

PANCREAS ARTIFICIALE: IDENTIFICAZIONE DI MODELLI
A SCATOLA NERA DEL METABOLISMO
GLUCOSIO-INSULINA

TESI DI LAUREA SPECIALISTICA
DI
MARCO FORGIONE

13 OTTOBRE 2009

UNIVERSITÀ DI PAVIA
FACOLTÀ DI INGEGNERIA

Sommario

Il diabete mellito è dovuto, in generale, ad una insufficiente produzione di insulina, un ormone di importanza cruciale per il metabolismo dell'organismo. Tale carenza causa parecchi scompensi metabolici, primo tra tutti una regolazione non ottimale del livello di glucosio (glicemia) nel sangue. Gli effetti di tali scompensi si ripercuotono tanto nel breve quanto nel lungo periodo peggiorando in maniera significativa la qualità della vita del malato. Nel caso di diabete di tipo 1, in particolare, la produzione endogena di insulina è nulla e in assenza di un'adeguata somministrazione esogena la malattia porta a morte certa.

Il numero di casi di diabete mellito nel mondo è stato stimato a 150 milioni nel 2000 ed è in costante aumento [6]. Di questi, il 90% circa è di tipo 2, un 5% è di tipo 1 mentre il restante 5% è diviso in altre tipologie secondarie. Oltre ai costi sociali in termini di sofferenza e perdite di vite umane, il diabete causa ingenti costi economici al servizio sanitario nazionale. Si stima che il costo totale pro capite per il paziente diabetico sia circa tre volte superiore alla media. La maggior parte di tali costi sono legati alle cure ospedaliere, che si rendono necessarie a causa delle complicanze legate alla malattia. Altri costi economici indiretti sono dovuti alle invalidità permanenti che può causare la malattia nel lungo periodo e alle ore lavorative perse per i trattamenti.

Non deve stupire dunque l'interesse crescente da parte della comunità scientifica e tecnologica e dalla politica riguardo la ricerca di nuove tecnologie nella terapia del diabete. L'insulina, un tempo estratta dagli animali ed ora sintetizzata con tecniche di ingegneria genetica, è disponibile come farmaco da lungo tempo. Una sua corretta somministrazione potrebbe in teoria riportare la qualità di vita dei pazienti vicino ai livelli degli individui sani se si riuscisse a replicare la naturale produzione endogena, che nell'individuo sano avviene nel pancreas.

Si tratta tuttavia di un problema di regolazione molto complesso che coinvolge parecchi fattori che pazienti e medici dovrebbero tenere costantemente in considerazione. Apporto calorico, sforzo, condizione fisica e molti altri fattori influenzano continuamente e in modo significativo la necessità insulinica dell'organismo ed è praticamente impossibile seguire il profilo ideale con le tecnologie tradizionali. Anche se riesce ad evitare i rischi più gravi ed immediati, una terapia non ottimizzata non evita del tutto quegli scompensi che producono danni all'organismo nel lungo termine. Essa richiede inoltre un'attenzione costante da parte del paziente, che di per sé è un fastidio considerevole.

Molti sforzi si stanno compiendo in questi anni per la realizzazione di un sistema di regolazione in anello chiuso per il controllo della glicemia: a tale dispositivo è stato dato il nome di *pancreas artificiale*. Si pensa ad un sistema che, utilizzando strumenti poco invasivi sia per la misurazione della glicemia che per l'infusione dell'insulina, possa occuparsi in modo praticamente autonomo della somministrazione ottimale del farmaco. La sua realizzazione è tuttavia una grossa sfida tecnologica che solo con una

stretta collaborazione tra medici, ingegneri ed industrie farmaceutiche potrà essere vinta.

Questa tesi ha come obiettivo la valutazione delle tecniche di identificazione a scatola nera per il metabolismo glucosio-insulina. Lo scopo dell'identificazione è migliorare le prestazioni dell'algoritmo di controllo fornendo un modello personalizzato del metabolismo del paziente.

La tesi è così organizzata

- Capitolo 1: introduzione al problema ed al progetto Pancreas Artificiale.
- Capitolo 2: descrizione del modello utilizzato per le sperimentazioni *in silico*.
- Capitolo 3: descrizione dell'algoritmo di controllo MPC attualmente implementato nel software.
- Capitolo 4: descrizione della architettura modulare secondo la quale è progettato il software.
- Capitolo 5: viene brevemente introdotta la teoria dell'identificazione ARX e vengono presentati dei dati sperimentali su pazienti virtuali, generati dal simulatore, e reali, provenienti da *trials* clinici.
- Capitolo 6: descrizione del nuovo protocollo clinico per i *trials*. Viene inoltre simulato *in silico* il nuovo protocollo di identificazione e ne sono mostrati i risultati sperimentali.
- Capitolo 7: conclusioni e possibili sviluppi futuri.

Indice

1	Introduzione	4
1.1	Il Sistema Endocrino	4
1.2	Il Pancreas e il Metabolismo dei Carboidrati	4
1.2.1	Insufficienza insulinica e diabete mellito	6
1.3	Diabete Mellito di Tipo 1	7
1.3.1	Terapia tradizionale	7
1.3.2	Pancreas Artificiale	8
1.4	Artificial Pancreas Project	9
2	Modello <i>in silico</i> del Metabolismo Glucosio-Insulina	11
2.1	Modello Dalla Man <i>et al.</i>	11
2.1.1	Sottosistema assorbimento del glucosio	13
2.1.2	Sottosistema glucosio	15
2.1.3	Sottosistema glucosio sottocutaneo	17
2.1.4	Sottosistema Insulina	18
2.1.5	Sottosistema Insulina Sottocutanea	19
2.1.6	Riassunto sul modello	20
2.1.7	Pazienti virtuali	20
2.2	Modello dell'hardware	23
2.2.1	Pompa Sottocutanea (CSII)	23
2.2.2	Sensore Sottocutaneo (CGM)	23
2.3	Piattaforma di test <i>in silico</i>	24
2.4	Metriche di Valutazione	25
2.4.1	Blood Glucose Index (BGI)	26
2.4.2	Control Variability Grid Analysis (CVGA)	26
2.5	Protocollo per il test di algoritmi di controllo	29
2.5.1	Visita di screening - Admit 1	30
2.5.2	Test in closed loop	31

3	Controllo Predittivo per il Pancreas Artificiale	33
3.1	Realizzazione del Controllo MPC	34
3.1.1	Modelli Lineari	34
3.1.2	Formulazione matriciale Input/Output	38
3.1.3	Generazione del profilo nominale	39
3.1.4	La funzione di costo	40
3.1.5	Filtro sui dati CGM	43
3.2	Procedura di Tuning	44
3.2.1	Cifra di merito	44
3.2.2	Calibrazione	45
3.2.3	Funzione di Tuning	48
4	Proposta di un'architettura modulare	54
4.1	Off Line Layer	55
4.2	Real Time Layer	55
4.3	Continuous Time Layer	56
4.4	Physical Layer (APS)	56
4.5	L'implementazione attuale	57
4.6	Vantaggi dell'Architettura Modulare	57
5	Identificazione di modelli a scatola nera	59
5.1	Modelli ARX	59
5.2	Predizione	61
5.2.1	Predizione ad un passo	61
5.2.2	Predittore a k passi in avanti	62
5.2.3	Simulazione	62
5.2.4	Metriche di valutazione	63
5.3	Stima dei Parametri	64
5.3.1	Prediction Error Method (PEM)	64
5.3.2	Eccitazione dell'input	66
5.3.3	Varianti dell'algoritmo	66
5.4	Scelta tra più classi ARX	69
5.4.1	Crossvalidazione	70
5.4.2	Criterio AIC	70
5.4.3	Vincoli di selezione	70
5.5	Risultati Sperimentali	71
5.5.1	Aspetti Tecnici	71
5.5.2	ARX su pazienti virtuali	72
5.5.3	ARX su pazienti reali	77
5.5.4	Conclusioni	85

6	Multicenter Trials: studio di fattibilità	89
6.1	Il protocollo Multicenter	89
6.1.1	Popolazione di studio e criteri di eleggibilità	89
6.1.2	Le fasi del protocollo	90
6.2	Identificazione di modelli ARX	93
6.3	Controllo MPC con modelli ARX	96
7	Conclusioni	100

Introduzione

1.1 Il Sistema Endocrino

Il sistema endocrino regola varie attività che hanno luogo all'interno del corpo umano tramite il rilascio di particolari molecole, gli ormoni, prodotti da un insieme di ghiandole. Gli ormoni raggiungono attraverso i vasi sanguigni gli organi periferici, dove compiono la loro funzione regolatoria influenzando in qualche direzione i processi metabolici, ovvero tutte le reazioni chimiche ed energetiche che avvengono nell'organismo che sono indispensabili alla vita. L'azione ormonale nell'organismo sano reagisce in modo piuttosto efficace a variazioni dell'ambiente esterno e dell'organismo stesso. Ciò che si osserva è che l'azione combinata di più ormoni riesce a mantenere i più importanti parametri fisiologici in un *range* sicuro nella maggior parte delle situazioni.

Dal un punto di vista ingegneristico, ciò che è stato descritto ricorda un sistema di controllo in anello chiuso: le reazioni metaboliche sono i processi sotto controllo, le ghiandole sono i regolatori in anello chiuso di cui gli ormoni sono i comandi in ingresso, mentre i parametri fisiologici sono le uscite. Una malattia metabolica può insorgere quando il sistema endocrino non funziona nel modo corretto e fallisce nel regolare qualche processo metabolico.

Il diabete mellito è una grave malattia dovuta ad una insufficiente produzione di un ormone, l'insulina, da parte del pancreas. Alcuni processi metabolici, tra cui il metabolismo dei carboidrati, ne risultano gravemente compromessi.

1.2 Il Pancreas e il Metabolismo dei Carboidrati

I carboidrati, di natura semplice o complessa, vengono naturalmente introdotti nell'organismo con il cibo. Una delle funzioni del tratto gastro-intestinale è di trasformare tutti i carboidrati introdotti in glucosio, che è l'unica forma in cui possono essere utilizzati dall'organismo come fonte di energia. La fase digestiva dura in genere circa

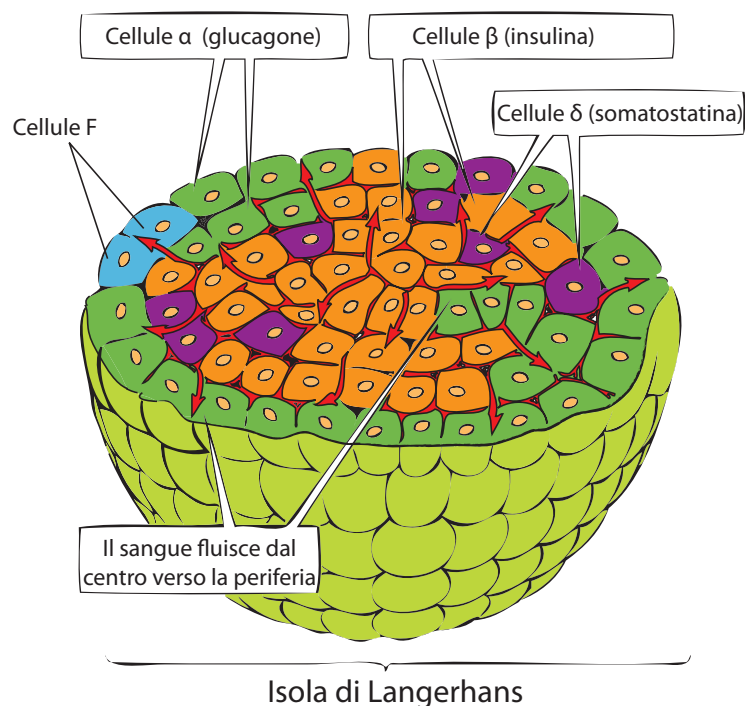


Figura 1.1: Struttura dell'isola di Langerhans

tre ore, con una certa variabilità dovuta al tipo di sostanze introdotte e alle quantità; in seguito il glucosio è assorbito a livello dell'intestino e raggiunge il flusso sanguigno. La concentrazione di glucosio nel sangue G è un parametro fisiologico di grande importanza e l'azione regolatrice del sistema endocrino, nell'organismo sano, riesce a tenere questo valore in un range piuttosto nella maggior parte delle situazioni.

Gli ormoni più importanti coinvolti nel metabolismo dei carboidrati sono secreti nel pancreas dalle isole di Langerhans. Queste strutture cellulari, dalla forma ovoidale, hanno dimensioni di circa $76 \times 175 \mu m$ e sono distribuite su tutto il pancreas, benchè ne costituiscano solamente il 2% del peso totale. Nell'organismo umano esse sono circa 1-2 milioni.

Ogni isola contiene quattro tipi di cellule: α (20%), β (60-75%), δ ed F. Le cellule α alfa producono glucagone, le β insulina, le δ somatostatina e le F polipeptide pancreatico. Insulina e glucagone svolgono le attività di regolazione più importanti e hanno un effetto complessivo opposto: l'insulina è anabolica e favorisce l'immagazzinamento del glucosio in riserve mentre il glucagone è catabolico ed induce la mobilitazione del glucosio dalle riserve nel flusso. L'azione insulinica ha due conseguenze principali: nel fegato, che ne assorbe circa il 60%, induce la produzione di glicogeno dal glucosio ed inibisce la secrezione di glucosio. Nei muscoli, nei grassi ed alcuni altri tessuti l'insulina favorisce l'ingresso di glucosio nelle cellule facendo au-

mentare il numero di trasportatori di glucosio nella membrana cellulare, permettendo così il suo utilizzo come fonte energetica. L'effetto complessivo dell'ormone è quindi una diminuzione del valore di G . Il glucagone, al contrario, favorisce la produzione di glucosio a partire dal glicogeno ed il suo conseguente rilascio nella circolazione provoca un aumento di G . La somatostatina ha un ruolo di regolazione nella secrezione stessa: quando la sua concentrazione è elevata inibisce la produzione degli altri ormoni nel pancreas [6]. L'effetto del polipeptide pancreatico nel metabolismo dei carboidrati è invece ancora parzialmente ignoto, e comunque presumibilmente di importanza inferiore.

Il meccanismo di *feedback* naturale che regola la produzione degli ormoni pancreatici nell'organismo sano è piuttosto efficiente: la glicemia G è mantenuta tra 80 e 120 (mg/dL) a digiuno e raramente supera 140 (mg/dL) dopo i pasti.

A digiuno viene prodotta una quantità approssimativamente costante di insulina, detta insulina basale, tale da permettere l'utilizzo di glucosio all'interno delle cellule. La produzione viene interrotta quando la concentrazione di glucosio scende al di sotto degli 80 (mg/dL). A questo livello glicemico approssimativamente inizia la secrezione di glucagone e di altri ormoni controregolatori. In corrispondenza dei pasti, al contrario, la produzione di insulina è aumentata.

1.2.1 Insufficienza insulinica e diabete mellito

Un insieme variegato di complicazioni insorge se l'azione insulinica è insufficiente e la malattia che ne consegue è chiamata diabete mellito. In presenza di diabete mellito il livello di glucosio nel sangue non è controllato in modo ottimale: raggiunge livelli molto ed impiega più tempo per ritornare al valore basale di equilibrio, che comunque è sensibilmente più elevato del normale: questa situazione è chiamata iperglicemia. L'effetto è dovuto in parte al ridotto utilizzo di glucosio nei tessuti, in parte al funzionamento anomalo del fegato, che con bassi livelli di insulina tende a mobilitare una quantità maggiore di glucosio dalle riserve sotto di glicogeno.

L'iperglicemia ha effetti dannosi per l'organismo nel breve e nel lungo periodo. In presenza di iperglicemia si ha glicosuria, ovvero un eccesso di glucosio nell'urina, quando la capacità di riassorbimento del glucosio dei reni viene superata. Per le proprietà osmotiche del glucosio viene persa una grande quantità di liquidi (poliuria). La conseguente disidratazione attiva i meccanismi di regolazione dell'acqua, da cui deriva una forte necessità di bere (polidipsia). Livelli glicemici particolarmente elevati (e.g. 400 mg/dL) costituiscono inoltre un grave rischio immediato che può portare al coma per ipermolarità del sangue. Livelli glicemici medi sostenuti (e.g. 200 mg/dL) non comportano in genere rischi immediati, ma nel lungo periodo comportano gravi complicazioni quali retinopatie, nefropatie e neuropatie.

A fronte dell'elevata glicemia plasmatica, la mancanza di attività insulina non permette l'ingresso di opportune quantità di glucosio all'interno delle cellule. Il catabolismo del glucosio, che in genere costituisce la fonte principale di energia per i processi cellulari, è sostituito in parte dal consumo di grassi e proteine. Lo spreco di

proteine in particolare comporta un indebolimento dell'organismo ed una maggiore sensibilità alle infezioni. Il catabolismo di grassi e proteine comporta inoltre la formazione di corpi chetonici che se presenti in grandi quantità nel sangue ne aumentano il pH. Questa condizione è nota come chetoacidosi diabetica e costituisce un grave fattore di rischio e può portare al coma se non viene curata adeguatamente.

I pazienti diabetici dipendono spesso almeno in parte dalla somministrazione di insulina esogona, ma un dosaggio preciso del farmaco è difficile. Capita occasionalmente nella terapia un'eccessiva somministrazione di insulina che porta all'ipoglicemia, ovvero ad una concentrazione di G troppo bassa. La ridotta disponibilità di glucosio causa problemi al sistema nervoso: al di sotto dei 60 (mg/dL) si verificano i primi sintomi di disfunzioni cognitive mentre attorno ai 40 (mg/dL) vi è il rischio concreto di coma.

1.3 Diabete Mellito di Tipo 1

Il Diabete Mellito di Tipo 1 (T1DM) è dovuto ad una reazione autoimmune che porta alla distruzione delle cellule β nel pancreas. La malattia si sviluppa di solito al di sotto dei 40 anni: per questa ragione è a volte chiamata diabete giovanile. Le cause del T1DM sono complesse ed ancora parzialmente ignote. Le influenze genetiche sono significative ma non cruciali: solo il 36% dei gemelli omozigoti di un diabetico sviluppa la malattia [12]. Si sospetta che in alcuni casi i fattori ambientali siano più significativi, ma non ci sono prove certe a riguardo.

1.3.1 Terapia tradizionale

Una cura per il diabete mellito di tipo 1 non esiste ad oggi. Quando diagnosticato in uno stadio primitivo, prima della distruzione di tutte le β -cellule, si riesce ad ottenere qualche risultato con terapie immunosoppressive [6]. Sono stati fatti tentativi di trapianto delle β -cellule, ma i risultati sono stati piuttosto scarsi fino ad ora.

La normale terapia prevede somministrazioni sottocutanee di insulina, attraverso due possibili regimi:

Multi-iniettivo: vengono utilizzate più iniezioni al giorno, in corrispondenza dei pasti e quando sono necessarie delle correzioni. Si utilizzano preparati di insulina ad azione rapida per contrastare la salita di G in corrispondenza dei pasti ed altri ad azione lenta per rilasciare una quantità di farmaco approssimativamente costante durante il giorno e simulare la secrezione basale.

Continuo: viene utilizzato un microinfusore in grado di somministrare insulina ad azione rapida secondo un profilo desiderato.

I dispositivi attualmente utilizzati per la somministrazione del farmaco sono siringhe, penne insuliniche e pompe sottocutanee. Siringhe e penne insuliniche vengono utilizzate in regime multi-iniettivo in quanto permettono una somministrazione solo discontinua del farmaco. Le penne insuliniche incorporano la fiala, l'ago e un sistema di dosaggio in un unico dispositivo, rendendo più agevole la vita del paziente.

Il regime continuo è invece possibile esclusivamente con le pompe sottocutanee, la tecnologia di infusione più avanzata attualmente disponibile a livello commerciale. Una pompa sottocutanea è un dispositivo che incorpora un catetere sottocutaneo per le infusioni, una riserva di insulina e un'elettronica di comando. L'utilizzatore può interagire con la pompa ed adattare il profilo alle sue particolari esigenze, in genere settando il livello basale e i boli in corrispondenza dei pasti. I modelli più recenti hanno funzioni più avanzate che assistono il paziente nella scelta di una terapia ottimale (e.g. calcolo dei boli e dell'insulina in circolazione IOB [18])

Tutti i dispositivi considerati somministrano l'insulina per via sottocutanea (sc). Altre vie possibili e clinicamente testate sono quella intramuscolare (im), intraperitoneale (ip) e intravenosa (iv).

1.3.2 Pancreas Artificiale

Il *pancreas artificiale* è un sistema in anello chiuso per la regolazione della glicemia. Come ogni altro sistema di controllo in anello chiuso, necessita di tre componenti di base:

- un attuatore
- un sensore
- software (algoritmi di controllo e supervisione)

Il primo dispositivo commerciale in grado di funzionare da pancreas artificiale fu il BiostatorTM, basato su sensori ed attuari intravenosi [17]. Questo approccio, detto in breve iv-iv, è in teoria il più efficace possibile in quanto le misurazioni plasmatiche sono le più accurate e le somministrazioni di insulina intravenose non soffrono dei ritardi e le incertezze che caratterizzano il percorso sottocutaneo. Il suo utilizzo è tuttavia limitato a circostanze particolari (e.g. interventi chirurgici) perché gli strumenti tecnologici necessari sono molto ingombranti. Altre soluzioni furono in seguito esplorate, ad esempio con misure intravenose e somministrazione insulinica intraperitoneale [16].

I recenti sviluppi nei settori di sensori sottocutanei [11], così come di nuovi analoghi rapidi dell'insulina [7] hanno aperto la strada per la realizzazione di un pancreas artificiale basato su tecnologia esclusivamente sottocutanea (sc-sc), caratterizzato dunque da un livello di invasività minimo. Un tale dispositivo, adatto alla vita di tutti i giorni del paziente, potrebbe cambiare in modo radicale la situazione attuale in fatto di terapia al diabete ed avrebbe inoltre un impatto commerciale considerevole.

Vi sono tuttavia parecchie difficoltà tecniche, sia legate all'hardware che al software, che ancora devono essere risolte.

1.4 Artificial Pancreas Project

L'Artificial Pancreas Project è stato lanciato nel 2006 dalla Juvenile Diabetes Research Foundation (JDRF), una delle principali fondazioni *no profit* per lo studio del diabete di tipo 1. L'obiettivo del progetto è di rendere disponibile entro il 2011 la tecnologia del pancreas artificiale, ed in seguito di promuoverne la diffusione commerciale. La JDRF ha dunque creato un consorzio di centri di ricerca ed università (Artificial Pancreas Consortium) che includono al momento l'Università di Cambridge, l'Università del Colorado, il MIT, Yale, Sansum, Stanford e l'Università della Virginia.

L'Università di Pavia collabora attualmente con l'Università della Virginia nello sviluppo dell'algoritmo di controllo. Dei *trials* clinici si sono svolti negli ospedali universitari di Padova (IT), di Montpellier (FR) e di Charlottesville (Virginia, USA). Una più ampia collaborazione è prevista in futuro per i *trials* secondo il protocollo Multicenter, esposto nel Capitolo 6. Uno schema di massima del pancreas artificiale utilizzato nelle prove *in vivo* è mostrato nella Figure 1.4. Le misure della concentrazione sottocutanea di glucosio vengono effettuate tramite il sensore Freestyle NavigatorTM prodotto dalla Abbott Diabetes Care^R.

I dati vengono passati in tempo reale ad un normale calcolatore tramite un dispositivo della stessa casa chiamato *cradle*. Sul calcolatore è implementato un algoritmo di controllo che decide la somministrazione di insulina ottimale: attualmente si tratta di controllo predittivo lineare senza vincoli (LMPC). La pompa Omnipod Insulin Management SystemTM riceve i comandi dal calcolatore (che comunque devono essere approvati dal personale medico) ed effettua le infusioni di insulina nel paziente.

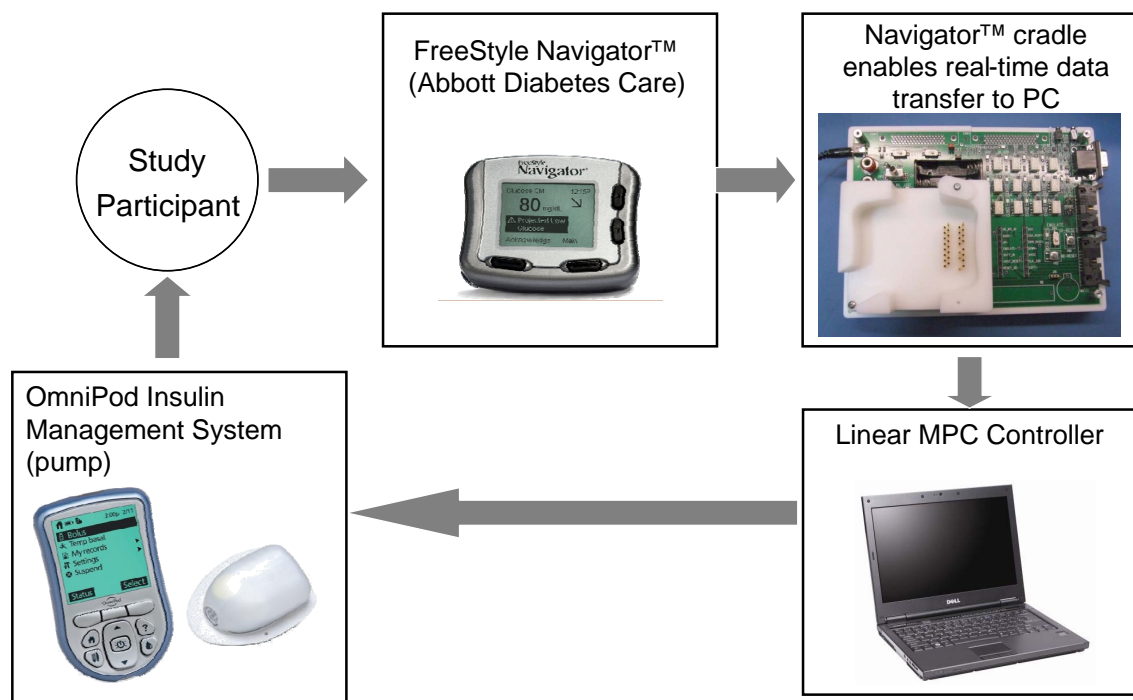


Figura 1.2: Schema del pancreas artificiale

Modello *in silico* del Metabolismo Glucosio-Insulina

Lo sviluppo del pancreas artificiale è stato enormemente agevolato dall'introduzione di un simulatore accurato del metabolismo umano detto Glucose-Insulin Model (GIM). Questo software implementa *in silico*, ovvero su un calcolatore, modelli matematici per il metabolismo del paziente diabetico e per l'hardware utilizzato nella terapia in anello chiuso, ovvero la pompa e il sensore CGM. GIM viene utilizzato dai ricercatori per sviluppare e validare gli algoritmi di controllo in modo economico e senza alcun rischio. Inoltre, esso è stato approvato dalla Food and Drug Administration (FDA), ente americano che controlla cibi e farmaci introdotti nel mercato nazionale, come sostituto alle prove sugli animali: dal Gennaio 2008 gli algoritmi di controlli validati sul modello *in silico* possono essere testati direttamente sugli esseri umani senza altri passaggi intermedi.

2.1 Modello Dalla Man *et al.*

Il modello matematico del metabolismo è la parte più importante del software GIM. Si basa su una versione leggermente modificata di quello proposto in [3] nel 2007 e facciamo riferimento ad esso come modello Dalla Man *et al.* da uno dei suoi ideatori nel gruppo di lavoro di Padova. È un modello fisiologico completo, inclusi i processi di ingestione ed assorbimento del glucosio. I suoi ingressi sono i pasti (glucosio) e le somministrazioni sottocutanee di insulina, l'uscita è la glicemia.

Non include tuttavia l'influenza di altri ormoni oltre all'insulina in modo esplicito, nè distingue tra carboidrati semplici e complessi: per questo motivo facciamo riferimento ad esso come un modello del metabolismo glucosio-insulina, piuttosto che del metabolismo dei carboidrati in generale. È composto da un certo numero di sottosistemi interconnessi che rappresentano fasi distinte del metabolismo (Figura 2.1)

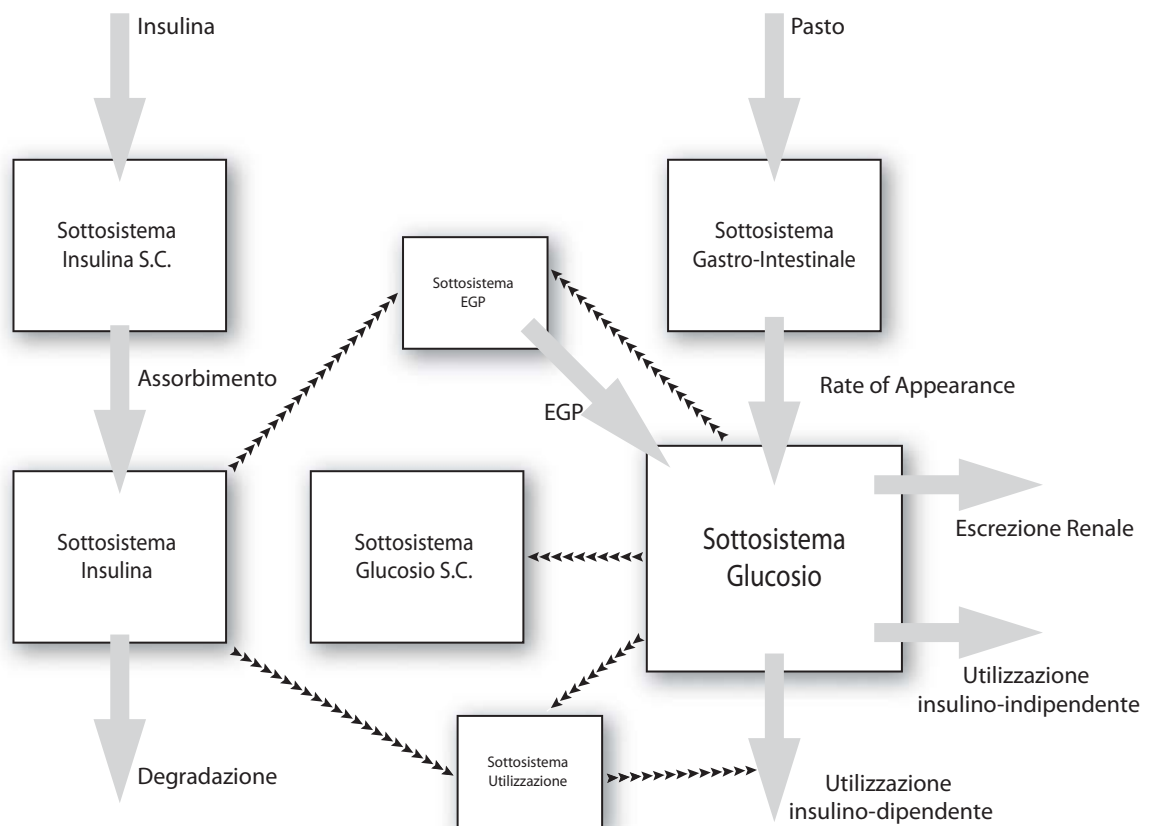


Figura 2.1: Sottosistemi del modello di Dalla Man *et.al.*

Originariamente era stato sviluppato su un dataset di 204 individui sani. Ad essi fu iniettato un pasto misto contenente un tracciante radioattivo e contemporaneamente gli furono iniettati due altri traccianti radioattivi per endovena. Questo permise di ottenere misure dei flussi di glucosio ed insulina rilevanti durante un pasto e per mezzo di questi i vari sottosistemi sono stati validati in modo praticamente indipendente. Senza l'utilizzo di questa tecnica, infatti, è possibile misurare solamente la glicemia e l'insulinemia, senza poter distinguere, per esempio, tra glucosio prodotto per via endogena dal fegato e glucosio introdotto con il pasto e dei modelli non corretti possono ugualmente produrre buone stime, se a livello complessivo gli errori su più sottosistemi si compensano.

Il modello è stato adattato solo successivamente per il paziente diabetico, sostituendo il modello di secrezione delle β -cellule con un blocco che rappresenta le iniezioni tramite la pompa e la dinamica sottocutanea dell'insulina. Un altro sottosistema è stato introdotto per rappresentare il glucosio sottocutaneo che è letto dai sensori CGM al posto della concentrazione plasmatica G . In seguito verranno descritte nel dettaglio le singole unità funzionali.

2.1.1 Sottosistema assorbimento del glucosio

Una catena di tre compartimenti è utilizzata per descrivere il transito del glucosio attraverso il tratto gastrointestinale. I primi due rappresentano la fase solida Q_{sto1} (mg) e triturrata Q_{sto2} (mg) nello stomaco, l'ultimo rappresenta la quantità di glucosio nell'intestino Q_{gut} (mg). L'ingresso al primo scompartimento è il glucosio dai pasti d (mg/min). Le equazioni di stato del sistema sono

$$\begin{aligned}\dot{Q}_{sto1}(t) &= -k_{gri} \cdot Q_{sto1}(t) + d(t) \\ \dot{Q}_{sto2}(t) &= -k_{empt}(Q_{sto}, D) \cdot Q_{sto2}(t) + k_{gri} \cdot Q_{sto1}(t) \\ \dot{Q}_{gut}(t) &= -k_{abs} \cdot Q_{gut}(t) + k_{empt}(Q_{sto}, D) \cdot Q_{sto2}(t)\end{aligned}\tag{2.1}$$

dove k_{gri} , k_{empt} e k_{abs} (1/min) sono i coefficienti frazionari di trasferimento (CFT).

