IFAC Journal of Systems and Control 36 (2026) 100406

Contents lists available at ScienceDirect

IFAC Journal of Systems and Control

journal homepage: www.elsevier.com/locate/ifacsc

Full length article

Knowledge-guided recurrent neural networks for glucose–insulin dynamics modeling☆,☆☆

Stefano De Carli *, Nicola Licini, Davide Previtali, Fabio Previdi, Antonio Ferramosca

Department of Management, Information and Production Engineering, University of Bergamo, 24044 Dalmine, Bergamo, Italy

ARTICLE INFO

Article history:
Received 17 October 2025
Received in revised form 9 February 2026
Accepted 3 March 2026
Available online 16 March 2026

Keywords:
Biomedical system modeling
Identification
And simulation
Artificial pancreas
Machine and deep learning for system identification
Physics informed and grey box model identification
Nonlinear system identification

ABSTRACT

Mathematical models of glucose–insulin dynamics are essential for managing type 1 diabetes. Their applications extend to closed-loop control, forecasting glucose trajectories, anticipating and detecting hypo- and hyperglycemia, and supporting real-time decision-making. In this work, we introduce the Compartmental Recurrent Neural Network (COMP-RNN) model, which advances the Biologically-Informed Recurrent Neural Network (BI-RNN) framework for glucose–insulin dynamics modeling. The COMP-RNN extends the data-driven strengths of the BI-RNN by embedding physiological structure directly into the model architecture. Specifically, it leverages structured RNNs aligned with canonical physiological compartments and incorporates prior physiological knowledge into training through an augmented cost function. The COMP-RNN is trained and validated on in silico cohorts. Compared to both BI-RNN and a benchmark linear model, the proposed approach achieves higher predictive accuracy and improved parameter efficiency, while better reflecting the underlying physiological principles.

1. Introduction

Type 1 Diabetes Mellitus (T1DM) is a chronic autoimmune disorder marked by the destruction of pancreatic β-cells, leading to absolute insulin deficiency and persistent hyperglycemia. Maintaining Blood Glucose (BG) within safe ranges is crucial yet challenging, as glucose regulation is influenced by diverse and rapidly changing factors such as insulin sensitivity, physical activity, nutrient absorption, and psychological stress (Katsarou et al., 2017). Mathematical modeling has long played a central role in understanding these dynamics and in supporting technologies designed to automate insulin delivery. However, capturing the nonlinear and time-varying nature of glucose–insulin interactions remains a major challenge. Variability arising from uncertain meal dynamics, fluctuations in insulin sensitivity, and circadian effects often limits the fidelity of existing models. Simplified approaches, such as compartmental models (Anderson, 2013), offer analytical tractability but struggle to represent the complexity and interdependence of biological processes, particularly under the high variability in real-life conditions.

To overcome these limitations, Machine Learning (ML) has increasingly been explored for glucose–insulin dynamics modeling. In particular, gated Recurrent Neural Networks (RNNs) such as Long Short-Term Memory (LSTM) (Hochreiter & Schmidhuber, 1997) and Gated Recurrent Unit (GRU) networks (Cho et al., 2014) have demonstrated remarkable capabilities in learning complex nonlinear systems by capturing intricate temporal dependencies. RNNs have therefore become established tools in the field of system identification (Ljung et al., 2020).

Despite the remarkable accuracies of RNNs in modeling dynamical systems, purely data-driven methods face critical barriers in clinical use (Jacobs et al., 2024): their black-box nature prevents physiological interpretability and limits robustness and generalizability, which is essential for medical-grade applications such as BG prediction, hypo-/hyperglycemia forecasting, and decision support. These limitations motivate the development of hybrid approaches that integrate physiological knowledge with the expressive power of recurrent neural networks (De Carli, Licini et al., 2025). In this context, biologically-informed frameworks have emerged as a promising direction, aiming to balance accuracy, interpretability, and reliability in glucose–insulin dynamics modeling.

☆ This article is part of a Special issue entitled: 'IFAC WC 2026 - TC 8.2' published in IFAC Journal of Systems and Control.

☆☆ This work was funded by the National Plan for NRRP Complementary Investments (PNC, established with the decree-law 6 May 2021, n. 59, converted by law n. 101 of 2021) in the call for the funding of research initiatives for technologies and innovative trajectories in the health and care sectors (Directorial Decree n. 931 of 06-06-2022) - project n. PNC0000003 - AdvaNced Technologies for Human-centrEd Medicine (ANTHEM).

* Corresponding author.
E-mail addresses: stefano.decarli@unibg.it (S. De Carli), nicola.licini@unibg.it (N. Licini), davide.previtali@unibg.it (D. Previtali), fabio.previdi@unibg.it (F. Previdi), antonio.ferramosca@unibg.it (A. Ferramosca).

https://doi.org/10.1016/j.ifacsc.2026.100406

Knowledge-guided machine learning aims to bridge this gap by embedding domain expertise directly into ML approaches (Karpatne et al., 2022). This is achieved through two main paradigms: (i) Knowledge-Guided Learning (KGL), which soft-incorporates knowledge during model training by augmenting loss functions, as in the well-known case of physics-informed neural networks (Raissi et al., 2019); (ii) Knowledge-Guided Architecture (KGA), where knowledge is hard-coded directly into the model structure. An example of KGA is demonstrated by De Giuli et al. (2024), where a ML model accounts for the inner topology of a district heating system. In a work by De Carli, Licini et al. (2025), the focus was on the KGL approach for BG prediction through gated RNNs, resulting in the Biological-Informed RNN (BI-RNN) framework. However, the complementary KGA approach, i.e., designing a specific ML model architecture to reflect the physiological structure, has yet to be applied to this problem.

In this work, we improve the BI-RNN framework by merging the KGL methodology with a novel KGA approach, resulting in the Compartmental RNN (COMP-RNN) model. This new model hard-codes prior knowledge by embedding the compartmental nature of glucose–insulin dynamics directly into its structure. We show that this architectural advancement leads to a model that is not only more accurate in its predictions but is also significantly more parameter-efficient than the standard BI-RNN. We validate these claims by developing and testing COMP-RNN configurations of varying size with the original BI-RNN and a linear model benchmark, confirming the benefits of our hybrid KGL-KGA strategy.

The paper is organized as follows. We begin in Section 2 by detailing the problem statement and our methodological framework. We then present the BI-RNN model in Section 3 before introducing the novel COMP-RNN architecture in Section 4. In Section 5, we evaluate the performance of our models through in silico validation. Finally, we conclude in Section 6 with a discussion of our findings and future research directions.

