=0)*sys(6);
149
150 % INSULIN ACTION ON GLUCOSE UTILIZATION
151 sys(7)=-parametros.p2u*x(7)+parametros.p2u*(It-parametros.Ib);
152
153 % INSULIN ACTION ON PRODUCTION
154 sys(8)=-parametros.ki*(x(8)-It);
155
156 sys(9)=-parametros.ki*(x(9)-x(8));
157
158
159 % INSULIN IN THE LIVER (pmol/kg)
160 sys(10)=-(parametros.m1+parametros.m30)*x(10)+parametros.m2*x(6);
161 sys(10)=(x(10)>=0)*sys(10);
162
163 % SUBCUTANEOUS INSULIN KINETICS
164 sys(11)=u(2)-(parametros.kal+parametros.kd)*x(11); %u2=insulina sc
165 sys(11)=(x(11)>=0)*sys(11);
166
167 sys(12)=parametros.kd*x(11)-parametros.ka2*x(12);
168 sys(12)=(x(12)>=0)*sys(12);
169
170 % SUBCUTANEOUS GLUCOSE
171 sys(13)=(-parametros.ksc*x(13)+parametros.ksc*x(4));
172 sys(13)=(x(13)>=0)*sys(13);
173
174
175 % end mdlDerivatives
176
177 %
178 %=====
179 % mdlOutputs
180 % Return the output vector for the S-function
181 %=====
182 %
183 function sys = mdlOutputs(t,x,u,parametros)
184
185 % Variacin intrapaciente
186 if parametros.variacion
187     parametros.Vmx = parametros.Vmx*(1+parametros.ampvmx*sin(2*pi/parametros.periodovmx*(t+
      parametros.fasevmx*60)));
188     parametros.kal = parametros.kal*(1+parametros.ampisc*sin(2*pi/parametros.periodoisc*(t+
      parametros.faseisc*60)));
189     parametros.ka2 = parametros.ka2*(1+parametros.ampisc*sin(2*pi/parametros.periodoisc*(t+
      parametros.faseisc*60)));
190     parametros.kd = parametros.kd*(1+parametros.ampisc*sin(2*pi/parametros.periodoisc*(t+
      parametros.faseisc*60)));
191 end
192
193 sys(1)=x(13)/parametros.Vg; % subcutaneous glucose
194 sys(2)=x(6)/parametros.Vi; % plasma Insulin
195 sys(3)=parametros.kp1-parametros.kp2*x(4)-parametros.kp3*x(9); % Production
196 sys(4)=parametros.f*parametros.kabs*x(3)/parametros.BW+u(6); % Rate of Appearance
197 Vmt=parametros.Vm0+parametros.Vmx*x(7);
198 Kmt=parametros.Km0; %
199 Uidt=Vmt*x(5)/(Kmt+x(5));
200 sys(5)=parametros.Fcns+Uidt; % Utilization
201 sys(6)=x(11)*parametros.kal+x(12)*parametros.ka2; % Rate of appearance dell'insulina
202 sys(7)=x(4)/parametros.Vg; % plasma glucose
203 sys(8:20)=x(1:13); % model states
204 sys(21)=x(1)+x(2); % Qsto
205 sys(22)=x(11)+x(12);

```

### A.3. Sujetos

```

1 function [sujeto v] = sujetos()
2
3 sujeto = what('Sujetos'); % lista de archivos MATLAB segun tipo dentro de la carpeta
4
5 [s,v] = listdlg('ListString',sujeto.mat,'SelectionMode','multiple','Name','Simulador UNLP','
      PromptString','Lista de sujetos a simular'); % Interfaz grafica
6
7 sujeto=sujeto.mat(s); % Arreglo de cadenas con el nombre de cada archivo correspondiente a los
      pacientes
8
9 end

```

### A.4. Cargar escenario

```

1 function [BGinit, Tmeals, Ameals, Dmeals, Tbolus, Abolus, basal, TIVD, AIVD, TIVINS, AIVINS, LAoptimo,
2     sensibilidad, deltat]=cargar_escenario(filename, parametros)
3
4 %default values
5 Qmeals=0;
6 LAoptimo=[];
7 sensibilidad='';
8 meals=[0 0];
9 QTmeals='min';
10 QTmealsV=1;
11 QTbolus=1;
12 Qbasal='fixed';
13 basal=0;
14 Tbolus=[];
15 Abolus=[];
16 Qbolus='total';
17 Tmeals=[];
18 Ameals=[];
19 Qmeals='total';
20 TIVD=[0];
21 AIVD=[0];
22 QIVD='total';
23 TIVINS=[0];
24 AIVINS=[0];
25 QIVins='total';
26 Dmeals=15;
27 deltat=0;
28 BGinit=[];
29
30 %read file
31 content=textread(['Escenarios\' filename], '%', 'delimiter', '\n');
32
33 %loop on the read lines, identify known headers, and extract data if need
34 %e
35 n=length(content);
36 if n~=0
37     for i=1:n
38         line=content{i};
39         if ~isempty(line)
40             if strcmp(line(1), '%')
41                 line=line(2:end);
42
43                 if ~isempty(strfind(line, 'QTsimul'))
44                     p=strfind(line, 'QTsimul');
45                     p2=1+strfind(line(p:end), '=');
46                     QTsimul=(line(p-1+p2:end));
47                     switch QTsimul
48                         case 'min'
49                             QTsimul=1;
50                         case 'hour'
51                             QTsimul=60;
52                         case 'day'
53                             QTsimul=1440;
54                         otherwise
55                             display('bad time unit for simulation length, minutes used')
56                             QTsimul=1;
57                     end
58                 elseif ~isempty(strfind(line, 'Tsimul'))
59                     p=strfind(line, 'Tsimul');
60                     p2=1+strfind(line(p:end), '=');
61                     Tsimul=str2double(line(p-1+p2:end));
62                     if Tsimul<1
63                         display('time of simulation is too short (Tsimul<1), using 500 min')
64                         Tsimul=500;
65                     end
66                 elseif ~isempty(strfind(line, 'QTclosed'))
67                     p=strfind(line, 'QTclosed');
68                     p2=1+strfind(line(p:end), '=');
69                     QTclosed=(line(p-1+p2:end));
70                     switch QTclosed
71                         case 'min'
72                             QTclosed=1;
73                         case 'hour'
74                             QTclosed=60;
75                         case 'day'
76                             QTclosed=1440;
77                         otherwise
78                             display('bad closed loop time unit, using minutes')
79                             QTclosed=1;
80                     end
81                 elseif ~isempty(strfind(line, 'Tclosed'))
82                     p=strfind(line, 'Tclosed');
83                     p2=1+strfind(line(p:end), '=');
84                     Tclosed=str2double(line(p-1+p2:end));
85                     if Tclosed<0
86                         display('time of closed loop is negative, using 0 instead')
87                         Tclosed=0;
88                     end
89                 elseif ~isempty(strfind(line, 'QTmeals'))
90                     p=strfind(line, 'QTmeals');
91                     p2=1+strfind(line(p:end), '=');
92                     QTmeals=(line(p-1+p2:end));
93                     switch QTmeals
94                         case 'min'

```

