Homework 4
Due Nov. 15

A common way to manage type I diabetes is through the use of insulin pumps. Currently, insulin pumps work by injecting a small basal level of insulin into the patient's blood. When eating a meal, the patient presses a button to obtain a large bolus of insulin to counteract the effects of the meal on their blood sugar.

An alternative strategy to maintain glucose levels in diabetic patients to use a closed-loop feedback system. To derive the closed-loop controller, we should first understand the open-loop system. The insulin system may be modeled in the following way.

Let G be the patient's blood glucose level, I be the patient's blood insulin level, and X be the amount of insulin dissolved in the patient's tissues. The input variable u would then be the insulin injection rate, into a blood volume of V_l. The corresponding equations follow:

$$\dot{G} = -P_1 G - P_4 X(G + G_b)$$
$$\dot{X} = -P_2 X + P_3 I$$
$$\dot{I} = -n(I + I_b) + u / V_l$$

Constants in the model have these numerical values:

- P_1 = .0287 per minute
- P_2 = .0283 per minute
- P_3 = 5.04×10^{-5} per minute
- P_4 = 1 L per mU-min. (the model I'm working from omitted this coefficient, but we need something here to get consistent units)
- G_b = 4.5 mMol per Liter
- n = 5/54 per minute
- V_l = 12 L

For this system:
Given that we are applying 15 milliUnits per Liter of insulin to the diabetic patient to maintain a basal level of insulin in the blood (i.e., I_b = 15)…

1. Find the equilibrium point of the nonlinear model;
2. Linearize around the equilibrium point;
3. Rewrite the linearized system in state space form;
4. Find the transfer function of the linearized system.

I recommend doing the work on paper in terms of the variables, then doing the computation using MATLAB or a symbolic solver. Do add relevant comments to your code.

Suggested exercises (not required for homework assignment):

a. Plot the step response of the linearized system;

b. Find the eigenvalues of the state matrix and comment on the system's stability;

c. Use MATLAB's ODE solver, such as `ode45()`, to solve the system of equations.