Software Engineering
Project: Insulin Pump

AY 2023/24

Enrico Tronci
Computer Science Department, Sapienza University of Rome
Via Salaria 113 - 00198 Roma - Italy

`tronci@di.uniroma1.it`

Version: 2023-12-29

1 Outline

This document outlines possible requirements for the *insulin pump* project. This is just a possibility and you may freely explore different approaches. If in doubt, just send me an email: `tronci@di.uniroma1.it`.

As usual we should identify the following four main components:

- Environment.
- System model.
- Functional requirements.
- Non-functional requirements.

In the following we outline (Sec. 2) some general approach to simulate continuous time systems (as those used to define *Virtual Patients*) along with the components of the pump model.

2 Simulation of Models defined through Ordinary Differential Equations

Many models for the system environment are defined through ODE (*Ordinary Differential Equations*) describing the system dynamics in continuous time.

In order to simulate a system of ODE in our software setting, we need to transform a system of ODE into a *Discrete Time System* (DTS). This is the topic of *Numerical Integration*, a central topic in the construction of simulation models at the core of many *Digital Twins*.

For our purposes, we can transform a system of ODE into a DTS as follows.

Let $x = [x_1, \ldots x_n]$, $u = [u_1, \ldots u_r]$, $\dot{x} = [\dot{x_1}, \ldots \dot{x_n}]$ be real valued functions of time.

Consider the system of ODE in Eq. 1 with initial condition $x(0) = x_0$ ($f : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}^n$):

$$\dot{x}(t) = f(x(t), u(t)) \tag{1}$$

In other words, $\dot{x_i}(t) = f_i(x(t), u(t))$ for $i = 1, \dots n$.

A DTS is defined through a system of equations as shown in Eq. 2 (with initial condition $z(0) = z_0$)

$$z(k+1) = F(z(k), v(k)) \tag{2}$$

Where: z is a function from $\mathbb{Z}$ to $\mathbb{R}^n$ and v is a function from $\mathbb{Z}$ to $\mathbb{R}^r$. Eq. 2 can be easily simulated with a computer program accordingly we wish to transform a system of ODE into a DTS.

Given a time step T we can *sample* our continuous time functions (x and u) by considering their values only at time instants: 0, T, $2T$, $3T$, ...

The DTS in Eq. 2 *approximates* the system of ODE in Eq. 1 if whenever z is computed using Eq. 2 with $z(0) = x_0$ and $v(k) = u(kT)$ ($k = 0, 1, \dots$) it holds that $x(kT) \sim z(k)$ ($k = 0, 1, \dots$) where x is the solution to Eq. 1 with initial condition $x(0) = x_0$. In other words, we can compute an approximate solution to the system of ODE in 1 through the DTS in Eq. 2.

Given a *small enough* time step T and setting $\hat{u}(k) = u(kT)$ and $\hat{x}(0) = x_0$, we can approximate the ODE in Eq. 1 with the following discrete time system (*Euler* approximation):

$$\hat{x}(k+1) = \hat{x}(k) + T \cdot f(\hat{x}(k), \hat{u}(k)) \tag{3}$$

Thus, $F(a, b) = a + T \cdot f(a, b)$ in Eq. 2.

Summing up, given a system of ODE as in Eq. 1, by choosing T *small enough* we can compute a solution for it using the DTS in Eq. 3.

3 System model

Our system consists of the following components:

1. The *environment* model.
2. The *patient* model.
3. The *pump* model.
4. The monitor (that will check our requirements).

In the following we outline the above models and show how they are interconnected.

4 Environment

The environment defines of the food ingested by the patient. The environment model has no inputs. The environment model output consists of a boolean variable evaluating to 1 when sugar is assumed and 0 otherwise.

For example, a diligent patient may ingest a moderate amount of carbohydrates every 8 hours or so, whereas a *greedy* patient may eat more and perhaps more frequently.

The equations A5 in the paper describing the Padova Type 2 Diabetes Simulator define the effect of the foot intake. In those equations $\delta(t)$ is an external input. So, this is to be provided by our environment. Here is an example of how $\delta(t)$ could be defined.

Let L be the meal duration (in minutes). For example, we may use $L = 60$ (i.e., one hour).

Let F be the fasting duration (among meals). For example, we may use $F = 480$ (i.e., 8 hours).

We may set:

$$\delta(t) = \mathrm{square}(0, L, F, t) \tag{4}$$

Where:

$$\mathrm{square}(a, b, c, t) = \begin{cases} 1 & \text{if } t \bmod (a+b+c) \in [a, a+b) \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Note that in the equations A5 in the above mentioned paper, $\delta(t)$ is multiplied by the amount (in mg/min) of glucose ingested during the meal (parameter Dose in A5). So the amount of glucose actually ingested during the day is $\int_0^{1440} \mathrm{Dose} \cdot \delta(t) dt$.