```

95         QTmealsV=1;
96     case 'hour'
97         QTmealsV=60;
98     case 'day'
99         QTmealsV=1440;
100     otherwise
101         display('bad meal times units, using minutes')
102         QTmealsV=1;
103     end
104     elseif ~isempty(strfind(line, 'Tmeals'))
105         p=strfind(line, 'Tmeals');
106         p2=1+strfind(line(p:end), '=');
107         Tmeals=str2num(str2mat(line(p-1+p2:end)));
108
109     elseif ~isempty(strfind(line, 'Dmeals'))
110         p=strfind(line, 'Dmeals');
111         p2=1+strfind(line(p:end), '=');
112         Dmeals=str2num(str2mat(line(p-1+p2:end)));
113     elseif ~isempty(strfind(line, 'QTbolus'))
114         p=strfind(line, 'QTbolus');
115         p2=1+strfind(line(p:end), '=');
116         QTbolus=line(p-1+p2:end);
117         switch QTbolus
118             case 'min'
119                 QTbolus=1;
120             case 'hour'
121                 QTbolus=60;
122             case 'day'
123                 QTbolus=1440;
124             otherwise
125                 display('bad bolus times units, using minutes');
126         end
127     elseif ~isempty(strfind(line, 'Tbolus'))
128         p=strfind(line, 'Tbolus');
129         p2=1+strfind(line(p:end), '=');
130         Tbolus=str2num(str2mat(line(p-1+p2:end)));
131     elseif ~isempty(strfind(line, 'Ameals'))
132         p=strfind(line, 'Ameals');
133         p2=1+strfind(line(p:end), '=');
134         Ameals=str2num(str2mat(line(p-1+p2:end)))*1e3;
135     elseif ~isempty(strfind(line, 'Qmeals'))
136         p=strfind(line, 'Qmeals');
137         p2=1+strfind(line(p:end), '=');
138         Qmeals=line(p-1+p2:end);
139         if ~strcmp(Qmeals, 'total') && ~strcmp(Qmeals, 'perkg')
140             display('badly formatted meal type, using %total%')
141             Qmeals='total';
142         end
143     elseif ~isempty(strfind(line, 'Abolus'))
144         p=strfind(line, 'Abolus');
145         p2=1+strfind(line(p:end), '=');
146         Abolus=str2num(str2mat(line(p-1+p2:end)));
147
148     elseif ~isempty(strfind(line, 'Qbolus'))
149         p=strfind(line, 'Qbolus');
150         p2=1+strfind(line(p:end), '=');
151         Qbolus=(line(p-1+p2:end));
152
153     elseif ~isempty(strfind(line, 'Qbasal'))
154         p=strfind(line, 'Qbasal');
155         p2=1+strfind(line(p:end), '=');
156         Qbasal=(line(p-1+p2:end));
157     elseif ~isempty(strfind(line, 'basal'))
158         p=strfind(line, 'basal');
159         p2=1+strfind(line(p:end), '=');
160         basal=str2double(line(p-1+p2:end));
161         if basal<0 || isnan(basal)
162             display('negative or badly formatted basal rate, using subject specific
163                 basal instead');
164             Qbasal='quest';
165             basal=0;
166         end
167     elseif ~isempty(strfind(line, 'TIVINS'))
168         p=strfind(line, 'TIVINS');
169         p2=1+strfind(line(p:end), '=');
170         TIVINS=str2num(str2mat(line(p-1+p2:end)));
171     elseif ~isempty(strfind(line, 'AIVINS'))
172         p=strfind(line, 'AIVINS');
173         p2=1+strfind(line(p:end), '=');
174         AIVINS=str2num(str2mat(line(p-1+p2:end)));
175     elseif ~isempty(strfind(line, 'QIVins'))
176         p=strfind(line, 'QIVins');
177         p2=1+strfind(line(p:end), '=');
178         QIVins=(line(p-1+p2:end));
179     elseif ~isempty(strfind(line, 'QIVD'))
180         p=strfind(line, 'QIVD');
181         p2=1+strfind(line(p:end), '=');
182         QIVD=(line(p-1+p2:end));
183     elseif ~isempty(strfind(line, 'TIVD'))
184         p=strfind(line, 'TIVD');
185         p2=1+strfind(line(p:end), '=');
186         TIVD=str2num(str2mat(line(p-1+p2:end)));
187     elseif ~isempty(strfind(line, 'AIVD'))
188         p=strfind(line, 'AIVD');

```

```

189         p2=1+strfind(line(p:end),'=');
190         AIVD=str2num(str2mat(line(p-1+p2:end)));
191     elseif ~isempty(strfind(line,'deltat'))
192         p=strfind(line,'deltat');
193         p2=1+strfind(line(p:end),'=');
194         deltat=str2double(line(p-1+p2:end));
195     elseif ~isempty(strfind(line,'BGinit'))
196         p=strfind(line,'BGinit');
197         p2=1+strfind(line(p:end),'=');
198         BGinit=str2double(line(p-1+p2:end));
199         if ~isa(BGinit,'numeric') || isnan(BGinit)
200             display('bad formatting of initial glucose, using patient fasting state')
201         end
202         BGinit=[];
203     elseif ~isempty(strfind(line,'LAoptimo'))
204         p=strfind(line,'LAoptimo');
205         p2=1+strfind(line(p:end),'=');
206         LAoptimo=line(p-1+p2:end);
207         if ~strcmp(LAoptimo,'on')
208             LAoptimo='off';
209         end
210     elseif ~isempty(strfind(line,'sensibilidad'))
211         p=strfind(line,'sensibilidad');
212         p2=1+strfind(line(p:end),'=');
213         sensibilidad=str2double(line(p-1+p2:end));
214     end
215 end
216 end
217 end
218 %-----
219 %check that size of time and value vectorsz are equal and set defaults
220 if ~isempty(Dmeals)
221     if length(Dmeals)==1
222         Dmeals=Dmeals*ones(size(Tmeals));
223     end
224 else
225     Dmeals=15*ones(size(Tmeals));
226 end
227 if isempty(basal)
228     basal=0;
229 end
230 meal_check=isempty(find((size(Tmeals)==size(Ameals))==0,1)) && ...
231 isempty(find((size(Tmeals)==size(Dmeals))==0,1)) && ...
232 isempty(find((size(TIVD)==size(AIVD))==0,1));
233 bolus_check=isempty(find((size(Tbolus)==size(Abolus))==0,1)) && ...
234 isempty(find((size(TIVINS)==size(AIVINS))==0,1));
235
236 %construct scenario
237 if meal_check && bolus_check
238
239 %-----construct meals vectors-----
240 if min(Tmeals)<=0
241     Tmeals(Tmeals<=0)=1;
242     warning_message('string',...
243         'meal detected before or at t=0, time changed to t=1');
244 end
245 if min(Ameals)<=0
246     Ameals(Ameals<=0)=0;
247     warning_message('string','negative meals were deleted');
248 end
249
250 [Tmeals,ind]=sort(QTmealsV*Tmeals);
251 Ameals=Ameals(ind);
252
253 %-----construct insulin vector-----
254
255 if min(Tbolus)<=0
256     Tbolus(Tbolus<=0)=1;
257     warning_message('string',...
258         'boluses detected before or at t=0, time changed to t=1');
259 end
260 if min(Abolus)<=0
261     Abolus(Abolus<=0)=0;
262     warning_message('string','negative boluses were deleted');
263 end
264 [Tbolus,ind]=sort(QTbolus*Tbolus);
265 Abolus=Abolus(ind);
266
267 else
268     error('The meals and/or bolus time and values vectors in the file do not match')
269 end
270 else
271     error('string','The file is empty or not in the proper directory')
272 end

```

## A.5. Creación de las comidas de escenario

