Bolus calculators using d, y, and dy/dt

In this project, you will develop a bolus calculator which uses the estimated meal carbohydrate content (d), the blood glucose concentration at the time of the meal (y), and the time derivative of the blood glucose concentration (dy/dt). The purpose is to administer more insulin if the blood glucose is already high or if it is rising, and conversely, administer less in the opposite cases.

Part 1 – analyzing the optimal boluses

1. Simulate the meal response (12 h) using the MVP model.
2. Evaluate the objective function, phi.
3. Repeat step 1. and 2. for different insulin boluses and meal carbohydrate contents (0-10 U and 0-150 g CHO).
4. Plot the objective function as a function of both insulin boluses and meal carbohydrate contents (use, e.g., Matlab's surf function).
5. For each meal carbohydrate content, compute the optimal bolus (e.g., in increments of 10 g CHO between 10 and 150 g CHO).
6. Create a set of states where 1) the blood glucose concentration is different from the target of 108 mg/dL and 2) the time derivative of the blood glucose concentration is different from zero (give a small meal or insulin bolus and simulate for, e.g., 30 min).
7. Repeat step 1.-5. where you use the states from step 6. in the initial condition.

Part 2 – developing the bolus calculator

8. Divide the meal carbohydrate content with the optimal bolus and take the average. The result is an estimate of the insulin-to-carb ratio (ICR).
9. Create a simple controller that administers bolus insulin based on the ICR (divide the meal carbohydrate content with ICR).
10. Create an extended version of the simple controller which also takes the blood glucose concentration into account (higher blood glucose concentration requires more insulin and vice versa).
11. Extend the extended controller to also take the rate of change (RoC) of the blood glucose concentration into account (a higher RoC requires more insulin and vice versa).
12. Tune the controllers to a single patient (the controller parameters do not need to be the same in all three versions).
13. Compare the three versions for a single patient.
14. Develop an I-controller (an integrator) for estimating the controller parameters in all three versions of the controller (start with the simple one).
15. Compare the three versions for 1, 10, and 100 virtual people.

Contact: John Bagterp Jørgensen, jbjo@dtu.dk, DTU Compute