NOTATION. Let $\mathbb{R}$ and $\mathbb{N}$ denote the sets of real and natural numbers, respectively, with $0 \notin \mathbb{N}$. The sets of positive and non-negative real numbers are denoted by $\mathbb{R}_{>0}$ and $\mathbb{R}_{\geq 0}$. For $n, m \in \mathbb{N}$, $\mathbb{R}^n$ represents the space of n-dimensional real column vectors, and $\mathbb{R}^{n\times m}$ is the space of $n \times m$ real matrices. The Hadamard (element-wise) product is denoted by $\circ$. For a set $\mathcal{S}$, its cardinality is written as $|\mathcal{S}|$. Discrete-time signals s_k are obtained by sampling continuous-time ones $s(t)$, $s : \mathbb{R}_{\geq 0} \to \mathbb{R}$, as $s_k = s(kT)$, where $k \in \mathbb{N}$ is the time step and $T \in \mathbb{R}_{>0}$ is the sampling time. $\bar{s} \in \mathbb{R}$ designates the sample mean of a finite signal s. Boldface denotes vectors. Component-wise sigmoid and hyperbolic tangent functions of $\boldsymbol{x}_k \in \mathbb{R}^n$ are denoted as $\boldsymbol{\sigma}(\boldsymbol{x}_k)$ and $\mathbf{tanh}(\boldsymbol{x}_k)$, respectively. The ceiling function $\lceil\cdot\rceil$ maps a real number to its smallest greater than or equal to integer. rand $(\mathcal{S}, n)$ denotes a subset of $\mathcal{S}$ composed of n elements chosen uniformly at random.

2. Preliminaries

In this Section, after the problem statement, we present GRU networks and their training procedure.

2.1. Problem statement

We consider the problem of modeling and identifying a discrete-time dynamical system with inputs $\boldsymbol{u}_k \in \mathbb{R}^{n_u}$, $n_u \in \mathbb{N}$, and outputs $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$, $n_y \in \mathbb{N}$. Specifically, we are interested in the modeling of glucose–insulin dynamics in the T1DM-patient system, consisting of $n_u = 2$ inputs $\boldsymbol{u}_k = [\iota_k, \mu_k]^\top$, i.e., the exogenous insulin delivery $\iota_k \in \mathbb{R}_{\geq 0}$ [U] and carbohydrate intake $\mu_k \in \mathbb{R}_{\geq 0}$ [g], and $n_y = 1$ measurable output y_k, namely the BG concentration $\gamma_k \in \mathbb{R}_{\geq 0}$ [mg/dL].

2.2. Gated recurrent unit networks

GRU networks, introduced by Cho et al. (2014), are a RNN architecture variant of LSTMs. They generally consist of $L \in \mathbb{N}$ stacked layers, each with $n_h^{(l)} \in \mathbb{N}, l \in \{1, \dots, L\}$, hidden units. In this work, we focus on single-layer GRU networks ($L = 1$); thus, we omit the layer-specific notation $^{(l)}$ for clarity. This choice is supported by prior studies on glucose–insulin dynamics modeling (De Carli, Licini et al., 2025; Iacono et al., 2023; Mongini et al., 2025), suggesting that additional layers do not significantly improve performance, whereas a properly regularized shallow architecture effectively captures nonlinear dynamics while preventing overfitting.

A GRU layer is a discrete-time nonlinear dynamical system represented in state–space form with inputs $\boldsymbol{u}_k$, outputs $\boldsymbol{y}_k$ and state $\boldsymbol{h}_k \in \mathbb{R}^{n_h}$, also known as the hidden state. Three gating operations control the dynamics of the hidden state: the reset gate $\boldsymbol{r}_k \in \mathbb{R}^{n_h}$, the update gate $\boldsymbol{z}_k \in \mathbb{R}^{n_h}$, and the candidate hidden state $\tilde{\boldsymbol{h}}_k \in \mathbb{R}^{n_h}$. The gate equations are:

$$\boldsymbol{r}_k = \boldsymbol{\sigma}\left(W_r \boldsymbol{u}_k + R_r \boldsymbol{h}_k + \boldsymbol{b}_r\right), \tag{1a}$$

$$\boldsymbol{z}_k = \boldsymbol{\sigma}\left(W_z \boldsymbol{u}_k + R_z \boldsymbol{h}_k + \boldsymbol{b}_z\right), \tag{1b}$$

$$\tilde{\boldsymbol{h}}_k = \mathbf{tanh}\left(W_{\tilde{h}} \boldsymbol{u}_k + \boldsymbol{r}_k \circ \left(R_{\tilde{h}} \boldsymbol{h}_k\right) + \boldsymbol{b}_{\tilde{h}}\right), \tag{1c}$$

where $W_g \in \mathbb{R}^{n_h \times n_u}$ are the input weights, $R_g \in \mathbb{R}^{n_h \times n_h}$ are the recurrent weights, and $\boldsymbol{b}_g \in \mathbb{R}^{n_h}$ are the biases for $g \in \mathcal{G} = \{r, z, \tilde{h}\}$. Afterwards, the hidden state $\boldsymbol{h}_k$ is mapped through a Fully Connected (FC) layer to the final output $\boldsymbol{y}_k$. The overall system behavior is described by the following state–space representation:

$$\begin{cases} \boldsymbol{h}_{k+1} = (\mathbf{1} - \boldsymbol{z}_k) \circ \tilde{\boldsymbol{h}}_k + \boldsymbol{z}_k \circ \boldsymbol{h}_k, & \text{(2a)} \\ \boldsymbol{y}_k = W_y \boldsymbol{h}_{k+1} + \boldsymbol{b}_y, & \text{(2b)} \end{cases}$$

where $\mathbf{1}$ (in bold) is the n_h-dimensional vector of ones, (2a) is the hidden state update equation, and (2b) describes the FC layer mapping the hidden state $\boldsymbol{h}_{k+1}$ to the output vector $\boldsymbol{y}_k$ with $W_y \in \mathbb{R}^{n_y \times n_h}$ and $\boldsymbol{b}_y \in \mathbb{R}^{n_y}$ being the weight matrix and bias vector. From a system-level perspective, we can rewrite Model (2) by representing it in a layered form:

$$\begin{cases} \boldsymbol{h}_{k+1} = \mathrm{GRU}(\boldsymbol{h}_k, \boldsymbol{u}_k), & \text{(3a)} \\ \boldsymbol{y}_k = \mathrm{FC}(\boldsymbol{h}_{k+1}). & \text{(3b)} \end{cases}$$

The GRU model in (3) relies on the following set of parameters:

$$\theta = \{W_g, R_g, \boldsymbol{b}_g : g \in \mathcal{G}\} \cup \{W_y, \boldsymbol{b}_y\}, \tag{4}$$

which are learned during training.

Remark 1. In this work, RNNs specifically denotes GRU networks unless otherwise specified. Nonetheless, the presented methodologies are applicable to any RNN architecture characterized by a hidden state $\boldsymbol{h}_k$ and an arbitrary set of gates $\mathcal{G}$.

