

Zadání úlohy MOS 9.11.2011: Regulace glukózy

Na cvičení jsme si ukazovali model regulace glukózy. Tento model máme implementován pouze v rovnicích.

```

model glucoseEquation
  parameter Real QL = 8400 "Glucose Input Flow Rate";
  parameter Real lambda = 2470 "constant";
  parameter Real v = 139000 "pokles na 20 % simuluje diabetes melitus 2. typu - inzulin independentni";
  parameter Real mi = 7200 "constant";
  parameter Real theta = 2.5 "Reabsorption Maximum";
  parameter Real phi = 0.51 "Threshold Glucose Concentration";
  parameter Real alpha = 7600 "Insulin Destruction Gain";
  parameter Real beta = 1430 "pokles na 20 % simuluje diabetes melitus 1. typu - inzulin dependentni";
  Real TURind "Tissue utilization rate - insulin independent";
  Real TURdep "Tissue utilization rate - insulin dependent";
  Real RLR "Renal loss rate";
  Real x "Blood Glucose Concentration (mg/mL)";
  Real y "Blood Insulin Concentration (mU/ML)";
  parameter Real CG = 15000 "Capacitance of Glucose";
  parameter Real CI = 15000 "Capacitance of Insulin";
  Modelica.Blocks.Interfaces.RealInput GlucoseInfusion
  Modelica.Blocks.Interfaces.RealOutput xOutput
  Modelica.Blocks.Interfaces.RealOutput yOutput
equation
  xOutput = x;
  yOutput = y;
  TURind = lambda * x;
  TURdep = v * x * y;
  if x <= theta then
    RLR = 0;
  else
    RLR = mi * (x - theta);
  end if;
  CG * der(x) = GlucoseInfusion + QL - TURind - TURdep - RLR;
  if x <= phi then
    CI * der(y) = -alpha * y;
  else
    CI * der(y) = -alpha * y + beta * (x - phi);
  end if;
end glucoseEquation;

```

Vášim úkolem je převést model do blokové, přehlednější podoby. Inspirací vám budiž přiložený text z knihy Michael C.K. Khoo: Physiological Control Systems. Uvedený text sice popisuje model ustáleného stavu (a tedy nikoli dynamický), nicméně princip rozdělení bude stejný.

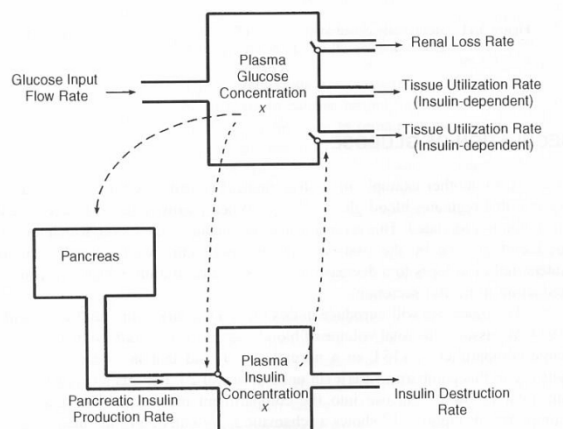
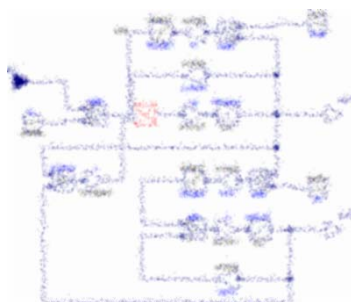


Figure 3.12 Schematic representation of the processes involved in the regulation of glucose and insulin.

Naše představa je, abyste došli k podobnému schématu , jako je na Fig. 3.12. Dnešní úkol tedy není ani tolik o implementaci, jako o porozumění rovnicím a abstrakci na jednotlivé subsystémy.

Bonus

V dnešním bonusu máte na výběr, zda si chcete hrát s Modelicou, či raději Simulinkem. Zadání je jednoduché: Implementujte model pomocí kauzálních bloků – tj. bez rovnic, čili stejně tak, jak by se to provedlo v Simulinku, a to buď v Modelice, pomocí knihovny Modelica.Blocks, či libovolně v Simulinku (+1bb “bonus bod”)



Obrázek 1: Ukázka možného vzhledu modelu

Upozorňuji, že pokud se rozhodnete dělat v Modelice, budete pravděpodobně muset použít OpenModelicu, nebo jinou implementaci jazyka, protože výsledný model již možná nepůjde spustit v Demo verzi Dymoly (nicméně nakreslit si to tam můžete).

Současná poslední revize OM má problém se simulací některých vstupních funkcí (typu step) z Modelica Library. Ty si samozřejmě můžete udělat vlastní, důležitá je kauzalita výpočtu – připodobnit schéma postupu výpočtu. Pak si jednotlivé bločky můžete udělat sami klidně všechny.

Bonus bonusů

Opět za použití OM a reprodukovatelný report případných problémů +1b.

Odevzdání

Kód spolu se zprávou odevzdejte do středy 16.11. do 12:00 do upload systému. Posunutí odevzdání je kvůli posunutí zadání, na cvičení si ale ukážeme správné řešení, proto budeme odevzdání po cvičení penalizovat.