Mentre k_{gri} e k_{abs} sono considerati costanti nel modello, k_{empt} è una funzione complessa della quantità di glucosio nello stomaco $Q_{sto} = Q_{sto1} + Q_{sto2}$ e della quantità totale dell'ultimo pasto D :

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

dove

$$D = \int_{t_i}^{t_f} d(t) dt$$

con t_i e t_f tempi di inizio e fine dell'ultimo pasto rispettivamente. Tale formulazione per k_{empt} è adatta a riprodurre il comportamento nonlineare di tipo bifasico dello svuotamento gastrico.

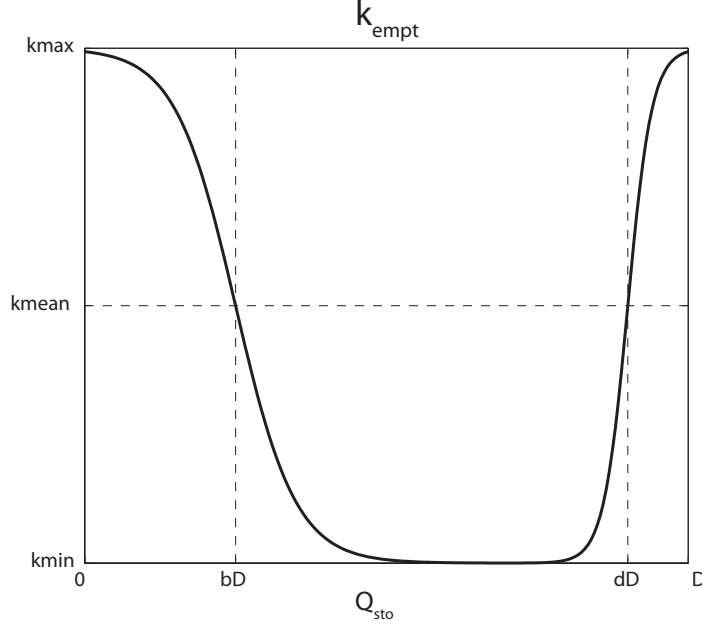


Figura 2.2: Andamento di $k_{empt}(Q_{sto}, D)$

L'andamento di tale coefficiente al variare di Q_{sto} è mostrato in Figura 2.2. All'inizio della digestione (i.e quando $Q_{sto} = D$) raggiunge il suo valore massimo, k_{max} , quindi decresce fino al valore minimo k_{min} e si avvicina di nuovo alla fine al valore massimo, quando lo stomaco è ormai quasi vuoto. I parametri b a d rappresentano le percentuali di pasto non ancora assorbite nei due istanti in cui $k_{empt} = \frac{k_{min} + k_{max}}{2}$ mentre α e β prendono il valore

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

$$\beta = \frac{5}{2 \cdot D \cdot d}$$

ed impongono la condizione $k_{empt} = k_{max}$ per $Q_{sto} = D$ e $Q_{sto} = 0$. Per garantire l'identificabilità *a priori*, è stato assunto che $k_{gri} = k_{max}$.

L'uscita del modello è la *rate of appearance* del glucosio nel plasma Rat (mg/min·Kg):

$$\text{Rat}(t) = f \cdot k_{abs} \cdot Q_{gut}(t) / BW \quad (2.2)$$

dove BW (Kg) è il peso del paziente ed f (adimensionale) è un parametro costante e rappresenta la frazione di glucosio intestinale che raggiunge effettivamente la circolazione. In Figura 2.3 è mostrato il processo di assorbimento non lineare sul paziente medio insieme con la sua approssimazione lineare, con k_{empt} fissato a $\frac{k_{max} + k_{min}}{2}$. Per il modello lineare è rappresentata la risposta impulsiva, per il non lineare la risposta ad un impulso di 80 grammi di glucosio opportunamente normalizzata dell'uscita Rat ed il parametro k_{empt} . Si nota che il processo ha una durata di circa tre ore e, nel caso non lineare, l'assorbimento ha un secondo picco dopo circa due ore e mezza.

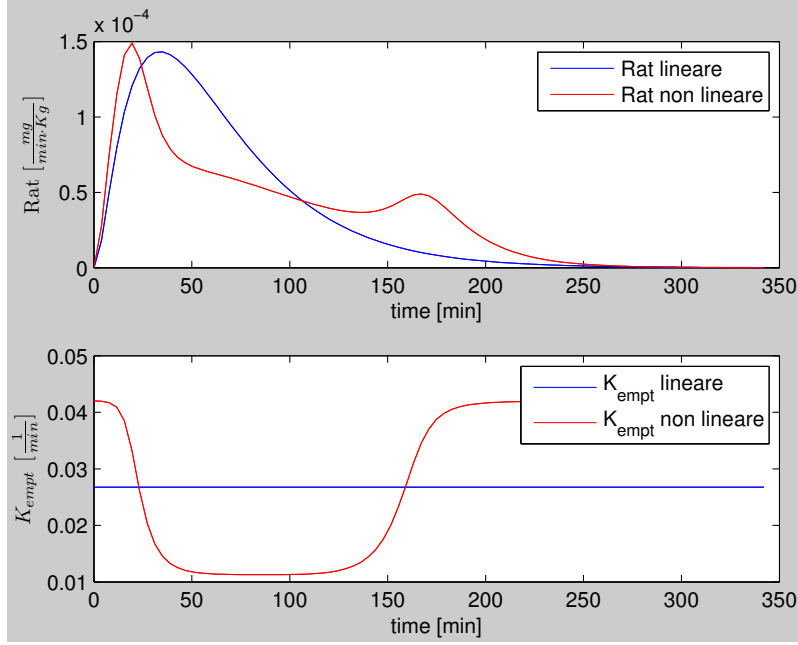


Figura 2.3: Assorbimento non lineare e lineare

2.1.2 Sottosistema glucosio

Il modello utilizzato per descrivere la dinamica del glucosio prevede due compartimenti:

$$\begin{aligned}\dot{G}_p(t) &= EGP(t) + \text{Rat}(t) - U_{ii}(t) - E(t) - k_1 \cdot G_p(t) + k_2 \cdot G_t(t) \\ \dot{G}_t(t) &= -U_{id}(t) + k_1 \cdot G_p(t) - k_2 \cdot G_t(t)\end{aligned}\quad (2.3)$$

dove G_p (mg/Kg) è la massa di glucosio nel plasma, G_t (mg/Kg) è la massa di glucosio nei tessuti, k_1 and k_2 (1/min) sono CFT costanti. I flussi di glucosio in ingresso al sottosistema sono la produzione endogena di glucosio $EGP(t)$ e la *rate of appearance* $\text{Rat}(t)$ mentre i flussi in uscita sono l'utilizzo insulino-dipendente $U_{id}(t)$ ed indipendente U_{ii} , e l'estrazione $E(t)$. L'utilizzo insulino-indipendente è una costante mentre tutti gli altri flussi sono variabili. L'unità di misura per i flussi è (mg/(kg·min)).

L'output del sottosistema è la concentrazione di glucosio nel plasma $G(t)$ (mg/dL):

$$G(t) = \frac{G_p(t)}{V_G} \quad (2.4)$$

dove V_G (dL/Kg) è il volume di distribuzione del glucosio nel plasma rapportato alla massa corporea. Il valore G è tralaltro l'uscita più importante del sistema ed è l'oggetto della regolazione.

L'equilibrio basale è lo stato stazionario del sistema caratterizzato da $d = 0$, ovvero condizione di digiuno, e glicemia costante G_b (mg/dL). Imponendo nella Equazione (2.3) le derivate temporali a zero:

$$\begin{cases} 0 = EGP_b + \text{Rat}_b - U_{ii} - E_b - k_1 G_{pb} + k_2 G_{td} \\ 0 = -U_{idb} + k_1 G_{pb} - k_2 G_{tb} \end{cases} \quad (2.5)$$

dove il pedice b indica per le varie grandezze il valore basale. Essendo in condizione di digiuno, si ha $\text{Rat}_b = 0$. Quindi

$$EGP_b = U_{ii} + U_{idb} + E_b \quad (2.6)$$

Andremo ora ad analizzare i flussi in ingresso ed uscita al sottosistema.

Rate of Appearance

La *rate of appearance* $\text{Rat}(t)$ è l'uscita del sottosistema glucosio-insulina dell'Equazione (2.2).

Endogenous Glucose Production EGP

La produzione endogena di glucosio avviene nel fegato a partire dalle riserve sotto forma di glicogeno. L'inibizione di EGP dipende in modo lineare dalla massa di glucosio plasmatico $G_p(t)$ e da un segnale di insulina ritardato $I_d(t)$. Il valore è comunque vincolato ad essere positivo.

$$EGP(t) = \max(0, k_{p1} - k_{p2} \cdot G_p(t) - k_{p3} \cdot I_d(t)) \quad (2.7)$$

dove k_{p1} (mg/Kg·min) è il valore di EGP estrapolato a $G_p = 0$, k_{p2} (1/min) e k_{p3} (mg/Kg·min)·(L/pmol) rappresentano l'azione del glucosio e dell'insulina sulla produzione endogena di glucosio EGP rispettivamente.

Il segnale di insulina ritardato $I_d(t)$ è generato dal seguente sistema dinamico

$$\begin{aligned} \dot{I}_1(t) &= k_i I(t) - k_i I_1(t) \\ \dot{I}_d(t) &= k_i I_1(t) + k_i I_d(t) \end{aligned} \quad (2.8)$$

dove I (pmol/L) è la concentrazione di insulina nel plasma e k_i (1/min) è un parametro costante. All'equilibrio basale abbiamo

$$EGP_b = k_{p1} - k_{p2} G_{pb} - k_{p3} I_b \quad (2.9)$$

dove I_b (pmol/L) è il valore basale della concentrazione di insulina nel plasma. Possiamo allora scrivere k_{p1} come

$$k_{p1} = EGP_b + k_{p2} G_{pb} + k_{p3} I_b \quad (2.10)$$

Estrazione Renale E

L'estrazione renale $E(t)$ è il flusso di glucosio eliminato dai reni quando la concentrazione G è sopra una certa soglia:

$$E(t) = \begin{cases} k_{e1} \cdot [G_p(t) - k_{e2}] & \text{se } G_p(t) > k_{e2} \\ 0 & \text{se } G_p(t) \leq k_{e2} \end{cases} \quad (2.11)$$

dove k_{e1} è il tasso di filtrazione e k_{e2} è un parametro di soglia. All'equilibrio basale, E è nulla nella quasi totalità dei casi.

Utilizzo del glucosio

L'utilizzo del glucosio ha due componenti: una costante U_{ii} , dovuta al cervello e ai globuli rossi ed una variabile U_{id} , che avviene in altri tessuti come i muscoli e dipende in modo non lineare da G_t ed X (pmol/L), un segnale che rappresenta l'insulina nel fluido interstiziale:

$$U_{id}(t) = \frac{V_m(X(t)) \cdot G_t(t)}{K_M \cdot X(t)} \quad (2.12)$$

dove $V_m(X(t))$ è una funzione lineare di $X(t)$:

$$V_m(X(t)) = V_{m0} + V_{mx} \cdot X(t) \quad (2.13)$$

ed il segnale $X(t)$ è generato dal seguente sistema lineare

$$\dot{X}(t) = -p_{2U} \cdot X(t) - p_{2U}[I(t) - I_b] \quad (2.14)$$

dove p_{2U} ($1/\text{min}$) è il *rate* dell'azione insulinica sull'utilizzazione periferica di glucosio. L'utilizzo basale insulino-dipendente è

$$U_{idb} = V_{m0} \frac{G_{tb}}{K_{m0} + G_{tb}} \quad (2.15)$$

facendo riferimanto alle Equazioni (2.5) e (2.6), possiamo scrivere

$$\begin{aligned} G_{tb} &= (U_{ii} - EGP_b + k_1 G_{pb} + E_b)/k_2 \\ V_{m0} &= (EGP_b - U_{ii} - E_b) \cdot (K_m + G_{tb})/G_{tb} \end{aligned} \quad (2.16)$$

2.1.3 Sottosistema glucosio sottocutaneo

I sensori CGM misurano la concentrazione sottocutanea di glucosio G_M che è, con una certa approssimazione, una versione ritardata del valore plasmatico G . Questa relazione è descritta da un sistema lineare del primo ordine, con costante k_{sc}

$$\dot{G}_M = -k_{sc} G_M + k_{sc} G(t) \quad (2.17)$$

Essendo $k_{sc} = 0.106$ per il paziente medio, il sistema sottocute si comporta come un filtro passa-basso del primo ordine di banda passante $k_{sc}/2\pi \approx 0.0169$ ($1/\text{min}$)

2.1.4 Sottosistema Insulina

Un sistema a due compartimenti è utilizzato per descrivere la cinetica dell'insulina. Il suo ingresso è il flusso di insulina $s(t)$ ($\text{pmol}/\text{min} \cdot \text{Kg}$) e la sua uscita è la concentrazione di insulina nel plasma I (pmol/L). Mentre nel sistema originale [3] era la secrezione descritta da un modello della β -cellula, nel nostro modello modificato $s(t)$ è l'uscita di un sistema lineare che rappresenta il percorso sottocutaneo, il cui ingresso è l'insulina dalla pompa. Le equazioni di stato del processo sono:

$$\begin{aligned}\dot{I}_p(t) &= -(m_2 + m_4) \cdot I_p(t) + m_1 \cdot I_l(t) + s(t) \\ \dot{I}_l(t) &= -(m_1 + m_3) \cdot I_l(t) + m_2 \cdot I_p(t)\end{aligned}\quad (2.18)$$

dove I_l (pmol/Kg) è la massa di insulina nel fegato, I_p (pmol/Kg) è la massa di glucosio nel plasma ed m_1 , m_2 , m_3 ed m_4 ($1/\text{min}$) sono parametri costanti. I parametri m_2 , m_3 ed m_4 sono dati da:

$$\begin{aligned}m_2 &= 0.6 \cdot \frac{CL_{ins}}{HE_b \cdot V_I \cdot BW} \\ m_3 &= m_1 \cdot \frac{HE_b}{1 - HE_b} \\ m_4 &= 0.4 \cdot \frac{CL_{ins}}{V_I \cdot BW}\end{aligned}\quad (2.19)$$

dove HE_b (adimensionale) è l'estrazione basale epatica di insulina e CL_{ins} ($1/\text{min}$) è la *clearance* dell'insulina, cioè il volume di plasma dal quale i reni sono in grado di eliminare l'insulina in un minuto. La concentrazione di insulina nel plasma I (pmol/L) è l'uscita:

$$I(t) = \frac{I_p}{V_I} \quad (2.20)$$

dove V_I (L/Kg) è il volume di distribuzione dell'insulina. L'insulina resta in circolazione per un periodo di circa 20 minuti. Si noti che il processo non è influenzato dalla concentrazione di glucosio plasmatico G . In Figura 2.4 è mostrata la risposta impulsiva a fronte di un ingresso $s(t)$ impulsivo.

All'equilibrio basale abbiamo:

$$\begin{aligned}0 &= -(m_2 + m_4)I_{pb} + m_1I_{lb} + s_b \\ 0 &= -(m_1 + m_3)I_{lb} + m_2I_{pb}\end{aligned}\quad (2.21)$$

e quindi:

$$\begin{aligned}I_{lb} &= \frac{m_2}{m_1 + m_3} I_{pb} \\ s_b &= (m_2 + m_4)I_{pb} - m_1I_{lb}\end{aligned}\quad (2.22)$$

in cui $I_{pb} = I_b \cdot V_I$.

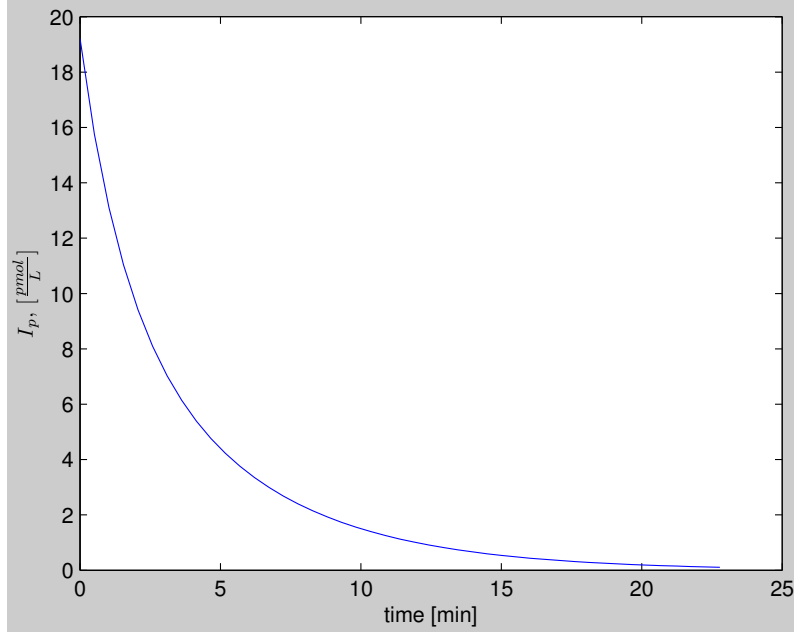


Figura 2.4: Risposta impulsiva del sottosistema insulina

2.1.5 Sottosistema Insulina Sottocutanea

Si riscontra un ritardo significativo tra le infusioni sottocutanee e la conseguente salita della concentrazione di insulina nel plasma. Questo tempo è necessario perché il preparato raggiunga i capillari, da cui è assorbito. Tale ritardo è uno dei problemi più difficili che la regolazione deve affrontare. La dinamica è descritta da un modello a due compartimenti

$$\begin{aligned}\dot{S}_1(t) &= -(k_{a1} + k_d)S_1(t) + u(t) \\ \dot{S}_2(t) &= k_d S_1(t) - k_{a2} S_2(t)\end{aligned}\tag{2.23}$$

e la sua uscita $s(t)$ è data da

$$s(t) = k_{a1} S_1(t) + k_{a2} S_2(t)\tag{2.24}$$

dove $u(t)$ ($\text{pmol}/\text{min}\cdot\text{Kg}$) è l'insulina dalla pompa, k_d ($1/\text{min}$) è detta costante di degradazione, k_{a1} e k_{a2} ($1/\text{min}$) sono le costanti di assorbimento. All'equilibrio basale abbiamo

$$\begin{cases} 0 &= -(k_{a1} + k_d)S_{1b} + u_b \\ 0 &= k_d S_{1b} - k_{a2} S_{2b} \\ s_b &= k_{a1} S_{1b} + k_{a2} S_{2b} \end{cases}$$

risolvendo il sistema otteniamo

$$\begin{cases} S_{1b} &= \frac{s_b}{k_d + k_{a1}} \\ S_{2b} &= \frac{k_d}{k_{a1}} S_{1b} \\ s_b &= u_b \end{cases}$$

Stima del ritardo di assorbimento

Essendo il sistema dell'insulina sottocutanea di Equazione (2.23) lineare, possiamo considerare la trasformata di Laplace dell'uscita s . Usando p come variabile di trasformata per evitare ambiguità con s , otteniamo da semplici calcoli:

$$\mathcal{L}[s(t)] = k_{a1} \frac{\mathcal{L}[u(t)]}{p + (k_{a1} + k_d)} + k_{a2} k_d \frac{\mathcal{L}[u(t)]}{(p + k_{a2})(p + (k_{a1} + k_d))} \quad (2.25)$$

Se l'ingresso $u(t)$ è un impulso, ovvero $u(t) = \delta$, l'uscita $s(t)$ è una combinazione lineare di esponenziali (Figura 2.5), il più lento dei quali ha costante di tempo, per il paziente medio:

$$T = \frac{1}{k_{a2}} \approx 50 \text{ min} \quad (2.26)$$

Dall'analisi della risposta impulsiva, in particolare, risulta evidente la somma di un sistema del primo ordine ed un sistema del secondo ordine (Equazione 2.25) e la durata del processo di circa 3 ore.

2.1.6 Riassunto sul modello

Il modello ha in tutto 13 stati, riportati in Tabella 2.1 e 36 parametri totali, riportati in Tabella 2.2. Cinque parametri, mostrati in grassetto, sono considerati costanti sulla intera popolazione mentre altri sei, riportati con un asterisco all'apice, sono parametri derivati, ovvero dipendono da altri parametri e costanti. Il singolo paziente è dunque caratterizzato dai valori di 25 parametri.

2.1.7 Pazienti virtuali

I parametri stimati sul dataset di 204 pazienti sono stati utilizzati per stimarne la distribuzione casuale, che è stata postulata essere di tipo lognormale. I parametri che caratterizzano univocamente tale distribuzione sono media e covarianza delle variabili: questa informazione è stata presa da media e covarianza campionaria all'interno del dataset.

Per riflettere le caratteristiche di un paziente adulto affetto da T1DM, la media campionaria dei parametri glucosio basale G_b e produzione di glucosio endogena basale EGP_b sono state aumentate rispettivamente, di 50 (mg/dL) e 0.7 (mg/kg·min), mentre la matrice di covarianza è rimasta inalterata. La distribuzione così ottenuta è stata utilizzata per descrivere la classe dei pazienti adulti. Per descrivere la classe

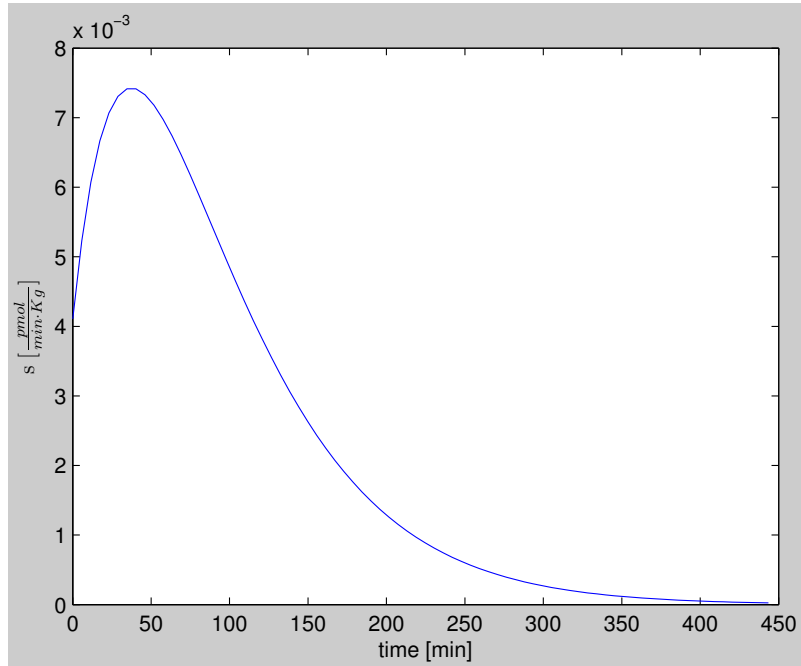


Figura 2.5: Risposta impulsiva del sottocute per il paziente medio

	Stato	Descrizione	Basale	Unità
x_1	Q_{sto1}	Fase solida nello stomaco	0	mg
x_2	Q_{sto2}	Fase triturata nello stomaco	0	mg
x_3	Q_{gut}	Glucosio nell'intestino	0	mg
x_4	G_t	Glucosio nel plasma	G_{pb}	mg/Kg
x_5	G_p	Glucosio nei tessuti	G_{tb}	mg/Kg
x_6	I_p	Insulina nel plasma	I_{pb}	pmol/Kg
x_7	X	Insulina remota	0	pmol/L
x_8	I_1	Segnale di insulina ritardata	I_b	pmol/L
x_9	I_d	Azione dell'insulina sull'EGP	I_b	pmol/L
x_{10}	I_l	Insulina nel fegato	I_{lb}	pmol/Kg
x_{11}	S_1	Primo compartimento insulina sottocutanea	S_{1b}	pmol/Kg
x_{12}	S_2	Secondo compartimento insulina sottocutanea	S_{2b}	pmol/Kg
x_{13}	G_M	Glucosio sottocutaneo	G_b	mg/Kg

Tabella 2.1: Stati del modello Dalla Man *et al.*

Sottosistema	Parametro	Valor Medio	Unità
Assorbimento Glucosio	k_{gri}^*	0.0422	$1/\text{min}$
	k_{abs}	0.2725	$1/\text{min}$
	k_{max}	0.0422	$1/\text{min}$
	k_{min}	0.0133	$1/\text{min}$
	b	0.772	adimensionale
	d	0.1532	adimensionale
Cinetica Glucosio	k_1	0.0772	$1/\text{min}$
	k_2	0.1169	$1/\text{min}$
	V_G	1.8563	dL/Kg
	\mathbf{f}	0.9	adimensionale
	G_b	120	mg/dL
Produzione Endogenea EGP	EGP_b	2.6613	mg/(Kg·min)
	k_{p1}^*	5.4314	mg/(Kg·min)
	k_{p2}	0.0067	$1/\text{min}$
	k_{p3}	0.0123	(L·mg)/(Kg·min·pmol)
	k_i	0.0114	$1/\text{min}$
Estrazione Renale	\mathbf{k}_{e1}	0.0005	$1/\text{min}$
	\mathbf{k}_{e2}	339	mg/Kg
Utilizzo Glucosio	\mathbf{U}_{ii}	1	mg/Kg·min
	V_{m0}^*	4.54	mg/(Kg·min)
	V_{mx}	0.0644	(L·mg)/(Kg·min·pmol)
	K_{m0}	230.34	mg/Kg
	p_{2U}	0.0427	$1/\text{min}$
Cinetica Insulina	\mathbf{HE}_b	0.6	L/Kg
	CL_{ins}	1.118	L/min
	m_1	0.199	$1/\text{min}$
	m_2^*	0.284	$1/\text{min}$
	m_3^*	0.2985	$1/\text{min}$
	m_4^*	0.1136	$1/\text{min}$
	V_I	0.0520	L/Kg
	I_b	104.94	pmol/L
Dinamica Sottocutanea	k_{a1}	0.0041	$1/\text{min}$
	k_{a2}	0.0184	$1/\text{min}$
	k_d	0.0168	$1/\text{min}$
	k_{sc}	0.106	$1/\text{min}$
Massa Corporea	BW	79.68	Kg

Tabella 2.2: Parametri del modello Dalla Man *et al.*

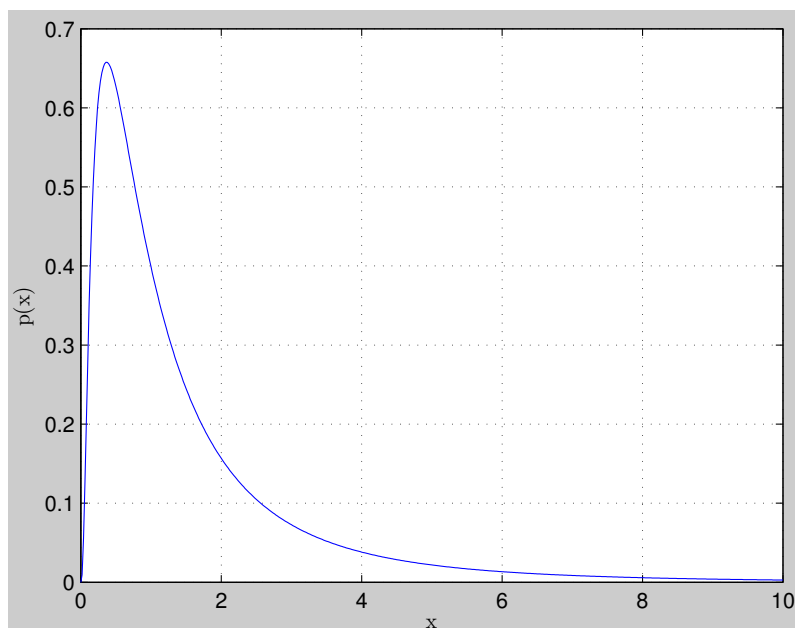


Figura 2.6: Distribuzione lognormale univariata

dei pazienti adolescenti e bambini è stata stimata la media dei parametri casuali su database più ristretti di pazienti, usando tuttavia la stessa covarianza campionaria stimata sugli adulti. Dalle distribuzioni dei parametri delle varie classi è possibile estrarre un numero arbitrario di pazienti virtuali, che non sono altro che particolari realizzazioni della variabile casuale dei parametri. In questa tesi è stato utilizzato un dataset di 100 pazienti virtuali estratti dalla popolazione virtuale degli adulti.

2.2 Modello dell'hardware

2.2.1 Pompa Sottocutanea (CSII)

La realizzazione virtuale della pompa o microinfusore di insulina è stata progettata in modo da tener conto dei reali vincoli fisici dell'attuatore, cioè portata minima e massima di insulina ammesse in uscita e la risoluzione in tale intervallo. A parte l'ovvia impossibilità di iniettare portate di insulina negativa, queste limitazioni non sembrano aver dato origine a particolari problemi.

2.2.2 Sensore Sottocutaneo (CGM)

Abbiamo già descritto la relazione intercorrente tra la concentrazione sottocutanea di glucosio G e la concentrazione sottocutanea G_M nella sezione sul sottosistema insulina sottocutanea. Il sensore CGM misura la concentrazione G_M con alcuni vincoli sul *range* di lettura (32-600 mg/dL) e, soprattutto, con un errore di misura ϵ_n non

trascurabile. In accordo con i dati empirici raccolti, il rumore è stato modellizzato con il seguente sistema di Johnson SU [8]:

$$\epsilon_n = \alpha + \lambda \sinh \left(\frac{e_n + \gamma}{\delta} \right) \quad (2.27)$$

dove e_n è l'uscita di un filtro autoregressivo di ordine 1, avente in ingresso del rumore bianco gaussiano ν a media nulla e varianza unitaria:

$$\begin{cases} e_1 = \nu_1 \\ e_n = 0.23e_{n-1} + 0.23\nu_n \end{cases} \quad (2.28)$$

2.3 Piattaforma di test *in silico*

Il modello del paziente, dell'hardware e dell'algoritmo di controllo è stato implementato *in silico*, ovvero sotto forma di software per un calcolatore. L'ambiente di sviluppo utilizzato è MATLAB-SimulinkTM, per la semplicità con cui è possibile gestire modelli complessi tramite il formalismo degli schemi a blocchi connessi.

Grazie alla popolazione di pazienti virtuali è possibile valutare come si comportano diverse soluzioni di controllo, sia in anello aperto che in anello chiuso, su differenti scenari clinici per una popolazione di pazienti con caratteristiche differenti. Il test *in-silico* deve fornire inoltre delle informazioni sulla robustezza a fronte incertezze di varia natura. Si generano e si testano a questo scopo, oltre allo scenario nominale, un certo numero di scenari perturbati.

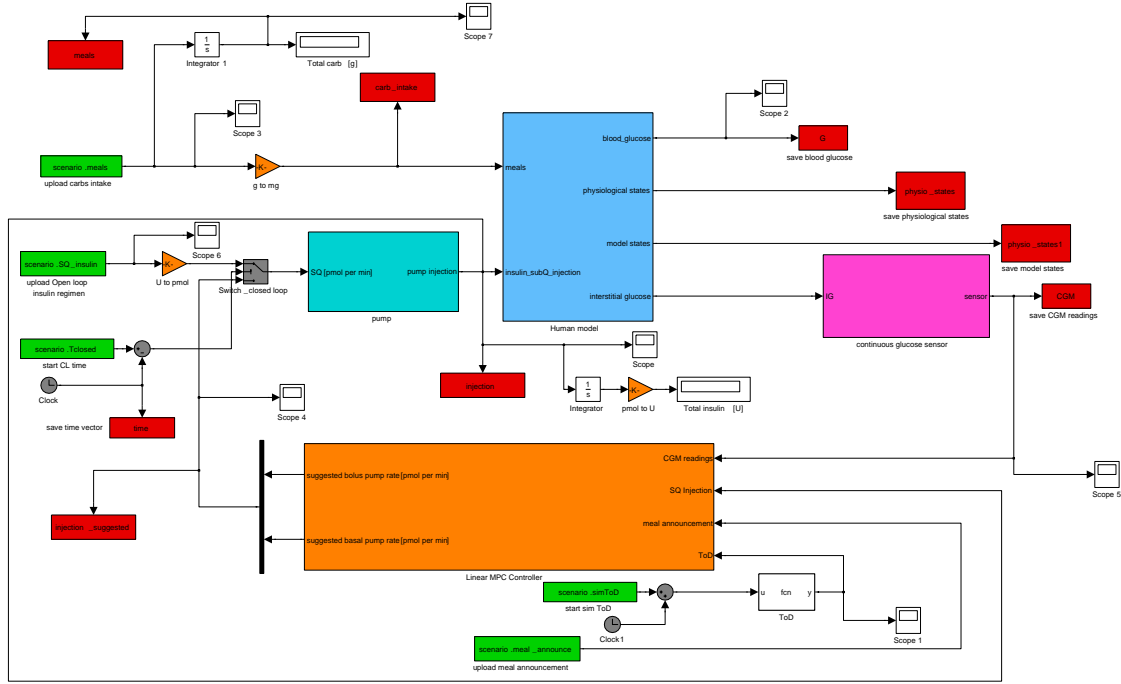


Figura 2.7: Piattaforma di sviluppo *in silico*

2.4 Metriche di Valutazione

Data la grande quantità di dati generati dal simulatore, sorge la necessità di definire metriche di valutazione oggettive per misurare le performance di una certa simulazione. Ricordando che, nell'individuo sano, la concentrazione di glucosio nel sangue G è normalmente nel range 80-120 (mg/dL), le metriche devono in qualche modo misurare la distanza rispetto a tale situazione ideale. Esistono nello specifico due tipi di misure: metriche puntuali che definiscono, per un dato valore G , il rischio associato, e metriche complessive, capaci di fornire una valutazione dell'andamento di G nel corso di una simulazione. Più formalmente, le prime sono funzioni (funzioni di rischio) dell'argomento reale G , le seconde sono funzionali il cui argomento è la funzione a valori reali $G(\cdot)$.