```

1 function [comidas] = comida(Tdosis,dosis,durcomida,t,paso)
2

```

```

3 comidas=zeros(1,size(t,2));
4 %Creacin del vector comidas
5 for i=1:size(t,2)
6     for j=1:size(Tdosis,2)
7         if i==floor((Tdosis(1,j))/paso)+1
8             if floor(durcomida(j)/paso)==0
9                 comidas(1,i)=dosis(1,j);
10            else
11                for k=0:floor(durcomida(j)/paso)-1
12                    comidas(1,i+k)=dosis(1,j)/durcomida(j);
13                end
14            end
15        end
16    end
17 end
18
19 end

```

## A.6. Creación de las comidas mixtas

```

1
2 function [comidas_mixtas,Tcomix,Acomix] = creacion_comidas_mixtas(comidas_mixtas,t,paso)
3 load('RaPoint')
4 load('CHOPoint')
5 load('Lista de comidas')
6 flag = 1;
7 j=1;
8 while flag
9     [k,v] = listdlg('ListString',Nombre,'SelectionMode','single','Name','Simulador UNLP','
10     PromptString','Lista de sujetos a simular');
11     T = str2num(char(inputdlg('Ingrese el horario de la comida (hs a partir de 00hs)','Comida
12     mixta')))*60;
13     ra=interp(1:5:425,RaPoint(k,:),1:paso:421);
14     f=(ra(end)-ra(end-1))/paso*t+ra(end)-(ra(end)-ra(end-1))/paso*421;
15     f=f(420/paso+1:end);
16     f(f <= 0)=0;
17     ra=horzcat(ra,f);
18     for i=1:(size(t,2)-T/paso)
19         comidas_mixtas(i+T/paso)=comidas_mixtas(i+T/paso)+ra(i);
20     end
21     %comidas_mixtas=comidas_mixtas(1:size(t,2));
22     Acomix(j)=CHOPoint(k)*1000;
23     Tcomix(j)=T;
24     pregunta = questdlg('Desea agregar otra comida?','Comidas mixtas','Si','No','No');
25     if strcmp(pregunta,'No')
26         flag = 0;
27     end
28     j=j+1;
29 end
30 [Tcomix ind]=sort(Tcomix);
31 Acomix=Acomix(ind);

```

## A.7. Creación de los bolos de escenario

```

1 function [insulina] = insulina (Tbolos,bolos,sensibilidad,ibasal,t,paso)
2
3 insulina=zeros(2,size(t,2));
4 durinsulina = 1;
5 bolos=6000*bolos;
6 for i=1:size(t,2)
7     for j=1:size(Tbolos,2)
8         if i==floor(Tbolos(1,j)/paso)+1
9             if floor(durinsulina/paso)==0
10                insulina(1,i)=bolos(1,j);
11            else
12                for k=0:floor(durinsulina/paso)-1
13                    insulina(1,i+k)=bolos(1,j)/durinsulina;
14                end
15            end
16        end
17    end
18 end
19
20 %nsulina basal
21 insulina(2,:) = 6000*ibasal/60;
22
23 end

```

## A.8. Creación de los suministros intravenosos de escenario

```

1 function [iv]=IV(Tivd,ivd,Tivins,ivins,t,paso)
2
3 iv=zeros(2,size(t,2));
4 for i=1:size(t,2)
5     for j=1:size(Tivd,2)
6         if i==floor(Tivd(1,j)/paso)+1
7             iv(1,i)=1e3*ivd(1,j);
8         end
9     for k=1:size(Tivins,2)
10        if i==floor(Tivins(1,k)/paso)+1
11            iv(2,i)=6000*ivins(1,k);
12        end
13    end
14 end
15
16
17 end

```

## A.9. Carga del hardware seleccionado