2.3. Standard model training

We consider a dataset $\mathcal{D} = \{\mathcal{D}^{(v)}\}_{v=1}^{V}$ of $V \in \mathbb{N}$ input–output sequences, where $\mathcal{D}^{(v)} = \{(\boldsymbol{u}_k^{(v)}, \boldsymbol{y}_k^{(v)})\}_{k=0}^{N^{(v)}-1}$ is the vth sequence with length $N^{(v)} \in \mathbb{N}$, $v \in \{1, \dots, V\}$. We divide $\mathcal{D}$ into mutually exclusive training ($\mathcal{D}_{\mathrm{tr}}$), validation ($\mathcal{D}_{\mathrm{val}}$), and test ($\mathcal{D}_{\mathrm{tst}}$) subsets, serving for model estimation and performance evaluation, respectively. We define the Mean Squared Error (MSE) as the loss function $\mathcal{L}$, which is minimized during training:

$$\mathrm{MSE}(\theta; \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{\mathcal{D}^{(v)} \in \mathcal{D}} \left[\frac{1}{N^{(v)}} \sum_{k=0}^{N^{(v)}-1} \left\| \boldsymbol{y}_k^{(v)} - \hat{\boldsymbol{y}}_k^{(v)}(\theta) \right\|_2^2 \right], \tag{5}$$

where $\hat{\boldsymbol{y}}_k^{(v)}(\theta)$ is the output of the network with parameters θ for the vth sequence, and $\mathcal{D}$ being $\mathcal{D}_{\text{tr}}$ in training. We train the RNN following established iterative gradient-based optimization methods from the literature (Goodfellow et al., 2016), i.e., by updating an initial parameter set θ_0 over $\kappa \in \mathbb{N}$ epochs, with each epoch processing $\mathcal{D}_{\text{tr}}$ in $\mathfrak{b} \in \mathbb{N}$ mini-batches ($\mathfrak{b} \leq |\mathcal{D}_{\text{tr}}|$). We use a learning rate $\mathfrak{l} \in \mathbb{R}_{>0}$, which is multiplied by a decay factor $\zeta \in (0, 1)$ every $\kappa_{\text{dc}} \in \mathbb{N}$ epochs, effectively reducing the size of parameter updates to aid parameter convergence. To improve generalization, an early stopping strategy is used: every $\kappa_{\text{val}} \in \mathbb{N}$ iterations, the MSE in (5) is computed on the validation set $\mathcal{D}_{\text{val}}$ and parameters are saved whenever validation error reaches a new minimum. Furthermore, dropout is applied, a regularization technique that prevents the model from overfitting by randomly ignoring a fraction $\delta \in [0, 1)$ of neuron outputs during each training iteration. The final model uses parameters θ^* obtained by the early stopping strategy with the best performance on $\mathcal{D}_{\text{val}}$.

3. Biological-informed RNN framework

This Section presents the BI-RNN framework (De Carli, Licini et al., 2025), detailing its underlying linear model, which also serves as a benchmark in Section 5.

3.1. Linear compartmental model

Compartmental models are a set of ordinary differential equations, aimed to turn physiology into clear, conservation-based dynamics: well-mixed subsystems exchange material, and states remain nonnegative under nonnegative initial conditions and inputs (Anderson, 2013). This simplicity keeps the model interpretable in various applications, such as the endocrine system description. In this work, we adopt the Linear-Time Invariant (LTI) state–space form of the glucose–insulin system presented by Ruan et al. (2017), which will later be used as a reference. A 5-state model is built through the vector $\boldsymbol{x}(t) = [\gamma(t), x^{\iota,1}(t), x^{\iota,2}(t), x^{\mu,1}(t), x^{\mu,2}(t)]^\top \in \mathbb{R}^5_{\geq 0}$, where $\gamma(t)$ [mg/dL] is blood glucose concentration, i.e. the measured output $y(t)$, $\boldsymbol{x}^\iota(t) = [x^{\iota,1}(t), x^{\iota,2}(t)]^\top$ [U/min] are the insulin compartments, and $\boldsymbol{x}^\mu(t) = [x^{\mu,1}(t), x^{\mu,2}(t)]^\top$ [g/min] are the meal compartments. The model inputs are exogenous insulin delivery ι (or, simply, insulin) and carbohydrate intake μ (or, simply, meals), which are combined in a vector as $\boldsymbol{u}(t) = [\iota(t), \mu(t)]^\top$. Fig. 1 sketches the signal flow: insulin and meals drive their two-compartment chains, which modulate glycemia. The transfer rate of each substance is modulated by the parameters $\mathcal{P} = \{p_0, \dots, p_5\}$, described and identified via regularized least squares as in Abuin et al. (2020). In addition, the model allows the estimation of the Insulin-On-Board (IOB), i.e., the amount of insulin still active in the body, as $\text{IOB}(t) = p_4(x^{\iota,1}(t) + x^{\iota,2}(t))$, and the rate of glucose appearance (Ra) in plasma due to meals as $\text{Ra}(t) = p_3 x^{\mu,1}(t)$ [mg/(dL·min)]. The dynamics read as:

$$\dot{\boldsymbol{x}}(t) = A\,\boldsymbol{x}(t) + B\,\boldsymbol{u}(t) + E, \tag{6a}$$

$$\boldsymbol{y}(t) = C\,\boldsymbol{x}(t), \tag{6b}$$

with

$$A = \begin{bmatrix} -p_1 & -p_2 & 0 & p_3 & 0 \\ 0 & -p_4^{-1} & p_4^{-1} & 0 & 0 \\ 0 & 0 & -p_4^{-1} & 0 & 0 \\ 0 & 0 & 0 & -p_5^{-1} & p_5^{-1} \\ 0 & 0 & 0 & 0 & -p_5^{-1} \end{bmatrix},$$

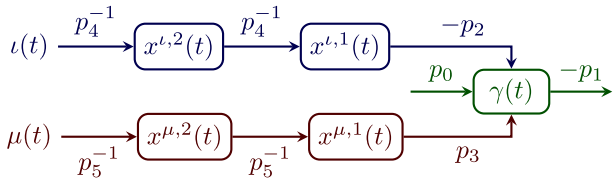

Fig. 1. Compartmental model for glucose–insulin dynamics.

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ p_4^{-1} & 0 \\ 0 & 0 \\ 0 & p_5^{-1} \end{bmatrix},$$

$$E = \begin{bmatrix} p_0 & 0 & 0 & 0 & 0 \end{bmatrix}^\top, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

For implementation, we discretize this continuous-time model via forward Euler, having the discretized state–space matrices as: $A_{\text{d}} = (I + TA)$, $B_{\text{d}} = TB$, and $E_{\text{d}} = TE$, where I is the identity matrix and T is the sampling time. This compact LTI formulation balances physiological meaning and simplicity, serving as a practical benchmark for comparison.