Le funzioni di rischio, in genere, assumono il valore minimo (solitamente 0) per un certo valore ottimale detto *target* e crescono allontanandosi da esso. Per quanto riguarda i funzionali, lo spazio delle possibilità è molto più ampio. Una semplice scelta potrebbe essere considerare il valore medio di G nel periodo della simulazione. I vantaggi di questa scelta sono la semplicità implementativa ed il significato fisiologico che questo dato implica: si ricordi infatti che molte patologie di lungo periodo legate al diabete sono connesse al valor medio troppo elevato della glicemia. Essa non tiene conto tuttavia delle oscillazione che la variabile può assumere nel corso della

simulazione: una situazione in cui ipo- e iper- glicemie si alternano compensandosi sarebbe ugualmente considerata ottimale, nonostante gli evidenti rischi connessi.

Altre scelte possibili sono la percentuale di tempo trascorsa in un *range* considerato accettabile, ad esempio tra 70 e 180 (mg/dL) o, al contrario, la percentuale di tempo trascorsa nei range pericolosi, o il numero di tali episodi. Ancora, si potrebbe andare a misurare in una qualche norma (e.g. massimo, L^2) la differenza tra l'andamento ideale e quello di simulazione.

2.4.1 Blood Glucose Index (BGI)

La Blood Glucose Index è una funzione, proposta da Kovatchev *et al.* [9], che permette di valutare il rischio associato ad un determinato valore di G . Associa ad un valore di glicemia un indice secondo la seguente funzione:

$$BGI(G) = 10(g[\ln^a G - b])^2 \quad (2.29)$$

dove a , b e g sono parametri. In particolare, essi sono stati scelti in modo che

$$BGI(70) = BGI(300) = 25 \text{ e } BGI(50) = BGI(400)$$

ottenendo

$$a = 1.44, \quad b = 10.07, \quad g = 0.75$$

Il valore di glicemia target, per il quale il rischio associato è uguale a zero, è dato da

$$G_0 = e^{\sqrt[a]{b}} = 148.31 \text{ mg/dL} \quad (2.30)$$

La funzione di rischio è convessa, non negativa, ed assume valore 0 solo in corrispondenza del target (Figura 2.8). È inoltre una funzione asimmetrica, allo scopo di penalizzare maggiormente la condizione di ipoglicemia rispetto all'iperglicemia.

2.4.2 Control Variability Grid Analysis (CVGA)

La Control Variability Grid Analysis è una metrica sia numerica che grafica e tiene conto dei valori minimi e massimi assunti dalla glicemia nel periodo dell'osservazione. Si tratta dunque di uno strumento in grado di valutare le proprietà complessive della regolazione. Il vantaggio principale di questo strumento è la possibilità di presentare in forma grafica i risultati di una simulazione non solo su un singolo paziente, ma su un'intera popolazione.

La CVGA si presenta come una griglia quadrata nella quale la simulazione sul singolo paziente è rappresentata con un punto le cui coordinate $[x \ y]'$ sono legate alla glicemia minima e massima ottenute rispettivamente. L'asse x relativo alla glicemia minima è invertito in modo che l'angolo in basso a destra della figura rappresenti la situazione ideale e muovendosi per x e y crescenti la situazione peggiori progressivamente, considerando che l'obiettivo del controllo è abbassare i massimi ed alzare i

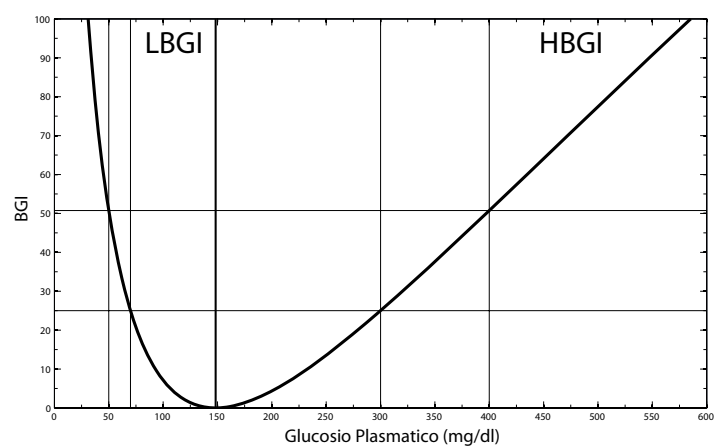


Figura 2.8: Funzione di rischio BGI

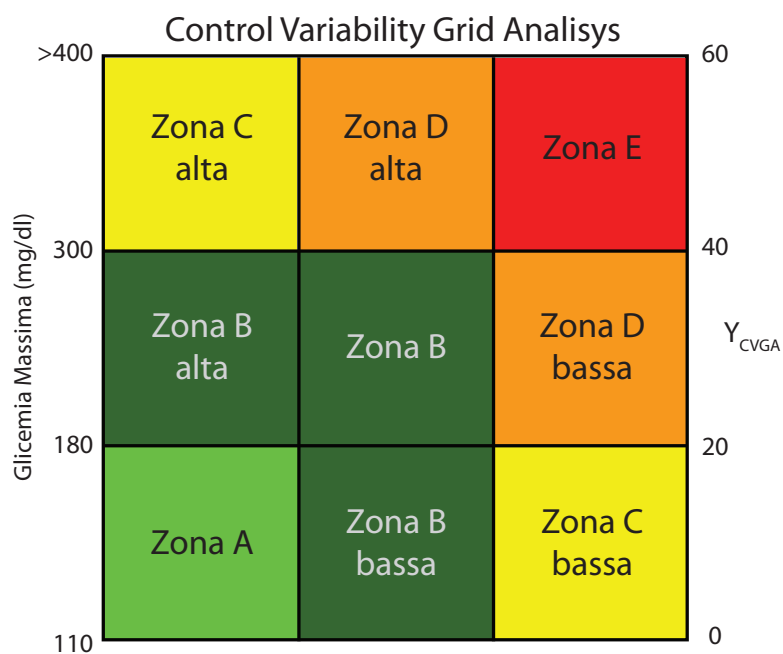


Figura 2.9: Control Variability Grid Analysis

minimi. Per penalizzare maggiormente le ipoglicemie rispetto alle iperglicemie, sull'asse delle ordinate è stata scelta una scala maggiore rispetto all'asse delle ascisse inoltre questa scala è nonlineare: in particolare è polinomiale di ordine 3 ed interpola i quattro punti (110, 0), (180, 20), (300, 40), (400, 60) contrassegnati con una label sull'asse y della CVGA (Figura 2.4.2). Nel caso che il punto si trovi al di fuori della griglia della CVGA, esso viene rappresentato sul bordo. La trasformazione che lega coordinate sulla CVGA e glicemie minime e massime è dunque

$$\begin{cases} X_{CVGA} &= \max(\min(110 - G_{MIN}, 60), 0) \\ Y_{CVGA} &= \max(\min(p(G_{MAX}, 60), 0) \end{cases} \quad (2.31)$$

dove $p(\cdot)$ è il polinomio interpolatore dei punti (110, 0), (180, 20), (300, 40) e (400, 60):

$$p(x) = 2.683 \cdot 10^{-6}x^3 - 2.210 \cdot 10^{-3}x^2 + 0.7540x - 59.44 \quad (2.32)$$

Per rendere più chiara l'interpretazione di un risultato, la griglia della CVGA è suddivisa in 9 zone, rappresentate nei grafici con diversi colori a seconda del rischio connesso.

Zona A ($G_{max} < 180$, $G_{min} > 90$): controllo ottimale della glicemia

Zona B alta ($180 < G_{max} < 300$, $G_{min} > 90$): trend benigno verso l'iperglicemia

Zona B bassa ($G_{max} < 180$, $70 < G_{min} < 90$): trend benigno verso l'ipoglicemia

Zona B ($180 < G_{max} < 300$, $70 < G_{min} < 90$): controllo accettabile

Zona C alta ($G_{max} > 300$, $G_{min} > 90$): controllo scarso per l'iperglicemia, ma ottimo per l'ipoglicemia.

Zona C bassa ($G_{max} < 180$, $G_{min} < 70$): controllo scarso per l'ipoglicemia, ma ottimo per l'iperglicemia.

Zona D alta ($G_{max} > 300$, $70 < G_{min} < 90$): pessimo controllo per l'iperglicemia.

Zona D bassa ($180 < G_{max} < 300$, $G_{min} < 70$): pessimo controllo per l'ipoglicemia.

Zona E ($G_{max} > 300$, $G_{min} < 70$): controllo errato

Un insieme di simulazioni su diversi pazienti è rappresentato sulla CVGA da una nuvola di punti che si collocano in 5 macro-zone A,B,C,D,E. Il numero di punti in ogni sezione è una metrica immediata per valutare le proprietà delle simulazioni di una popolazione.

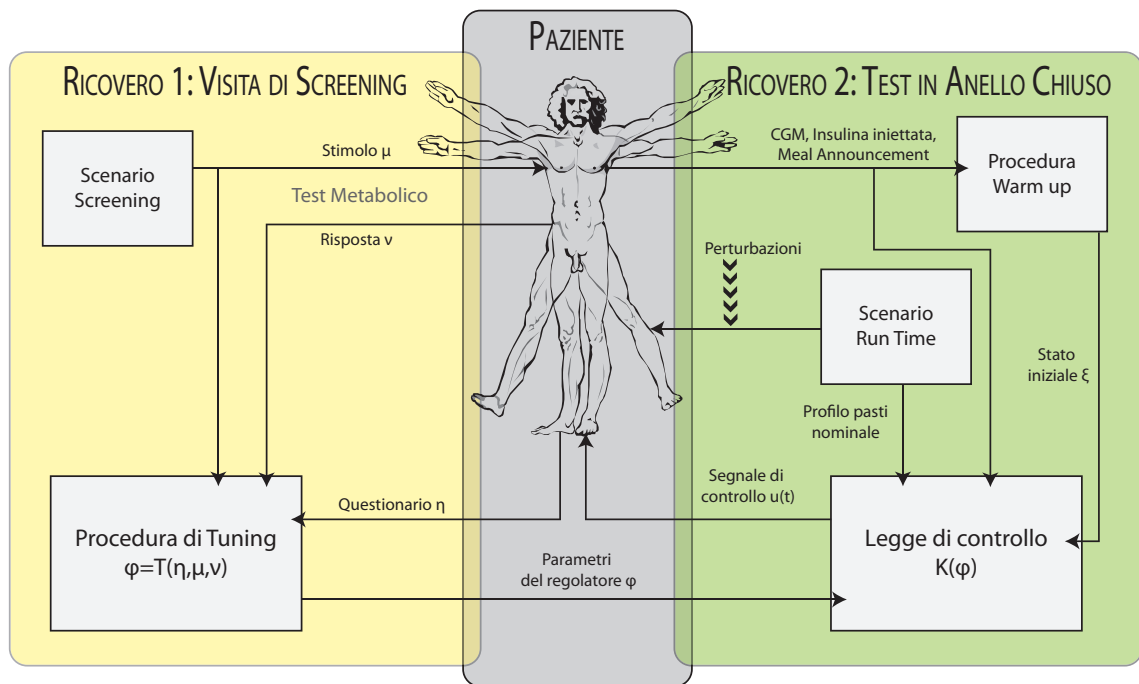


Figura 2.10: Rappresentazione schematica del protocollo

2.5 Protocollo per il test di algoritmi di controllo

Il protocollo virtuale per il test di algoritmi sul simulatore è stato studiato per ri-specchiare il più possibile i *trials* effettuati su pazienti reali negli ospedali di Padova, di Montpellier e presso l'UVA. In questa sezione presentiamo quindi parallelamente il protocollo virtuale e quello reale, sottolineando solo le differenze tra i due. È da sottolineare che i *trials* futuri verranno effettuati con un protocollo modificato e migliorato, tuttavia vale la pena di presentare il vecchio protocollo su cui sono basate molte delle simulazioni e tutti i risultati clinici. Il nuovo protocollo sarà invece presentato nel Capitolo 6.

Il test è composto essenzialmente di due parti:

- visita di screening - Admit 1
- test in closed loop - Admit 2

2.5.1 Visita di screening - Admit 1

La visita di screening ha due funzioni: raccogliere informazioni utili alla sintesi del controllore ed ottenere un riferimento con il quale misurare le prestazioni in closed loop. Le informazioni vengono raccolte in due modi:

Questionario di screening η

Il paziente compila alcune informazioni fisiologiche come peso, età ed altezza e alcune informazioni relative alla sua terapia insulinica di base (e.g. TDI). Sono dati che ogni paziente diabetico ha ben presente, ed è quindi possibile pensare di utilizzarli senza perdere di generalità sul simulatore.

Età (AGE): l'età in anni del paziente. Per i pazienti virtuali viene generata in maniera casuale con media dipendente dalla classe di appartenenza (bambini, adolescenti e adulti).

Massa Corporea (BW): il peso del paziente in Kg. È un parametro del modello di Dalla Man *et al.* che si può considerare noto.

Insulina basale (I_b): insulina plasmatica basale.

Glucosio basale (G_b): glicemia basale. È un parametro del modello di Dalla Man *et al.* che si considera noto.

Massimo Drop di Glucosio (MD): in ($\text{mg}/(\text{dL}\cdot\text{UI})$). Si calcola come la differenza tra punto di minimo e valore basale della glicemia che si osserva dopo la somministrazione a digiuno di un 1 UI di insulina.

Bolo Ottimale (CR): in (UI/g). Rappresenta il massimo bolo di insulina per grammo di carboidrato (CHO), somministrato contemporaneamente ad un pasto, che permette una sottoelongazione postprandiale rispetto alla glicemia basale non superiore al 5% del valore di G_b e rappresenta la quantità ottima di insulina da somministrare per far fronte ad un pasto contenente 1 grammo di CHO. Nel modello *in silico* è ottenuto tramite una procedura iterativa di ottimizzazione. Nella realtà è un parametro noto dal paziente.

Insulina Totale Giornaliera (TDI): la quantità media, in unità insuliniche UI, di insulina assunta nell'arco di 24 ore. Viene calcolata come media dell'insulina somministrata durante una simulazione in anello aperto della durata di 3 giorni, utilizzando una dieta da 2.7g di CHO al giorno per ogni Kg di peso corporeo distribuiti su tre pasti (0.6 g/kg, 1 g/kg e 1.1 g/kg). La terapia insulinica usata prevede una somministrazione basale pari a I_b , più dei boli in contemporanea con i pasti.

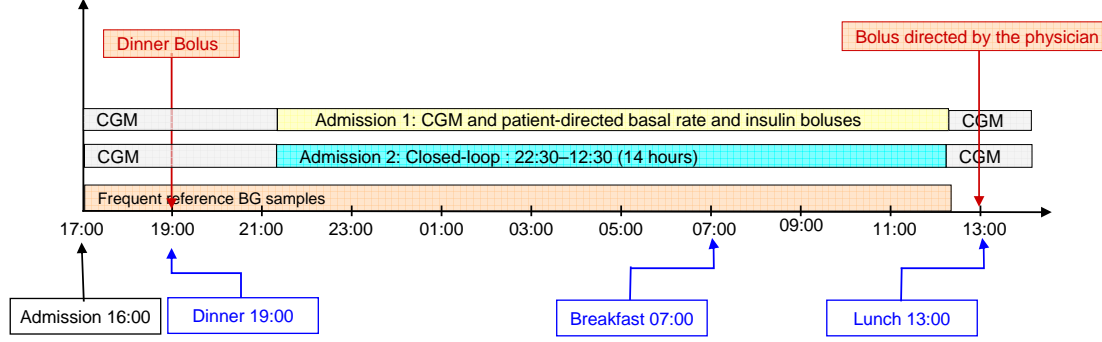


Figura 2.11: Protocollo per i test *in vivo*

Test Metabolico (μ, ν) : il paziente è sottoposto ad un test metabolico in anello aperto. Dato un input μ , costituito dai pasti $d(t)$ e dal profilo di insulina $u(t)$, si misurano le uscite rumorose ν affette da errore di misura, nel nostro caso le letture dai sensori CGM della concentrazione sottocutanea G_M e l'insulina plasmatica I . Il protocollo utilizzato è mostrato in Figura 2.11.

Il periodo in ospedale va dalle 16:00 alle 13:00 del giorno successivo. Alle ore 19:00 vi è la cena di 85 g di glucosio, alle ore 7:00 la colazione di 50 g e alla 13:00 un pranzo da 75 g. Le misurazioni CGM iniziano alle ore 17:30 del primo giorno (Figura 2.11). La terapia insulinica prevede boli contemporanei ai pasti sommati al basale costante. I pazienti sono a riposo durante il *trial* clinico ed analogamente il modello *in silico* non prende in considerazione lo sforzo fisico. Per quanto riguarda i pazienti reali, si possono avere significative variazioni nello scenario dovute a scelte particolari dei medici (e.g. boli di insulina correttivi) mentre nel caso di rischio serio di ipoglicemia sono previste somministrazioni di glucosio aggiuntive dette *hypo treatment*. Le informazioni contenute nella risposta alla visita in open loop possono essere ulteriormente elaborate per ottenere parametri utili alla sintesi del regolatore. In astratto, il controllore K dipende da un insieme di parametri ϕ che possono essere raffinati utilizzando le informazioni contenute in η, ν, μ :

$$\phi = T(\eta, \nu, \mu) \quad (2.33)$$

dove T è la *funzione di tuning* del regolatore.

2.5.2 Test in closed loop

La seconda visita prevede l'utilizzo dell'algoritmo di controllo in closed loop. I tempi di ammissione e gli orari dei pasti risultano invariati rispetto alla visita in open loop, ma dalle ore 22:30 alle ore 12:30 del giorno successivo l'algoritmo di controllo in closed loop sostituisce la terapia in open loop. Il tempo intercorrente tra l'ammissione in ospedale e l'inserimento del closed loop è chiamata fase di *warm up* del regolatore. Nel caso dei pazienti virtuali si utilizza questo intervallo temporale per verificare il corretto

funzionamento del sistema di controllo e per portare il paziente in uno stato il più possibile di equilibrio nell'istante in cui si commuta al closed loop e per inizializzare gli stati del controllore. La fase di *warm up* è prevista anche nel caso *in silico* allo stesso scopo di inizializzazione. Le metriche di valutazione del closed loop ignorano l'intervallo di tempo immediatamente successivo alla commutazione detto *intervallo di commutazione* in quanto eccessivamente dipendente dalla regolazione in open loop.

Sono quindi definite in tutto tre fasi: warm-up, commutazione e regolazione vera e propria. Nel caso dei pazienti reali le decisioni del controllore devono comunque per ragioni di sicurezza e legali essere approvate dai medici, che possono decidere di modificare la terapia in ogni istante, anche se nella pratica ciò è avvenuto in pochissimi casi. In caso di rischio di ipoglicemia, tuttavia, è ancora previsto l'utilizzo di *hypo treatment*.

Controllo Predittivo per il Pancreas Artificiale

Algoritmi di controllo MPC sono stati ampiamente utilizzati in ambito industriale negli ultimi 30 anni e solo in seguito sono emersi in ambito accademico i risultati teorici a riguardo. L'idea di base del controllo MPC è di usare un modello del processo sotto controllo per predire il suo comportamento futuro e, basandosi su tali predizioni, minimizzare una certa funzione di costo che riguarda i valori futuri di ingressi e uscite del sistema. La funzione di costo rappresenta gli obiettivi del controllo: questi sono, in generale, di mantenere le variabili di interesse il più possibile vicine ad una traiettoria nominale.

I problemi di controllo MPC sono posti naturalmente nel tempo discreto: ad ogni istante di campionamento t si risolve un problema di ottimizzazione che riguarda le variabili future di ingresso e uscita nell'intervallo $[t, t + N]$, dove N è detto orizzonte di predizione, sottoposte al vincolo della dinamica imposto dal modello e con stato iniziale x . La soluzione al problema di ottimizzazione è una sequenza di comandi ottima $U^o(k) = u^o(t) \dots u^o(t+N-1)$. Tuttavia, solo il primo elemento della sequenza ottima è realmente applicato al tempo t . All'istante $t + 1$ l'orizzonte di predizione si sposta di un campione in avanti ed il problema di ottimizzazione viene risolto di nuovo nell'intervallo temporale $[t + 1, t + N + 1]$, e con lo stato del modello opportunamente aggiornato a seconda della misura effettuate al tempo $t + 1$. Come prima, solo il primo elemento della sequenza ottima viene applicato, e così via: questo modo di operare è noto nella teoria del controllo come principio *Receding Horizon* (RH). Grazie al principio RH ad ogni istante di campionamento si tiene conto della misura corrente, e l'algoritmo di controllo è dunque in anello chiuso. L'algoritmo di controllo attualmente implementato nel progetto è un Linear Model Predictive Control (LMPC), ovvero è basato su un modello lineare del metabolismo glucosio-insulina.

3.1 Realizzazione del Controllo MPC

Le componenti fondamentali del controllore LMPC sono

- un modello lineare
- un profilo nominale
- una funzione di costo quadratica
- una tecnica di ottimizzazione

3.1.1 Modelli Lineari

L'algoritmo di controllo LMPC ha bisogno di un modello del processo sotto controllo, nel nostro caso il metabolismo glucosio-insulina. La qualità del modello utilizzato è uno dei fattori fondamentali per ottenere buona performance. Tre scelte differenti sono state considerate:

Modello Lineare Medio

Un singolo modello lineare viene generato linearizzando il modello di Dalla Man *et al.* descritto nel Capitolo 2. Si fissano i valori dei parametri del modello sul valor medio della popolazione e si linearizza il modello attorno all'equilibrio basale. In seguito, il sistema viene discretizzato, ridotto di ordine e viene trasformato in forma input-output.

Linearizzazione Si considera il modello di Dalla Man *et al.* fissando i parametri al valore medio su tutta la popolazione e fissando il parametro

$$k_{empt} = \frac{k_{max} + k_{min}}{2}$$

Otteniamo un sistema lineare con 13 stati continui che rappresenta il paziente medio sulla popolazione:

$$\begin{aligned} \dot{x} &= f(x(t), u(t), d(t)) \\ y &= Cx(t) \end{aligned} \tag{3.1}$$

dove $x(t) \in \mathcal{R}^{13}$ è il vettore di stato, $u(t)$ e $d(t)$ sono i vettori degli ingressi insulina e pasto rispettivamente e $y(t) = x_{13}(t)$ è la concentrazione sottocutanea di glucosio G_M . La trasformazione di uscita è quindi lineare e non dipende direttamente dagli ingressi:

$$C = [0 \ \dots \ 1] \in \mathcal{R}^{1 \times 13}$$

Linearizzando il sistema attorno all'equilibrio basale otteniamo il seguente sistema lineare:

$$\begin{aligned}\delta\dot{x}(t) &= A\delta x(t) + B\delta u(t) + Md(t) \\ \delta y(t) &= C\delta x(t)\end{aligned}\tag{3.2}$$

dove δx , δy and δu sono le differenze delle variabili rispetto all'equilibrio basale:

$$\begin{aligned}\delta x &= x - x_b \\ \delta y &= y - y_b \\ \delta u &= u - u_b\end{aligned}$$

e le matrici A , B ed M sono calcolate come:

$$\begin{aligned}A &= \frac{\partial f(x, u, d)}{\partial x} & B &= \frac{\partial f(x, u, d)}{\partial u} & M &= \frac{\partial f(x, u, d)}{\partial d} \\ A &\in \mathcal{R}^{13 \times 13} & B &\in \mathcal{R}^{13 \times 1} & M &\in \mathcal{R}^{13 \times 1}\end{aligned}$$

Discretizzazione Il sistema lineare a tempo continuo dell'Equazione (3.2) è in seguito discretizzato al tempo di campionamento T_s . Nell'implementazione attuale, il tempo di campionamento è posto a 15 minuti.

$$\begin{aligned}\delta x(t+1) &= A_D\delta x(t) + B_D\delta u(t) + M_Dd(t) \\ \delta y(t) &= C_D\delta x(t)\end{aligned}\tag{3.3}$$

Ci sono in generale diversi metodi per ottenere le matrici che descrivono la dinamica a tempo discreto A_D , B_D , M_D con C_D con diverse proprietà. Utilizzando il metodo ZOH si garantisce la stessa risposta negli istanti di campionamento tra il sistema in tempo discreto e il sistema a tempo continuo quando l'ingresso è costante a tratti negli intervalli $(t_i, t_i + T_s)$. Tramite semplici calcoli si giunge alla formulazione per le matrici a tempo discreto

$$A_D = e^{AT_s}, \quad B_D = \int_0^{T_s} e^{A\tau} B d\tau, \quad M_D = \int_0^{T_s} e^{A\tau} M d\tau, \quad C_D = C \tag{3.4}$$

Riduzione di Ordine e Bilanciamento Il modello che risulta dalla linearizzazione risulta essere sovrapparametrizzato, nel senso che un certo numero di stati vengono utilizzati per descrivere parti poco osservabili o poco raggiungibili del sistema, che hanno poca influenza nella relazione ingresso-uscita del sistema. È possibile inoltre trascurare le dinamiche troppo veloci del sistema, non essenziali per la sintesi del regolatore.

Per valutare in maniera quantitativa questi aspetti è però necessario ottenere una rappresentazione numericamente bilanciata del sistema. Il bilanciamento del sistema asintoticamente stabile di Equazione (3.3), con gramiani di controllabilità e

di osservabilità W_c e W_o non singolari, consiste nel trovare una trasformazione di stato $\hat{x} = T\delta x$, con T invertibile, tale che il sistema equivalente:

$$\begin{aligned}\hat{x}(t+1) &= \hat{A}\hat{x}(t) + \hat{B}\delta u(t) + \hat{M}d(t) \\ \delta y(t) &= \hat{C}\hat{x}(t)\end{aligned}\tag{3.5}$$

con

$$\begin{cases} \hat{A} &= T A_D T^{-1} \\ \hat{B} &= T B_D \\ \hat{M} &= T M_D \\ \hat{C} &= C_D T^{-1} \end{cases}\tag{3.6}$$

abbia gramiani di controllabilità ed osservabilità

$$\begin{aligned}\hat{W}_c &= T W_c T' \\ \hat{W}_o &= T'^{-1} W_o T^{-1}\end{aligned}\tag{3.7}$$

diagonali ed uguali:

$$\Sigma = \hat{W}_c = \hat{W}_o = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{13}), \quad 0 \leq \sigma_1 \leq \sigma_2 \leq \dots \leq \sigma_{13}\tag{3.8}$$

I valori sulla diagonale σ_i sono detti *valori singolari di Hankel* (HSV). Gli stati a cui sono associati HSV in modulo minori sono associati a parti poco osservabili e/o raggiungibili del sistema e possono essere eliminati dalla realizzazione bilanciata. La funzione MATLAB `balred` implementa una procedura per calcolare la realizzazione bilanciata di un sistema lineare. Specificando l'opzione `Truncate`, inoltre, è possibile ridurre il sistema ad un ordine specificato eliminando gli stati a cui sono associati HSV minori. Da considerazioni empiriche, è risultato nel nostro caso sufficiente una realizzazione di ordine 3:

$$\begin{aligned}x_t &= A_t x_t(t) + B_t \delta u(t) + M_t d(t) \\ \delta y(t) &= C_t d(t)\end{aligned}\tag{3.9}$$

in cui $x_t = [\hat{x}_1 \ \hat{x}_2 \ \hat{x}_3]'$ e le matrici A_t , B_t , M_t e C_t si ottengono eliminando le righe e le colonne corrispondenti agli stati da 4 a 13 dalle matrici \hat{A} , \hat{B} , \hat{M} e \hat{C} rispettivamente.

Realizzazione Input-output La legge di controllo MPC è basata su una retroazione dello stato: in genere, è necessaria la presenza di un osservatore (e.g. un filtro di Kalman) per tenere aggiornata l'informazione dello stato in base alle uscite misurate. Per evitare questa ulteriore complessità è possibile ricondursi ad una realizzazione input-output in cui il vettore di stato è composta dagli ingressi ed uscite passate e presenti. Dato il sistema lineare dopo la riduzione di ordine, si considera la trasformata Zeta:

$$\begin{aligned}\delta Y(z) &= C_t(zI - A_t)^{-1} B_t \delta U(z) + C_t(zI - A_t)^{-1} M_t D(z) \\ &= \frac{\text{Num}_B(z)}{\text{Den}(z)} \delta U(z) + \frac{\text{Num}_M(z)}{\text{Den}(z)} D(z)\end{aligned}\tag{3.10}$$

con

$$\begin{aligned}\text{Num}_B(z) &= b_1 z^2 + b_2 z + b_3 \\ \text{Num}_M(z) &= m_1 z^2 + m_2 z + m_3 \\ \text{Den}(z) &= z^3 - a_1 z^2 - a_2 z - a_3\end{aligned}\tag{3.11}$$

dove le lettere maiuscole indicano la trasformata Zeta delle variabili corrispondenti. Quando si parte da una realizzazione bilanciata, i denominatori delle due funzioni di trasferimento tra δU e δY e tra D e δY coincidono. Nel dominio del tempo il sistema può essere quindi riscritto come:

$$\begin{aligned}\delta y(t) &= a_1 \delta y(t-1) + a_2 \delta y(t-2) + a_3 \delta y(t-3) + b_1 \delta u(t-1) + b_2 \delta u(t-2) \\ &\quad + b_3 \delta u(t-3) + m_1 d(t-1) + m_2 d(t-2) + m_3 d(t-3)\end{aligned}\tag{3.12}$$

Modello Linearizzato Personalizzato

Dato un modello di un paziente virtuale, è possibile considerare il modello linearizzato utilizzando i suoi parametri invece che quelli del modello medio. A parte questa differenza, la procedura è tecnicamente identica alla quella della sezione precedente e dopo analoghi passaggi di discretizzazione, bilanciamento e riduzione di ordine e realizzazione input-output si arriva ad una formulazione del tipo di Equazione (3.12). Questa strategia è ovviamente possibile soltanto con i pazienti virtuali, su cui i 25 parametri conosciuti. L'identificazione di tali parametri sui pazienti reali sarebbe infatti troppo complessa. È tuttavia utile sperimentare questa soluzione sui pazienti virtuali per valutare le performance ottenibili in relazione al modello medio ed i modelli identificati.

Modelli Lineari Identificati (ARX)

Un modello lineare del paziente può essere ottenuto tramite procedure di identificazione. Questa strategia è applicabile sia sui pazienti reali che su quelli virtuali. Nel primo caso è necessaria una visita del paziente in ospedale prima del closed-loop per raccogliere le informazioni necessarie. Nel secondo, la visita stessa è un esperimento *in silico*. Questa strada è stata provata esclusivamente sui pazienti virtuali fino ad ora mentre sui pazienti reali non si è ancora provato a modificare la strategia di controllo in closed loop in base ai modelli identificati in open loop. L'argomento principale di questa tesi è valutare le performance di questa soluzione per introdurre una procedura di identificazione sui pazienti reali. Il problema è trattato in dettaglio nel Capitolo 5 in cui si presentano alcuni risultati su pazienti virtuali e reali.

I modelli ARX identificati hanno la forma:

$$\begin{aligned}\delta y(t) &= a_1 \delta y(t-1) + \dots + a_{n_a} \delta y(t-n_a) + b_1 \delta u(t-1) + \dots + b_{n_b} \delta u(t-n_b) + \\ &\quad + m_1 d(t-1) + \dots + m_{n_m} d(t-n_m)\end{aligned}\tag{3.13}$$

e sono già in forma input-output, inoltre l'ordine del modello è già ridotto opportunamente per le scelte effettuate in identificazione: non sono dunque necessari i passaggi effettuati sui modelli linearizzati. Si noti che i modelli linearizzati riformulati in forma input/output rientrano nella classe più generale dei modelli ARX dell'equazione precedente, per cui la trattazione può procedere utilizzando i modelli ARX senza perdere di generalità.

3.1.2 Formulazione matriciale Input/Output

Qualunque sia l'origine del modello, esso può essere rappresentato nella forma matriciale:

$$\begin{aligned}\xi(t+1) &= A_{IO}\xi(t) + B_{IO}\xi(t) + M_{IO}\xi(t) \\ y(t) &= C_{IO}\xi(t)\end{aligned}\tag{3.14}$$

con

$$\xi(t) = \begin{bmatrix} \delta y(t) \\ \delta y(t-1) \\ \vdots \\ \delta y(t-n_a+1) \\ \delta u(t-1) \\ \delta u(t-2) \\ \vdots \\ \delta u(t-n_b+1) \\ d(t-1) \\ d(t-2) \\ \vdots \\ d(t-n_m+1) \end{bmatrix} \in \mathcal{R}^M, \quad B_{IO} = \begin{bmatrix} b_1 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad M_{IO} = \begin{bmatrix} m_2 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathcal{R}^M$$

$$A_{IO} = \begin{bmatrix} a_1 & a_2 & \dots & a_{n_a-1} & a_{n_a} & b_2 & b_3 & \dots & b_{n_b-1} & b_{n_b} & m_1 & m_2 & \dots & m_{n_m-1} & m_{n_m} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ddots & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \in R^{M \times M}$$

$$C_{IO} = [1 \ 0 \ 0 \ \dots \ 0] \in R^M$$

dove

$$M = n_a + (n_b - 1) + (n_m - 1) = n_a + n_b + n_m - 2$$

Questa formulazione matriciale sarà utile per la soluzione del problema di ottimizzazione MPC. La rappresentazione è di certo non minima in quanto utilizza $n_a + n_b + n_m - 2$ stati quanto ne sarebbero sufficienti solo n_a . Ha tuttavia il vantaggio di avere un vettore di stato composto esclusivamente da ingressi ed uscite passate, rendendo così banale la realizzazione di un osservatore dello stato.