```

1 function hardware=load_hardware(nombre_bomba,nombre_sensor,hardware,bomba,sensor)
2 if bomba
3     content=textread(['hardware\' nombre_bomba '.pmp'],'%','delimiter','\n');
4     n=length(content);
5     if n~=0;
6         for i=1:n
7             line=content{i};
8             if ~isempty(line)
9                 if strcmp(line(1),'%')
10                    line=line(2:end);
11                    if strfind(line,'minbolus')
12                        p=strfind(line,'minbolus');
13                        p2=1+strfind(line(p:end),'=');
14                        hardware.pump_bolus_min=str2double(line(p-1+p2:end))*6000;
15                    elseif strfind(line,'maxbolus')
16                        p=strfind(line,'maxbolus');
17                        p2=1+strfind(line(p:end),'=');
18                        hardware.pump_bolus_max=str2double(line(p-1+p2:end))*6000;
19                    elseif strfind(line,'incbolus')
20                        p=strfind(line,'incbolus');
21                        p2=1+strfind(line(p:end),'=');
22                        hardware.pump_bolus_inc=str2double(line(p-1+p2:end))*6000;
23                    elseif strfind(line,'minbasal')
24                        p=strfind(line,'minbasal');
25                        p2=1+strfind(line(p:end),'=');
26                        hardware.pump_basal_min=str2double(line(p-1+p2:end))*100;
27                    elseif strfind(line,'maxbasal')
28                        p=strfind(line,'maxbasal');
29                        p2=1+strfind(line(p:end),'=');
30                        hardware.pump_basal_max=str2double(line(p-1+p2:end))*100;
31                    elseif strfind(line,'incbasal')
32                        p=strfind(line,'incbasal');
33                        p2=1+strfind(line(p:end),'=');
34                        hardware.pump_basal_inc=str2double(line(p-1+p2:end))*100;
35                    %
36                    %
37                    %
38                    %
39                    %
40                    %
41                    %
42                    %
43                    %
44                    %
45                    %
46                    %
47                    %
48                    %
49                    %
50                    %
51                    %
52                    %
53                    %
54                    %
55                    %
56                    %
57                    %
58                    %
59                    %
60                    %
61                    %
62                    %
63                    %
64                    %
65                    %
66                    %
67                    %
68                    %
69                    %
70                    %
71                    %
72                    %
73                    %
74                    %
75                    %
76                    %
77                    %
78                    %
79                    %
80                    %
81                    %
82                    %
83                    %
84                    %
85                    %
86                    %
87                    %
88                    %
89                    %
90                    %
91                    %
92                    %
93                    %
94                    %
95                    %
96                    %
97                    %
98                    %
99                    %
100                   %
101                   %
102                   %
103                   %
104                   %
105                   %
106                   %
107                   %
108                   %
109                   %
110                   %
111                   %
112                   %
113                   %
114                   %
115                   %
116                   %
117                   %
118                   %
119                   %
120                   %
121                   %
122                   %
123                   %
124                   %
125                   %
126                   %
127                   %
128                   %
129                   %
130                   %
131                   %
132                   %
133                   %
134                   %
135                   %
136                   %
137                   %
138                   %
139                   %
140                   %
141                   %
142                   %
143                   %
144                   %
145                   %
146                   %
147                   %
148                   %
149                   %
150                   %
151                   %
152                   %
153                   %
154                   %
155                   %
156                   %
157                   %
158                   %
159                   %
160                   %
161                   %
162                   %
163                   %
164                   %
165                   %
166                   %
167                   %
168                   %
169                   %
170                   %
171                   %
172                   %
173                   %
174                   %
175                   %
176                   %
177                   %
178                   %
179                   %
180                   %
181                   %
182                   %
183                   %
184                   %
185                   %
186                   %
187                   %
188                   %
189                   %
190                   %
191                   %
192                   %
193                   %
194                   %
195                   %
196                   %
197                   %
198                   %
199                   %
200                   %
201                   %
202                   %
203                   %
204                   %
205                   %
206                   %
207                   %
208                   %
209                   %
210                   %
211                   %
212                   %
213                   %
214                   %
215                   %
216                   %
217                   %
218                   %
219                   %
220                   %
221                   %
222                   %
223                   %
224                   %
225                   %
226                   %
227                   %
228                   %
229                   %
230                   %
231                   %
232                   %
233                   %
234                   %
235                   %
236                   %
237                   %
238                   %
239                   %
240                   %
241                   %
242                   %
243                   %
244                   %
245                   %
246                   %
247                   %
248                   %
249                   %
250                   %
251                   %
252                   %
253                   %
254                   %
255                   %
256                   %
257                   %
258                   %
259                   %
260                   %
261                   %
262                   %
263                   %
264                   %
265                   %
266                   %
267                   %
268                   %
269                   %
270                   %
271                   %
272                   %
273                   %
274                   %
275                   %
276                   %
277                   %
278                   %
279                   %
280                   %
281                   %
282                   %
283                   %
284                   %
285                   %
286                   %
287                   %
288                   %
289                   %
290                   %
291                   %
292                   %
293                   %
294                   %
295                   %
296                   %
297                   %
298                   %
299                   %
300                   %
301                   %
302                   %
303                   %
304                   %
305                   %
306                   %
307                   %
308                   %
309                   %
310                   %
311                   %
312                   %
313                   %
314                   %
315                   %
316                   %
317                   %
318                   %
319                   %
320                   %
321                   %
322                   %
323                   %
324                   %
325                   %
326                   %
327                   %
328                   %
329                   %
330                   %
331                   %
332                   %
333                   %
334                   %
335                   %
336                   %
337                   %
338                   %
339                   %
340                   %
341                   %
342                   %
343                   %
344                   %
345                   %
346                   %
347                   %
348                   %
349                   %
350                   %
351                   %
352                   %
353                   %
354                   %
355                   %
356                   %
357                   %
358                   %
359                   %
360                   %
361                   %
362                   %
363                   %
364                   %
365                   %
366                   %
367                   %
368                   %
369                   %
370                   %
371                   %
372                   %
373                   %
374                   %
375                   %
376                   %
377                   %
378                   %
379                   %
380                   %
381                   %
382                   %
383                   %
384                   %
385                   %
386                   %
387                   %
388                   %
389                   %
390                   %
391                   %
392                   %
393                   %
394                   %
395                   %
396                   %
397                   %
398                   %
399                   %
400                   %
401                   %
402                   %
403                   %
404                   %
405                   %
406                   %
407                   %
408                   %
409                   %
410                   %
411                   %
412                   %
413                   %
414                   %
415                   %
416                   %
417                   %
418                   %
419                   %
420                   %
421                   %
422                   %
423                   %
424                   %
425                   %
426                   %
427                   %
428                   %
429                   %
430                   %
431                   %
432                   %
433                   %
434                   %
435                   %
436                   %
437                   %
438                   %
439                   %
440                   %
441                   %
442                   %
443                   %
444                   %
445                   %
446                   %
447                   %
448                   %
449                   %
450                   %
451                   %
452                   %
453                   %
454                   %
455                   %
456                   %
457                   %
458                   %
459                   %
460                   %
461                   %
462                   %
463                   %
464                   %
465                   %
466                   %
467                   %
468                   %
469                   %
470                   %
471                   %
472                   %
473                   %
474                   %
475                   %
476                   %
477                   %
478                   %
479                   %
480                   %
481                   %
482                   %
483                   %
484                   %
485                   %
486                   %
487                   %
488                   %
489                   %
490                   %
491                   %
492                   %
493                   %
494                   %
495                   %
496                   %
497                   %
498                   %
499                   %
500                   %
501                   %
502                   %
503                   %
504                   %
505                   %
506                   %
507                   %
508                   %
509                   %
510                   %
511                   %
512                   %
513                   %
514                   %
515                   %
516                   %
517                   %
518                   %
519                   %
520                   %
521                   %
522                   %
523                   %
524                   %
525                   %
526                   %
527                   %
528                   %
529                   %
530                   %
531                   %
532                   %
533                   %
534                   %
535                   %
536                   %
537                   %
538                   %
539                   %
540                   %
541                   %
542                   %
543                   %
544                   %
545                   %
546                   %
547                   %
548                   %
549                   %
550                   %
551                   %
552                   %
553                   %
554                   %
555                   %
556                   %
557                   %
558                   %
559                   %
560                   %
561                   %
562                   %
563                   %
564                   %
565                   %
566                   %
567                   %
568                   %
569                   %
570                   %
571                   %
572                   %
573                   %
574                   %
575                   %
576                   %
577                   %
578                   %
579                   %
580                   %
581                   %
582                   %
583                   %
584                   %
585                   %
586                   %
587                   %
588                   %
589                   %
590                   %
591                   %
592                   %
593                   %
594                   %
595                   %
596                   %
597                   %
598                   %
599                   %
600                   %
601                   %
602                   %
603                   %
604                   %
605                   %
606                   %
607                   %
608                   %
609                   %
610                   %
611                   %
612                   %
613                   %
614                   %
615                   %
616                   %
617                   %
618                   %
619                   %
620                   %
621                   %
622                   %
623                   %
624                   %
625                   %
626                   %
627                   %
628                   %
629                   %
630                   %
631                   %
632                   %
633                   %
634                   %
635                   %
636                   %
637                   %
638                   %
639                   %
640                   %
641                   %
642                   %
643                   %
644                   %
645                   %
646                   %
647                   %
648                   %
649                   %
650                   %
651                   %
652                   %
653                   %
654                   %
655                   %
656                   %
657                   %
658                   %
659                   %
660                   %
661                   %
662                   %
663                   %
664                   %
665                   %
666                   %
667                   %
668                   %
669                   %
670                   %
671                   %
672                   %
673                   %
674                   %
675                   %
676                   %
677                   %
678                   %
679                   %
680                   %
681                   %
682                   %
683                   %
684                   %
685                   %
686                   %
687                   %
688                   %
689                   %
690                   %
691                   %
692                   %
693                   %
694                   %
695                   %
696                   %
697                   %
698                   %
699                   %
700                   %
701                   %
702                   %
703                   %
704                   %
705                   %
706                   %
707                   %
708                   %
709                   %
710                   %
711                   %
712                   %
713                   %
714                   %
715                   %
716                   %
717                   %
718                   %
719                   %
720                   %
721                   %
722                   %
723                   %
724                   %
725                   %
726                   %
727                   %
728                   %
729                   %
730                   %
731                   %
732                   %
733                   %
734                   %
735                   %
736                   %
737                   %
738                   %
739                   %
740                   %
741                   %
742                   %
743                   %
744                   %
745                   %
746                   %
747                   %
748                   %
749                   %
750                   %
751                   %
752                   %
753                   %
754                   %
755                   %
756                   %
757                   %
758                   %
759                   %
760                   %
761                   %
762                   %
763                   %
764                   %
765                   %
766                   %
767                   %
768                   %
769                   %
770                   %
771                   %
772                   %
773                   %
774                   %
775                   %
776                   %
777                   %
778                   %
779                   %
780                   %
781                   %
782                   %
783                   %
784                   %
785                   %
786                   %
787                   %
788                   %
789                   %
790                   %
791                   %
792                   %
793                   %
794                   %
795                   %
796                   %
797                   %
798                   %
799                   %
800                   %
801                   %
802                   %
803                   %
804                   %
805                   %
806                   %
807                   %
808                   %
809                   %
810                   %
811                   %
812                   %
813                   %
814                   %
815                   %
816                   %
817                   %
818                   %
819                   %
820                   %
821                   %
822                   %
823                   %
824                   %
825                   %
826                   %
827                   %
828                   %
829                   %
830                   %
831                   %
832                   %
833                   %
834                   %
835                   %
836                   %
837                   %
838                   %
839                   %
840                   %
841                   %
842                   %
843                   %
844                   %
845                   %
846                   %
847                   %
848                   %
849                   %
850                   %
851                   %
852                   %
853                   %
854                   %
855                   %
856                   %
857                   %
858                   %
859                   %
860                   %
861                   %
862                   %
863                   %
864                   %
865                   %
866                   %
867                   %
868                   %
869                   %
870                   %
871                   %
872                   %
873                   %
874                   %
875                   %
876                   %
877                   %
878                   %
879                   %
880                   %
881                   %
882                   %
883                   %
884                   %
885                   %
886                   %
887                   %
888                   %
889                   %
890                   %
891                   %
892                   %
893                   %
894                   %
895                   %
896                   %
897                   %
898                   %
899                   %
900                   %
901                   %
902                   %
903                   %
904                   %
905                   %
906                   %
907                   %
908                   %
909                   %
910                   %
911                   %
912                   %
913                   %
914                   %
915                   %
916                   %
917                   %
918                   %
919                   %
920                   %
921                   %
922                   %
923                   %
924                   %
925                   %
926                   %
927                   %
928                   %
929                   %
930                   %
931                   %
932                   %
933                   %
934                   %
935                   %
936                   %
937                   %
938                   %
939                   %
940                   %
941                   %
942                   %
943                   %
944                   %
945                   %
946                   %
947                   %
948                   %
949                   %
950                   %
951                   %
952                   %
953                   %
954                   %
955                   %
956                   %
957                   %
958                   %
959                   %
960                   %
961                   %
962                   %
963                   %
964                   %
965                   %
966                   %
967                   %
968                   %
969                   %
970                   %
971                   %
972                   %
973                   %
974                   %
975                   %
976                   %
977                   %
978                   %
979                   %
980                   %
981                   %
982                   %
983                   %
984                   %
985                   %
986                   %
987                   %
988                   %
989                   %
990                   %
991                   %
992                   %
993                   %
994                   %
995                   %
996                   %
997                   %
998                   %
999                   %
1000                  %

```