3.2. Biological-Informed RNN (BI-RNN)

The BI-RNN model, as introduced by De Carli, Licini et al. (2025), employs a standard KGL approach by modifying the prediction target and minimizing an augmented, physiologically-related loss. In this work, we follow the same core implementation, with some minor modifications to the loss computation and the integration of a final suitable activation function on the output. Instead of solely predicting glucose, BI-RNN models jointly estimate glucose and the linear model's insulin and meal states, leading to the augmented output vector $\tilde{\boldsymbol{y}}_k = [\gamma_k, \boldsymbol{x}_k^{\iota\top}, \boldsymbol{x}_k^{\mu\top}]^\top$, which is consistent with the LTI model in Section 3.1 but implemented using a standard RNN. Training minimizes an augmented loss function, which accounts for both prediction error and physiological knowledge from the linear model, ensuring the learned dynamics adhere to biological constraints. We define the loss function as:

$$\mathcal{L}(\theta; \mathcal{P}, \mathcal{D}_{\text{tr}}) = \alpha_{\text{D}}\mathcal{L}^{\text{D}}(\theta; \mathcal{D}_{\text{tr}}) + \alpha_{\text{B}}\mathcal{L}^{\text{B}}(\theta; \mathcal{P}, \mathcal{D}_{\text{tr}}) + \alpha_{\text{A}}\mathcal{L}^{\text{A}}(\theta; \mathcal{D}_{\text{tr}}), \tag{7}$$

where the data loss $\mathcal{L}^{\text{D}}$ is the MSE, as in (5), between predicted and real glucose γ. The biological loss $\mathcal{L}^{\text{B}}$ and auxiliary loss $\mathcal{L}^{\text{A}}$ augment the loss function to regularize the BI-RNN, promoting adherence to biologically-meaningful behavior. $\alpha_{\text{D}}, \alpha_{\text{B}}, \alpha_{\text{A}} \in \mathbb{R}_{\geq 0}$ weigh the contribution of each term in (7). In detail, $\mathcal{L}^{\text{B}}$ measures the misalignment between the BI-RNN's predictions and the biological LTI model in (6), computed as:

$$\mathcal{L}^{\text{B}}(\theta; \mathcal{P}, \mathcal{D}_{\text{tr}}) = \frac{1}{|\mathcal{D}_{\text{tr}}|} \sum_{\mathcal{D}^{(v)} \in \mathcal{D}_{\text{tr}}} \ell^{\text{B},(v)}(\theta; \mathcal{P}, \mathcal{D}_{\text{tr}}), \tag{8}$$

$$\ell^{\text{B},(v)}(\theta; \mathcal{P}, \mathcal{D}_{\text{tr}}) = \frac{1}{N^{(v)}} \sum_{k=0}^{N^{(v)}-2} \left\| A_{\text{d}}\hat{\tilde{\boldsymbol{y}}}_k^{(v)}(\theta) + B_{\text{d}}\boldsymbol{u}_k^{(v)} + E_{\text{d}} - \hat{\tilde{\boldsymbol{y}}}_{k+1}^{(v)}(\theta) \right\|_2^2.$$

The auxiliary loss $\mathcal{L}^{\text{A}}$ encourages the predicted states to follow the LTI model's response:

$$\mathcal{L}^{\text{A}}(\theta; \mathcal{D}_{\text{tr}}) = \frac{1}{|\mathcal{D}_{\text{tr}}|} \sum_{\mathcal{D}^{(v)} \in \mathcal{D}_{\text{tr}}} \ell^{\text{A},(v)}(\theta; \mathcal{D}_{\text{tr}}), \tag{9}$$

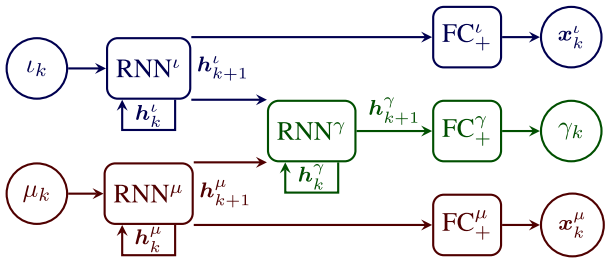

Fig. 2. COMP-RNN architecture.

$$\ell^{\mathrm{A},(v)}(\theta;\mathcal{D}_{\mathrm{tr}}) = \frac{1}{\left|\tilde{\mathcal{N}}^{(v)}\right|}\sum_{k\in\tilde{\mathcal{N}}^{(v)}}\left\|\boldsymbol{x}_k^{\iota,\mu,(v)} - \hat{\boldsymbol{x}}_k^{\iota,\mu,(v)}(\theta)\right\|_2^2,$$
$$\tilde{\mathcal{N}}^{(v)} = \mathrm{rand}\left(\{1,\dots,N^{(v)}-1\}, \lceil \xi N^{(v)}\rceil\right),$$

where $\boldsymbol{x}_k^{\iota,\mu,(v)} = [\boldsymbol{x}_k^{\iota,(v)\top}, \boldsymbol{x}_k^{\mu,(v)\top}]^\top$ from the LTI model in Section 3.1 is purposely added to the dataset $\mathcal{D}$, $\hat{\boldsymbol{x}}_k^{\iota,\mu,(v)}(\theta)$ is its COMP-RNN prediction, and $\tilde{\mathcal{N}}^{(v)}$ is a randomly sampled subset of points, comprising a fraction $\xi \in [0,1]$ of the considered sequence. $\tilde{\mathcal{N}}^{(v)}$ is recomputed for each sequence during every $\mathcal{L}^{\mathrm{A}}$ calculation.

Finally, to enforce non-negative predictions, the BI-RNN architecture incorporates a custom fully connected layer (FC$_+$) that applies a modified rectified linear unit activation: $\tilde{\boldsymbol{y}}_k = \mathrm{FC}_+(\boldsymbol{h}_{k+1}) = \max(\tilde{\boldsymbol{0}}, \mathrm{FC}(\boldsymbol{h}_{k+1}))$, where $\tilde{\boldsymbol{0}}$ is the standardized zero threshold (assuming data is standardized), computed component-wise as the negative of the mean divided by the standard deviation of the output data.

4. Compartmental RNN (COMP-RNN)

Now, we introduce the COMP-RNN model, an evolution of the BI-RNN model described in Section 3.2. The key change lies in the COMP-RNN architectural structure, reflecting physiological compartmental topology (Ruan et al., 2017) by employing individual RNN model compartments. Specifically, we denote with RNN$^\phi$ an RNN system assigned to a physiological compartment ϕ; the same is done for FC$_+^\phi$. This architectural shift incorporates a KGA approach into BI-RNNs, leading to models that leverage both KGL and KGA methodologies. As detailed in Section 3.1, the glucose–insulin system is partitioned into three primary subsystems: insulin (ι), meal (μ), and glucose (γ), as depicted in Fig. 2. We denote the set of compartments as $\Phi = \{\iota, \mu, \gamma\}$. The insulin and meal compartments process only their respective input signals. In contrast, the glucose compartment receives and elaborates processed signals from both the insulin and meal compartments, and mixes their effects. Consistent with the BI-RNN model, the COMP-RNN input is $\boldsymbol{u}_k = [\iota_k, \mu_k]^\top$, and the output is $\tilde{\boldsymbol{y}}_k = [\gamma_k, \boldsymbol{x}_k^{\iota\top}, \boldsymbol{x}_k^{\mu\top}]^\top$. Each compartment's state $\boldsymbol{h}_{k+1}^\phi$, $\phi \in \Phi$, is then mapped to its corresponding output via a dedicated FC$_+^\phi$ layer, as illustrated in Fig. 2. The state–space formulation associated with the COMP-RNN amounts to:

$$\begin{cases}
\boldsymbol{h}_{k+1}^\iota = \mathrm{RNN}^\iota(\boldsymbol{h}_k^\iota, \iota_k), & (10a)\\
\boldsymbol{h}_{k+1}^\mu = \mathrm{RNN}^\mu(\boldsymbol{h}_k^\mu, \mu_k), & (10b)\\
\boldsymbol{h}_{k+1}^\gamma = \mathrm{RNN}^\gamma\left(\boldsymbol{h}_k^\gamma, \left[\boldsymbol{h}_{k+1}^{\iota\top}, \boldsymbol{h}_{k+1}^{\mu\top}\right]^\top\right), & (10c)\\
\boldsymbol{x}_k^\iota = \mathrm{FC}_+^\iota\left(\boldsymbol{h}_{k+1}^\iota\right), & (10d)\\
\boldsymbol{x}_k^\mu = \mathrm{FC}_+^\mu\left(\boldsymbol{h}_{k+1}^\mu\right), & (10e)\\
\gamma_k = \mathrm{FC}_+^\gamma\left(\boldsymbol{h}_{k+1}^\gamma\right). & (10f)
\end{cases}$$

Eqs. (10a)–(10c) describe how inputs are processed within their respective compartments, with corresponding updates in their hidden states. We then obtain the COMP-RNN outputs separately in Eqs. (10d)–(10f) through FC$_+$ layers, mapping each hidden state to its output.

Thanks to this compartmental design, COMP-RNNs not only enable independent modeling of insulin and meal dynamics but also allow assigning an arbitrary number of hidden units n_h^ϕ to each compartment according to its complexity. In this work, however, we suggest setting n_h^ϕ equal for each $\phi \in \Phi$ to ensure a balanced learning capacity across each physiological subsystem. Following the rationale in Section 2.2, each RNN$^\phi$ consists of a single-layer GRU; since a shallow network is sufficient to model the overall glucose–insulin interaction, the same depth is deemed appropriate for the individual compartments. Moreover, the FC$_+^\iota$ and FC$_+^\mu$ layers can be interpreted as neural encoders: they extract the corresponding linear model states from each compartment's hidden state. During inference, if only the glucose prediction is needed, these FC$_+$ layers can be discarded, yielding a leaner, modular network architecture.

The parameter set θ of the COMP-RNN model is the union of the parameters from each compartmental RNN and its associated FC$_+$ layer, defined as:

$$\theta = \bigcup_{\phi\in\Phi, g\in\mathcal{G}} \{W_g^\phi, R_g^\phi, \boldsymbol{b}_g^\phi\} \cup \{W_y^\phi, \boldsymbol{b}_y^\phi\}. \quad (11)$$

We optimize θ according to the training procedure in Section 2.3, employing the same augmented loss function $\mathcal{L}$ as the BI-RNN model in (7).

Remark 2. For a given number of total hidden units, the COMP-RNN has fewer parameters than a standard BI-RNN. This efficiency comes from the compartmental division of the recurrent weights, which mitigates their otherwise quadratic scaling of having a single RNN. Assuming an equal distribution of hidden units across compartments, the COMP-RNN in (10) has roughly 60% the number of trainable parameters as the BI-RNN in Section 3.2, as later shown in Section 5.4.

Remark 3. The linear model in (6) serves as a simplified yet physiologically consistent reference for both the BI-RNN and COMP-RNN architectures. More detailed formulations (e.g., Bergman, 1989, or Dalla Man et al., 2006) could be employed, but would increase the number of compartments and model parameters, thus complicating both the training process and the resulting network structure. One of the long-term goals of this framework is its integration within real-time control strategies; the selected LTI model helps maintain an effective balance between physiological fidelity and architectural simplicity.

5. Simulation results and discussion

This Section describes the framework utilized to assess the improvement of the COMP-RNN w.r.t. the BI-RNN, including the methodology for dataset generation, training procedures, and performance assessment.

5.1. Experimental setup

We use in silico data generated from the simulated cohort of 10 virtual adult patients from the commercial version of the UVA/Padova T1DM simulator (Dalla Man et al., 2014), which incorporates circadian variations in insulin sensitivity according to the model proposed by Toffanin et al. (2013); we apply uniform input conditions to all patients and sample data at 5-minute intervals. The training ($\mathcal{D}_{\mathrm{tr}}$) and testing ($\mathcal{D}_{\mathrm{tst}}$) datasets adhere to

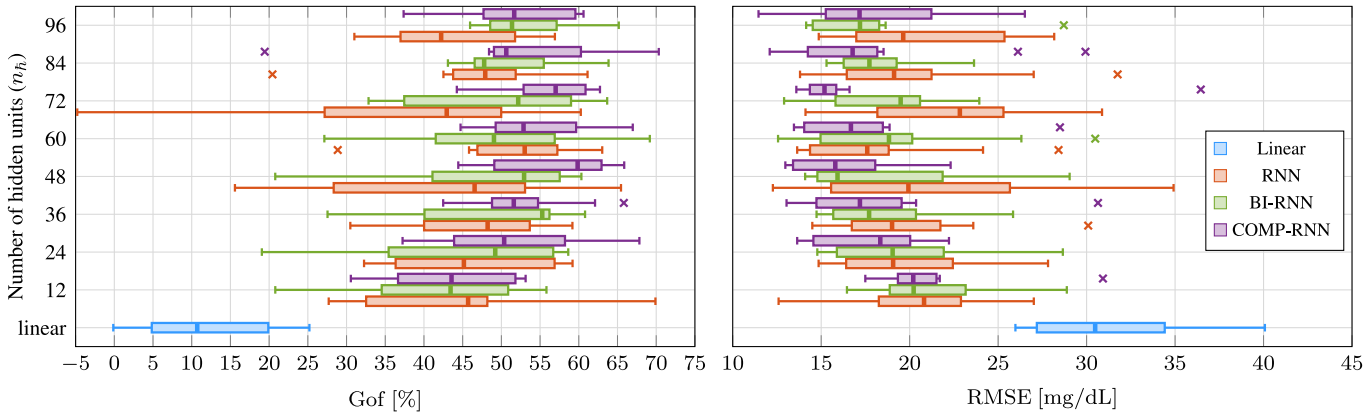

Fig. 3. Gof and RMSE across the 10 simulated patients between COMP-RNNs, BI-RNNs, RNNs and LTI models on test data.

the scenarios described in Messori et al. (2019) (Table 2, Table 3). We derive the validation dataset ($\mathcal{D}_{\text{val}}$) from a 14-day simulation of three structured daily meals: 60 g carbohydrates at 7:00 and 12:00 and 80 g at 18:00, with a duration of 30 min for breakfast and lunch, and 40 min for dinner. To enhance realism, we introduce daily variability by randomly adjusting meal timings by ±20 min, portions by ±20%, and durations by ±10 min each day. The insulin administration strategy is based on standard basal/bolus therapy, where constant insulin is delivered at basal, fasting conditions, and a corrective post-prandial injection (bolus) is administered after a meal (Howorka, 2012). We deliberately introduce clinical realism by miscalculating bolus doses (both over- and under-estimations) and delaying their administration by 5 to 30 min after the start of a meal.