3.1.3 Generazione del profilo nominale

Il funzionamento dell'algoritmo MPC richiede, ad ogni istante di campionamento, la conoscenza dei valori di riferimento per le grandezze d , δu , dy all'interno dell'orizzonte di predizione. Nella implementazione attuale, l'orizzonte di predizione N è pari ad 8 campioni, ovvero 2 ore considerando che il tempo di campionamento T_S è di 15 minuti.

Profilo nominale per $d(t)$ Il glucosio assunto con i pasti rappresenta il principale disturbo agente sul sistema e l'algoritmo MPC permette di tenere in conto esplicitamente questa informazione: questa caratteristica è il vero vantaggio rispetto ad algoritmo di controllo reattivi basati sul solo *feedback* sulle variabili di uscita come i PID. In uno scenario reale, non è realistico conoscere con precisione assoluta gli istanti di arrivo dei pasti e le rispettive quantità. Si può però ipotizzare che esista una dieta nominale che il paziente normalmente segue in un determinato scenario (e.g. 3 pasti in 24 ore). Ipotizziamo che questa informazione sia contenuta in un vettore diet

di dimensione opportuna: ad ogni istante t possiamo costruire un vettore $D(t)$, che rappresenta il valore di riferimento per d all'interno dell'orizzonte di predizione:

$$D(t) = \begin{bmatrix} \text{diet}(t) \\ \text{diet}(t+1) \\ \vdots \\ \text{diet}(t+N-1) \end{bmatrix} \quad (3.15)$$

Data l'incertezza presente nell'informazione, è stato introdotto un segnale di *meal confirmation* $\delta(t)$ che deve essere attivato 30 minuti prima del pasto, con la relativa quantità. Se non avviene alcuna conferma, il pasto è considerato saltato.

Profilo nominale per $u(t)$ Il profilo nominale per l'insulina è basato sulla strategia di controllo in anello aperto ed è basato sulla combinazione di un basale costante u_b e di boli contemporanei ai pasti proporzionali alla quantità di glucosio assunta attraverso una fattore CR, costante moltiplicativa ottimizzata su ogni paziente.

Profilo nominale per $y(t)$ Il profilo nominale per $y(t)$ non è costante ma varia a seconda dell'ora del giorno in accordo con le informazioni contenute nello scenario. Durante il giorno il valore del set point è di 130 (mg/dL) ma si alza a 180 (mg/dL) in corrispondenza dei pasti per una durata di due ore. Durante le ore notturne invece è portato al valore di 110 (mg/dL).

Vettori di riferimento Definiamo $\delta y_o(t)$ e $\delta u_o(t)$ come $y_o(t) - y_b$ e $u_o(t) - u_b$ rispettivamente. Tornerà utile per la formulazione del problema MPC definire i vettori:

$$\Delta Y_o(t) = \begin{bmatrix} \delta y_o(t+1) \\ \delta y_o(t+2) \\ \vdots \\ \delta y_o(t+N-1) \\ \delta y_o(t+N) \end{bmatrix}, \quad \Delta U_o(t) = \begin{bmatrix} \delta u_o(t) \\ \delta u_o(t+1) \\ \vdots \\ \delta u_o(t+N-2) \\ \delta u_o(t+N-1) \end{bmatrix} \quad (3.16)$$

3.1.4 La funzione di costo

Al generico istante t , la seguente funzione di costo quadratica viene considerata:

$$J(\xi(t), \delta u(\cdot)) = \sum_{i=0}^{N-1} \left(q(\delta y_o(t+i+1) - \delta y(t+i+1))^2 + (\delta u_o(t+i) - \delta u(t+i))^2 \right) \quad (3.17)$$

sottoposta alla dinamica del sistema di Equazione (3.14). La funzione di costo rappresenta gli obiettivi del controllore di seguire la traiettoria nominale per l'uscita y_o utilizzando una terapia insulinica il più possibile uguale a quella in anello aperto.

Ogni deviazione dalla situazione ideale contribuisce ad alzare il valore della funzione di costo; il controllore deve trovare la sequenza di controllo ottima con lo scopo di minimizzarla.

I due obiettivi possono essere contrastanti: il parametro regolabile q stabilisce l'importanza relativa tra i due. Quando il valore di q è elevato, la deviazione di δy ha un peso maggiore nella funzione di costo rispetto ad una deviazione dell'insulina δu : di conseguenza il controllore diventa più aggressivo, nel senso che è disposto a modificare in modo sostanziale la terapia insulinica per seguire meglio l'uscita di riferimento y_o . Il problema della scelta della q è di fondamentale importanza e diverse tecniche sono state sviluppate. Esse verranno analizzate in dettaglio nella Sezione 3.2.

Il controllore deve dunque risolvere il seguente problema

$$\Delta U^O(t) = \arg \min_{\Delta U \in \mathcal{R}^N} J(\xi(t), \Delta U) \quad (3.18)$$

la cui soluzione $\Delta U^O(t)$ è un vettore $\in \mathcal{R}^N$ che rappresenta la sequenza di controllo ottima da t fino a $t + N - 1$:

$$\Delta U^O(t) = \begin{bmatrix} \delta u^O(t) \\ \delta u^O(t+1) \\ \vdots \\ \delta u^O(t+N-2) \\ \delta u^O(t+N-1) \end{bmatrix} \quad (3.19)$$

Ottimizzazione

Il problema di minimizzazione non vincolata del problema di Equazione (3.18) ha una semplice soluzione analitica in forma chiusa: questo è il vantaggio principale del controllo LMPC.

Considerando ancora l'Equazione (3.14) abbiamo, dalla formula di Lagrange:

$$\xi(t+i) = A^i \xi(t) + \sum_{j=0}^{i-1} A^{i-j-1} (B \delta u(t+j) + M d(t+j)), \quad \forall i \geq 0 \quad (3.20)$$

definendo allora i vettori

$$\Delta Y(t) = \begin{bmatrix} \delta y(t+1) \\ \delta y(t+2) \\ \vdots \\ \delta y(t+N) \end{bmatrix}, \quad \Delta U(t) = \begin{bmatrix} \delta u(t) \\ \delta u(t+1) \\ \vdots \\ \delta u(t+N-1) \end{bmatrix}, \quad D(t) = \begin{bmatrix} d(t+1) \\ d(t+2) \\ \vdots \\ d(t+N-1) \end{bmatrix}$$

e le matrici

$$\mathcal{B}_c = \begin{bmatrix} CB & 0 & 0 & \dots & 0 & 0 \\ CAB & CB & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ CA^{N-2}B & CA^{N-3}B & CA^{N-4}B & \dots & CB & 0 \\ CA^{N-1}B & CA^{N-2}B & CA^{N-3}B & \dots & CAB & CB \end{bmatrix}$$

$$\mathcal{M}_c = \begin{bmatrix} CM & 0 & 0 & \dots & 0 & 0 \\ CAM & CM & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ CA^{N-2}M & CA^{N-3}M & CA^{N-4}M & \dots & CM & 0 \\ CA^{N-1}M & CA^{N-2}M & CA^{N-3}M & \dots & CAM & CM \end{bmatrix}$$

risulta che

$$\Delta Y(t) = \mathcal{A}_c \xi(t) + \mathcal{B}_c \Delta U(t) + \mathcal{M}_c D(t) \quad (3.21)$$

La funzione di costo di Equazione (3.17) può essere scritta come

$$J(\xi(t), \Delta U(t)) = (\Delta Y_o(t) - \Delta Y(t))' \mathcal{Q} (\Delta Y_o(t) - \Delta Y(t)) + \\ + (\Delta U_o(t) - \Delta U(t))' \mathcal{R} (\Delta U_o(t) - \Delta U(t))$$

con $\mathcal{Q} = qI_N$ e $\mathcal{R} = I_N$, dove I_N è la matrice identità di dimensione n . Sostituendo $\Delta Y(t)$ dall' Equazione (3.21) otteniamo la funzione quadratica definita positiva in $\Delta U(t)$:

$$J(\xi, \Delta U) = \Delta U' \mathcal{B}'_c \mathcal{B}_c \Delta U + 2(\xi' \mathcal{A}'_c \mathcal{Q} \mathcal{B}_c + D' \mathcal{M}'_c \mathcal{Q} \mathcal{B}_c - \Delta Y'_o \mathcal{Q} \mathcal{B}_c - \Delta U'_o \mathcal{R}) \Delta U + C$$

dove la costante C rappresenta tutti i termini non dipendenti da ΔU e la dipendenza da t è stata omessa per brevità di notazione. Imponendo a zero tutte le derivate rispetto a ΔU otteniamo l'unica soluzione:

$$\Delta U^O(t) = K_y \Delta Y_o(t) - K_\xi \xi(t) - K_D D(t) + K_U \Delta U_o(t) \quad (3.22)$$

con

$$\begin{cases} K_y &= (\mathcal{B}'_c \mathcal{Q} \mathcal{B}_c + \mathcal{R})^{-1} \mathcal{B}'_c \mathcal{Q} \\ K_\xi &= (\mathcal{B}'_c \mathcal{Q} \mathcal{B}_c + \mathcal{R})^{-1} \mathcal{B}'_c \mathcal{Q} \mathcal{A}_c \\ K_D &= (\mathcal{B}'_c \mathcal{Q} \mathcal{B}_c + \mathcal{R})^{-1} \mathcal{B}'_c \mathcal{Q} \mathcal{M}_c \\ K_U &= (\mathcal{B}'_c \mathcal{Q} \mathcal{B}_c + \mathcal{R})^{-1} \mathcal{R} \Delta U_o \end{cases} \quad (3.23)$$

In accordo con il principio *receding horizon*, solo il primo termine della sequenza ottima è davvero applicato all'istante t :

$$\delta u^O(t) = \Delta U^O(1)$$

ed il comando di insulina proposto dal regolatore è dunque

$$u^O(t) = \delta u^O(t) + u_b \quad (3.24)$$

Si noti che i vettori K_y , K_ξ , K_D e K_U sono tempo-invarianti e possono essere calcolati off-line in una fase di inizializzazione del regolatore. L'ottimizzazione in real-time è così risolta con semplici moltiplicazioni matriciali con uno sforzo computazionale costante e trascurabile per qualsiasi calcolatore moderno, ed evitando rischi di problemi numerici relativi all'uso di algoritmi di ottimizzazione di natura iterativa.

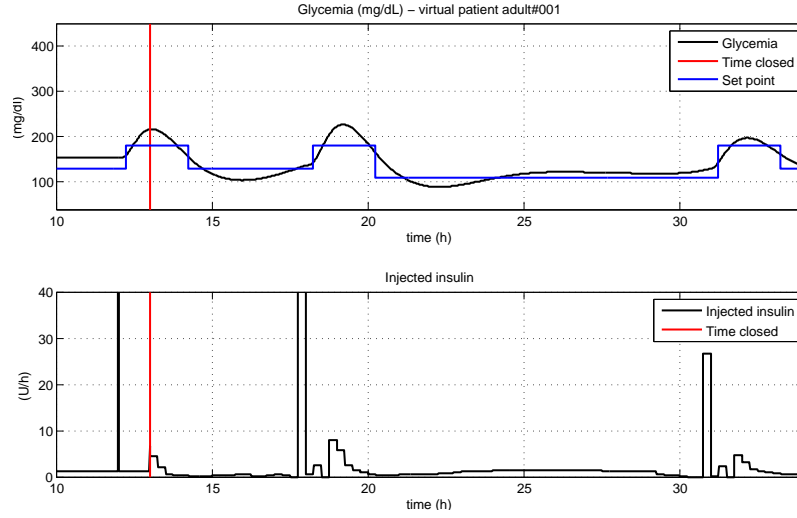


Figura 3.1: Paziente virtuale 001 in closed loop senza rumore sui sensori CGM

3.1.5 Filtro sui dati CGM

Al fine di effettuare una reiezione dei disturbi sui dati in ingresso, è stato inserito un filtro ai segnali CGM in ingresso. Invece dei segnali y segnali grezzi, il controllore riceve i segnali y_f ottenuti dal seguente filtro lineare a tempo discreto

$$Y_f(z) = \frac{az}{z + (a - 1)}Y(z) \quad (3.25)$$

il cui tempo di campionamento in ingresso è 1 minuto. Posto il parametro $a = 0.4$, il filtro ha un tempo di salita di circa 10 minuti a fronte di un ingresso a scalino.

In Figura 3.1 è mostrato l'andamento temporale di una simulazione in closed loop sul paziente virtuale 001, in assenza di rumore sui sensori CGM e con MPC basato sul modello medio e q ottimizzata tramite una procedura di calibrazione descritta nella sezione seguente. In Figura 3.2 è mostrata una simulazione basata sullo stesso controllore, ma con rumore sui sensori.

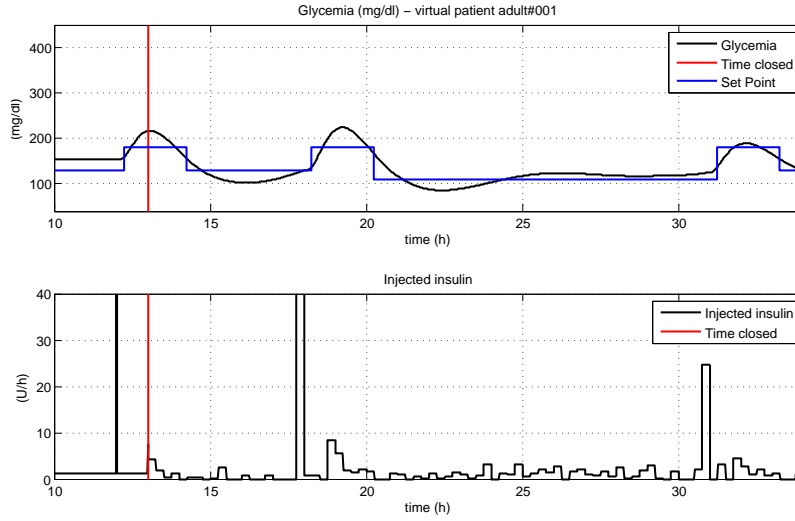


Figura 3.2: Paziente virtuale 001 in closed loop con rumore sui sensori CGM

3.2 Procedura di Tuning

L'unico parametro libero del regolatore è il peso q sull'errore dell'uscita rispetto al setpoint. Come già detto, la regolazione del parametro determina l'aggressività del controllo. Quando è troppo basso, il regolatore non effettua correzioni considerevoli rispetto all'open loop e perde di utilità nella reiezione dei disturbi. Quando è troppo elevato, il regolatore diventa troppo aggressivo, ma non potendo effettuare controregolazione espone il paziente al rischio di ipoglicemia.

L'obiettivo del *tuning* è trovare una funzione $q = T(\eta, \mu, \nu)$ che approssimi il valore ottimo per il parametro q^o dati i parametri relativi al paziente ricavati dal questionario di screening e/o dal test metabolico. Si noti che, una volta trovata la funzione T , sarà possibile ricavare le stime per q^o sia per i pazienti reali che per quelli virtuali. Il primo aspetto da considerare è la definizione di una metrica per la definizione di un valore del parametro ottimo q^o .

3.2.1 Cifra di merito

La cifra di merito utilizzata è basata sulla CVGA, introdotta nella Sezione 2.4.2. Eseguendo uno scenario clinico calibrazione con un certo parametro q , si ricava il vettore $[x \ y]'$ delle coordinate della CVGA considerando esclusivamente il *periodo di regolazione*. Si ricordi che sulla CVGA la situazione migliore si ha nell'angolo in basso a destra, di coordinate $[0 \ 0]'$. Ci si riduce in seguito ad una più comoda misura scalare considerandone la norma- ∞ :

$$C = \left\| \begin{bmatrix} x \\ y \end{bmatrix} \right\|_{\infty} = \max(|X_{CVGA}|, |Y_{CVGA}|) \quad (3.26)$$

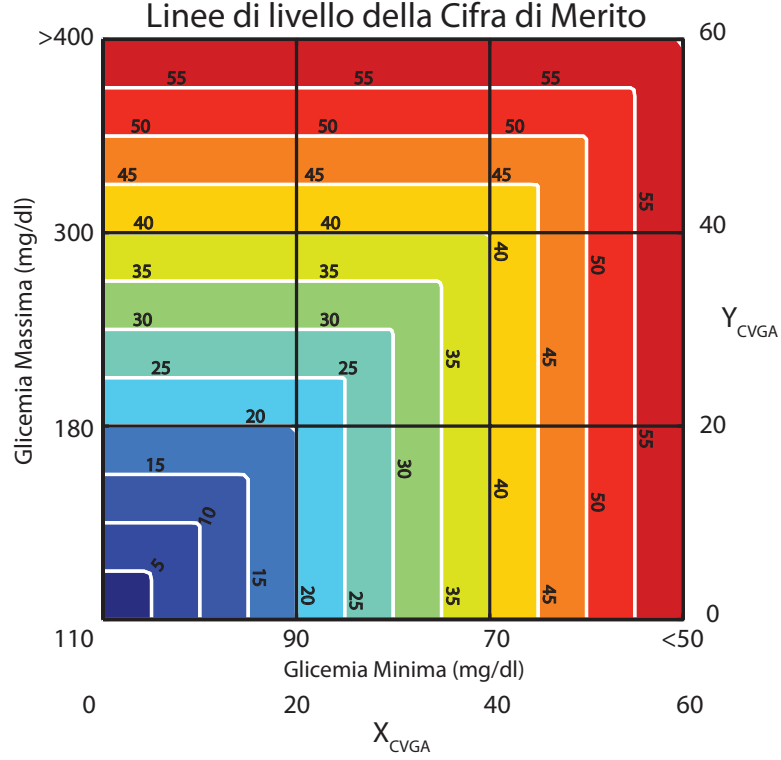


Figura 3.3: Curve di livello sulla CVGA

Questa scelta equivale a considerare linee di livello quadrate sulla CVGA (Figura 3.3). Il valore di C su un certo paziente ed un certo scenario, fissati gli altri parametri del controllore, dipende dal solo q . Possiamo definire concettualmente:

$$q^o = \arg \min_{q \in \mathcal{R}} C(q) \quad (3.27)$$

3.2.2 Calibrazione

Anche sui pazienti virtuali, di cui sono noti tutti i parametri, non si riesce a ricavare la relazione $C(q)$, e risulta quindi impossibile risolvere il problema di minimo in modo esplicito. Fissato uno scenario, $C(q)$ dipende solo dai parametri del modello del paziente: la funzione di tuning ideale sarebbe quindi data dalla soluzione del problema di minimo di $C(q)$ e dipenderebbe dai parametri del modello del paziente. In realtà non conosciamo la funzione $C(q)$ e, nel caso dei pazienti reali, neanche i parametri del modello. Per i pazienti virtuali è sviluppata una tecnica iterativa, chiamata calibrazione, che nella maggior parte dei casi converge al valore q^o con ottima precisione.

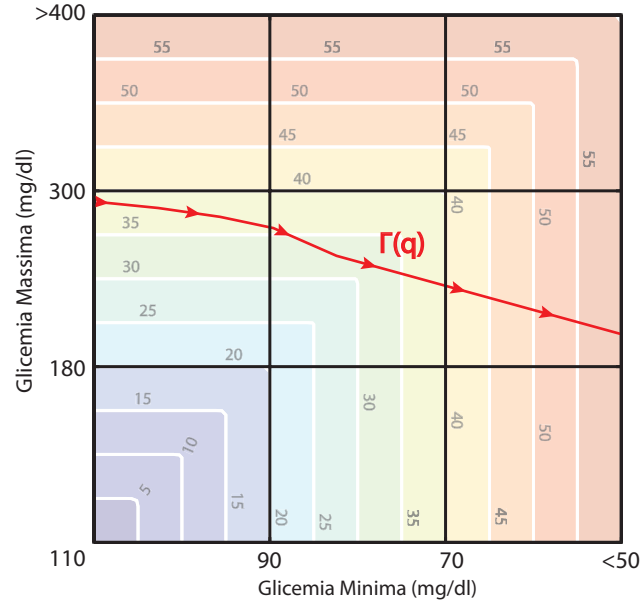


Figura 3.4: Tipica curva di calibrazione

Provando in simulazione valori crescenti di q , si vede che il corrispettivo punto sul grafico della CVGA si muove lungo una curva parametrica, detta *curva di calibrazione*:

$$\Gamma(q) = \begin{cases} x &= X_{CVGA}(q) \\ y &= Y_{CVGA}(q) \end{cases} \quad (3.28)$$

anche se non si conosce la formula analitica per la curva $\Gamma(q)$, si nota che nella maggior parte dei casi essa ha l'andamento di Figura 3.4, ovvero percorre la CVGA dall'estremo superiore sinistro verso l'angolo inferiore destro. Ciò corrisponde al fatto che ad un valore maggiore di q corrisponde un utilizzo maggiore di insulina e l'effetto è di abbassare minimo e massimo del profilo glicemico in uguale misura, lasciando approssimativamente invariato il range glicemico (Figura 3.5).

Se la glicemia minima e massima sono funzioni decrescenti del parametro q , la funzione di costo $C(q)$ ha un minimo di costo globale per il valore di q in cui Γ interseca la diagonale della CVGA (Figura 3.2.2). Più formalmente, il valore q^o è tale per cui:

$$X_{CVGA}(q^o) = Y_{CVGA}(q^o) \quad (3.29)$$

La tecnica numerica implementate sfruttano questa proprietà per trovare una approssimazione di q^o che chiameremo q^c :

Posto $\tilde{q} = \log_{10}(q)$, l'algoritmo compie i seguenti passi:

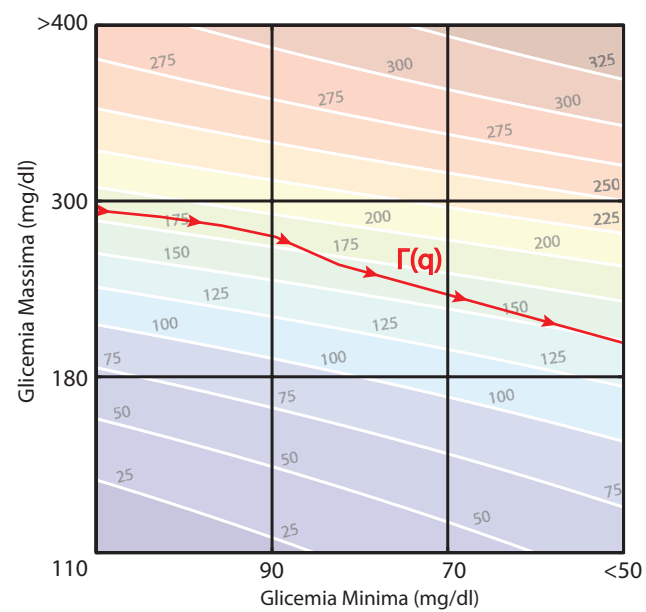


Figura 3.5: Curva di calibrazione con linee isorange

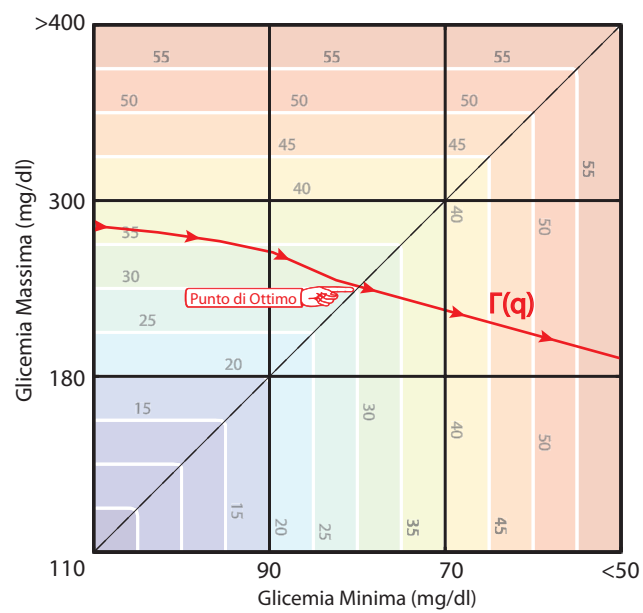


Figura 3.6: Punto di ottimo sulla CVGA

1. Per ogni paziente, sono state eseguite 6 simulazioni dello scenario clinico, utilizzando i valori per \tilde{q} corrispondenti alla seguente griglia: $-5, -4, -3, -2, -1, 0$. Per ogni simulazione viene calcolato il valore C_i della cifra di merito.
2. Vengono effettuati due passi di raffinamento della griglia in corrispondenza del minimo C_i tra tutti i punti simulati sino a quel momento, per un totale di 4 simulazioni. Si ottengono così in totale 10 punti che rappresentano un campionamento della curva di calibrazione: questo campionamento è più fitto nelle vicinanze del minimo della cifra di merito.
3. Sono state ricavate le funzioni $X_{CVGA}(\tilde{q})$ e $Y_{CVGA}(\tilde{q})$ interpolando con una funzione lineare a tratti tutti i punti ottenuti nei passi precedenti.
4. È stata calcolata la soluzione \tilde{q}^* dell'equazione $X_{CVGA}(\tilde{q}) = Y_{CVGA}(\tilde{q})$ (si veda Figura 5.8). Il valore trovato viene provato in simulazione, ottenendo anche in questo caso un valore della cifra di merito C^* .
5. Assumiamo come valore ottimo per \tilde{q} quello a cui corrisponde la cifra di merito minima tra gli 11 punti C^*, C_1, \dots, C_{10} .

3.2.3 Funzione di Tuning

Tramite la calibrazione abbiamo ottenuto dei valori q^c , praticamente identici ai parametri q^o , per la popolazione virtuale di 100 soggetti adulti. Sui pazienti reali non possiamo seguire questo approccio: la procedura sarebbe infatti rischiosa, richiedendo di provare diversi valori di q di ordini di grandezza differenti, quindi presumibilmente poco sicuri. Una procedura di calibrazione ammissibile su pazienti reali, basata sul controllo run-to-run, è stata sviluppata in [15], ma non verrà presentata in questa tesi.

Un altro approccio possibile è identificare una funzione che da un insieme di parametri θ relativi al paziente ricavati dal questionario di screening η e dal test metabolico (ν, μ) , restituisca un'approssimazione q^r del valore ottimo q^o :

$$q = T(\theta) \quad (3.30)$$

Per ottenere la relazione che intercorre tra i parametri clinici e il peso di calibrazione q^c abbiamo utilizzato lo stimatore lineare:

$$\hat{q}_i = T(\theta_i) = \bar{q}^c + \text{Cov}(Q^c, \Theta) \cdot \text{Var}(\Theta)^{-1} \cdot (\theta_i - \bar{\theta}) \approx q_i^o \quad (3.31)$$

dove \bar{q}^c è la media delle q di calibrazione sulla popolazione, θ_i è il vettore dei parametri del paziente i , $\bar{\theta}$ è il vettore dei parametri medi sulla popolazione, Θ e Q^c sono i parametri e le q di calibrazione su tutta la popolazione e gli operatori Cov e Var sono la covarianza e la varianza campionarie rispettivamente. Sotto l'ipotesi di distribuzione gaussiana di tali variabili, la funzione precedente è lo stimatore ottimo secondo il

criterio MS (Mean Square) della variabile casuale q_i^c , nel senso che minimizza la somma dei quadrati dei residui:

$$SSR = \sum_{i=1}^N (q_i^r - q_i^c)^2 \quad (3.32)$$

in cui la sommatoria è estesa a tutta la popolazione dei pazienti virtuali. Se la distribuzione dei parametri non è gaussiana, lo stimatore è comunque ottimo all'interno della classe degli stimatori lineari. Dato che le variabili casuali dei pesi e dei parametri sembravano distribuite lognormalmente, lo stimatore è stato calcolato rispetto al logaritmo delle quantità, che risultano quindi distribuite normalmente.

Scelta dei parametri θ

I parametri θ vengono scelti da un sottoinsieme di indici estratti dal questionario η o ricavati da semplici elaborazioni sulla risposta del paziente alla visita di screening (ν, μ) . Sono state considerate due scelte: l'utilizzo dei soli parametri biometrici presenti in η e l'utilizzo dei parametri biometrici e metabolici ricavati dalla visita di screening (ν, μ) .

Parametri biometrici

- Massa corporea BW
- Insulina basale I_b
- Bolo ottimale CR

Parametri metabolici

- G_b Glucosio basale
- I_b Insulina basale
- AUC_G Area sotto la curva di $G \quad \int_{t_i}^{t_f} G(t) dt$
- AUC_I Area sotto la curva di I
- AUC_{Gob} Area sotto la curva $G - G_b \quad \int_{t_i}^{t_f} (G(t) - G_b) dt$
- AUC_{Iob} area sotto la curva $I - I_b \quad \int_{t_i}^{t_f} (I(t) - I_b) dt$
- $\frac{AUC_G}{AUC_I}$
- $\frac{AUC_{Gob}}{AUC_{Iob}}$
- G_{max} massimo di $G(t)$

- I_{max} massimo di $I(t)$
- ΔG massima variazione di G rispetto al basale $G_{max} - G_b$
- ΔI massima variazione di I rispetto al basale $I_{max} - I_b$
- $\frac{G_{max}}{I_{max}}$
- $\frac{\Delta G}{\Delta I}$

Nel caso dei modelli *ARX* identificati, si è tentato di inserire alcuni parametri dei modelli tra i regressori:

- guadagno da δu a δy
- guadagno da d a δy
- risposta impulsiva da δu a δy dopo 2 ore
- risposta impulsiva da d a δy dopo 2 ore

Una procedura di *stepwise regression* è stata utilizzata per scegliere il sottoinsieme dei parametri da utilizzare tra tutti i parametri biometrici e metabolici presi in considerazione. Tale procedura ha natura iterativa: partendo da uno stimatore iniziale, e ad ogni passo esamina tutti gli stimatori che si possono ottenere aggiungendo e togliendo un parametro. Quando si aggiunge un parametro la SSR non può diminuire mentre quando ne si toglie uno non può che crescere: per questo è necessario un test statistico, nel nostro caso il test R^2 , per determinare se l'aggiunta o l'eliminazione del parametro è una *buona* scelta dal punto di vista del criterio adottato, nel senso che migliora l'indice statistico utilizzato. La modifica che porta il beneficio maggiore viene adottata all'iterazione i e viene poi analogamente analizzata all'iterazione $i + 1$. L'algoritmo termina quando tutti i modelli che differiscono da quello corrente per un solo parametro non provocano miglioramenti dell'indice statistico utilizzato. Si noti che anche in questa situazione non c'è alcuna garanzia di ottimalità della stepwise regression: è possibile ad esempio che un miglioramento statistico si verificherebbe modificando contemporaneamente due o più regressori, ma l'algoritmo non prenderà mai in considerazione questa possibilità.

Nella implementazione attuale sui pazienti reali, vengono utilizzati esclusivamente i parametri biometrici. La funzione di *tuning* sul regolatore è in definitiva:

$$q^r = \exp(r_0 + r_1 \ln(BW) + r_2 \ln(I_b) + r_3 \ln(CR)) \quad (3.33)$$

dove i parametri r_j sono ricavati dalla (3.31).

La Figura 3.7 mostra un confronto sulla CVGA tra le prestazioni ottenute con le q^c di calibrazione e le q^r di regressione. Si osserva una dispersione della nuvola della CVGA, comunque limitata nelle zone sicure A e B. Nella Figure 3.8 sono mostrate

invece le prestazioni che si ottengono introducendo il modello del rumore sui segnali CGM. In entrambe i casi i controllori sono sintetizzati sul modello medio linearizzato: i risultati ottenuti utilizzando modelli personalizzati saranno esporati nel Capitolo 6.