```

70         elseif strfind(line, 'xi')
71             p=strfind(line, 'xi');
72             p2=1+strfind(line(p:end), '=');
73             hardware.sensor_xi=str2double(line(p-1+p2:end));
74         elseif strfind(line, 'lambda')
75             p=strfind(line, 'lambda');
76             p2=1+strfind(line(p:end), '=');
77             hardware.sensor_lambda=str2double(line(p-1+p2:end));
78
79         elseif strfind(line, 'min')
80             p=strfind(line, 'min');
81             p2=1+strfind(line(p:end), '=');
82             hardware.sensor_min=str2double(line(p-1+p2:end));
83         elseif strfind(line, 'max')
84             p=strfind(line, 'max');
85             p2=1+strfind(line(p:end), '=');
86             hardware.sensor_max=str2double(line(p-1+p2:end));
87         elseif strfind(line, 'sampling')
88             p=strfind(line, 'sampling');
89             p2=1+strfind(line(p:end), '=');
90             hardware.sensor_sampling=str2double(line(p-1+p2:end));
91         end
92     end
93 end
94 end
95 end
96 end

```

## A.10. Carga de los parámetros de los pacientes a simular

```

1 function [parametros] = carga_sujeto(sujeto, parametros)
2
3 load(sujeto)
4
5 %Carga de parámetros del sujeto
6 parametros.BW=BW;
7 parametros.EGPb=EGPb;
8 parametros.Gb=Gb;
9 parametros.Ib=Ib;
10 parametros.kabs=kabs;
11 parametros.kmax=kmax;
12 parametros.kmin=kmin;
13 parametros.b=b;
14 parametros.f=f;
15 parametros.c=d;
16 parametros.Vg=Vg;
17 parametros.Vi=Vi;
18 parametros.Gpb=Gpb;
19 parametros.Ipb=Ipb;
20 parametros.Vmx=Vmx;
21 parametros.Km0=Km0;
22 parametros.k1=k1;
23 parametros.k2=k2;
24 parametros.Fcns=Fsc;
25 parametros.Gtb=Gtb;
26 parametros.Vm0=Vm0;
27 parametros.Rdb=Rdb;
28 parametros.PCRb=PCRb;
29 parametros.p2u=p2u;
30 parametros.m1=m1;
31 parametros.m30=m30;
32 parametros.m4=m4;
33 parametros.m2=m2;
34 parametros.m5=m5;
35 parametros.CL=CL;
36 parametros.HEb=HEb;
37 parametros.Ilb=Ilb;
38 parametros.ki=ki;
39 parametros.kp1=kp1;
40 parametros.kp2=kp2;
41 parametros.kp3=kp3;
42 parametros.ke1=ke1;
43 parametros.ke2=ke2;
44 parametros.ksc=ksc;
45 parametros.kd=kd;
46 parametros.kal=ka1;
47 parametros.ka2=ka2;
48 parametros.dosekempt=dosekempt;
49 parametros.x0=x0;
50 parametros.u2ss=u2ss;
51 parametros.ueq=[0 u2ss];
52 parametros.xeq=parametros.x0;
53 parametros.isc1ss=isc1ss;
54 parametros.isc2ss=isc2ss;
55
56 end

```



## A.11. Creación de bolo óptimo por paciente

```

1 function [insulina] = creacion_bolos_optimo(Tdosis, dosis, sensibilidad, t, paso, deltat, sujeto, v,
2     parametros, mix, Tcomix, Acomix)
3 insulina=zeros(2, size(t,2));
4 durinsulina = 1;
5 if mix
6     Tdosis=[Tdosis Tcomix];
7     dosis=[dosis Acomix];
8     [Tdosis ind]=sort(Tdosis);
9     dosis=dosis(ind);
10 end
11 if isempty(sensibilidad)
12     load OB
13     for j=1:size(CR,2)
14         if strcmp(char(sujeto(v)), [CR(j).names '.mat'])
15             sensibilidad = CR(j).values*1000;
16         end
17     end
18 end
19 for i=1:size(dosis,2)
20     bolos(i) = 6000*dosis(1,i)/sensibilidad; %6000 pasa unidades internacionales a pmol
21     Tbolos=Tdosis-deltat;
22 end
23 for i=1:size(t,2)
24     for j=1:size(Tbolos,2)
25         if i==floor(Tbolos(1,j)/paso)+1
26             if floor(durinsulina/paso)==0
27                 insulina(1,i)=bolos(1,j);
28             else
29                 for k=0:floor(durinsulina/paso)-1
30                     insulina(1,i+k)=bolos(1,j)/durinsulina;
31                 end
32             end
33         end
34     end
35 end
36
37 %ansulina basal
38 insulina(2,:) = ones(1, size(t,2))*parametros.u2ss*6000/60;
39
40
41 end

```

## A.12. Cálculo de condiciones iniciales