With our schema of 3 meals lasting L minutes we have $GD = 105 * 3 * 60 = 18.9g$

As an example, with the above schema (3 meals of $L = 60$ minutes each) we may chose Dose $= 105$. This leads to a daily glucose intake of $105*3*60 = 18.9g$ which is below the limit of $24g$ [$36g$] per day for women [men].

We may increase [decrease] the value of Dose to simulate a patient taking more [less] sugar per day than the *diligent* patient above. Furthermore, we may change the feeding schema (for example, 5 meals lasting 30 minutes each).

5 Patient model

The patient model takes as input the insulin dose provided by the pump and the food intake from the environment.

The patient model output is the blood glucose concentration.

Of course it is not our goal here to elaborate a model linking insulin, food and glucose. We can just use a model from the literature. You may use the one from the recent paper:

Roberto Visentin, Claudio Cobelli, Chiara Dalla Man. *The Padova Type 2 Diabetes Simulator from Triple-Tracer Single Meal Studies: In Silico Trials also possible in Rare but Not-So-Rare Individuals.* Diabetes Technology and Therapeutics, 2020.

This paper is included in this directory.

This is a paper modelling type 2 diabetes, whereas insulin pumps presently are mainly used for type 1 diabetes (indeed, it is much harder to certify software for type 2 diabetes). However the model above is self-contained and contains all elements we are interested in.

The appendix in the paper contains the equations for the model. The food intake is modelled through the variables Dose and $delta(t)$ in equations A5, A6, A7, A8 of the above paper as explained in Sec. 4. This connects the patient model to the environment. Note that the Dose is part of the patient model whereas $delta(t)$ is a patient input.

Of course, you do not need to enter into the details of the model. It suffices to model the equations in the paper.

Parameters for the equations are in the tables in the above paper.

The patient model rests, as usual, on ODE (*Ordinary Differential Equations*) describing the system dynamics in continuous time.

We can transform the system of ODE in the above paper into a discrete time system using the approach in Sec. 2.

Considering that the time is measured in minutes in the paper describing the Padova Type 2 Diabetes Simulator, we may chose $T = 0.1$, that is T is 0.1 minutes (i.e., 6 seconds).

The patient model takes as input the amount $u(t)$ of insulin from the pump. To account for such input the model in the above paper can be modified as follows.

First, the plasma insulin concentration $I(t)$ in Eq. A2 of the paper changes from $I(t) = \frac{I_p(t)}{V_I}$ to

$$I(t) = \frac{I_p(t) + u(t)}{V_I} \tag{6}$$

Second, the equation A2 in the paper defining the plasma insulin mass $I_p(t)$ changes from $\dot{I}_p(t) = LHS$ to

$$\dot{I}_p(t) = LHS + u(t) \tag{7}$$

6 Pump model

The pump model (actually the model for the control software) takes as input the glucose level from the patient and returns as output the amount of insulin to be injected (which, in turn, will go as input to the patient model).

For the pump you may use rules as those in the book, or any strategy that you like. You may use an approach similar to the one shown during our classes to optimise the parameters of the control strategy. You may use a sampling time of 5 minutes.

The pump input is the glucose concentration in the blood, that is $G(t)$ in Eq. A1 of the paper describing the patient model.

The pump output is the amount of insulin to be injected. This is $u(t)$ in the patient model outlined in Sec. 5.

7 Monitors

We will have a monitor for each requirement are will be verifying. Each monitor takes as input the system variables needed to evaluate the monitored property.

A monitor for a functional requirement returns 0 if the requirement is satisfied and 1 otherwise.

A monitor for a non-functional requirement return a value assessing the requirement under evaluation.

The following monitors should be developed.

7.1 Functional requirements

- *Safety*: glucose should never drop below 50 mg/dL (hypoglycemia, very dangerous).
- *Liveness*: glucose should stay as close as possible to 100 mg/dL.

7.2 Non-functional requirements

- The amount of insulin injected daily should be minimized. The monitor will evaluate how much insulin per day the pump injects. You may then explore different strategies to satisfy the functional requirements while minimizing the injected insulin.
- The control software sampling time should be maximized (5 minutes will do, but a strategy working with a sampling time of 10 minutes would be even better). No monitor needs to be developed for this non-functional requirement. Just experiment with different sampling times and report your findings.

8 Final remarks

Start model development from the interfaces. So, just develop models that take inputs and output and just return dummy numbers. This will allow you to test if the models are properly connected. Then, you will focus on developing correct models for each single component.