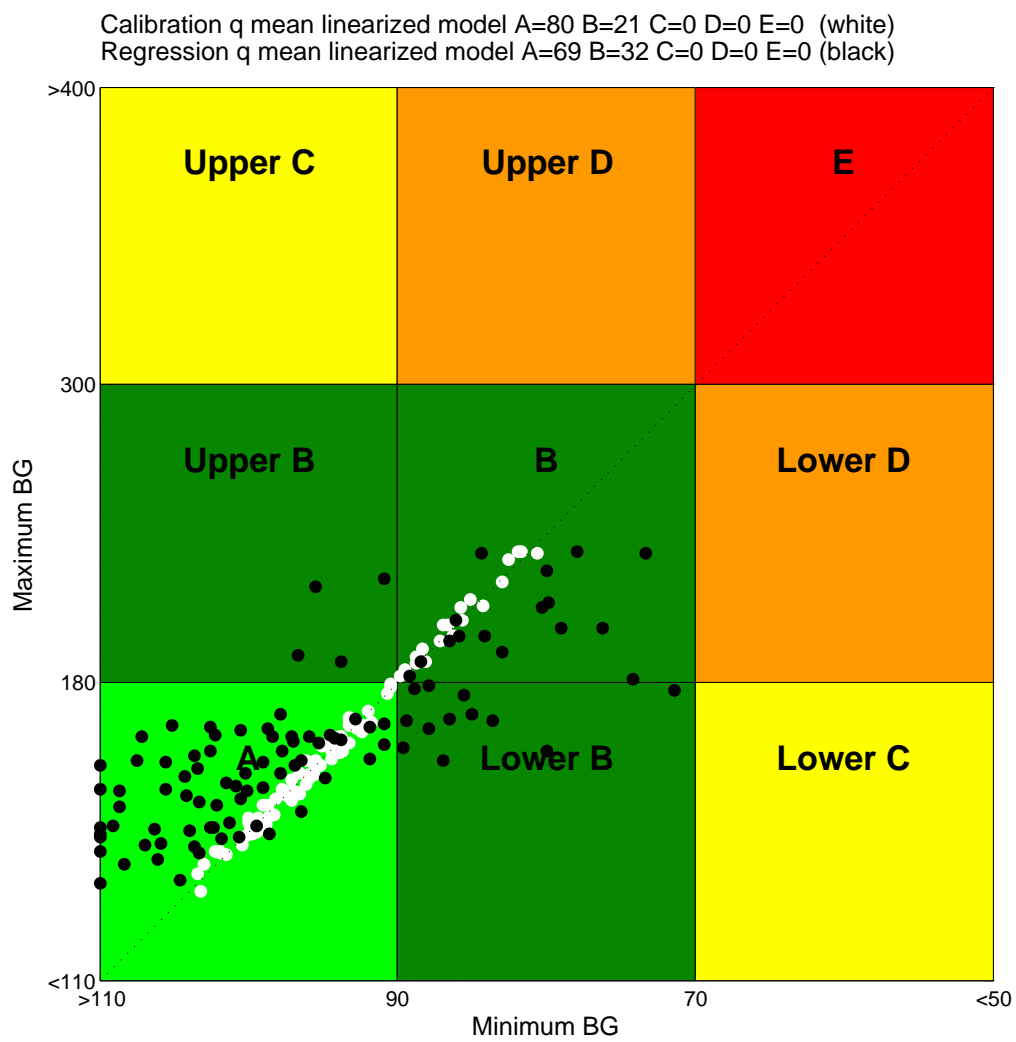


Figura 3.7: Confronto tra le prestazioni ottenute con le q da calibrazione e le q di regressione in assenza di rumore

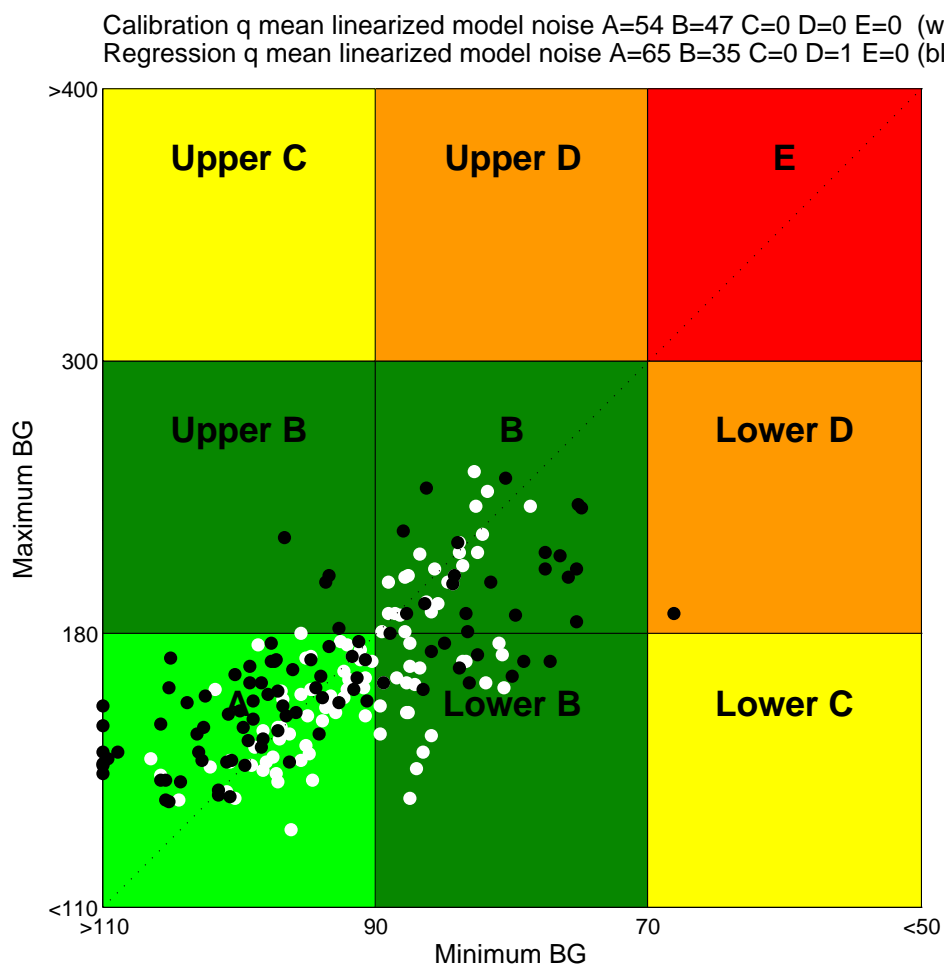


Figura 3.8: Contronto tra le prestazioni ottenute con le q da calibrazione e le q di regressione con rumore sui sensori CGM

Proposta di un'architettura modulare

È stata recentemente proposta un'architettura modulare per lo sviluppo del software di controllo del pancreas artificiale [10].

L'idea chiave di tale architettura è la suddivisione del software in più livelli (layer) indipendenti (Figura 4), ma in grado di interagire per mezzo di interfacce di comunicazioni ben note e documentate. All'interno di ogni layer un insieme di moduli si occupa di una specifica funzionalità. I tre layer superiori implementano l'algoritmo di controllo mentre il layer inferiore fornisce un servizio di astrazione dall'hardware sottostante. L'azione di controllo è suddivisa nei tre strati superiori secondo un criterio di diverse scale temporali. Il Real-Time Layer e il Continuous Time Layer sono internamente composti di tre moduli che implementano le funzionalità di Signal Management, Estimation e Control.

L'architettura è concepita per essere il più generale possibile in quanto definisce le funzioni e le interazioni tra i livelli, non una particolare implementazione per essi. È possibile ad esempio sostituire una particolare implementazione di un livello con una differente ed il sistema dovrebbe continuare a funzionare, anche se probabilmente si otterranno diverse prestazioni.

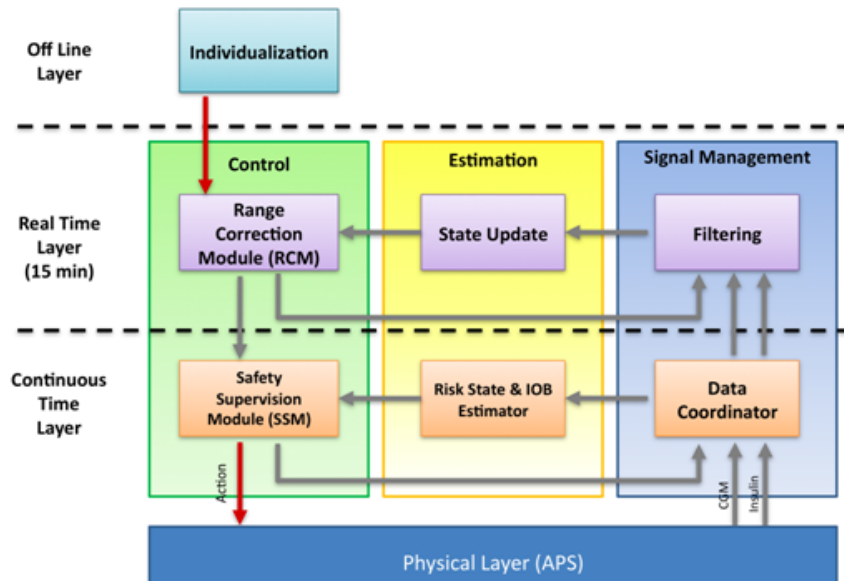


Figura 4.1: Architettura modulare per il pancreas artificiale

4.1 Off Line Layer

L'Off Line Layer si occupa di tutti i calcoli necessari all'inizializzazione dell'algoritmo di controllore e dunque entra in funzione una sola volta, fornendo alcune informazioni agli strati di controllo sottostanti. Il compito minimo che esso deve svolgere è ricavare un profilo nominale per il paziente, che il Real Time Layer utilizzerà come riferimento. L'Off Line Layer è inoltre responsabile dell'eventuale funzione di *tuning* dei regolatori di livello inferiore, ovvero può fornire informazioni utili che ne migliorano le prestazioni.

4.2 Real Time Layer

Il Real Time layer opera su una scala temporale adatta a catturare la dinamica del processo (e.g. 15 min) e ad ogni intervallo di tempo propone un comando agli attuatori con l'obiettivo di seguire il meglio possibile il profilo nominale fornito dall' Off-Line Layer

Il Filtering Module compie le funzionalità di signal management: riceve i segnali di interesse (e.g. le letture dal sensore CGM) dal Data Coordinator e ne fornisce in uscita una versione campionata alla frequenza opportuna (e.g. 15 min), compiendo anche eventualmente alcune semplici operazioni di filtraggio allo scopo di ridurre il rumore. Ci si aspetta tuttavia da questo livello una bassa complessità computazionale (e.g. filtraggio passa-basso, media mobile).

Lo State Update Module riceve dal Filtering Module i dati filtrati e campionati e compie su di essi elaborazioni più sofisticate. Lo scopo di questo blocco è mantenere una rappresentazione aggiornata dello stato del paziente (e.g. un filtro di Kalman).

Il Range Correction Module implementa le funzionalità di controllo vere e proprio: riceve le informazioni elaborate dallo State Update Module e, con l'obiettivo di mantenere il paziente il più vicino possibile al profilo nominale, propone una correzione al valore nominale di insulina. Qualsiasi algoritmo di controllo può essere utilizzato a tale scopo: l'architettura di per sé non pone particolari vincoli.

4.3 Continuous Time Layer

Il Continuous Time Layer è uno strato di sicurezza volto a prevenire le situazioni di maggiore rischio, ovvero le ipoglicemie. Esso opera alla massima velocità permessa dall'hardware utilizzato, ovvero entra in funzione ogni volta che un nuovo dato viene fornito dallo strato fisico (e.g. ad ogni lettura dai sensori CGM).

Il Data Coordinator ed il Risk State & IOB Estimator compiono a questo livello le funzionalità dal Filtering e lo State Update Module al livello superiore. Il Data Coordinator Module riceve i dati grezzi dallo strato fisico e si occupa di indirizzarli correttamente verso il Risk State & IOB Estimator Module (così come verso lo stato superiore). Il Risk State & IOB Estimator Module compie delle elaborazioni su tali dati allo scopo di prevedere il rischio di ipoglicemia. Sono state utilizzate, ad esempio, tecniche di indagine statistica come funzioni di rischio sui valori di glicemia misurati o la stima dell'insulina ancora presente in circolazione.

Il Safety Supervision Module (SSM) riceve tutte le informazioni elaborate sullo stato di rischio e, quando questo supera una certa soglia, riduce fino anche ad annullare la somministrazione di insulina.

4.4 Physical Layer (APS)

Il Physical Layer è lo strato software che permette di condurre esperimenti senza dover interagire in modo diretto con l'hardware sottostante. Esso implementa infatti i protocolli di comunicazione verso sensore e pompa fornendo un servizio di astrazione tramite il quale i livelli superiori possono interagire in modo trasparente, ovvero senza necessità di modifiche al codice.

Lo stesso hardware può essere in realtà virtuale, quando si svolgono simulazioni al computer con modelli del paziente, del sensore e della pompa: in questo modo gli stessi algoritmi di controllo possono essere testati *in silico* ed *in vivo* senza rischi di incongruenze.

4.5 L'implementazione attuale

Nell'implementazione attualmente utilizzata, l'Off Line layer fornisce i profili nominali di glucosio e insulina ed un parametro, q , utilizzato per regolare l'aggressività del RCM tramite una funzione di *tuning* T . Quest'ultima informazione è ottenuta tramite regressione lineare sui parametri biometrici ricavati dal questionario (η) e da alcuni parametri metabolici ricavati dalla risposta (μ, ν). Si sta studiando la possibilità di includere a questo livello l'identificazione di modelli personalizzati del paziente: per raggiungere questo scopo, tuttavia, un'apposita visita di screening con particolari caratteristiche sembra indispensabile.

Il Real Time Layer implementa l'algoritmo MPC descritto nel Capitolo 3. La maggior parte dei calcoli è svolta dal Range Correction Module: utilizza i profili nominali di glucosio e insulina come riferimento e, sulla base del parametro q , determina l'azione di controllo. Lo State Update Module rimane molto semplice in questa implementazione in quanto il modello usato dall'MPC è in forma ingresso-uscita: non è necessario un vero osservatore, ma è sufficiente registrare i valori passati di ingressi e uscite mentre il Filter Module è implementato da un semplice filtro lineare del primo ordine.

Il modulo SSM implementato nel Continuous Time Layer è basato sulla funzione di rischio definita da:

$$\text{Risk}(BG) = 22.765(\max(0, -(\log(BG)^{1.084} - 5.381)))^2 \quad (4.1)$$

dove BG è una stima della concentrazione di glucosio plasmatica. Il comando di insulina inviato effettivamente agli attuatori è modulato dal valore della funzione di rischio:

$$u_{SCM} = \frac{1}{1 + \text{Risk}(BG)} \quad (4.2)$$

dove u è il comando suggerito dal modulo RCM.

4.6 Vantaggi dell'Architettura Modulare

L'introduzione di un'architettura modulare è di cruciale importanza per diverse ragioni. In primo luogo, facilita il coordinamento tra i vari gruppi di ricerca che cooperano in quanto risulta più facile dividere i compiti di sviluppo e in seguito integrare i diversi sforzi in un'unica soluzione. Differenti moduli possono quindi essere sviluppati e rilasciati separatamente da diversi centri di ricerca. L'architettura è stata ideata per essere il più generale possibile: non vincola in alcun modo l'implementazione di un particolare modulo, ma ne definisce le funzionalità che deve fornire e le interfacce con gli altri moduli. Diverse soluzioni possono essere dunque assemblate in modo semplice combinando, ad esempio, diverse implementazioni dei moduli RCM e SSM, permettendo dunque di testare diversi algoritmi in un ambiente di sviluppo standard.

In secondo luogo, la divisione delle funzionalità renderà più agevole in futuro il rilascio di applicazioni commerciali legate al pancreas artificiale secondo standard ben definiti.

Identificazione di modelli a scatola nera

La grossa variabilità osservata sui pazienti virtuali e reali suggerisce che le prestazioni del controllo MPC possano essere migliorate significativamente utilizzando modelli personalizzati del metabolismo al posto del modello medio. Non si è tentato tuttavia di identificare un modello fisiologico complesso come quello introdotto nel Capitolo 2 in quanto sarebbe praticamente impossibile stimare i suoi parametri dalle brevi e rumorose osservazioni input-output in open loop. Inoltre, la strategia di controllo attualmente utilizzata prevede un modello lineare a tempo discreto.

Sono stati considerati invece modelli a scatola nera (black box), in particolare modelli lineari a tempo discreto di dimensione finita [14]. Il tempo di campionamento per l'identificazione è stato posto a 15 min, come nel Capitolo 3. Tra le varie strutture di modelli in questa classe (e.g. ARX, ARMAX, OE), nessuna ha portato a prestazioni significativamente migliori degli ARX. Questa struttura è stata quindi preferita perché la stima dei parametri è in questo caso computazionalmente molto semplice. Un altro vantaggio di questa classe di modelli è che essi sono naturalmente posti in forma ingresso-uscita, nella stessa forma in cui viene attualmente implementato il controllo MPC. In seguito verrà brevemente introdotta la teoria della stima direttamente applicata ad un modello ARX del metabolismo glucosio-insulina.

5.1 Modelli ARX

La struttura di un modello ARX per il metabolismo glucosio-insulina è

$$\begin{aligned} \delta y(t) = & a_1 \delta y(t-1) + \dots + a_{n_a} \delta y(t-n_a) + b_1 \delta u(t-1) + \dots + b_{n_b} \delta u(t-n_b) + \\ & + m_1 d(t-1) + \dots + m_{n_m} d(t-n_m) + e(t) \end{aligned} \quad (5.1)$$

dove

- $\delta y(t) = y(t) - y_b$, $y(t)$ è la glicemia al tempo t , y_b è la glicemia basale
- $\delta u(t) = u(t) - u_b$, $u(t)$ è l'insulina in ingresso al tempo t , u_b è l'insulina basale

- $d(t)$ è il pasto in ingresso al tempo t
- $e(t)$ è un termine di errore bianco gaussiano
- a_i e b_i , m_i sono i coefficienti delle parti autoregressive ed esogene rispettivamente

Introducendo l'operatore di anticipo unitario come

$$qx(t) = x(t+1) \quad (5.2)$$

e l'operatore di ritardo unitario come

$$q^{-1}x(t) = x(t-1) \quad (5.3)$$

possiamo riscrivere l'Equazione (5.1) come

$$A(q)y(t) = B(q)u(t) + M(q)d(t) + e(t) \quad (5.4)$$

dove $A(q)$, $B(q)$, $M(q)$ sono i polinomi

$$\begin{aligned} A(q) &= 1 - a_1q^{-1} - a_2q^{-2} - \dots - a_{n_a}q^{-n_a} \\ B(q) &= b_1q^{-1} + b_2q^{-2} + \dots + b_{n_b}q^{-n_b} \\ M(q) &= m_1q^{-1} + m_2q^{-2} + \dots + m_{n_m}q^{-n_m} \end{aligned} \quad (5.5)$$

preferiremo in futuro questa sintassi, quando possibile. Una classe di modelli ARX (i.e. tutti i possibili modelli ARX con una certa struttura) è quindi specificata da 3 indici interi: faremo riferimento, ad esempio ad

$$\mathcal{A} = ARX(n_a, n_b, n_m)$$

per la classe di modelli ARX con n_a termini nella parte autoregressiva, n_b e n_m nelle parti esogene relative agli ingressi δu e d rispettivamente.

Un particolare modello all'interno della classe \mathcal{A} è determinato da $M = n_a + n_b + n_m$ coefficienti che per comodità vengono raccolti in un unico vettore θ :

$$\theta = [a_1 \dots a_{n_a} b_1 \dots b_{n_b} m_1 \dots m_{n_m}]'$$

ed appartengono, in generale, a un certo domainio D , sottoinsieme di \mathcal{R}^M :

$$\theta \in D \subseteq \mathcal{R}^M \quad (5.6)$$

Il modello avente parametri θ può essere indicato come $\mathcal{A}(\theta)$.

Considereremo tre problemi principali legati all'utilizzo degli ARX

- predizione di futuri valori date delle osservazioni
- stima dei paramedi un modello all'interno di una classe
- scelta di un modello tra più classi

5.2 Predizione

Supponendo di avere già un modello ARX, cioè una struttura con i suoi parametri $\mathcal{A}(\bar{\theta})$ appartenente ad una certa classe di modelli ARX $\mathcal{A} = ARX(n_a, n_b, n_m)$. Supponiamo inoltre di avere registrato delle osservazioni passate di $\delta y(i)$, $\delta u(i)$ e $d(i)$ per $i \leq t$. Il modello può essere utilizzato per fare delle predizioni $\delta \hat{y}(t+k|t)$ di $\delta y(t+k)$ date le osservazioni passate e dei futuri ingressi.

Dato che abbiamo a che fare con processi casuali, non potremo predire il valore esatto, ma possiamo usare il valore atteso $E[\delta y(t+k|t)]$ al tempo t come migliore stima di $\delta y(t+k)$:

$$\delta \hat{y}(t+k|t) = E[\delta y(t+k|t)]$$

5.2.1 Predizione ad un passo

L'Equazione (5.1) può essere scritta come

$$\delta y(t) = \frac{B(q)}{A(q)}\delta u(t) + \frac{M(q)}{A(q)}d(t) + v(t) \quad (5.7)$$

I primi due termini sono deterministici, mentre l'ultimo è un processo casuale

$$v(t) = \frac{1}{A(q)}e(t) \quad (5.8)$$

che rappresenta il disturbo sul processo. Il valore atteso di δy è quindi la somma della parte deterministica e del valore atteso di v . Possiamo scrivere l'Equazione 5.8 come

$$v(t) = a_1 v(t-1) + a_2 v(t-2) + \dots + a_{n_a} v(t-n_a) + e(t)$$

Essendo $e(t)$ rumore bianco, il suo valore atteso è 0:

$$\hat{v}(t|t-1) = E[v(t|t-1)] = a_1 v(t-1) + a_2 v(t-2) + \dots + a_{n_a} v(t-n_a) = [1 - A(q)]v(t)$$

Dunque

$$\delta \hat{y}(t|t-1) = E[\delta y(t|t-1)] = \frac{B(q)}{A(q)}\delta u(t) + \frac{M(q)}{A(q)}d(t) + [1 - A(q)]v(t) \quad (5.9)$$

Dall'Equazione (5.7) possiamo scrivere $v(t)$ come

$$v(t) = \delta y(t) - \frac{B(q)}{A(q)}\delta u(t) - \frac{M(q)}{A(q)}d(t)$$

sostituendo nell'Equazione (5.9):

$$\delta \hat{y}(t|t-1) = [1 - A(q)]\delta y(t) + B(q)\delta u(t) + M(q)d(t) \quad (5.10)$$

Introducendo il vettore

$$\phi(t) = [\delta y(t-1) \dots \delta y(t-n_a) \delta u(t-1) \dots \delta u(t-n_b) d(t-1) \dots d(t-n_m)]'$$

la precedente Equazione si può scrivere come

$$\delta \hat{y}(t|t-1) = \phi'(t)\theta \quad (5.11)$$

Ed il predittore è dunque il prodotto scalare tra un termine noto al tempo t , $\phi(t)$, ed il vettore dei parametri θ del modello ARX. Questa importante proprietà renderà particolarmente semplice la stima dei parametri, come vedremo nella Sezione 5.3.

5.2.2 Predittore a k passi in avanti

Analogamente alla predizione ad un passo, è possibile effettuare predizioni a più passi in avanti indicate come $\delta \hat{y}(t+k|t)$ in cui si considerano note le uscite fino al tempo t e gli ingressi fino al tempo $t+k-1$.

Si consideri ad esempio il predittore ottimo a 2 passi in avanti: scrivendo l'equazione ARX nel dominio del tempo si ha:

$$\begin{aligned} \delta y(t+2|t) = & a_1 \delta y(t+1) + \dots + a_{n_a} \delta y(t-n_a+2) + b_1 \delta u(t+1) + \dots \\ & + b_{n_b} \delta u(t-n_b+2) + m_1 d(t+1) + \dots + m_{n_m} d(t-n_m+2) + e(t) \end{aligned}$$

Dovendo considerarne il valore atteso, si ha che $\delta y(i)$ sono noti per $i \leq t$, $e(t)$ è rumore bianco imprevedibile mentre il valore atteso per $\delta y(t+1)$ all'istante t è lo stimatore ottimo $\delta \hat{y}(t+1|t)$ già introdotto. Quindi:

$$\begin{aligned} \delta \hat{y}(t+2|t) = & a_1 \phi'(t+1)\theta + a_2 \delta y(t) + \dots + a_{n_a} \delta y(t-n_a) + b_1 \delta u(t+1) + \dots \\ & + b_{n_b} \delta u(t-n_b+2) + m_1 d(t+1) + \dots + m_{n_m} d(t-n_m+2) \end{aligned} \quad (5.12)$$

5.2.3 Simulazione

Ignorando le passate osservazioni dell'uscita, si può utilizzare il modello ARX come simulatore: ad ogni istante di tempo la predizione è data da:

$$A(q)\delta \hat{y}(t) = B(q)\delta u(t) + M(q)d(t) \quad (5.13)$$

che, scritta nel dominio del tempo, diventa

$$\begin{aligned} \delta \hat{y}(t) = & a_1 \delta \hat{y}(t-1) + \dots + a_{n_a} \delta \hat{y}(t-n_a) + b_1 \delta u(t-1) + \dots + \\ & + b_{n_b} \delta u(t-n_b) + m_1 d(t-1) + \dots + m_{n_m} d(t-n_m) \end{aligned} \quad (5.14)$$

in cui si nota che la predizione è funzione delle uscite predette passate, non delle vere uscite passate. Si fa riferimento talvolta alla simulazione come predizione a ∞ passi in avanti.

5.2.4 Metriche di valutazione

Supponiamo di avere un dataset, generato in qualsiasi modo, costituito dalla sequenza completa di osservazioni di ingressi e uscite:

$$Y_N = \begin{bmatrix} \delta y(1) \\ \delta y(2) \\ \vdots \\ \delta y(N) \end{bmatrix}, \quad U_N = \begin{bmatrix} \delta u(1) \\ \delta u(2) \\ \vdots \\ \delta u(N) \end{bmatrix}, \quad D_N = \begin{bmatrix} d(1) \\ d(2) \\ \vdots \\ d(N) \end{bmatrix} \quad (5.15)$$

e un modello ARX $\mathcal{A}(\theta)$ con parametri noti. Possiamo valutare quanto il modello ARX è adatto a calcolare le predizioni a k passi: si calcola la sequenza

$$Y_N^k = \begin{bmatrix} \delta \hat{y}(1| -k) \\ \vdots \\ \delta \hat{y}(N|N-k) \end{bmatrix} \quad (5.16)$$

e la si compara alla sequenza dei valori misurati Y_N .

Si noti che, a rigore, sarebbe necessario conoscere le osservazioni per tempi negativi, non presenti nel dataset. Nello specifico, sarebbe necessario avere informazioni sul sistema a partire da

$$p = k + \max(n_a, n_b, n_m) - 1$$

campioni prima dell'istante iniziale per effettuare correttamente la predizione del primo campione $\delta \hat{y}(1| -k)$. Per ovviare al problema si pongono i termini δy , δu e d a 0 per $t < 1$, oppure si escludono dalla stima i primi p termini.

Diverse metriche possono essere utilizzate per valutare la distanza tra le osservazioni Y_N e le predizioni Y_N^k . La scelta più semplice è considerare la Sum of Squared Residuals (SSR):

$$\text{SSR} = \sum_{i=1}^N (\delta y(t) - \delta \hat{y}(t))^2 = \|Y_N - Y_N^k\|^2 \quad (5.17)$$

In alternativa, si può calcolare il Root Mean Square Error (RMSE):

$$\text{RMSE} = \left[\frac{1}{N} \sum_{i=1}^N (\delta y(t) - \delta \hat{y}(t))^2 \right]^{\frac{1}{2}} = \frac{1}{\sqrt{N}} \|Y_N - Y_N^k\| \quad (5.18)$$

Un altro indice comunemente utilizzato è il fit, definito da

$$\text{fit} = \left(1 - \frac{\|Y_N - Y_N^k\|}{\|Y - \bar{Y}\|} \right) \cdot 100\% \quad (5.19)$$

dove \bar{Y} è un vettore di dimensione opportuna i cui elementi sono uguali al valor medio di Y .

Una scelta simile è il coefficiente di determinazione:

$$R^2 = \left(1 - \frac{\|Y_N - Y_N^k\|^2}{\|Y_N - \bar{Y}_N\|^2}\right) \cdot 100\% \quad (5.20)$$

L'indice fit è spesso utilizzato per la semplicità con cui può essere interpretato. Se le previsioni Y_N^k sono uguali ai alle uscite Y_N , il valore dell'indice è 100% e si dice, in gergo, che il modello spiega perfettamente i dati. Se si utilizzasse invece un predittore costante, pari al valor medio del dato osservato, si otterrebbe un valore di fit = 0%. Si noti che l'indice può anche assumere valori negativi quando la predizione è peggiore di quella fornita dal valor medio costante. La qualità del modello, ovviamente, cresce al crescere del fit ed empiricamente si può dire che un modello è già abbastanza buono quando l'indice è almeno al 60%.

L'indice R^2 condivide con il fit molte caratteristiche: anche in questo caso una predizione perfetta porta a $R^2 = 100\%$ mentre una basata sul valor medio porta a $R^2 = 0\%$. Sono entrambe metriche relative, nel senso che sono adimensionali e per questo indipendenti dall'unità utilizzata per misurare l'uscita. La metrica fit è spesso preferita in quanto più sensibile di R^2 quando $\|Y_N - Y_N^k\|$ è vicina a zero perché il termine non è al quadrato.

L'indice RMSE, al contrario, è una metrica assoluta: rappresenta una stima della grandezza dell'errore di predizione ed ha la stessa unità di misura dell'output (nel nostro caso, mg/dL).

5.3 Stima dei Parametri

Supponiamo di aver scelto una struttura ARX \mathcal{A} e che sia disponibile un dataset di osservazioni:

$$Y_N = \begin{bmatrix} \delta y(1) \\ \delta y(2) \\ \vdots \\ \delta y(N) \end{bmatrix}, \quad U_N = \begin{bmatrix} \delta u(1) \\ \delta u(2) \\ \vdots \\ \delta u(N) \end{bmatrix}, \quad D_N = \begin{bmatrix} d(1) \\ d(2) \\ \vdots \\ d(N) \end{bmatrix} \quad (5.21)$$

Il problema generale della stima parametrica è trovare un modello $\mathcal{A}(\hat{\theta}) \in \mathcal{A}$, o equivalentemente trovare un vettore $\hat{\theta}$ per il quale il modello $\mathcal{A}(\hat{\theta})$ spiega al meglio il vettore delle osservazioni.

5.3.1 Prediction Error Method (PEM)

Abbiamo visto in precedenza che lo stimatore ottimo a un passo $\hat{y}(t|t-1)$ per il modello $\mathcal{A}(\theta)$ è dato dal prodotto scalare

$$\delta \hat{y}(t|t-1) = \phi'(t)\theta \quad (5.22)$$

Date le osservazioni Y_N , U_N e D_N possiamo scrivere l'errore di predizione per il generico modello $\mathcal{A}(\theta)$ come funzione del parametro incognito θ :

$$\epsilon(t, \theta) = \delta y(t) - \delta \hat{y}(t|t-1) = \delta y(t) - \phi'(t)\theta \quad (5.23)$$

La sequenza degli errori $\epsilon(t)$, $t = 1 \dots N$ può essere vista come un vettore $\mathcal{E}(\theta) \in \mathcal{R}^N$.

L'idea base dell'algoritmo PEM è di scegliere il parametro $\hat{\theta}$ tale per cui il modello $\mathcal{A}(\hat{\theta})$ produce la sequenza di errori più piccola possibile. Data una norma del vettore $\mathcal{E}(\theta)$, generalmente nella forma

$$V_N(\theta) = \frac{1}{N} \sum_{t=1}^N l(\epsilon(t)) \quad (5.24)$$

in cui l è una funzione scalare positiva, si sceglie il parametri θ che risolve il problema di minimo:

$$\hat{\theta} = \arg \min_{\theta \in D} V_N(\theta) \quad (5.25)$$

Una scelta molto comune che rende particolarmente semplice la minimizzazione è utilizzare la norma euclidea $\|\mathcal{E}(\theta)\|_2$ e risolvere il problema con $D = \mathcal{R}^M$, ovvero senza vincoli sui coefficienti:

$$\hat{\theta} = \arg \min_{\theta \in \mathcal{R}^M} \|\mathcal{E}(\theta)\|_2 = \arg \min_{\theta \in \mathcal{R}^M} \sum_{t=1}^N (\delta y(t) - \phi'(t)\theta)^2 \quad (5.26)$$

In questo caso, la funzione da minimizzare è una forma quadratica in θ , per la quale è possibile trovare una soluzione analitica in forma chiusa. Definendo la matrice

$$\Phi(t) = \begin{bmatrix} \phi'(1) \\ \phi'(2) \\ \vdots \\ \phi'(N) \end{bmatrix} \quad (5.27)$$

possiamo scrivere il quadrato della norma euclidea al quadrato $\|\mathcal{E}(\theta)\|_2^2$ come

$$\|\mathcal{E}(\theta)\|_2^2 = (Y_N - \Phi\theta)'(Y_N - \Phi\theta) \quad (5.28)$$

Imponendo le derivate rispetto a θ a zero troviamo la soluzione non vincolata del problema di ottimizzazione:

$$\Phi'\Phi\theta = \Phi'Y_N \quad (5.29)$$

L'equazione precedente, nota come equazione normale, ricorre spesso nei problemi di identificazione parametrica. Se la matrice $\Phi'\Phi$ è invertibile esiste un'unica soluzione al problema, ovvero

$$\theta^{LS} = (\Phi'\Phi)^{-1}\Phi'Y_N \quad (5.30)$$

Si noti tuttavia che spesso si evita di invertire effettivamente la matrice $\Phi'\Phi$ per motivi di stabilità numerica: in particolare vengono utilizzate tecniche quali la fattorizzazione QR [14].

5.3.2 Eccitazione dell'input

La qualità del dataset con cui un modello viene identificato influenza fortemente la qualità dei modelli identificati. In molti ambiti è possibile sollecitare gli ingressi del sistema in maniera quasi indipendente, ad esempio con sequenze PRBS [14]. Nella terapia del diabete, tuttavia, è comune somministrare insulina in contemporanea ai pasti secondo un fattore di proporzionalità costante (CR). Avere ingressi linearmente dipendenti mette in difficoltà gli algoritmi di identificazione; nel caso della procedura PEM su modelli ARX la matrice $\Phi'\Phi$ può essere non invertibile o mal condizionata. L'algoritmo può in questo caso fallire nel senso che può non trovare una soluzione, o comunque non essere affidabile nella stima. L'influenza dell'eccitazione dell'input è un problema centrale per il problema dell'identificazione di modelli del metabolismo glucosio-insulina ed è stato studiato in [5].