```

1 function [x0] = condiciones_iniciales(parametros, BGini)
2
3 %Se recalculan las condiciones iniciales si se cambio los niveles de glucosa en plasma iniciales
4
5 if ~isempty(BGini)
6     Gpop=BGini*parametros.Vg;
7     GGta=-parametros.k2-parametros.Vmx*parametros.k2/parametros.kp3;
8     GGtb=parametros.k1*Gpop-parametros.k2*parametros.Km0-parametros.Vm0+parametros.Vmx*
9         parametros.Ib+(parametros.Vmx*(parametros.k1+parametros.kp2)*Gpop-parametros.Vmx*
10            parametros.kp1+parametros.Vmx*parametros.Fcns)/parametros.kp3;
11     GGtc=parametros.k1*Gpop*parametros.Km0;
12     Gtop=(-GGtb-sqrt(GGtb^2-4*GGta*GGtc))/(2*GGta);
13     Idop=(-(parametros.k1+parametros.kp2)*Gpop+parametros.k2*Gtop+parametros.kp1-parametros.Fcns
14         )/parametros.kp3;
15     Ipop=Idop*parametros.Vi;
16     ILOp=parametros.m2*Ipop/(parametros.m1+parametros.m30);
17     Xop=Ipob/parametros.Vi-parametros.Ib;
18     iscl0p=((parametros.m2+parametros.m4)*Ipob-parametros.m1*ILOp)/(parametros.kal+parametros.kd
19         );
20     isc20p=parametros.kd*iscl0p/parametros.ka2;
21     u20p=(parametros.kal+parametros.kd)*iscl0p;
22     x0=[0 0 0 Gpop Gtop Ipop Xop Idop Idop ILOp iscl0p isc20p Gpop];
23 else
24     x0=parametros.x0;
25 end
26 end

```

## A.13. Configuración de la simulación

```

1 function [parametros_simu] = config_sim()
2
3 %Configuracin de la simulacin de simulink
4 parametros_simu.SimulationMode = 'normal';
5 parametros_simu.SaveState = 'on';

```

```

6  parametros_simu.StateSaveName = 'estados';
7  parametros_simu.SaveOutput    = 'on';
8  parametros_simu.OutputSaveName = 'salidas';
9  parametros_simu.StartTime     = 'parametros.ti';
10 parametros_simu.StopTime      = 'parametros.tf';
11 parametros_simu.SaveTime      = 'on';
12 parametros_simu.TimeSaveName   = 'tout';
13 parametros_simu.SolverType     = 'Fixed-step';
14 parametros_simu.Solver        = 'ode4';
15 parametros_simu.Fixedstep      = 'parametros.paso';
16 parametros_simu.LimitDataPoints = 'off';
17
18 end
19
20 %Solvers de paso fijo
21 % ode3 (Bogacki-Shampine)
22 % ode8 (Dormand-Prince RK8(7))
23 % ode5 (Dormand-Prince)
24 % ode4 (Runge-Kutta)
25 % ode2 (Heun)
26 % ode1 (Euler)
27 % ode14x (extrapolation)
28
29 %Para mas informacin ver:
30 % http://www.mathworks.com/help/simulink/gui/solver-pane.html#bq7cmsp-1
    _1

```

## A.14. Crear ruido

```

1
2  if settings.ruido
3      parametros.ruido=[t' zeros(size(t'))];
4      % create a normally distributed AR(1) time series with ruido mean 0 and variance 1
5      v=randn(floor(parametros.tf/parametros.paso/15),1);
6      e(1)=v(1);
7      for i=2:parametros.tf/parametros.paso/15
8          e(i)=hardware.sensor_PACF*v(i)+hardware.sensor_PACF*e(i-1);
9      end
10     % transform the standard normally distributed TS to obtain proper sensor
11     % ruido distribution using Johnson family of distributions.
12     parametros.ruido(:,2)=interp(0:15:(length(e)-1)*15,Johnson.transform(hardware.sensor_type
13         ,...
14         hardware.sensor_gamma,hardware.sensor_delta,hardware.sensor_lambda,hardware.sensor_xi,e)
15         ,0:parametros.paso:parametros.tf,'linear','extrap');
16 else
17     parametros.ruido=[t' zeros(size((t')))];
18 end

```

## A.15. Guardado de variables

```

1  function [data,resultados,archi]=guardar(data,resultados,i,sujeto,parametros,salidas,estados,
2      escenario,t)
3  archi = [];
4  data(i).paciente = char(sujeto(i));
5  data(i).glucosa = salidas(:,7);
6  data(i).t = t;
7  data(i).salidas = salidas;
8  data(i).estados = estados;
9  data(i).parametros = parametros;
10 data(i).escenario = escenario;
11
12 resultados(i).paciente = char(sujeto(i));
13 resultados(i).numhipos = 0;
14 resultados(i).hipo = 100*sum(salidas(:,7)<parametros.rango(1))/length(salidas(:,7));
15 resultados(i).hiper = 100*sum(salidas(:,7)>parametros.rango(2))/length(salidas(:,7));
16 resultados(i).gmedia = mean(salidas(:,7));
17 resultados(i).gmax = max(salidas(:,7));
18 if min(salidas(:,7))>0
19     resultados(i).gmin = min(salidas(:,7));
20 else
21     resultados(i).gmin = 0;
22 end
23 for j=1:length(t)-1
24     if data(i).glucosa(j)>parametros.rango(1) && data(i).glucosa(j+1)<parametros.rango(1)
25         resultados(i).numhipos=resultados(i).numhipos+1;
26     end
27 end
28 if i==length(sujeto)
29     cd Sim_data
30     archi = inputdlg('Ingrese el nombre del archivo','Guardar simulacin',1,{ 'data' });
31     archi=char(archi);
32     save(archi,'data');

```

```

33     save('resultados','resultados');
34
35 %Excel
36 if strcmp(archi,'data')
37     comp=fopen('resultados.xlsx');
38     if comp~=1
39         fclose(comp);
40         if size(xlsread('resultados.xlsx'),1)>0
41             filename = cd;
42             filename = strcat(filename,'\resultados.xlsx');
43             %Open Excel as a COM Automation server
44             Excel = actxserver('Excel.Application');
45             %Open Excel workbook
46             Workbook = Excel.Workbooks.Open(filename);
47             %Clear the content of the sheet
48             tam = strcat('A2:G',num2str(size(xlsread('resultados.xlsx'),1)));
49             Workbook.Worksheets.Item('Hojal').Range(tam).ClearContents;
50             %Now save/close/quit/delete
51             Workbook.Save;
52             Excel.Workbook.Close;
53             invoke(Excel,'Quit');
54             delete(Excel);
55         end
56     end
57     filename = 'resultados.xlsx';
58 else
59     filename = archi;
60 end
61
62 A = {'Paciente','Glucosa media','Glucosa mxima','Glucosa mnima','# hipos','Tiempo en hipo %','
63     Tiempo en hiper %'};
64 for i=1:length(sujeto)
65     A(i+1,:) = {resultados(i).paciente,resultados(i).gmedia,resultados(i).gmax,resultados(i).
66         gmin,resultados(i).numhipos,resultados(i).hipo,resultados(i).hiper};
67 end
68 xlsxwrite(filename,A);
69
70 cd ..
71 end

```

## A.16. Gráficos