5.2. RNN training

We standardize all data prior to network training and estimate the parameters following the procedure in Section 2.3. We train standard RNNs, BI-RNNs and COMP-RNNs for comparison, all configured as GRU networks with total hidden units ranging from 12 to 96 in increments of 12. We denote configurations by their total number of hidden units (e.g., COMP-RNN$_{96}$). For the COMP-RNNs, we distribute these units uniformly across their compartments as described in Section 4 (e.g., a COMP-RNN$_{96}$ has $n_h^{\phi} = 32$ for each compartment $\phi \in \Phi$). Training runs for $\kappa = 1250$ epochs using $\mathfrak{b} = 2$ mini-batches with a learning rate of $\mathfrak{l} = 0.02$, decay factor $\zeta = 0.8$, decay interval $\kappa_{\text{dc}} = 500$, and dropout rate of $\delta = 0.1$. We employ the augmented loss in (7) with hyper-parameters $[\alpha_{\text{D}}, \alpha_{\text{B}}, \alpha_{\text{A}}] = [0.5, 0.25, 0.25]$ and $\xi = 0.5$. Validation is carried out every $\kappa_{\text{val}} = 15$ iterations. These settings for BI-RNN and COMP-RNN training are taken from De Carli, Licini et al. (2025).

5.3. Metrics for evaluation

We evaluate the models' performance on a test sequence $\mathcal{D}^{(v)} \in \mathcal{D}_{\text{tst}}$ by means of the Root Mean Squared Error (RMSE) and the Goodness of Fit (Gof) computed on the glucose prediction. The RMSE$^{(v)}$ corresponds to the square root of the MSE in (5) for a single sequence on predicted γ, while the Gof$^{(v)}$ is defined as:

$$\text{Gof}^{(v)}\left(\theta^*\right) = 100\left(1 - \frac{\sqrt{\sum_{k=0}^{N^{(v)}-1}\left(\gamma_k^{(v)} - \hat{\gamma}_k^{(v)}\left(\theta^*\right)\right)^2}}{\sqrt{\sum_{k=0}^{N^{(v)}-1}\left(\gamma_k^{(v)} - \bar{\gamma}^{(v)}\right)^2}}\right), \qquad (12)$$

5.4. Results

The proposed COMP-RNN architecture demonstrates a clear superiority across both RMSE and Gof metrics on the test dataset. To assess statistical significance given the limited cohort size (10 patients) and non-normal distribution of paired errors, we employed a one-tailed Wilcoxon Signed-Rank test[1] (Wilcoxon, 1992). As illustrated in Fig. 3 and detailed in Table 1, COMP-RNNs not only surpass the baseline linear model but also achieve statistically significant RMSE reductions (p-value < 0.05) compared to the BI-RNNs (marked with *) and RNNs (marked with †) in various configurations. Performance for each network plateaus at the mid-range hidden unit size ($n_h = 48$), with no significant improvement at the largest configuration, showing good performance results at around 24 total hidden units. Fig. 4 illustrates this remarkable performance, comparing each network type with 24 hidden units using the test scenario for the simulator patient number 9.

However, the most critical advantage of the COMP-RNN is revealed in its physiologically-plausible internal states. Fig. 4 also displays the IOB and Ra from the predicted states of the BI-RNN and COMP-RNN, showing good resemblance of the in silico equivalent for the COMP-RNN: it is evident, particularly during the third day of simulation, that the effect of meals is also influencing the IOB prediction in the BI-RNN, which is not reasonable as IOB should only depend on insulin for physiological consistency. The COMP-RNN does not suffer from this unphysicological crosstalk: its compartmental design enforces a decoupling of insulin and meal dynamics, allowing physiological-compliant treatment of its inputs.

To further validate the ability of COMP-RNN$_{24}$ to accurately infer the input–output relationships, we simulated specific input sequences and examined the resulting glucose trends. As shown in Fig. 5, the glucose level decreases in response to increased insulin administration (i.e., after bolus injection) and increases after carbohydrate intake, according to a correct physiological response. This behavior is consistently observed across all virtual patients in the cohort.

The last notable aspect is the COMP-RNN parameter efficiency: as previously mentioned in Remark 2, COMP-RNN have about 60% the number of parameters of the corresponding BI-RNN, which is depicted in Fig. 6.

[1] Symmetry was confirmed by ensuring the skewness coefficient remained below 1 for all tested configurations, consistent with the thresholds for moderate symmetry defined by Bulmer (1979).