5.3.3 Varianti dell'algoritmo

Introduzione di ritardi

La struttura ARX può essere modificata per prendere in considerazione ritardi più che unitari sui due ingressi:

$$A(q)\delta y(t) = q^{-k_u+1}B(q)\delta u(t) + q^{-k_d+1}M(q)d(t) + e(t) \quad (5.31)$$

dove k_u e k_d sono i ritardi associati a δu e d rispettivamente. L'implementazione del metodo PEM non cambia di molto : è sufficiente modificare opportunamente il vettore di regressione:

$$\phi(t) = [\delta y(t-1) \dots \delta y(t-n_a) \delta u(t-k_u) \dots \delta u(t-k_u-n_b) d(t-k_d) \dots d(t-k_d-n_m)]'$$

mentre i calcoli rimangono equivalenti. Si noti che aggiungendo questo grado di libertà le classi di modelli ARX dipendono da 5 parametri. Potremmo utilizzare ad esempio la sintassi:

$$\mathcal{A}_D = ARX_D(n_a, n_b, k_u, n_m, k_d) \quad (5.32)$$

per indicare una classe di modelli ARX con n_a coefficienti per la parte regressiva, n_u e n_m coefficienti per le parti esogene relative agli ingressi δu e d , k_u e k_d ritardi relativi agli ingressi δu e d .

La trattazione della Sezione 3.1.2 non perde comunque di generalità in quanto la classe \mathcal{A}_D sopra definita è un sottoinsieme di una classe senza ritardi di ordine più elevato:

$$\mathcal{A}_D \subset ARX(n_a, n_b + k_u, n_m + k_d) \quad (5.33)$$

in cui i primi k_u e k_d termini delle parti esogene sono posti a zero. Tenere conto dei ritardi in modo esplicito è una delle tecniche per inserire conoscenza a priori in in algoritmo a scatola nera.

L'utilizzo di questa tecnica porta qualche beneficio specie per quanto riguarda l'ingresso insulina, interessato da un considerevole ritardo in relazione al tempo di campionamento utilizzato.

Stima dell'equilibrio

Sui dati reali i valori di u_b e y_b si sono verificati inaffidabili; abbiamo dunque utilizzato delle stime \hat{u}_b e \hat{y}_b . Utilizziamo \hat{u}_b come la media dell'insulina basale effettivamente iniettata durante lo scenario di identificazione e \hat{y}_b come la media della glicemia in uscita. Dato che l'approssimazione può non essere corretta, si può stimarne l'errore introducendo un termine di disturbo costante γ .

Anche in questo caso la modifica è facilmente implementabile nell'algoritmo: si considera un terzo ingresso di valore sempre unitario $o(t) = 1, \forall t$

$$A(q)\delta y(t) = B(q)\delta u(t) + M(q)d(t) + \gamma o(t) + e(t) \quad (5.34)$$

Il vettore dei parametri θ per questo modello diventa

$$\theta = [a_1 \dots a_{n_a} \ b_1 \dots b_{n_b} \ m_1 \dots m_{n_m} \ \gamma]^\top \quad (5.35)$$

e il vettore di regressione

$$\phi(t) = [\delta y(t-1) \dots \delta y(t-n_a) \ \delta u(t-k_u) \dots \delta u(t-k_u-n_b) \ d(t-1) \dots d(t-k_d-n_m) \ 1]^\top$$

L'errore di predizione a un passo è ancora dato dal prodotto scalare tra ϕ e θ e la soluzione è sempre ricavabile dalle equazioni normali.

Input-output prefiltering

Il metodo PEM standard minimizza una norma della sequenza degli errori di predizione a un passo, $\epsilon(t)$. I modelli identificati con questa tecnica sono quasi sempre buoni quando utilizzati effettivamente come predittori a un passo. Tuttavia, molto spesso le prestazioni degradano rapidamente quando li si utilizza invece come predittori a più passi. Nell'applicazione del pancreas artificiale sono richieste predizioni accurate su un orizzonte di predizioni piuttosto lungo (2 ore, equivalenti ad 8 passi).

Talvolta è possibile incrementare sensibilmente le prestazioni in predizione su un orizzonte più lungo, o anche in simulazione, considerando la norma degli errori di predizione filtrata attraverso un opportuno sistema lineare stabile:

$$\epsilon_f(t) = L(q)\epsilon(t) \quad (5.36)$$

e scegliendo il parametro che minimizza la norma su tale sequenza. L'implementazione di questa tecnica non porta particolari difficoltà tecniche: è sufficiente impiegare in fase di calcolo i segnali filtrati $\delta y_f(t) = L(q)\delta y(t)$, $\delta u_f = L(q)\delta u(t)$ e $d_f(t) = L(q)d(t)$. Per maggiori dettagli su questa tecnica, si può fare riferimento a [14].

Scelta della norma

Un'altra possibilità è considerare un'altro tipo di norma al posto della norma euclidea standard $\|\cdot\|_2$. Una scelta spesso utilizzata, dipendente dal numero di elementi nel

dataset N , è ad esempio:

$$V_N(\theta) = \frac{1}{N} \sum_{i=1}^N \lambda^{t-i} \epsilon(i)^2, \quad 0 < \lambda \leq 1 \quad (5.37)$$

dove il termine λ è detto *forgetting factor*.

Questa scelta può essere utilizzata per effettuare stime on-line, man mano che i dati vengono generati. All'arrivo della nuova osservazione all'istante t ($\delta y(t), \delta u(t), d(t)$) la procedura di identificazione PEM viene effettuata di nuovo tenendo in considerazione il nuovo dato ma pesando le osservazioni precedenti con un termine che decade in modo esponenziale col fattore λ , permettendo così di dare maggiore un peso ai dati più recenti. Questa tecnica permette di seguire caratteristiche di un modello tempo-variante ed è stata sperimentata su dati di pazienti reali in [4]. La scelta del parametro λ rappresenta un compromesso tra la velocità con cui si possono seguire i cambiamenti del modello e il numero di dati presi in considerazione nella minimizzazione.

Scritta in forma matriciale, la funzione da minimizzare diventa, con lo stesso significato delle variabili della 5.28

$$\|\mathcal{E}(\theta)\|_{\Lambda} = (Y_N - \Phi\theta)' \Lambda (Y_N - \Phi\theta) \quad (5.38)$$

dove $\Lambda = \text{diag}(\lambda^{N-1}, \lambda^{N-2} \dots \lambda^0)$.

Sempre imponendo le derivate rispetto a θ a zero, si trova la condizione:

$$\Phi' \Lambda \Phi \theta = \Phi' \Lambda Y_N \quad (5.39)$$

e la soluzione, quando esiste ed è unica, è analogamente data da:

$$\theta^{RE} = (\Phi' \Lambda \Phi)^{-1} \Phi' \Lambda Y_N \quad (5.40)$$

Si noti che le condizioni di risolubilità $\Phi' \Lambda \Phi$ invertibile e $\Phi' \Phi$ invertibile sono di fatto equivalenti.

È possibile anche utilizzare altre norme, che non portano alla minimizzazione di una forma quadratica, tuttavia raramente si segue questa strada perché non è semplice trovare una soluzione in forma chiusa come nel caso precedente.

Vincoli sui Parametri

È possibile risolvere il problema di ottimizzazione (5.25) ponendo dei vincoli sui parametri da stimare. Dal punto di vista concettuale non vi è alcuna differenza particolare, ma dal punto di vista risolutivo non esiste più una soluzione in forma chiusa ed un ottimizzatore numerico è necessario per ottenere una soluzione approssimata. L'obiettivo di porre dei vincoli sui parametri è di inserire conoscenze fisiche ad un algoritmo di per sé a scatola nera.

Una possibile scelta è porre dei vincoli sui guadagni in continua per i due ingressi δu e d , facilmente esprimibili nei termini dei coefficienti a_i , b_i ed m_i :

$$\begin{aligned}\frac{\Delta y}{\Delta u} &= \frac{\sum_{i=1}^{n_b} b_i}{1 - \sum_{i=1}^{n_a} a_i} \\ \frac{\Delta y}{\Delta d} &= \frac{\sum_{i=1}^{n_b} m_i}{1 - \sum_{i=1}^{n_a} a_i}\end{aligned}\tag{5.41}$$

dove Δy nelle due equazioni è la variazione finale di uscita dovuta a una variazione degli ingressi δu e d rispettivamente. È naturale imporre quantomeno che il guadagno relativo a δu sia negativo e quello relativo a d positivo. I dati di identificazione, infatti, hanno talvolta un input talmente poco eccitante da portare a stime con guadagni aventi segni errati.

Sarebbe opportuno inoltre imporre la stabilità dei modelli identificati, dato che il sistema glucosio-insulina è stabile. Il vincolo sul guadagno in continua, d'altra parte, non ha senso se il modello identificato non è stabile. Per imporre la stabilità del modello sarebbe sufficiente, in teoria, richiedere che tutte le radici del polinomio $q^{n_a}A(q)$ abbiano modulo minore di uno, condizione difficile da esprimere in termini dei coefficienti polinomiali.

PEM a più passi

È possibile in teoria impostare la procedura PEM per minimizzare l'errore di predizione a più passi in avanti ed ottimizzare così le predizioni a lungo termine. Questa scelta porta tuttavia ad un problema non lineare nei parametri per il quale non è prevista una soluzione in forma chiusa.

5.4 Scelta tra più classi ARX

Mentre la scelta di un modello all'interno di una classe è abbastanza immediata tramite l'algoritmo PEM, la scelta un modello all'interno di più classi è un problema più complesso. Anche limitandosi alla struttura ARX, restano da scegliere gli ordini n_a , n_b ed n_m e, se si vuole tenere conto esplicitamente dei ritardi sulle due linee in input, k_u e k_d . Inoltre, quando si ha a che fare con classi di modelli di questo tipo, non ha molto senso scegliere il modello che minimizza gli errori di predizione a un passo: questa scelta porterebbe per forza al modello migliore all'interno della classe con più parametri, rendendo di fatto inutile la valutazione delle classi con meno parametri.

Usare un modello sovrapparametrizzato porta un duplice svantaggio: oltre a complicare inutilmente i calcoli, i parametri di troppo tendono a modellizzare la particolare realizzazione del rumore presente nel dataset. Quando il modello è usato per fare predizioni su un altro dataset, le prestazioni sono peggiori di quelle che si sarebbero ottenute con modelli più semplici. Questo problema, comune in molti ambiti dell'identificazione parametrica, è noto come *overfitting*. Esistono numerose tecniche per scegliere il modello più accurato evitando problemi di overfitting.

5.4.1 Crossvalidazione

Quando si hanno parecchie osservazioni a disposizione, vale la pena di dividere il dataset originale in tre parti:

- dataset di identificazione
- dataset di validazione
- dataset di test

L'algoritmo PEM, in una delle sue possibili varianti, viene applicato al dataset di identificazione utilizzando differenti classi di modelli ARX (i.e. si provano differenti combinazioni degli ordini dei modelli); per ogni classe si trova il miglior modello secondo il criterio PEM. Il modello viene quindi utilizzato come predittore (a uno o più passi) o come simulatore sul dataset di validazione: si sceglie infine il modello che mostra il minor indice SSR (o equivalentemente maggiore fit) nel dataset di validazione. Si valuta infine la qualità del modello identificato per mezzo degli stessi indici sull'ultimo dataset di test.

Questa tecnica è particolarmente semplice sia dal punto di vista concettuale che applicativo. Purtroppo però i dati sui pazienti reali a nostra disposizione sono spesso pochi e difficilmente si può sprecare un dataset per la sola validazione. Sui pazienti *in silico*, d'altra parte, possiamo svolgere un gran numero di esperimenti, ed è dunque possibile esplorare questa tecnica.

5.4.2 Criterio AIC

Utilizzando questo criterio non è necessario avere un dataset di validazione. Considerando un certo insieme di classi di modelli, si seleziona quello che minimizza l'indice AIC:

$$\text{AIC} = \frac{2M}{N} + \ln(\text{SSR}) \quad (5.42)$$

che rappresenta un compromesso tra la complessità del modello (il termine $\frac{2M}{N}$) e l'accuratezza con cui si seguono i dati (il termine $\ln(\text{SSR})$).

5.4.3 Vincoli di selezione

Se non si pongono vincoli esplicitamente nell'algoritmo PEM capita spesso che il modello selezionato, sia usando la crossvalidazione che criteri come l'AIC, presenti caratteristiche non desiderabili in quanto non corrispondenti alla conoscenza a priori che abbiamo del metabolismo. La procedura di selezione del modello è quindi stata modificata per scartare:

- modelli instabili

- modelli con guadagni invertiti (i.e. guadagno di δu positivo o guadagno di d negativo)
- modelli con segno della risposta all'impulso invertito dopo un'ora

Queste scelte sono di natura essenzialmente empirica: si è verificato che per mezzo di esse, nel più dei casi, i modelli identificati avevano proprietà accettabili. Non si è scelto, ad esempio, di scartare i modelli con risposta impulsiva inversa dal primo passo perché in molti casi accadeva che i modelli con questa caratteristica, per quanto indesiderata e non corrispondente all'intuito fisico, permettevano comunque buone prestazioni.

Anche se questa tecnica non è tanto efficace quanto lo sarebbe porre questi vincoli nella funzione di stima PEM, è decisamente più facile da implementare. Intere classi ARX non vengono considerate perché il modello trovato dall'algoritmo PEM non vincolato è scartato nella fase di model selection, mentre è possibile che l'ottimo vincolato all'interno della classe abbia le caratteristiche cercate. Se si pongono vincoli troppo stringenti nella fase di selezione, del resto, può capitare anche che nessun modello rispetti i requisiti. È tuttavia decisamente più semplice inserire i vincoli nella fase di model selection che nell'algoritmo PEM.

5.5 Risultati Sperimentali

5.5.1 Aspetti Tecnici

L'identificazioni dei modelli è stata effettuata con l'aiuto del System Identification ToolboxTM di MATLAB. Questo software, potente e flessibile, implementa vari algoritmi di identificazione parametrica tra cui il PEM da noi utilizzato per i modelli ARX oltre a strumenti di visualizzazione dei dati che aiutano l'utente nella scelta. Non sono previste tuttavia la possibilità di settare esplicitamente vincoli sui coefficienti nell'algoritmo PEM, né la possibilità di effettuare la stima ottimizzando l'errore di predizione a più passi, che pure porterebbe a un problema di ottimizzazione non risolubile in forma chiusa. È possibile invece specificare un prefiltro da utilizzare sui dati.

Un certo lavoro di preprocessing sui dati è necessario per importare i dati nel toolbox nel formato corretto. I dati sugli esperimenti reali sono raccolti in clinica su fogli di lavoro excel: sono stati quindi sviluppati algoritmi per estrarle automaticamente da questi documenti in modo da evitare un lavoro manuale noioso e incline ad errori. Le misurazioni CGM, inoltre, sono talvolta registrate ad intervalli irregolari, mentre la teoria dell'identificazione da noi analizzata richiede misure ad intervalli costanti. È stato dunque necessario interpolare le misurazioni e ricampionare a frequenza costante (15 min) tali dati. È stata utilizzata la funzione di MATLAB `interp1`, che effettua una interpolazione lineare tra i punti adiacenti.

Le misure della pompa non si prestano ad essere ricampionate con questa strategia, in quanto l'input è discontinuo in corrispondenza dei boli di insulina. Si è quindi considerata la media dell'integrale in ogni intervallo di campionamento.

5.5.2 ARX su pazienti virtuali

Sui pazienti virtuali è possibile effettuare un gran numero di esperimenti *in silico* con diverse possibilità nella scelta dell'input. Non avrebbe molto senso, d'altro canto, proporre scenari di identificazione con ingressi fortemente eccitati ma rischiosi, che non sarebbero mai accettati da medici su pazienti reali. Il lavoro sui pazienti virtuali ha dunque come scopo trovare il giusto compromesso tra sicurezza del paziente ed eccitazione dell'input e misurare quali risultati in termini di identificazione questo permette di ottenere.

I risultati sui pazienti virtuali dovrebbero quindi guidare lo sviluppo di protocolli specifici per l'identificazione da usare in seguito sui pazienti reali. L'eccitazione dell'input si ottiene in genere anticipando e/o posticipando i boli di insulina, che nella terapia tradizionale sono somministrati in concomitanza dei pasti.

Considerazioni sul modello

Alcune semplici considerazioni sul modello nonlineare da identificare possono essere fatte per interpretare i risultati che si ottengono con i modelli ARX identificati. Le Figure 5.1 e 5.2 mostrano delle risposte all'impulso normalizzate dagli ingressi glucosio $d(t)$ ed insulina $\delta u(t)$ all'uscita misurabile G_M per diversi valori possibili dell'ingresso sul modello nonlineare di Dalla Man *et al.* medio. La risposta normalizzata ad impulsi di diverse quantità di insulina sono sensibilmente diverse: questo perché il modello dell'utilizzo insulino-dipendente è fortemente non lineare (si veda l'Equazione 2.12). È quindi evidente che un modello lineare, in cui gli impulsi normalizzati coincidono, non possa spiegare perfettamente il comportamento del sistema quando questo è sottoposto ad ingressi $\delta u(t)$ molto differenti: per questa ragione risulta difficile valutare la bontà dei modelli ARX identificati in generale e le prestazioni possono cambiare utilizzando uno scenario di test differente.

La risposta normalizzata ad impulsi su $d(t)$ è invece praticamente identica utilizzando pasti di diverse entità. È tuttavia presente un picco deformato nell'uscita G_M , dovuto all'andamento bifasico della *rate of appearance* (Figura 2.3). Questa caratteristica rende problematica l'identificazione di modelli basata su boli ritardati, che d'altra parte è la strategia più accettabile dal punto di vista medico. L'azione dell'insulina si sovrappone in tutto e in parte al picco dovuto all'assorbimento ritardato del glucosio e l'algoritmo PEM può identificare modelli con guadagni non corretti o con un comportamento oscillante.

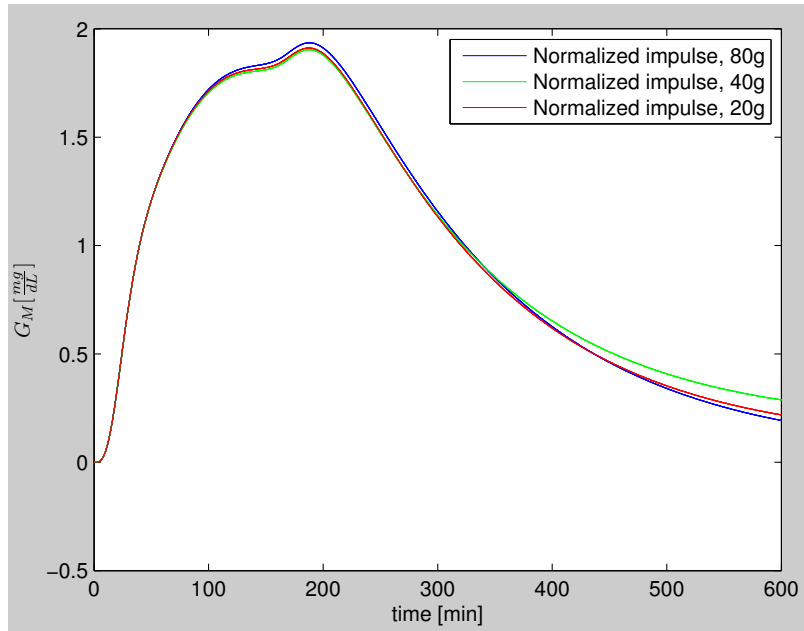


Figura 5.1: Risposte ad impulsi da d a G_M normalizzate

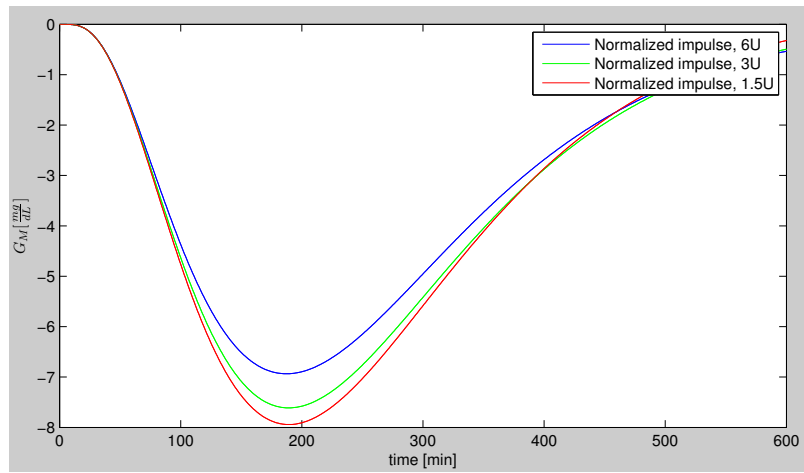


Figura 5.2: Risposte ad impulsi da δu a G_M normalizzate

ARX con crossvalidazione

La situazione ideale per l'identificazione è avere a disposizione due dataset separati per l'identificazione e la validazione, generati con scenari abbastanza differenti: questa strategia è stata utilizzata in [15]. Per garantire buone proprietà di eccitazione dell'input i boli, normalmente somministrati in concomitanza ai pasti, sono stati divisi in due parti, mezz'ora prima e mezz'ora dopo del pasto. Lo scenario di identificazione utilizzato è una giornata con tre pasti da 50, 75, e 85 grammi di glucosio. La terapia insulinica è stata modificata e prevede il normale valore basale costante e due boli per ogni pasto, di entità rispettivamente pari al 60% e 40% del valore nominale, somministrate rispettivamente mezz'ora prima e mezz'ora dopo il pasto (Figura 5.3).

Lo scenario usato per la validazione è analogo al precedente, ma le entità dei boli sono invertite. L'algoritmo di stima utilizzato è il PEM con minimizzazione dell'errore di predizione a un passo. Un prefiltraggio dei dati è stato utilizzato per ottimizzare le prestazioni in simulazione. Il toolbox fornisce un filtro ottimizzato a tale scopo, ottenuto tramite un'analisi dello spettro dell'input [13]. È sufficiente settare la proprietà `Focus a Simulation` nella funzione di stima ARX. Sono state considerate le classi ARX con ordini $2 \geq n_a \leq 5$, $2 \geq n_b \leq 5$, $2 \geq n_m \leq 5$, $1 \geq k_u \leq 2$, $k_d = 1$. Per brevità di notazione, diremo che la ricerca avviene nell'insieme di classi $ARX_D(2:5, 2:5, 1:2, 2:5, 1)$. Per ognuna delle $4 \times 4 \times 4 \times 2 = 128$ classi di modelli ARX è stato trovato lo stimatore ottimo e tra questi stato selezionato il modello con valore di fit più alto (o equivalentemente minor RMSE) nello scenario di validazione scartando quelli che non rispettavano i vincoli di selezione.

Le prestazioni del modello sono state considerate in simulazione (Sezione 5.2.3). I risultati per il paziente virtuale 001 sono mostrati in termine di fit su uno scenario di test, costituito dalla giornata nominale con boli contemporanei ai pasti (Figura 5.4). Viene mostrato inoltre l'andamento del modello medio linearizzato sui parametri del paziente. Non è riportato l'andamento temporale del modello linearizzato sul paziente medio in quanto le sue prestazioni sono nettamente inferiori: il valore di fit che si ottiene con questo modello è di -243.6% e la glicemia simulata arriva ad assumere valori negativi. Questo risultato non deve stupire: il modello medio è, in generale, un pessimo simulatore. I dati di sintesi sulla popolazione di 100 pazienti virtuali sono riportati in Tabella 5.3.

Le prestazioni in simulazione dei modelli ARX sono nettamente migliori anche dei modelli linearizzati sui parametri del paziente. Questo risultato, non del tutto scontato, è comunque comprensibile: il modello linearizzato è infatti la migliore approssimazione del non lineare solo quando ingressi e uscite sono in un intorno del punto di equilibrio, che può essere anche molto ristretto. È dunque possibile che per oscillazioni più ampie esistano modelli lineari che seguono i dati in maniera più fedele i dati generati dal simulatore rispetto al modello linearizzato.

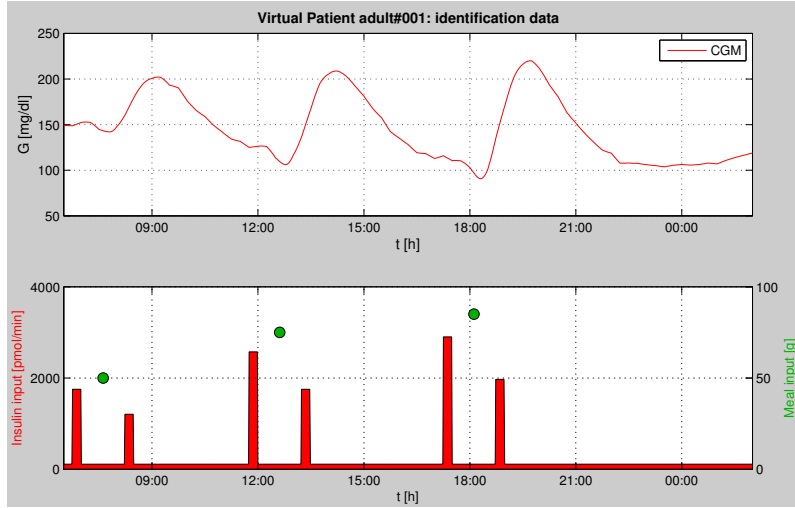


Figura 5.3: Paziente virtuale 001: scenario di identificazione con crossvalidazione

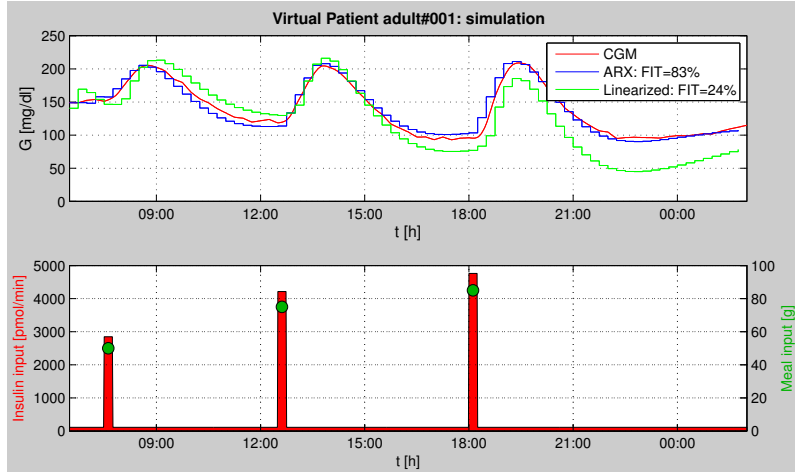


Figura 5.4: Paziente virtuale 001: scenario di test con crossvalidazione

	1	2	3	4	5
n_a	-	18	18	38	26
n_b	-	30	17	14	39
n_m	-	12	20	18	50
k_u	45	55	-	-	-

Tabella 5.1: Modelli ARX con crossvalidazione: frequenza dei vari ordini del modello identificato nei 100 pazienti analizzati

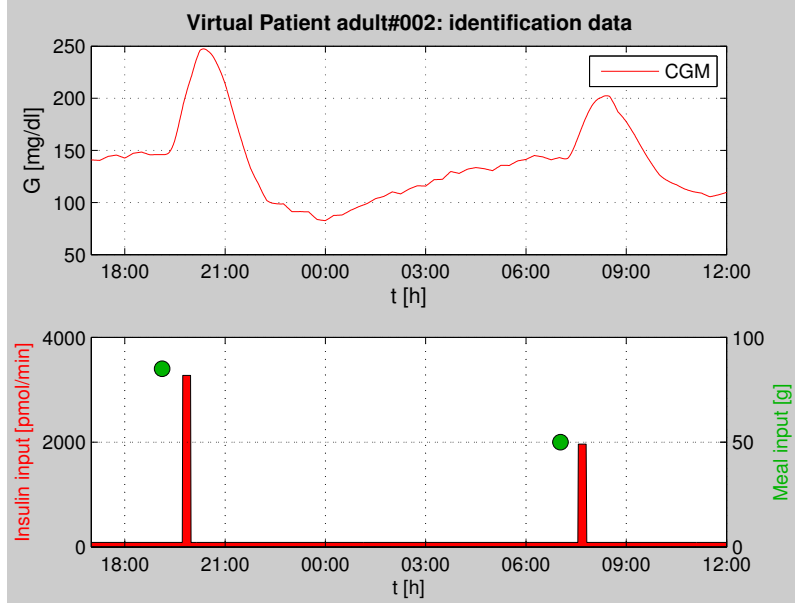


Figura 5.5: Paziente virtuale 003: scenario di identificazione

ARX con AIC

Si è testata sui pazienti virtuali la situazione in cui i dati a disposizione sono pochi, e non è ragionevole utilizzare un dataset per la sola crossvalidazione. Vi sono dunque solamente uno scenario di identificazione ed uno scenario di test. Lo scenario di identificazione è lungo 19 ore e comprende due pasti: una cena di 85 grammi e una colazione di 50. La terapia insulinica è costituita dalla combinazione del basale nominale e di boli di quantità nominale ritardati di 30 minuti. L'andamento temporale di tale scenario per il paziente virtuale 003 è riportato in Figura 5.5. Lo scenario di test è costituito da 24 ore con pranzo, cena e colazione di 75, 85 e 50 grammi con boli di insulina contemporanei ai pasti per pranzo e cena e un bolo ritardato di un'ora per la colazione. Lo scenario di test è invece equivalente al precedente.

È stato applicato l'algoritmo PEM con minimizzazione di predizione ad un passo nelle classi di modelli ARX con ordini $2 \geq n_a \leq 5$, $2 \geq n_b \leq 5$, $1 \geq k_u \leq 2$, $2 \geq n_m \leq 5$, $1 \geq k_u \leq 2$ e con prefiltraggio dell'input, settando l'opzione **Focus** a **Simulation** nella funzione di stima PEM. È stato selezionato il modello migliore secondo il criterio AIC scartando quelli che non rispettavano i vincoli di selezione.

In Figura 5.6 sono mostrati i profili di input ed output per il paziente virtuale 003 sullo scenario di test insieme all'andamento del simulatore ARX. Il risultato in termini di fit è ancora soddisfacente anche se inferiore al caso con crossvalidazione. Dal profilo temporale, comunque, si vede che il modello ha colto bene la dinamica del processo e le prestazioni sono ancora superiori a quelle del modello linearizzato. Si nota inoltre che la perturbazione dello scenario nominale necessaria per dare una sufficiente eccitazione all'input non è stata troppo rischiosa per il paziente in termini

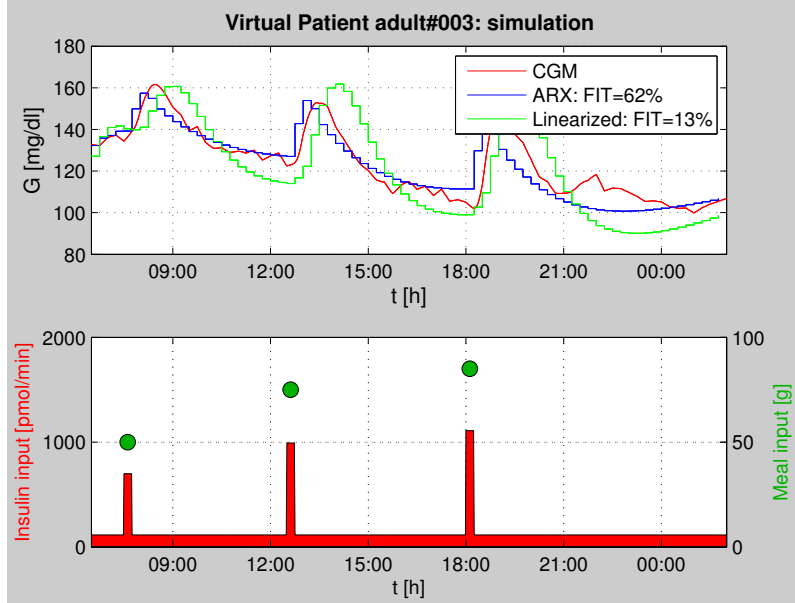


Figura 5.6: Paziente virtuale 003: scenario di test

	1	2	3	4	5
n_a	-	9	21	42	28
n_b	-	12	12	22	54
n_m	-	25	12	23	48
k_u	39	61	-	-	-

Tabella 5.2: Modelli ARX con AIC: frequenza dei vari ordini del modello identificato nei 100 pazienti analizzati

di ipo- e iper- glicemia.

L'identificazione ARX con criterio AIC porta ancora a risultati più che accettabili sull'intera popolazione virtuale con entrambe i protocolli: i dati di sintesi sono mostrati in Tabella 5.3.

5.5.3 ARX su pazienti reali

Tutto il lavoro sui pazienti virtuali ha come scopo capire quali prestazioni è possibile raggiungere sui pazienti reali. I dataset JDRF a nostra disposizione finora non sono pensati esplicitamente per l'identificazione: la prova in open loop (Admit1) è pensata come metro di paragone per valutare il closed loop (Admit2) e non prevede in generale pasti e boli disallineati. In un certo numero di casi, tuttavia, alcuni eventi accidentali (e.g. boli correttivi e hypo-treatment) rendono i dataset sufficientemente eccitati da permettere una discreta identificabilità: su questi casi “fortunati” possiamo utilizzare

Modello	fit in test (media \pm sd)	RMSE in test (media \pm sd)
ARX crossvalidazione	82.4 ± 8.5	4.6 ± 1.6
ARX con AIC	60.9 ± 18.7	10.6 ± 18.7
linearizzato	2.9 ± 53.8	25.5 ± 10.7
linearizzato medio	-235 ± 439.9	92 ± 109

Tabella 5.3: Identificazione ARX su pazienti virtuali

Orizzonte	fit in Admit1	fit in Admit2
1	91	90
2	77	77
4	54	54
8	31	24
∞	68	52

Tabella 5.4: Trial 303: dati di sintesi

le routine di identificazione. Avendo a disposizione comunque pochi dati, non è stato possibile effettuare la crossvalidazione, e per effettuare la scelta dell'ordine del modello si è utilizzato il criterio AIC come nella Sezione 5.5.2. Più precisamente, è stato utilizzato Admit1 come dataset di identificazione e Admit2 come dataset di test.