```

1 function [] = Graficos(archivo)
2 addpath(genpath(cd))
3 load(archivo);
4 n=length(data);
5
6 [settings boton] = settingsdlg(...
7     'title','Grficos',...
8     'separator','Grficos individuales',...
9     {'Glucosa en plasma';'gp'},false,...
10    {'Glucosa intersticial';'gi'},false,...
11    {'Glucosa intersticial (sensor)';'sensor'},false,...
12    {'Inyección de insulina subcutnea (seal)';'isc'},false,...
13    {'Inyección de insulina subcutnea (bomba)';'bomb'},false,...
14    {'IOB';'iob'},false,...
15    {'Tasa de aparicin gluc/ins y uti/prod de gluc';'fisio'},false,...
16    {'Estado del paciente';'aggr'},false,...
17    'separator','Grficos poblacionales',...
18    {'Glucosa media + std + excursin';'gm'},false,...
19    {'CVGA';'cvga'},false);
20
21 if ~strcmp(boton,'OK')
22     return;
23 end
24
25 for i=1:n
26     if (settings.gp||settings.gi||settings.sensor)&&(settings.isc||settings.bomb)
27         figure
28         subplot(2,1,1)
29         if settings.gp
30             plot(data(i).t/60,data(i).glucosa,'r','DisplayName','Glucosa en plasma')
31             legend('-DynamicLegend');
32             hold all
33         end
34         if settings.gi
35             plot(data(i).t/60,data(i).salidas(:,1),'g','DisplayName','Glucosa intersticial')
36             legend('-DynamicLegend');
37             hold all
38         end
39         if settings.sensor
40             plot(data(i).t/60,data(i).salidas(:,21),'b','Displayname','Glucosa sensor')
41             legend('-DynamicLegend');
42             hold all
43         end
44         xlim([data(i).t(1)/60 data(i).t(end)/60])
45         title(['Niveles de glucosa en plasma, intersticial y en salida de sensor de ',data(i).
46             paciente]);ylabel('Nivel de glucosa [mg/dl]');xlabel('tiempo [horas]');

```

```

46     grid on
47     subplot(2,1,2)
48     if settings.bomb
49         plot(data(i).t/60,data(i).salidas(:,23),'Displayname','Iny. sc. bomba')
50             legend('-DynamicLegend');
51         hold all
52     end
53     if settings.isc
54         plot(data(i).t/60,data(i).salidas(:,24),'Displayname','Iny. sc. seal')
55             legend('-DynamicLegend');
56         hold all
57     end
58     title(['Inyección de insulina subcutánea de ',data(i).paciente]),ylabel('Insulina [pmol/
59         min]'),xlabel('tiempo [horas]'),xlim([data(i).t(1)/60 data(i).t(end)/60]);
60 elseif (settings.gp||settings.gi||settings.sensor)
61     figure
62     if settings.gp
63         plot(data(i).t/60,data(i).glucosa,'r','Displayname','Glucosa plasma')
64             legend('-DynamicLegend');
65         hold all
66     end
67     if settings.gi
68         plot(data(i).t/60,data(i).salidas(:,1),'g','Displayname','Glucosa intersticial')
69             legend('-DynamicLegend');
70         hold all
71     end
72     if settings.sensor
73         plot(data(i).t/60,data(i).salidas(:,21),'b','Displayname','Glucosa sensor')
74             legend('-DynamicLegend');
75         hold all
76     end
77     xlim([data(i).t(1)/60 data(i).t(end)/60])
78     title(['Niveles de glucosa en plasma, intersticial y en salida de sensor de ',data(i).
79         paciente]);ylabel('Nivel de glucosa [mg/dl]');xlabel('tiempo [horas]');
80 elseif (settings.isc||settings.bomb)
81     figure
82     if settings.bomb
83         plot(data(i).t/60,data(i).salidas(:,23),'Displayname','Iny. sc. bomba')
84             legend('-DynamicLegend');
85         hold all
86     end
87     if settings.isc
88         plot(data(i).t/60,data(i).salidas(:,24),'Displayname','Iny. sc. seal')
89             legend('-DynamicLegend');
90         hold all
91     end
92     title(['Inyección de insulina subcutánea de ',data(i).paciente]),ylabel('Insulina [pmol/
93         min]'),xlabel('tiempo [horas]'),xlim([data(i).t(1)/60 data(i).t(end)/60]);
94 end
95 if settings.iob
96     figure
97     plot(data(i).t/60,data(i).salidas(:,22)*data(i).parametros.BW/6000),ylabel('IOB [U/min]')
98         ,xlabel('tiempo [horas]'),xlim([data(i).t(1)/60 data(i).t(end)/60]);
99     title(['IOB de ',data(i).paciente])
100    grid on
101 end
102 if settings.fisio
103     figure
104     subplot(2,2,1),plot(data(i).t/60,data(i).salidas(:,4)),xlim([data(i).t(1)/60 data(i).t(
105         end)/60])
106     title(['Tasa de aparición de glucosa de ',data(i).paciente]);ylabel('Ra [mg/kg/min]');
107     xlabel('tiempo [horas]'),xlim([data(i).t(1)/60 data(i).t(end)/60]);
108     grid on
109     subplot(2,2,2),plot(data(i).t/60,data(i).salidas(:,5)),xlim([data(i).t(1)/60 data(i).t(
110         end)/60])
111     title(['Utilización de glucosa de ',data(i).paciente]);ylabel('Ug [mg/kg/min]');xlabel('
112         tiempo [horas]'),xlim([data(i).t(1)/60 data(i).t(end)/60]);
113     grid on
114     subplot(2,2,3),plot(data(i).t/60,data(i).salidas(:,6)),xlim([data(i).t(1)/60 data(i).t(
115         end)/60])
116     title(['Tasa de aparición de la insulina de ',data(i).paciente]);ylabel('Rai [mg/kg/min]')
117         ;xlabel('tiempo [horas]'),xlim([data(i).t(1)/60 data(i).t(end)/60]);
118     grid on
119     subplot(2,2,4),plot(data(i).t/60,data(i).salidas(:,3)),xlim([data(i).t(1)/60 data(i).t(
120         end)/60])
121     title(['Producción de glucosa de ',data(i).paciente]);ylabel('EGP [mg/kg/min]');xlabel('
122         tiempo [horas]'),xlim([data(i).t(1)/60 data(i).t(end)/60]);
123     grid on
124 end
125 if settings.aggr
126     if data(i).t/60<3
127         warning('Tiempo de simulación escaso para graficar el nivel clínico del sujeto por
128             hora');
129     else
130         figure
131         T1=[];
132         T2=[];
133         T3=[];
134         for j=1:(60/data(i).parametros.paso):length(data(i).t)-60/data(i).parametros.paso-
135             mod(length(data(i).t),60) %glucosa media por hora y comparación de
136             la media con los límites fijados

```