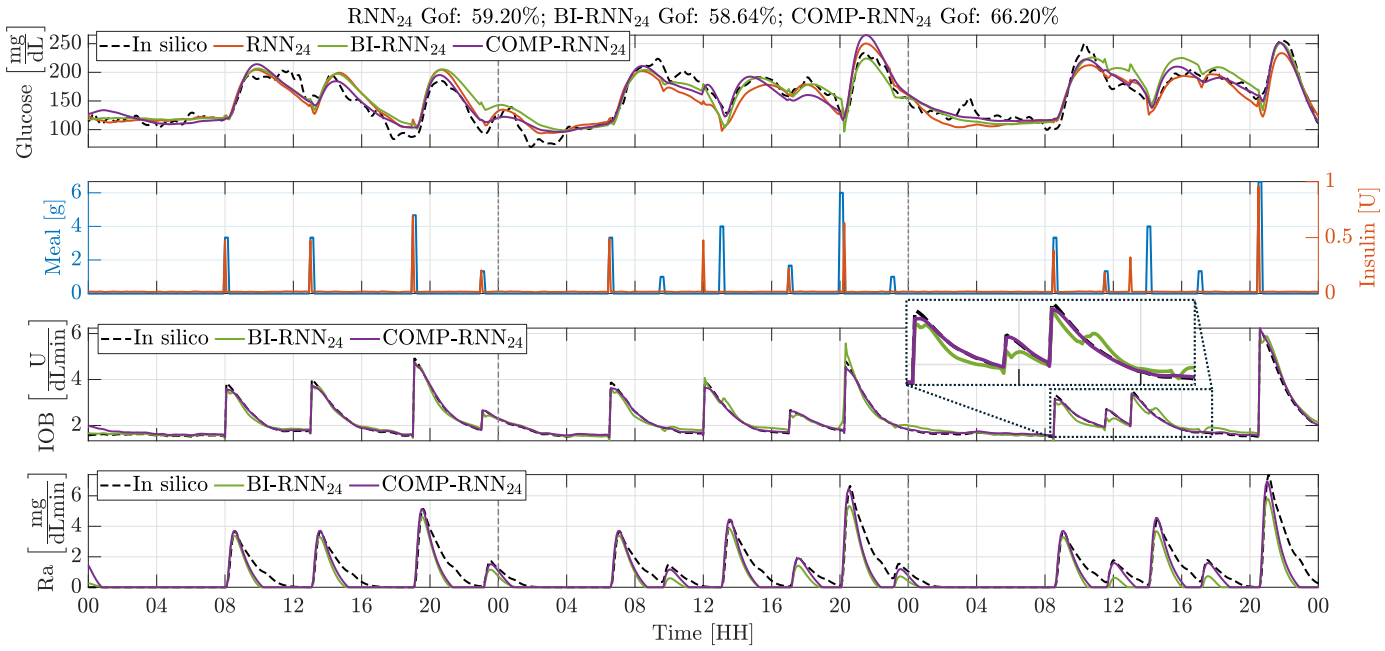

Fig. 4. Comparison of glucose predictions of COMP-RNN$_{24}$, BI-RNN$_{24}$, and the linear model for the simulated patient number 9. Subplots show predicted and in silico glucose concentrations (top), insulin and meal inputs utilized during the simulation (middle), and the predicted and in silico IOB and Ra values (bottom two).

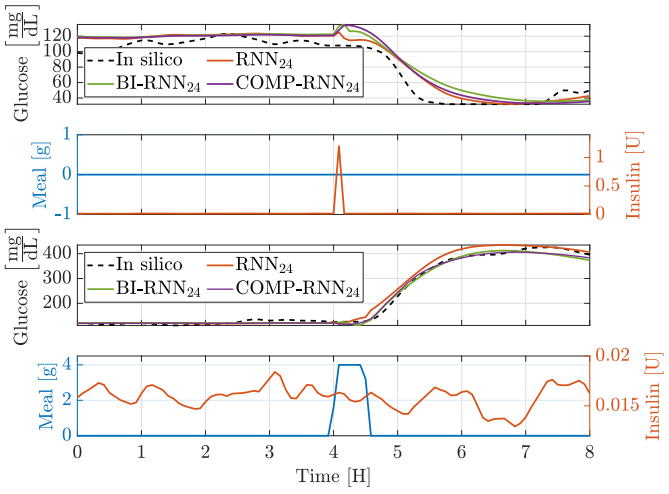

Fig. 5. Physiological validation for Patient 9. The first two subplots illustrate the glucose response of the in silico patient and network predictions to a bolus of insulin, while the last two to a meal with active basal insulin.

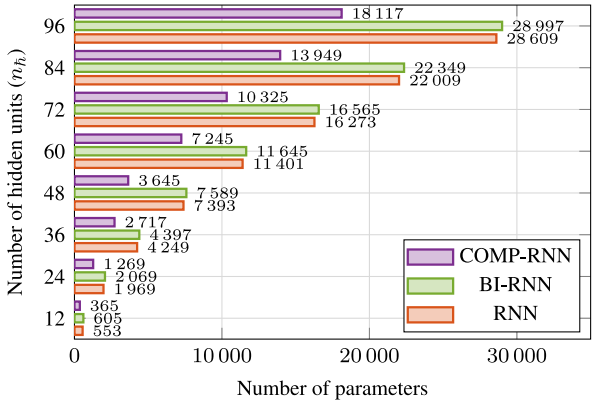

Fig. 6. Number of parameters for each network configuration.

Table 1
Mean ± standard deviation of Gof [%] and RMSE [mg/dL] on test data for different network configurations (best in bold). The baseline linear model achieves Gof = 13.2 ± 8.7% and RMSE = 34.3 ± 10.7 mg/dL. Symbols denote statistical significance (p-value < 0.05) of the RMSE reduction achieved by COMP-RNNs compared to BI-RNNs (*) and standard RNNs (†) using a one-tailed Wilcoxon Signed-Rank test.

n_h	Metric	RNN	BI-RNN	COMP-RNN
12	GoF	**45.2 ± 12.7**	43.4 ± 11.5	44.6 ± 8.4
	RMSE	**20.9 ± 4.2**	21.7 ± 3.9	21.3 ± 3.6
24	GoF	47.6 ± 10.4	47.1 ± 13.3	**52.5 ± 10.3**
	RMSE	20.2 ± 4.5	20.3 ± 4.9	**18.1 ± 3.0***
36	GoF	47.4 ± 8.7	50.1 ± 10.9	**53.2 ± 6.8**
	RMSE	20.4 ± 4.5	19.1 ± 4.0	**18.3 ± 5.0**†
48	GoF	43.7 ± 16.2	49.5 ± 12.4	**57.5 ± 7.6**
	RMSE	22.1 ± 8.0	19.6 ± 5.5	**16.3 ± 3.0***†
60	GoF	52.3 ± 9.9	49.9 ± 11.9	**55.2 ± 6.6**
	RMSE	18.5 ± 4.6	19.4 ± 5.5	**17.5 ± 4.4**†
72	GoF	39.6 ± 19.4	50.5 ± 11.4	**56.8 ± 5.7**
	RMSE	22.9 ± 5.5	18.9 ± 3.3	**17.2 ± 6.8**†
84	GoF	47.0 ± 10.7	52.0 ± 6.7	**52.5 ± 13.6**
	RMSE	20.6 ± 5.3	18.5 ± 2.8	**18.3 ± 5.5**
96	GoF	44.9 ± 9.2	**53.9 ± 5.9**	52.9 ± 7.5
	RMSE	21.4 ± 4.6	**17.9 ± 4.2**	18.4 ± 4.3†

6. Conclusion

In this work, we evolve the KGL approach of the BI-RNN framework by incorporating the KGA methodology through a compartmental architecture, resulting in a hybrid COMP-RNN model that better mirrors the nature of glucose–insulin dynamics. Our in silico validation demonstrates the COMP-RNN superior predictive performance alongside its greater parameter efficiency over the prior BI-RNN model, rendering it a preferred candidate for further developments. Furthermore, the COMP-RNN's compartmental architecture offers a distinct advantage: by enforcing a separation between meal and insulin dynamics, it eliminates the unphysiological crosstalk that compromises the standard BI-RNN model. Although the cohort size is limited to 10 patients,

this corresponds to the entire adult population available in the current commercial version of the UVA/Padova T1DM simulator (Dalla Man et al., 2014). Despite this limitation, formal paired statistical analysis indicates that the observed performance improvements are statistically significant across subjects, suggesting that the benefits of the proposed architecture are consistent at the population level rather than driven by isolated cases. While the resulting impact on downstream clinical decision-making may be incremental, these findings support the relevance of the proposed modeling approach. The main limitation of this study resides in the current COMP-RNN topology, which is, by design, a simplification of the true physiological system. This presents a clear path for future developments: the model modularity allows for direct extension, such as adding compartments related to the physical-activity chain (Licini et al., 2024) or other metabolic factors to create a more comprehensive whole-body representation (Bergman, 1989). Further enhancements will focus on applying specific properties to RNNs, such as stability guarantees (De Carli, Previtali et al., 2025), and implementing these reduced-order models in control frameworks.