Si sono generati per ogni paziente due modelli: uno da usare in predizione ed uno in simulazione. I modelli per la predizione sono stati stimati senza prefiltraggio dell'input, settando l'opzione **Focus** a **Prediction** nella funzione di stima PEM, mentre i modelli usati in simulazione sono stati prefiltrati settando l'opzione a **Simulation**, come descritto in precedenza. In entrambi i casi, è stato selezionato il modello migliore secondo il criterio AIC scartando i modelli che non rispettavano i vincoli di selezione. I modelli stimati senza prefiltraggio sono stati utilizzati come predittori a 1, 2, 4 ed 8 passi in avanti. I modelli sono stati inoltre stimanti utilizzando le misurazioni di glucosio plasmatiche invece di quelle dei sensori CGM perchè meno influenzate dal rumore. Data l'incertezza sui valori basali y_b e u_b , essi sono stati stimati come descritto nella Sezione 5.3.3 ed è stata stimata la componente di errore costante γ .

Trial 303 Montpellier

Il trial 303 Admit1 è caratterizzato da una ottima eccitazione dell'input a causa dei boli correttivi relativi alla cena e alla colazione e alla presenza di *hypo treatment* somministrati per contrastare una grave ipoglicemia notturna (Figura 5.7). Le procedure di identificazione ottengono in questo caso particolare risultati piuttosto buoni. In Figura 5.8 è mostrato l'andamento in Admit2, ovvero in closed loop, della glicemia del paziente insieme al simulatore ARX. Dai dati di sintesi di Tabella 5.4 si vede che il modello spiega sufficientemente bene sia i dati di identificazione che i dati di test.

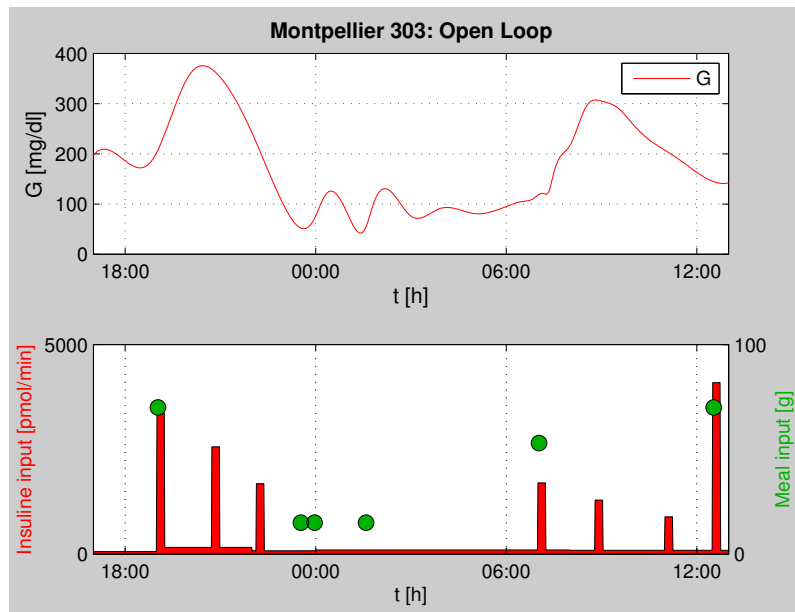


Figura 5.7: Trial 303 Montpellier: Admit1

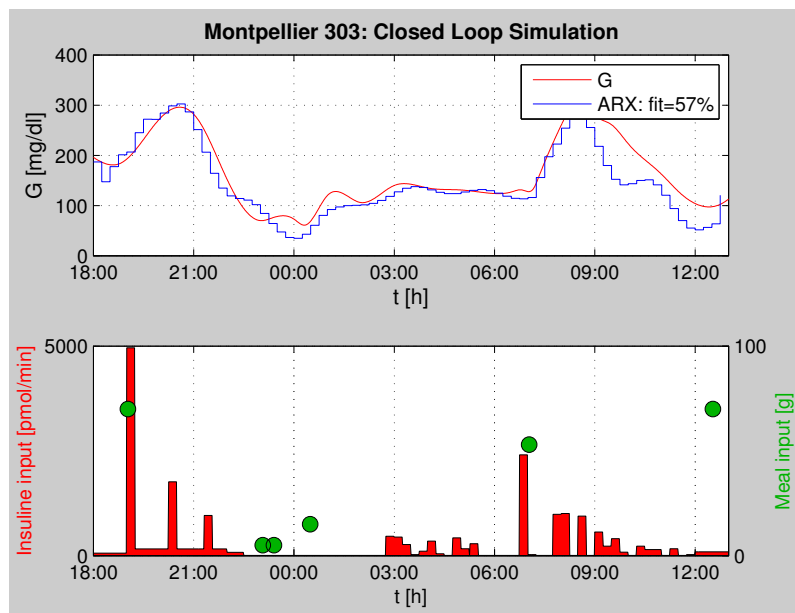


Figura 5.8: Trial 303 Montpellier: ARX in simulazione, Admit2

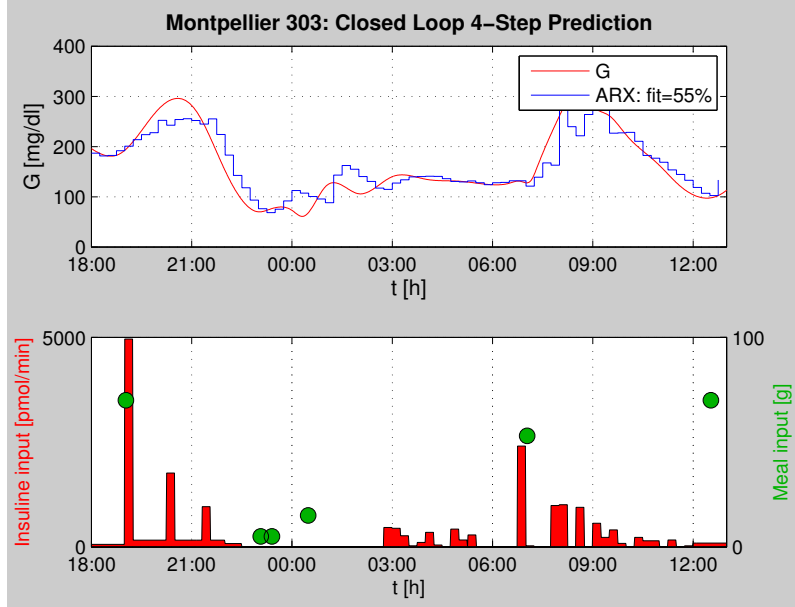


Figura 5.9: Trial 303 Montpellier: ARX in predizione a 4 passi, Admit2

Orizzonte	fit in Admit1	fit in Admit2
1	83	84
2	60	58
4	25	21
8	7	2
∞	11	-10

Tabella 5.5: Trial 306: dati di sintesi

Trial 306 Montpellier

Il trial 306 Montpellier, analogamente al 306, è dotato di una sufficiente eccitazione dell'input in open loop. Al bolo contemporaneo al pranzo seguono infatti due boli correttivi mentre la notte è caratterizzata da un numero considerevole di *hypo treatment* (Figura 5.10).

I dati sintetici di fit per questo trial sono riportati nella Tabella 5.5. Nonostante la buona proprietà dell'input i risultati su questo *trial* sono meno incoraggianti. Si notano risultati meno buoni anche considerando la fit sullo scenario Admit1, su cui gli ARX vengono identificati. Questo può significare che non esiste per questo paziente un modello lineare che spiega accuratamente dati.

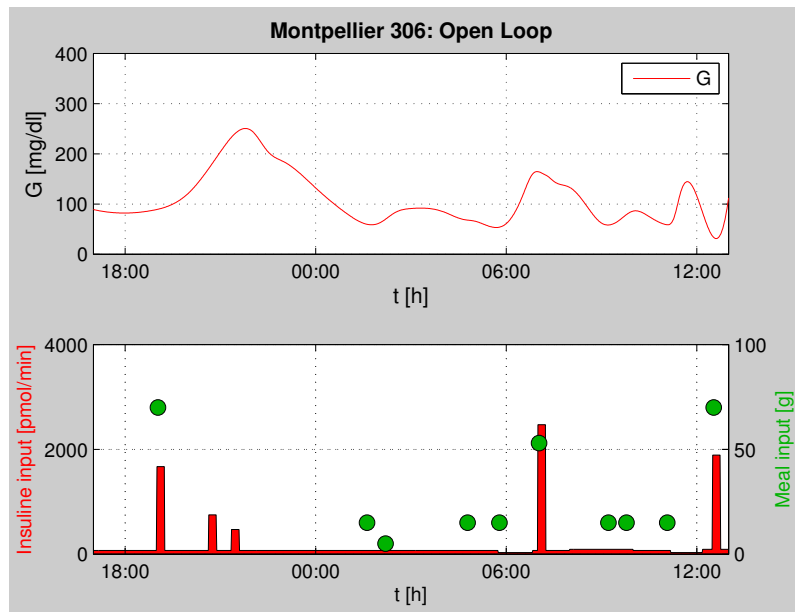


Figura 5.10: Trial 306 Montpellier: Admit1

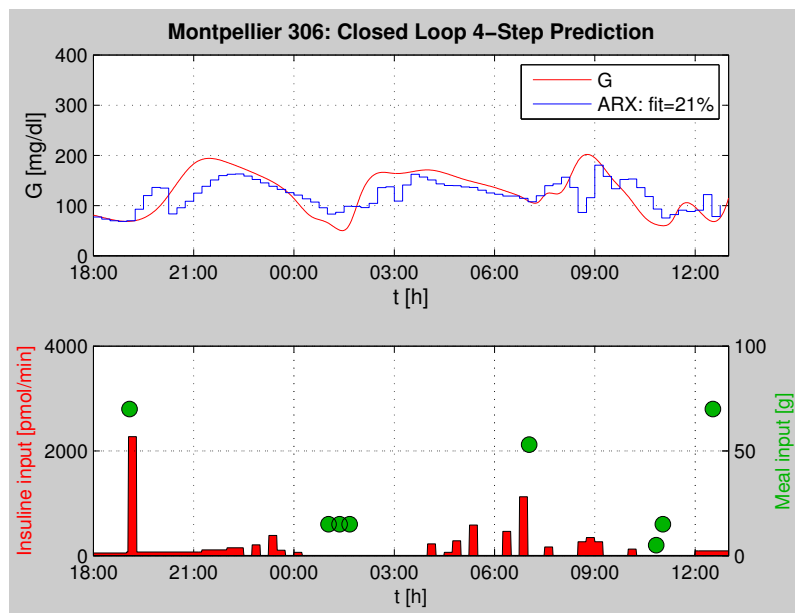


Figura 5.11: Trial 306 Montpellier: ARX in predizione a 4 passi, Admit2

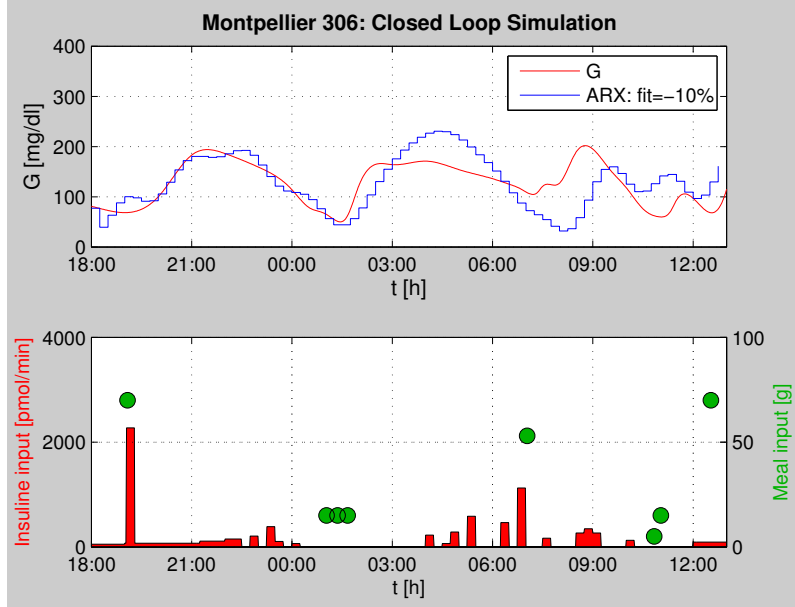


Figura 5.12: Trial 306 Montpellier: ARX in simulazione, Admit2

Trial 301 Montpellier

Il trial 301 è caratterizzato anch'esso da un'ottima eccitazione dell'input in open loop (Figura 5.13). I risultati in termini di fit sono riportati nella Tabella 5.6. Si nota per questo *trial* che i risultati sono insoddisfacenti per quanto riguarda le predizioni sul dataset di validazione, ma sono ottime nel dataset di test. Sembra dunque che la risposta del paziente sia spiegata molto bene da un modello lineare nel primo caso, ma lo stesso modello lineare diventa pessimo se utilizzato nel secondo scenario. Questo suggerisce che il modello stesso del paziente sia tempo-variante. L'ipotesi è corroborata da un'analisi comparativa dei due dataset: in open loop vengono somministrate in tutto 43 U di insulina, si verificano 3 *hypo treatment* e il valor medio di G è 137 (mg/dL). In closed loop vengono somministrate 70U di insulina, non vi è alcun *hypo treatment*, ma il livello medio di G è di 209 (mg/dL).

La sensibilità insulinica sembra decisamente ridotta nella seconda visita. È difficile supporre che un solo modello lineare tempo-invariante possa spiegare in modo soddisfacente due risposte tanto diverse.

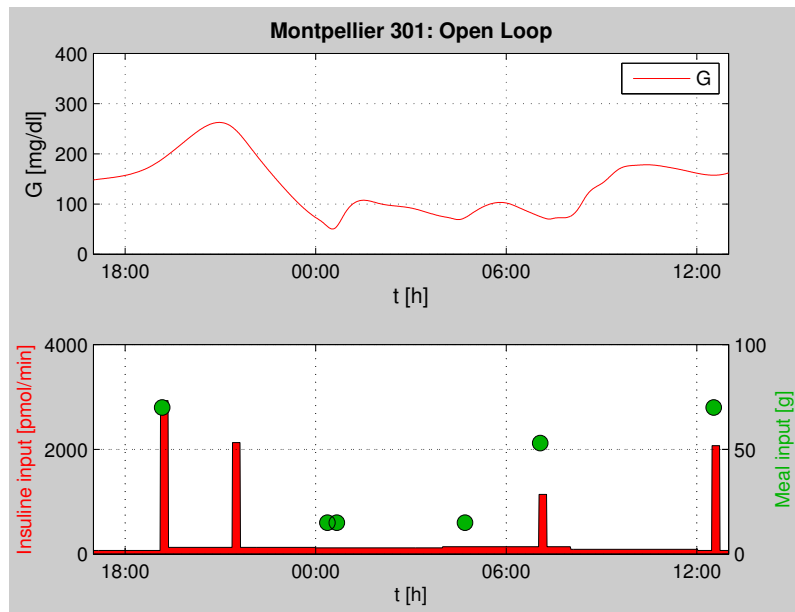


Figura 5.13: trial 301: Admit 1

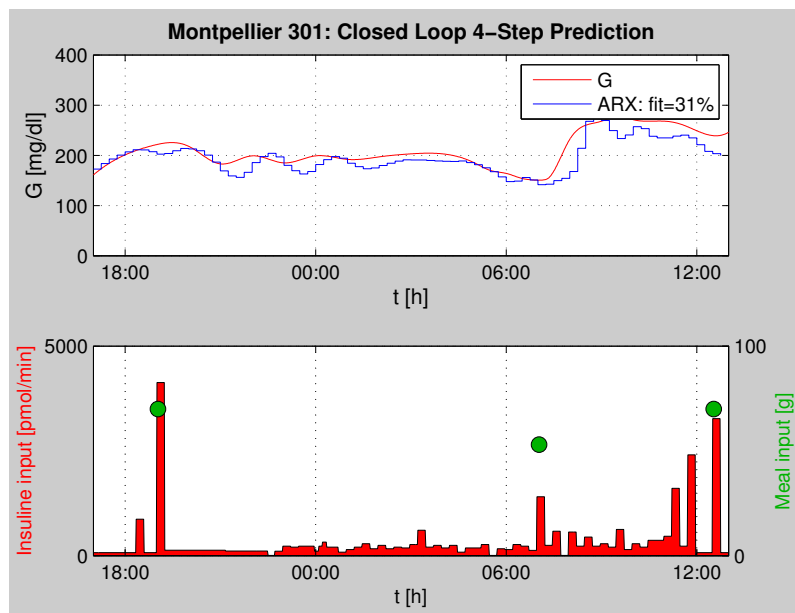


Figura 5.14: Trial 301 Montpellier: ARX in predizione a 4 passi, Admit2

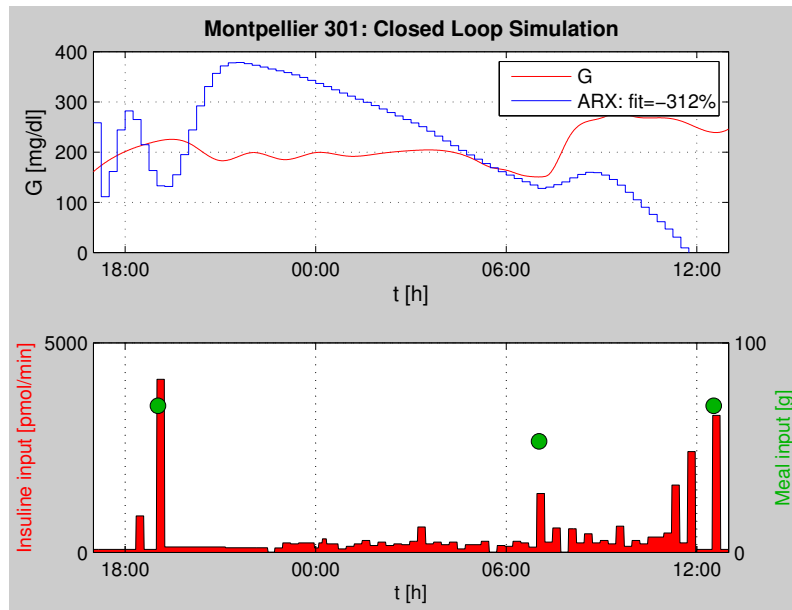


Figura 5.15: Trial 301 Montpellier: ARX in simulazione, Admit2

Orizzonte	fit in Admit1	fit in Admit2
1	94	91
2	84	73
4	65	30
8	37	-48
∞	68	-312

Tabella 5.6: Trial 301: dati di sintesi

5.5.4 Conclusioni

I risultati in termine di fit per i pazienti virtuali sembrano essere decisamente positivi se confrontati con il modello medio ed il linearizzato medio. Nonostante questo, i modelli identificati hanno caratteristiche spesso non desiderabili. Alcuni di essi presentano risposte oscillanti, guadagni e tempi di salita non realistici. A volte questi errori si compensano molto bene sia sullo scenario di identificazione che sullo scenario di test portando a buoni risultati di fit, ma i modelli ottenuti contengono informazioni non corrette sulla dinamica del sistema e, se usati nella sintesi del controllore MPC, possono portare a risultati non accettabili in closed loop.

Si guardi ad esempio la risposta impulsiva del miglior ARX per il paziente virtuale 025 stimato con crossvalidazione: esso rispetta tutti i vincoli di selezione che abbiamo posto, ma non ha alcuna attinenza con il modello fisico sottostante. La risposta è oscillante ed ha una durata di oltre 50 ore, tuttavia il modello ottiene un ottimo risultato di fit = 83.2 e RMSE = 4.0. In altri casi si ottengono modelli più realistici, come nel caso del paziente virtuale 090 (Figura 5.17). Il risultato in termini di fit e RMSE per questo paziente è di 88.2 e 3.9 rispettivamente, solo leggermente migliore al precedente.

Nelle Figure 5.18 e 5.19 sono mostrate le CVGA in simulazione con MPC sintetizzato sul modello medio e sui modelli ARX personalizzati ottenuti con crossvalidazione, in presenza ed assenza di rumore rispettivamente. Le prestazioni ottenute con i modelli ARX sono sensibilmente inferiori rispetto a quelle con il modello medio nel primo caso, sostanzialmente equivalenti nel secondo.

Le tecniche ed i protocolli di identificazione utilizzati, in definitiva, sono da migliorare e ancora non soddisfacenti anche nel caso più semplice dei pazienti virtuali. Sui pazienti reali non abbiamo ancora a disposizione un campione significativo per giungere ad una conclusione: dallo studio preliminare sono emerse difficoltà anche maggiori a causa del comportamento tempo-variante del paziente. In ogni caso, sono necessari protocolli clinici pensati specificatamente per l'identificazione: senza di essi, non sembra possibile ottenere risultati significativi.

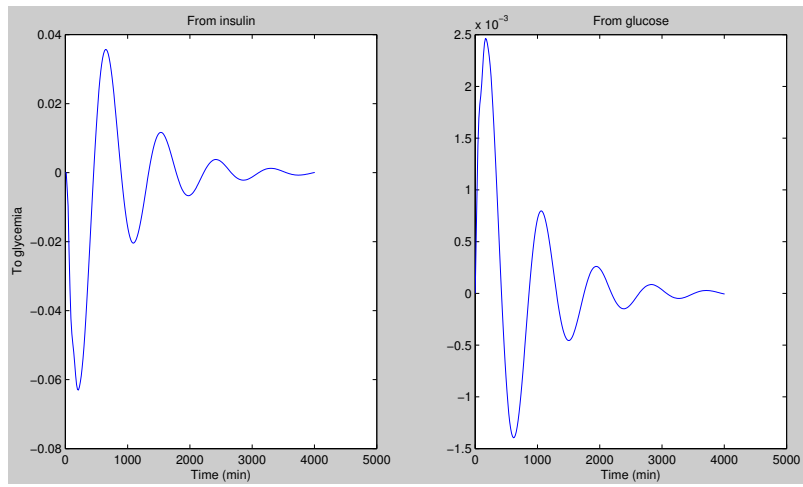


Figura 5.16: Risposta impulsiva del modello ARX 025 ottenuto con crossvalidazione

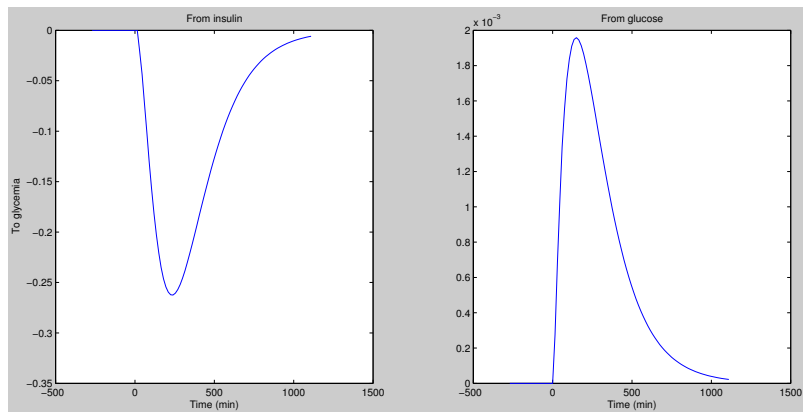


Figura 5.17: Risposta impulsiva del modello ARX 090 ottenuto con crossvalidazione

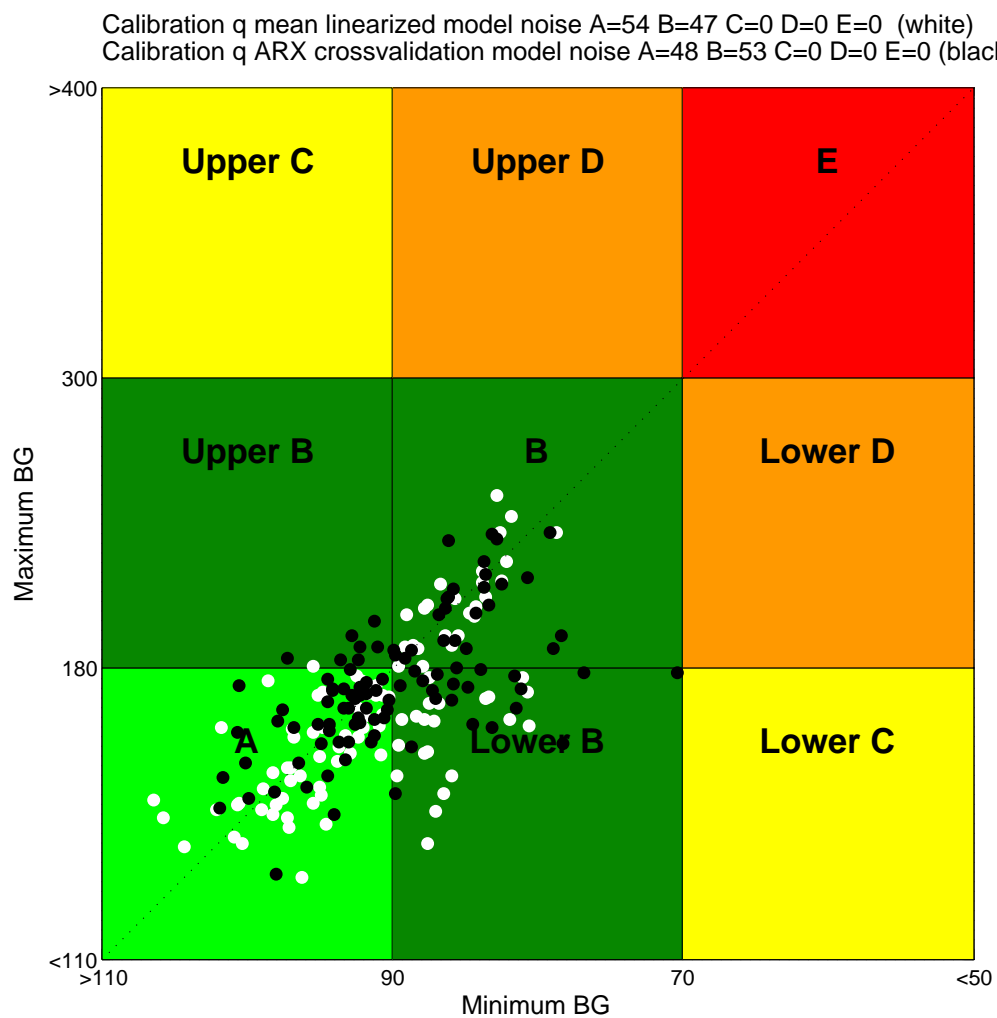


Figura 5.18: Confronto tra le prestazioni ottenute con MPC basato sul modello medio e sui modelli ARX con crossvalidazione con rumore

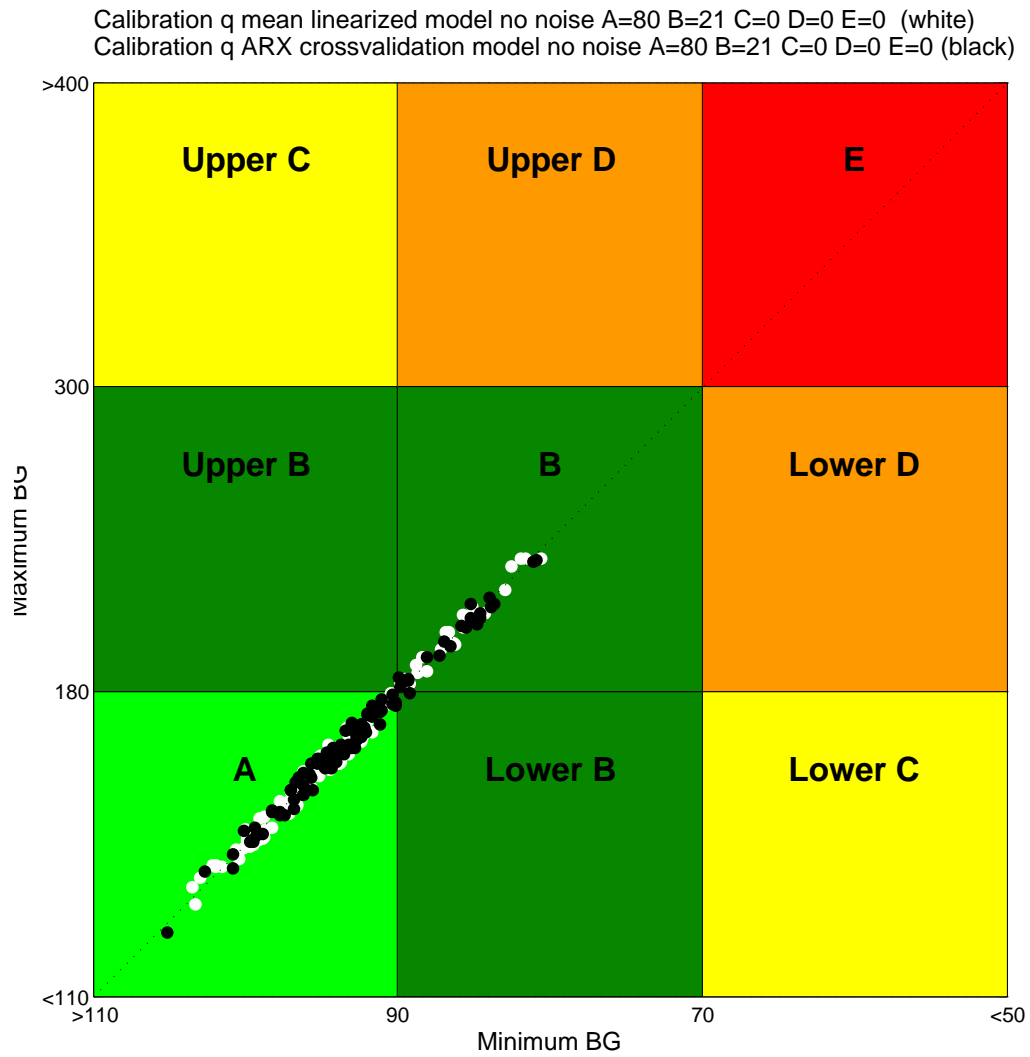


Figura 5.19: Confronto tra le prestazioni ottenute con MPC basato sul modello medio e sui modelli ARX con crossvalidazione in assenza di rumore

Multicenter Trials: studio di fattibilità

6.1 Il protocollo Multicenter

È in fase di studio un protocollo per i futuri *trials* su pazienti reali. Frutto di una razionalizzazione del protocollo precedenti, esso si presterà meglio ad essere eseguito nei diversi centri che parteciperanno al progetto in condizioni standard e su una popolazione di pazienti dalle caratteristiche ben definite. L'obiettivo di questi nuovi test è ottenere una quantità di dati statisticamente rilevante per valutare la possibilità di introdurre in tempi brevi un pancreas artificiale a livello commerciale.

6.1.1 Popolazione di studio e criteri di eleggibilità

I pazienti dovranno indossare sensori CGM da almeno una settimana prima di una *visita di screening* in ospedale. In base ai dati raccolti in questa settimana e ad ulteriori analisi svolte durante la *visita di screening* si valuterà la possibilità di ammettere il paziente nello studio.

È prevista una popolazione di studio di circa 108 soggetti, che saranno selezionati e divisi in due classi A e B a seconda delle loro caratteristiche. I criteri per partecipare allo studio saranno:

- Età tra 12 e 65 anni
- T1D1 diagnosticato da almeno 1 anno e utilizzo di pompa da almeno 6 mesi
- Studio A: $HbA1c \geq 7.0\%$ con meno del 60 % delle misurazioni tra 71 e 180 (mg/dL) nei 7 giorni precedenti
- Studio B: $HbA1c < 7.0\%$ con almeno un evento di ipoglicemia nei 7 giorni precedenti

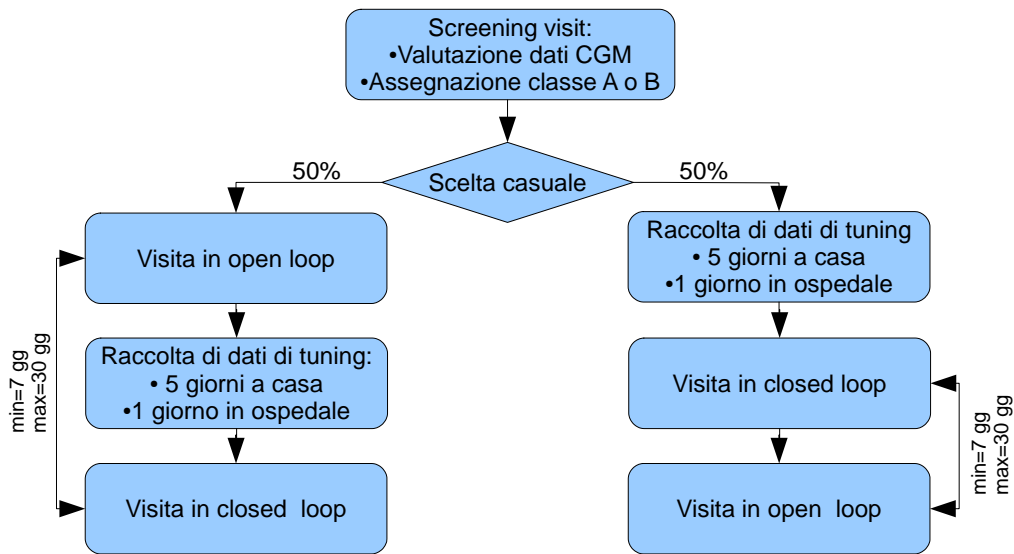


Figura 6.1: Protocollo Multicenter

6.1.2 Le fasi del protocollo

Il protocollo Multicenter prevede due fasi di studio: una in open loop ed una in closed loop. L'ordine in cui queste fasi verranno eseguite sarà scelto in modo casuale con eguale probabilità per ogni paziente in modo da non introdurre vizi statistici.