```

125         if mean(data(i).glucosa(j:(j+60/data(i).parametros.paso-1)))<data(i).parametros.
            rango(1)                                %Los vectores Ti poseen las horas
            donde se desarroll c/zona clinica
126         T1=[T1 (j*data(i).parametros.paso-1)/60];
127     elseif mean(data(i).glucosa(j:(j+60/data(i).parametros.paso-1)))<=data(i).
        parametros.rango(2)
128         T2=[T2 (j*data(i).parametros.paso-1)/60];
129     elseif mean(data(i).glucosa(j:(j+60/data(i).parametros.paso-1)))>data(i).
        parametros.rango(2)
130         T3=[T3 (j*data(i).parametros.paso-1)/60];
131     end
132     end
133     if ~isempty(T1)
134         plot(T1,-1,'ksquare','MarkerFaceColor',[1 0 0])
135     end
136     hold on
137     if ~isempty(T2)
138         plot(T2,0,'ksquare','MarkerFaceColor',[0 1 0])
139     end
140     if ~isempty(T3)
141         plot(T3,1,'ksquare','MarkerFaceColor',[1 1 0])
142     end
143     plot([0 find(data(i).t*data(i).parametros.paso,1,'last')],[-1 -1],':k')
144     plot([0 find(data(i).t*data(i).parametros.paso,1,'last')],[0 0],':k')
145     plot([0 find(data(i).t*data(i).parametros.paso,1,'last')],[1 1],':k')
146     title('Glucose Aggregated Trace subj: ' data(i).paciente])
147     xlabel('tiempo [horas]')
148     ylabel('Zona clinica')
149     axis([0 max([T1 T2 T3]) -1.5 1.5])
150     set(gca,'YTick',[-1 0 1],'YtickLabel',{'Hipoglucemia','Normoglucemia','Hiperglucemia'})
151     drawnow
152     end
153     end
154 end
155 if settings.gm
156     figure
157     for i=1:n
158         G(i,:)=data(i).glucosa;
159     end
160     mg=mean(G,1);
161     sdg=std(G,0,1);
162     maxg=max(G);
163     ming=min(G);
164
165     plot(data(1).t/60,mg,'g','LineWidth',2)
166     hold on
167     plot(data(1).t/60,sdg+mg,'-b')
168     plot(data(1).t/60,mg-sdg,'-b')
169     plot(data(1).t/60,maxg,'-r')
170     plot(data(1).t/60,ming,'-r')
171     title('Glucosa media \pm 1 STD (azul) y valores min/max (rojo)')
172     xlabel('tiempo [horas]')
173     ylabel('Glucosa en plasma [mg/dl]')
174     xlim([data(i).t(1)/60 data(i).t(end)/60])
175     grid on
176     drawnow
177     figure
178     sdg1=sdg;
179     for k=1:length(data(i).t)
180         if mod(data(i).t(k),50)~=0
181             sdg1(k)=NaN;
182         end
183     end
184     errorbar(data(1).t/60,mg,sdg1)
185     hold on
186     plot(data(1).t/60,maxg,'-r')
187     plot(data(1).t/60,ming,'-r')
188     title('Glucosa media \pm 1 STD y valores min/max (rojo)',xlim([data(i).t(1)/60 data(i).t(
        end)/60]),xlabel('tiempo [horas]'),ylabel('Glucosa en plasma [mg/dl]')
189     grid on
190 end
191 if settings.cvga
192     figure
193     load CVGA
194     imagesc(CVEG)
195     set(gca,'Ytick',[10 833 1653 2472],'YtickLabel',[400 300 180 110])
196     axis([0 2482 0 2482])
197     set(gca,'Xtick',[10 832 1652 2472],'XtickLabel',[110 90 70 50])
198     set(gca,'Box','off')
199     hold on
200     xlabel(' [mg/dl] ')
201     ylabel(' [mg/dl] ')
202     for i=1:n
203         T=data(i).t;
204         G=data(i).glucosa;
205         [f,x]=ecdf(G);
206         mini(i)=min([110 max([50 x(find(f<=0.025,1,'last'))])]);
207         maxi(i)=max([110 min([400 x(find(f<=0.975,1,'last'))])]);
208     end
209     plot(bmin(1)*mini+bmin(2),bmax(1)*maxi.^3+bmax(2)*maxi.^2+bmax(3)*maxi+bmax(4),'o','
        MarkerFaceColor',[0 0 0])
210     A=round(100*sum(maxi<=180 & mini>=90)/length(maxi));
211     B=round(100*sum((maxi >180 & maxi<=300 & mini>=70) | (mini<90 & maxi<=180 & mini>=70))/
        length(maxi));

```

```
212 C=round(100*sum((maxi >300 & mini >=90) | (mini <70 & maxi <=180))/length(maxi));
213 D=round(100*sum((maxi >300 & mini <90 & mini >=70) | (mini <70 & maxi >180 & maxi <300))/length(
maxi));
214 E=round(100*sum((maxi >300 & mini <70))/length(maxi));
215 if A+B+C+D+E <100
216     t=find(max([A B C D E])==[A B C D E],1,'first');
217     switch t
218         case 1
219             A=A+100-A-B-C-D-E;
220         case 2
221             B=B+100-A-B-C-D-E;
222         case 3
223             C=C+100-A-B-C-D-E;
224         case 4
225             D=D+100-A-B-C-D-E;
226         otherwise
227     end
228 end
229 title(['A zone ' num2str(A) '%, B zone ' num2str(B) '%, C zone ' num2str(C) '%, D zone '
num2str(D) '%E zone ' num2str(E) '%])
230 drawnow
231 end
```



# Bibliografía

- [1] J. Bondía, J. Vehí, C. Palerm, P. Herrero (2010) El páncreas artificial: Control automático de infusión de insulina en Diabetes Mellitus Tipo I, *Revista Iberoamericana de Automática e Informática Industrial* 7 (2) 5-20.
- [2] M. Wilinska, R. Hovorka (2008) Simulation models for in silico testing of closed-loop glucose controllers in type I diabetes, *Drug Discovery Today: Disease Models* 5 (4) 289-298.
- [3] P. Colmegna, R. Sánchez Peña (2014) Analysis of three T1DM simulation models for evaluating robust closed-loop controllers, *Computer Methods and Programs in Biomedicine* 113 371-382.
- [4] M. Wilinska, L. Chassin, C. Acerini, J. Allen, D. Dunger, R. Hovorka (2010) Simulation environment to evaluate closed-loop insulin delivery systems in type 1 diabetes, *Journal of Diabetes Science and Technology* 4 (1) 132-144
- [5] C. Dalla Man, M. Camilleri, C. Cobelli (2006) A system model of oral glucose absorption: Validation on gold standard data, *IEEE Transaction on Biomedical Engineering* 53 (12) 2472-2478.
- [6] C. Dalla Man, R. Rizza, C. Cobelli (2007) Meal simulation model of the glucose-insulin system, *IEEE Transaction on Biomedical Engineering* 54 1740-1749.
- [7] C. Dalla Man, D. Raimondo, R. Rizza, C. Cobelli (2007) GIM, Simulation Software of Meal Glucose-Insulin Model, *Journal of Diabetes Science and Technology* 1 (3) 323-330.
- [8] K. Andersen, M. Højbjerg (2003) A bayesian approach to Bergman's minimal model, C.M. Bishop, B.J. Frey (Eds.), *Proceedings of the Ninth International Workshop on Artificial Intelligence*, <http://research.microsoft.com/conferences/ais-tats2003/proceedings/183.pdf>
- [9] F. León - Vargas (2013) Design and Implementation of a Closed-Loop Blood Glucose Control System in Patients with Type 1 Diabetes, Phd thesis, Universitat de Girona.
- [10] C. Palerm (2011) Physiologic insulin delivery with insulin feedback: A control systems perspective. *Computer Methods and Programs in Biomedicine*, 102(2):130-137
- [11] J. T. Sorensen (1985) A physiological model of glucose metabolism in man and its use to design and assess improved insulin therapies for diabetes, Phd thesis, Massachusetts Institute of Technology.
- [12] <http://www.mathworks.com/matlabcentral/fileexchange/26312-settings-dialog>
- [13] <http://www.wikipedia.org/>



[14] <http://www.webfisio.es/fisiologia/endocrino/textos/insulina.htm>

[15] <http://www.mathworks.com/>

[16] <http://www.ni.com/>