CRediT authorship contribution statement

Stefano De Carli: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Nicola Licini:** Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Formal analysis. **Davide Previtali:** Writing – review & editing, Validation, Supervision. **Fabio Previdi:** Writing – review & editing, Supervision. **Antonio Ferramosca:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abuin, P., Rivadeneira, P. S., Ferramosca, A., & González, A. H. (2020). Artificial pancreas under stable pulsatile MPC. *Journal of Process Control*, *92*, 246–260. http://dx.doi.org/10.1016/j.jprocont.2020.06.009.

Anderson, D. H. (2013). *vol. 50, Compartmental modeling and tracer kinetics*. Springer Science & Business Media.

Bergman, R. N. (1989). Toward physiological understanding of glucose tolerance. *Diabetes*, *38*, http://dx.doi.org/10.2337/diab.38.12.1512.

Bulmer, M. (1979). Principles of statistics. In *Dover books on mathematics*, Newburyport: Dover Publications.

Cho, K., Van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proc. EMNLP* (pp. 1724–1734).

Dalla Man, C., Camilleri, M., & Cobelli, C. (2006). A system model of oral glucose absorption: validation on gold standard data. *IEEE Transactions on Biomedical Engineering*, *53*(12), 2472–2478. http://dx.doi.org/10.1109/tbme.2006.883792.

Dalla Man, C., Micheletto, F., Lv, D., Breton, M., Kovatchev, B., & Cobelli, C. (2014). The UVA/PADOVA type 1 diabetes simulator. *Journal of Diabetes Science and Technology*, *8*(1), 26–34. http://dx.doi.org/10.1177/1932296813514502.

De Carli, S., Licini, N., Previtali, D., Previdi, F., & Ferramosca, A. (2025). Integrating biological-informed recurrent neural networks for glucose-insulin dynamics modeling. *IFAC-PapersOnLine*, *59*(2), 91–96. http://dx.doi.org/10.1016/j.ifacol.2025.06.016.

De Carli, S., Previtali, D., Pitturelli, L., Mazzoleni, M., Ferramosca, A., & Previdi, F. (2025). Infinity-norm-based input-to-state-stable long short-term memory networks. In *Proceedings of the 2025 European control conference, ECC2025* (pp. 911–916). http://dx.doi.org/10.23919/ECC65951.2025.11187211.

De Giuli, L. B., La Bella, A., & Scattolini, R. (2024). Physics-informed neural network modeling and predictive control of district heating systems. *IEEE Transactions on Control Systems Technology*, *32*(4), 1182–1195. http://dx.doi.org/10.1109/TCST.2024.3355476.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Cambridge, Mass: The MIT Press.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. http://dx.doi.org/10.1162/neco.1997.9.8.1735.

Howorka, K. (2012). *Functional insulin treatment*. Springer Science & Business Media.

Iacono, F., Magni, L., & Toffanin, C. (2023). Personalized LSTM-based alarm systems for hypoglycemia and hyperglycemia prevention. *Biomedical Signal Processing and Control*, *86*, Article 105167. http://dx.doi.org/10.1016/j.bspc.2023.105167.

Jacobs, P. G., Herrero, P., Facchinetti, A., Vehi, J., Kovatchev, B., Breton, M. D., Cinar, A., Nikita, K. S., Doyle, F. J., Bondia, J., Battelino, T., Castle, J. R., Zarkogianni, K., Narayan, R., & Mosquera-Lopez, C. (2024). Artificial intelligence and machine learning for improving glycemic control in diabetes. *IEEE Reviews in Biomedical Engineering*, *17*, 19–41. http://dx.doi.org/10.1109/rbme.2023.3331297.

Karpatne, A., Kannan, R., & Kumar, V. (2022). *Knowledge-Guided machine learning*. Boca Raton: Chapman and Hall/CRC.

Katsarou, A., Gudbjörnsdottir, S., Rawshani, A., Dabelea, D., Bonifacio, E., Anderson, B. J., Jacobsen, L. M., Schatz, D. A., & Lernmark, A. (2017). Type 1 diabetes mellitus. *Nature Reviews Disease Primers*, *3*(1), 17016. http://dx.doi.org/10.1038/nrdp.2017.16.

Licini, N., Sonzogni, B., Abuin, P., Previdi, F., Gonzalez, A. H., & Ferramosca, A. (2024). Artificial pancreas under stable pulsatile model predictive control: including the physical activity effect. In *Proceedings of the 2024 IEEE conference on decision and control, CDC2024* (pp. 4028–4033). http://dx.doi.org/10.1109/CDC56724.2024.10886334.

Ljung, L., Andersson, C., Tiels, K., & Schön, T. B. (2020). Deep learning and system identification. *IFAC-PapersOnLine*, *53*(2), 1175–1181. http://dx.doi.org/10.1016/j.ifacol.2020.12.1329.

Messori, M., Toffanin, C., Del Favero, S., De Nicolao, G., Cobelli, C., & Magni, L. (2019). Model individualization for artificial pancreas. *Computer Methods and Programs in Biomedicine*, *171*, 133–140. http://dx.doi.org/10.1016/j.cmpb.2016.06.006.

Mongini, P. A., Drecogna, M., & Toffanin, C. (2025). New physics-LSTM hybrid models for control-oriented glucose prediction in type 1 diabetes. In *Proc. amer. control conf.* (pp. 662–667). Denver, CO, USA: IEEE, http://dx.doi.org/10.23919/ACC63710.2025.11107934.

Raissi, M., Perdikaris, P., & Karniadakis, G. (2019). Physics-informed neural networks. *Journal of Computational Physics*, *378*, 686–707. http://dx.doi.org/10.1016/j.jcp.2018.10.045.

Ruan, Y., Wilinska, M. E., Thabit, H., & Hovorka, R. (2017). Modeling day-to-day variability of glucose–Insulin regulation over 12-week home use of closed-loop insulin delivery. *IEEE Transactions on Biomedical Engineering*, *64*(6), 1412–1419. http://dx.doi.org/10.1109/tbme.2016.2590498.

Toffanin, C., Zisser, H., Doyle, F. J., & Dassau, E. (2013). Dynamic insulin on board: incorporation of circadian insulin sensitivity variation. *Journal of Diabetes Science and Technology*, *7*(4), 928–940. http://dx.doi.org/10.1177/193229681300700415.

Wilcoxon, F. (1992). Individual comparisons by ranking methods. In S. Kotz, & N. L. Johnson (Eds.), *Breakthroughs in statistics* (pp. 196–202). New York, NY: Springer New York, http://dx.doi.org/10.1007/978-1-4612-4380-9_16.