Fase in open loop È composta da una visita di durata 24 ore, dalle ore 10:00 di un giorno alle ore 10:00 del successivo. Il paziente riceverà tre pasti: un pranzo e una cena alle ore 12:00 e 19:00 del primo giorno, una colazione alle ore 8:00 del secondo giorno. La terapia insulinica sarà composta come al solito da un basale costante e di boli in corrispondenza dei pasti. Il paziente utilizzerà in questa visita la sua pompa con gli stessi parametri che utilizzerebbe normalmente a casa, il personale sanitario prenderà semplicemente nota di tali dati. In casi particolari (e.g. ipoglicemie) i medici potranno comunque intervenire per la sicurezza del paziente. Oltre alle misurazioni continue di glucosio sottocutaneo tramite sensori CGM, verranno effettuate misure plasmatiche ogni 20 minuti con un apparecchiatura YSI o equivalente.

Per i pazienti appartenenti alla classe B sarà prevista, in alcuni casi, un'attività fisica di 60 minuti alle ore 17:00. In questa fase verranno monitorati il battito cardiaco del paziente e la potenza sviluppata.

Fase in closed loop La fase di studio in closed loop è divisa in due parti:

- raccolta dati di *tuning*

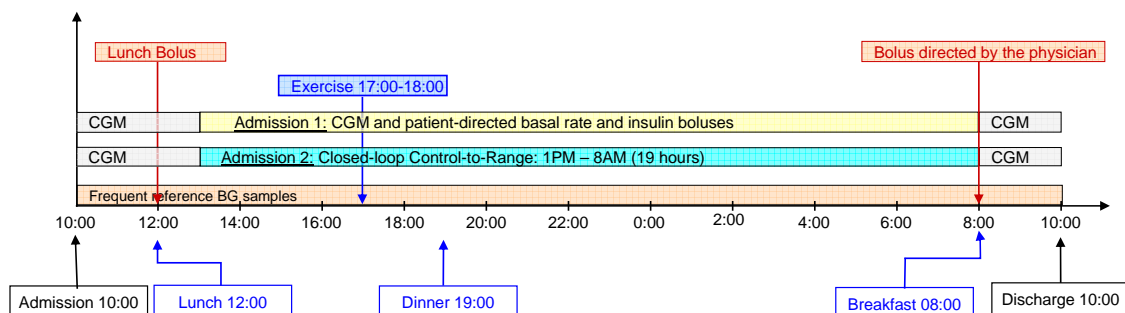


Figura 6.2: Protocollo Multicenter: open e closed loop

- visita in closed loop

La visita in closed loop vera e propria sarà preceduta da una raccolta di dati clinici della durata di ben 6 giorni (Figura 6.3). Le informazioni raccolte saranno utilizzate per effettuare il *tuning* del regolatore. Il protocollo utilizzato in questa parte del test è pensato per rendere possibile l'identificazione di modelli personalizzati, oltre che la regolazione del parametro q del regolatore.

I primi 5 giorni il paziente sarà a casa propria e seguirà una vita normale, ad eccezione del fatto che gli sarà chiesto di modificare la terapia insulinica normale posticipando di due ore alcuni boli di insulina allo scopo di ottenere un dataset con buone proprietà di eccitazione. Il paziente terrà nota inoltre degli orari dei pasti con una stima delle quantità, senza le quali i dati raccolti sarebbero di poca utilità.

L'ultimo giorno il paziente sarà ammesso in clinica e sottoposto a un'ulteriore test. Alle ore 9:30 gli sarà somministrato un bolo di insulina a digiuno in modo da indurre ipoglicemia, che sarà trattata solo quando le misure di glucosio plasmatico raggiungeranno i 60 (mg/dL). A seguire ci sarà un pasto alle ore 13:00 e il relativo bolo di insulina due ore dopo, seguito da un ulteriore *snack*. Il regime seguito l'ultimo giorno conterrà presumibilmente le informazioni più preziose per la modellizzazione, tuttavia è abbastanza rischioso da non poter essere effettuato a casa dal paziente in modo autonomo.

La visita in closed loop vera e propria avverrà in clinica e dovrà ricalcare negli orari dei pasti e nelle dosi il protocollo dell'open loop. Gli orari di ammissione e dei pasti rimarranno invariati, così come l'attività fisica, nel caso fosse presente in open loop. Dalle ore 13:00 del primo giorno alle 08:00 del successivo sarà però attivato il controllo in closed loop (Figure 6.2).

Il software di controllo, chiamato APS, sarà progettato secondo l'architettura modulare presentata nel capitolo 4 e gli algoritmi utilizzati, compresa la procedura di tuning del regolatore. L'hardware stesso che verrà utilizzato sarà sempre lo stesso: la pompa Omnipod Insulin Management SystemTM (Insulet Corp., Bedford, MA) con l'utilizzo di insulina ad azione rapida Lispro (Humalog) ed il sensore CGM FreeStyle NavigatorTM.

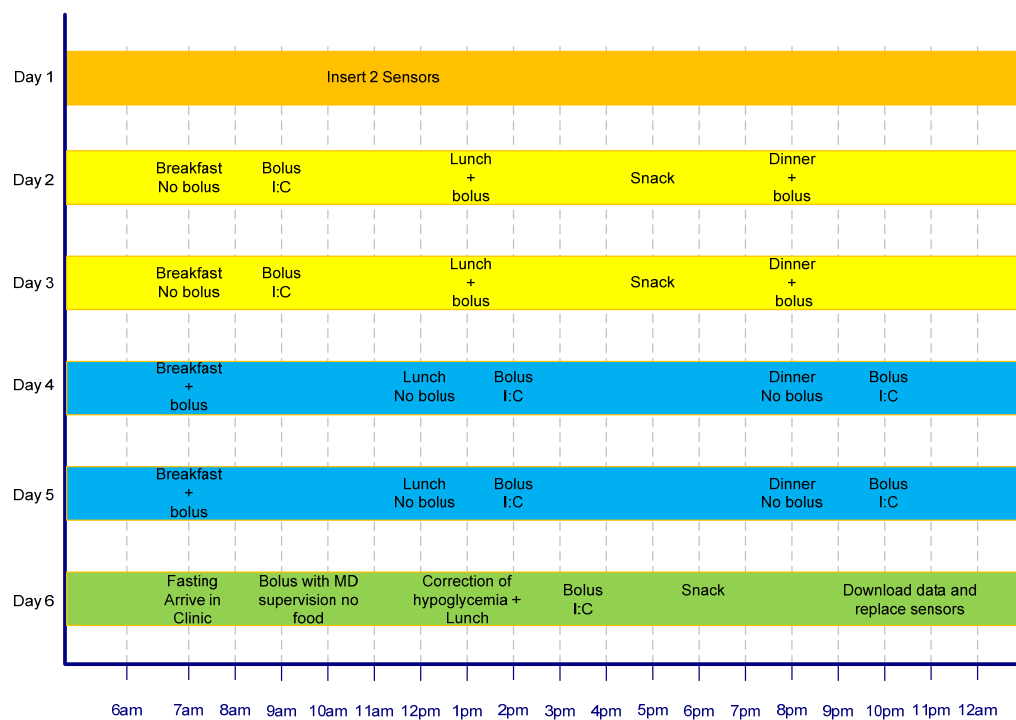


Figura 6.3: Protocollo Multicenter: raccolta dati

La differenza fondamentale rispetto ai *trials* JDRF è che il tuning del regolatore non avviene sui dati clinici in open loop, ma attraverso un protocollo specifico per la raccolta di dati. La visita in open loop nei *trials* Multicenter servirà esclusivamente come metro di paragone con cui valutare la visita in closed loop e dovrà dunque seguire le condizioni nominali, con boli di insulina contemporanei ai pasti. Il regime seguito nella fase di raccolta dati introdurrà invece modifiche alla terapia standard volte ad generare una sufficiente eccitazione dell'input, ma necessariamente queste modifiche peggioreranno il controllo della glicemia. Non sarebbe dunque corretto pensare ad un solo protocollo perché gli obiettivi da raggiungere sono in netto contrasto.

6.2 Identificazione di modelli ARX

Il protocollo di raccolta dati è molto migliore di quelli considerati in precedenza. Valuteremo in questa sezione i miglioramenti che esso porta all'identificazione di modelli di pazienti *in silico*.

La parte più preziosa dei dati raccolti è quella relativa all'ultimo giorno in cui, diversamente dai casi precedenti, si somministra un bolo di insulina senza alcun pasto in concomitanza. Data l'abbondanza di dati disponibili possiamo utilizzare la tecnica della crossvalidazione senza problemi. Utilizzeremo il giorno 6 per generare un dataset di identificazione, il giorno 5 per generare un dataset di validazione e giorno 2 per generare un dataset di test. Se si avesse a che fare con dati clinici le scelte potrebbero essere forzate dalla presenza in altri giorni di anomalie quali malfunzionamenti dei sensori o pasti non registrati correttamente dal paziente. Fissiamo in tutti i dataset le quantità di 45 g per le colazioni, 60 g per i pranzi, 80 g per le cene e 15 g per gli snack. Il bolo di insulina somministrato il sesto giorno a digiuno è stato personalizzato per ogni paziente ed ottenuto come $U = 20 \cdot CR$, ovvero la quantità di insulina ottimale per un pasto di 20 g di glucosio. Questa scelta empirica ha permesso di ottenere una significativa variazione dell'uscita G_M , senza comunque mettere a rischio la vita del paziente. L'algoritmo utilizzato è dunque lo stesso della Sezione 5.5.2.

In Figura 6.2 è mostrato lo scenario di identificazione Multicenter sul paziente virtuale 007 mentre in Figura 6.2 è mostrato lo scenario di test con il miglior modello ARX utilizzato come simulatore. In Tabella 6.2 è riportato il risultato in termini di fit e RMSE in simulazione per il modello ARX stimato con il protocollo multicenter ed i modelli ARX già stimati nel Capitolo 5 sullo scenario di test del protocollo multicenter. I modelli stimati con il nuovo protocollo ottengono risultati migliori rispetto ai precedenti, inoltre dall'analisi delle risposte impulsive si nota che essi hanno caratteristiche più realistiche. In Figura 6.2 è mostrata la risposta impulsiva del modello ARX per il paziente virtuale 025, che con i protocolli precedenti aveva dato luogo a modelli oscillanti.

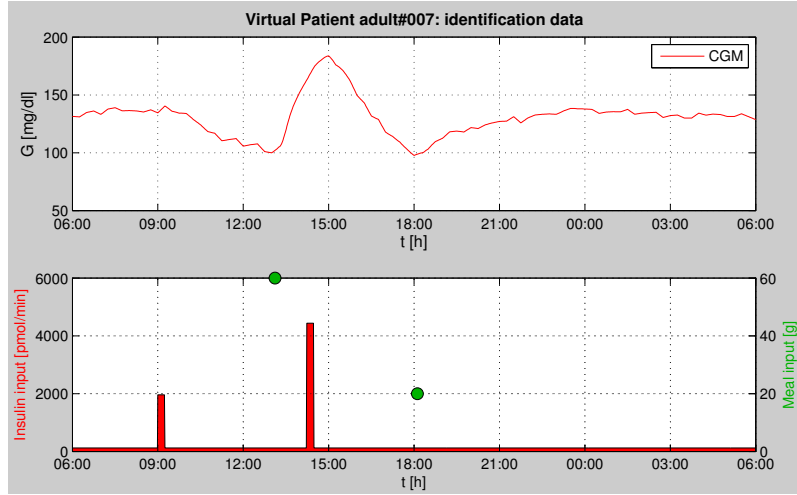


Figura 6.4: Paziente virtuale 007: scenario di identificazione multicenter

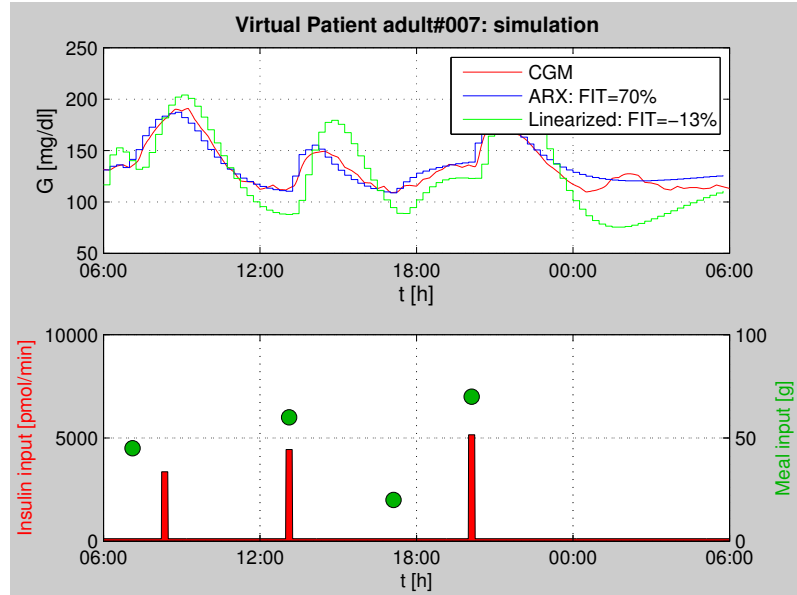


Figura 6.5: Paziente virtuale 007: scenario di test multicenter

	1	2	3	4	5
n_a	-	11	20	21	48
n_b	-	57	13	8	22
n_m	-	6	5	12	77
k_u	75	25	-	-	-

Tabella 6.1: Modelli ARX Multicenter: frequenza dei vari ordini del modello identificato nei 100 pazienti analizzati

Modello	fit in test (media \pm sd)	RMSE in test (media \pm sd)
ARX multicenter	65.2 ± 34.98	9.6 ± 8.07
ARX crossvalidazione	57.6 ± 25.7	11.9 ± 6.0
ARX AIC	55.4 ± 21.1	12.6 ± 5.8
linearizzato	31.4 ± 27.7	19.4 ± 7.8
linearizzato medio	-162.0 ± 282.6	74.2 ± 83.2

Tabella 6.2: Modelli ARX sul protocollo di test multicenter: sintesi

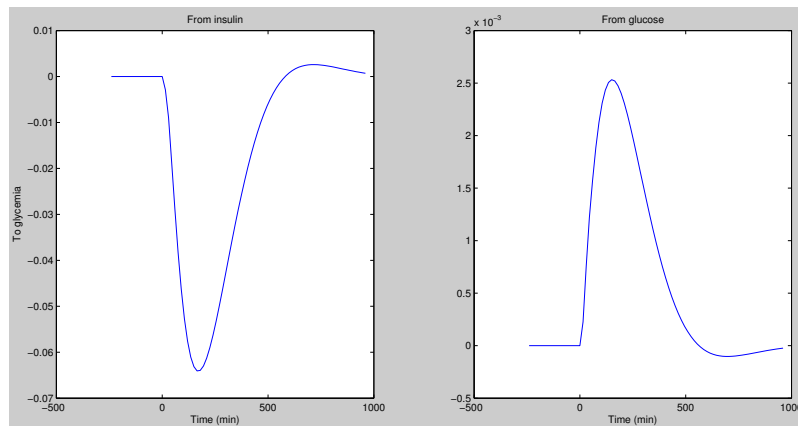


Figura 6.6: Risposta impulsiva del modello ARX 025 stimato con il protocollo multicenter

6.3 Controllo MPC con modelli ARX

Lo scopo dell'identificazione di modelli ARX è sintetizzare un controllore MPC basato sul modello personalizzato del paziente, piuttosto che sul modello medio linearizzato. La trattazione del Capitolo 3 è generale ed entrambe le scelte sono possibili nel software di simulazione.

La CVGA permette di valutare in maniera rapida le prestazioni delle differenti soluzioni in maniera intuitiva sull'intera popolazione di pazienti virtuali.

In Figura 6.7 è mostrata la calibrazione della q per la popolazione di pazienti virtuali con controllore sintetizzato sul modello medio linearizzato, sul modello linearizzato sui parametri del paziente e sul modello ARX identificato con il protocollo Multicenter. Come si può vedere dal conteggio delle simulazioni nelle varie zone, le prestazioni del modello medio e del linearizzato sono approssimativamente equivalenti, quelle dei modelli ARX Multicenter sono leggermente inferiori.

In Figura 6.8 sono riportati i risultati in simulazione con gli stessi modelli del caso precedente. Vengono utilizzate le q ottenute dalla calibrazione senza rumore e viene inserito il rumore al segnale CGM ricevuto dal controllore.

In questo caso i modelli migliori sono il modello medio linearizzato e l'ARX Multicenter, mentre quello linearizzato sui parametri del paziente ottiene il risultato peggiore, con una forte tendenza all'ipoglicemia. In nessuna circostanza, tuttavia, i modelli ARX multicenter ottengono prestazioni significativamente superiori rispetto agli altri modelli, e lo sforzo impiegato per l'identificazione pare non giustificato. Il miglioramento ottenuto con il nuovo protocollo di identificazione è comunque verificabile confrontando le simulazioni ottenute dagli ARX con crossvalidazione della Sezione 5.5.2 e gli ARX Multicenter (Figura 6.9).

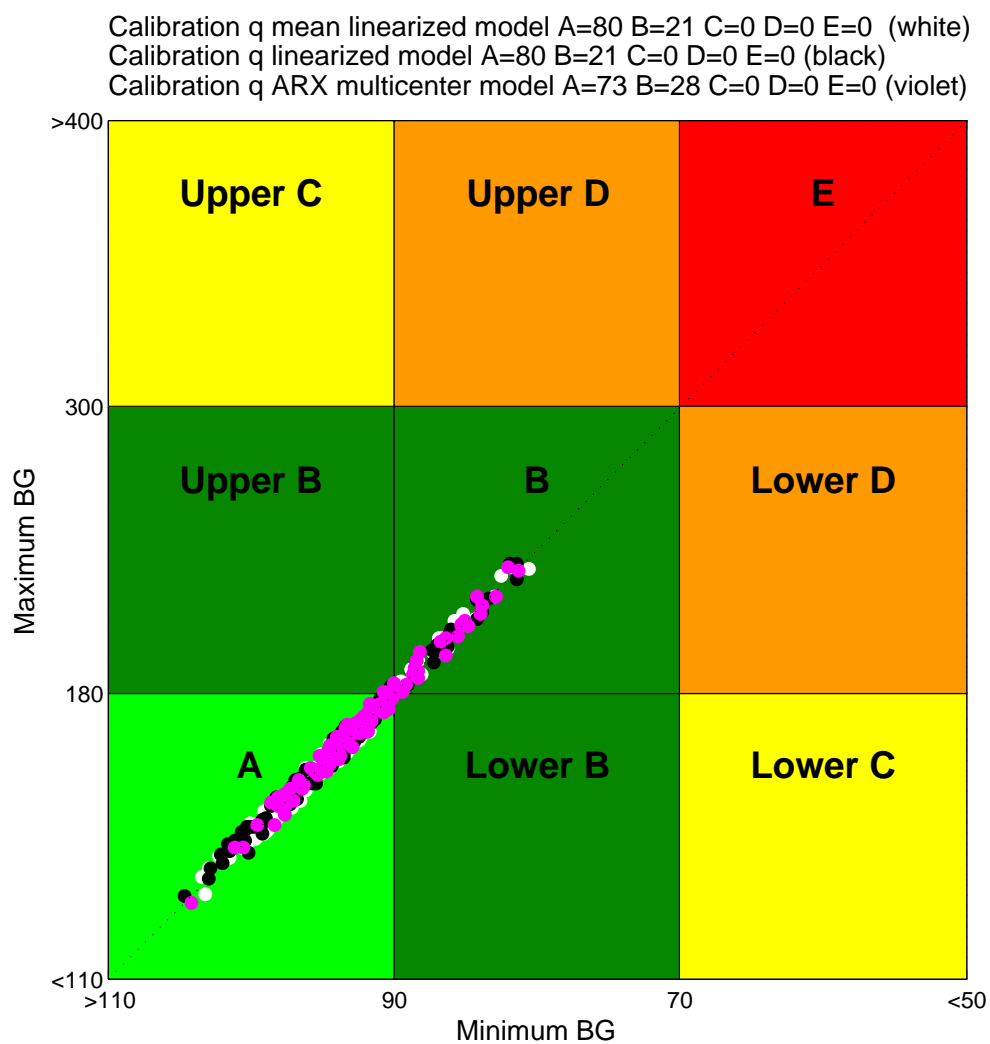


Figura 6.7: Calibrazione con modello linearizzato medio, linearizzato sul paziente, ARX multicenter

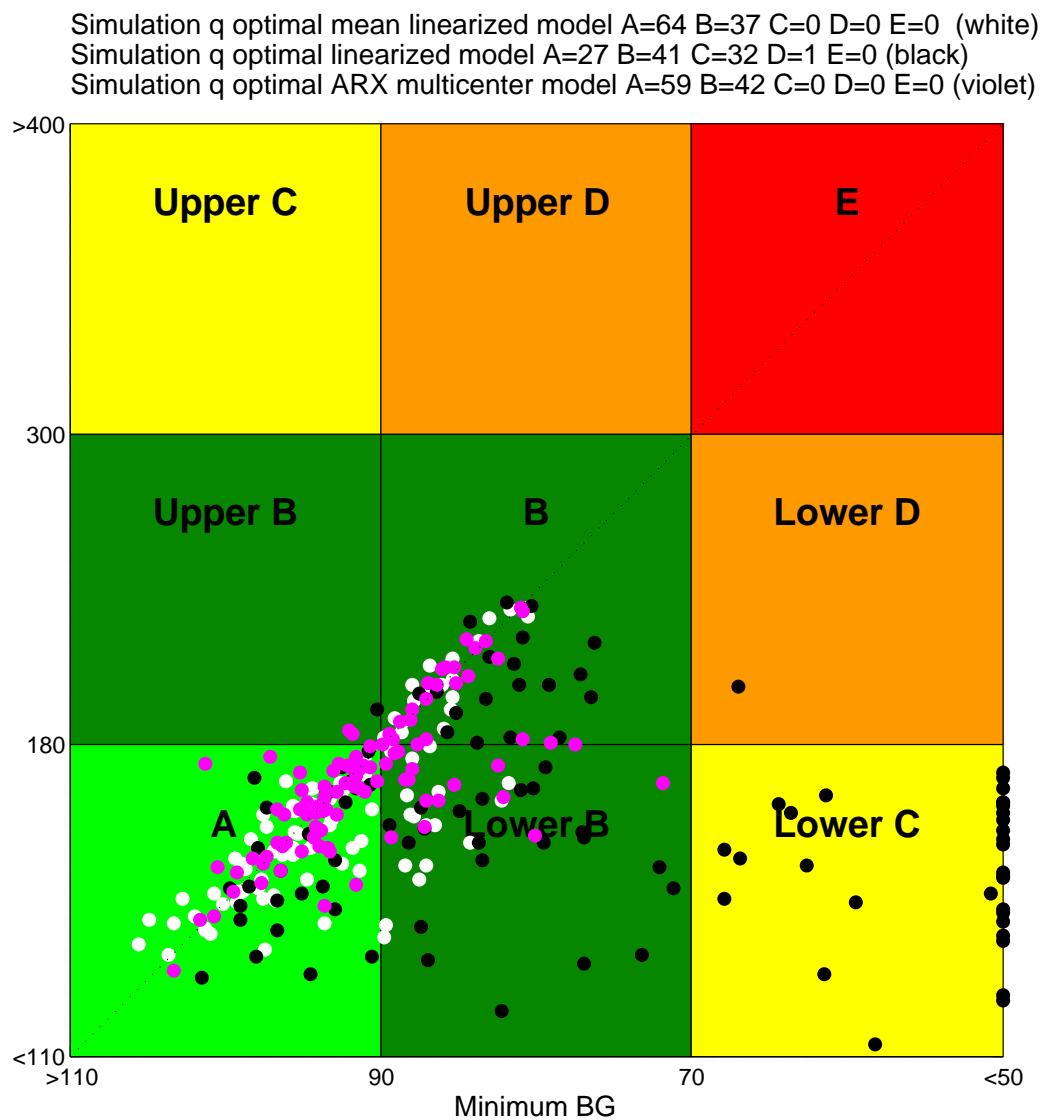


Figura 6.8: Simulazione con modello medio, linearizzato sul paziente, ARX multicenter

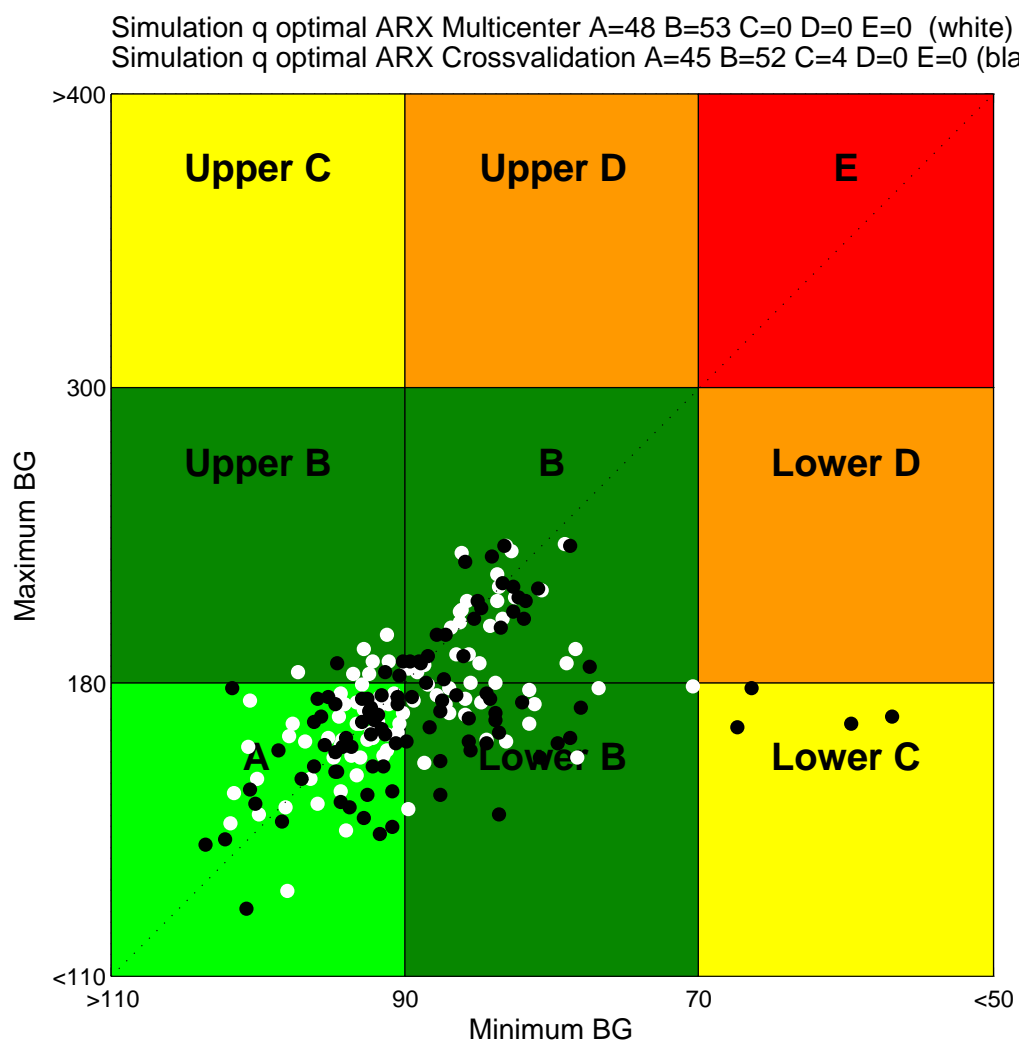


Figura 6.9: Simulazione con modelli ARX Multicenter ed ARX con protocollo vecchio (Crossvalidazione)

Conclusioni

In questa tesi sono state valutate le classiche tecniche di identificazione a scatola nera applicate al metabolismo glucosio-insulina. Questa strada è stata esplorata data la grande variabilità osservata nelle popolazioni di pazienti virtuali e reali. I modelli ARX identificati si sono dimostrati molto migliori del modello medio quando usati off-line in predizione ed in simulazione, anche se negli esperimenti necessari per la loro identificazione si deve affrontare un delicato compromesso tra eccitazione dell'input e sicurezza del paziente. Ci si aspettava che la sintesi di controllori MPC basata sui modelli personalizzati avrebbe portato a miglioramenti significativi anche nelle prestazioni di controllo in anello chiuso. I risultati sperimentali sui pazienti virtuali non hanno tuttavia portato ai risultati attesi; in alcuni casi si è notato anzi un degrado delle prestazioni del controllo.

Le tecniche di adattamento della q già sviluppate sembrano quindi sufficienti e lo sforzo di identificazione pare ingiustificato. Sui *trials* reali, anche nei pochi casi in cui l'eccitazione dell'input era buona, l'identificazione è ulteriormente complicata dal comportamento tempo-variante del paziente, non modellizzato nel simulatore.

Questi risultati suggeriscono che lo stato della ricerca sui modelli individualizzati non sia sufficientemente maturo per il loro utilizzo sulla sintesi di controllori da usare negli studi clinici su pazienti reali, dato che l'individualizzazione del controllo basata sul modello medio fornisce risultati migliori.

Bibliografia

- [1] A. Albisser, B. Leibel, T. Ewart, Z. Davidovac, and W. Zingg. An artificial endocrine pancreas. *Diabetes*, 23:389–404, 1974.
- [2] C. Dalla Man, M. Camilleri, and C. Cobelli. A system model of oral glucose absorption: Validation on gold standard data. *IEEE Transactions on Biomedical Engineering*, 53(12):2472–2478, December 2006.
- [3] C. Dalla Man, R. A. Rizza, and C. Cobelli. Meal simulation model of the glucose-insulin system. *IEEE Transactions on Biomedical Engineering*, 54(10), October 2007.
- [4] D. Finan, F. J. Doyle III, C. Palerm, W. Bevier, H. Zisser, L. Jovanovic, and D. Seborg. Experimental evaluation of recursive model identification techniques for type 1 diabetes. *Journal of diabetes science and technology*, 3, September 2009.
- [5] D. A. Finan, C. C. Palerm, F. J. Doyle III, D. E. Seborg, H. Zisser, W. C. Bevier, and L. Jovanovic. Effect of input excitation on the quality of empirical dynamic models for type 1 diabetes. *AiChE Journal*, 2009.
- [6] W. F. Ganong. *Review of Medical Physiology*. Pretince-Hall International Inc., 22nd edition, 2005.
- [7] C. Homko, A. Deluzio, C. Jimenez, J. Kolaczynski, and G. Boden. Comparison of insulin aspart and lispro. *Diabetes Care*, pages 2027–2031, 2003.
- [8] N. Johnson. *Continuous Univariate Distributions*. John Wiley and Sons, 1995.
- [9] B. Kovatchev, W. Clarke, M. Breton, K. Brayman, and A. McCall. Quantifying temporal glucose variability in diabetes via continous monitoring: mathematica methods and clinical application. *Diabetes Technology & Therapeutics* 7, 6:849–862, 2005.

- [10] B. Kovatchev, S. Patek, E. Dassau, F. J. Doyle III, L. Magni, G. De Nicolao, and C. Cobelli. Control-to-range for diabetes: functionality and modular architecture. *Journal of diabetes science and technology*, 3, September 2009.
- [11] L. Laffel and A. Kowalski. State of the art: Continuous glucose monitoring. *American Academy of Pediatric - section on Endocrinology Newsletter*, 16, 2007.
- [12] R. Leslie, N. Lazarus, and D. Vergani. Aetiology of insulin-dependent diabetes. *British Medical Bulletin*, 45(1), 1989.
- [13] L. Ljung. *System identification toolbok user's guide*. Natick: The MathWorks, Inc, 1998-2004.
- [14] L. Ljung. *System Identification: Theory for the User*. Pretince Hall PTR, 1999.
- [15] L. Magni, M. Forgione, C. Toffanin, C. Dalla Man, B. Kovatchev, G. De Nicolao, and C. Cobelli. Run-to-run tuning of model predictive control for type 1 diabetes subjects: In silico trial. *Journal of diabetes science and technology*, 3, September 2009.
- [16] E. Renard. Implantable closed-loop glucose-sensing and insulin delivery: the future for insulin pump therapy. *Curr Opin Pharmacol*, 2:708–16, 2002.
- [17] J. Santiago, H. Clemens, W. Clarke, and D. Kipnis. Closed-loop and open-loop devices for blood glucose control in normal and diabetic subject. *Diabetes*, 28:71–84, 1979.
- [18] H. Zisser, L. Robinson, W. Bevier, E. Dassau, C. Ellingsen, F. J. Doyle III, and L. Jovanovic. Bolus calculator: A review of four "smart" insulin pumps. *Diabetes Technology and Therapeutics*, 10(6), 2008.