**Smart Multidrug Delivery for Anesthesia**

**Introduction and Model**

Anesthesia is a medical treatment that prevents patients from feeling pain during procedures like surgery, certain screenings, and diagnostic tests. General anesthesia affects the whole body, making patients unconscious and unable to move. In particular, the clinical procedure consists of inducing 3 conditions: hypnosis, analgesia, and neuromuscular blockade. The first condition is ensured by the injection of Propofol, the second by the Remifentanil and the last by the Atracurium. The obtained outputs are respectively: Bispectral Index (BIS), Richmond Agitation-Sedation Score (RASS) and Neuromuscular Blockade (NBM).

BIS processes the EEG signals in order to obtain a value, which is an integer number, that reflects the level of consciousness of the patient on a scale 0 to 100, where 100 means the patient is completely awake and 0 means that there’s no brain activity recorded. During the surgery the BIS target value is 50 (general anesthesia).

RASS is a 10-point scale of integer numbers that goes from -5 (unarousable) to +4 (combative) and describes the agitation or sedation of the patient; during the surgery the goal is to maintain RASS around level -4 (deep sedation).

NBM defines the lack of movement, and the aims is to maintain an adequate level of paralysis during the surgery; the value range goes from 0 up to 100% where 0 means total paralysis and 100% means total muscular activity. For surgery the preferred level of NBM is 15%.

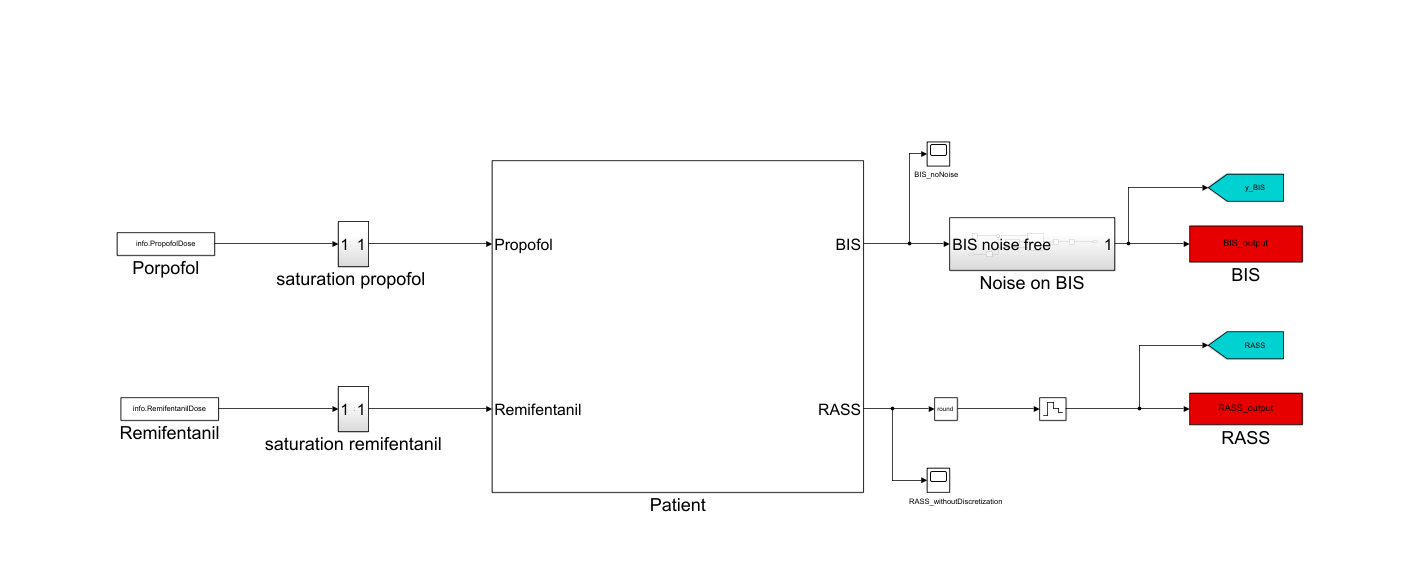
Because we are two-person group we consider only two inputs: Propofol (drug 1) and the Remifentanil (drug 2) and we want to reach and maintain a steady state of

Figure : Schema of the control system.



in less than 60s via a closed-loop delivery. In particular the inputs of the system are the dose of Propofol and Remifentanil injected, while the outputs are respectively the BIS and RASS signals as reported in Figure 1[[1]](#footnote-1).

The Simulink Non-Linear model is reported below. After the input block there is a saturation block both for the Propofol and the Remifentanil, this is used in order define an upper and a lower limit for the injection:

* [0 - 5] mg/(kg\*min) for Propofol;
* [0 - 2.5] mg/(kg\*min) for Remifentanil.

We have available biological parameters of 24 different subjects and one set of biological parameters calculated as the mean value among all the subjects.

Figure : Non-Linear Simulink model.

The outputs of this system for the average patient are reported in the graphs below.

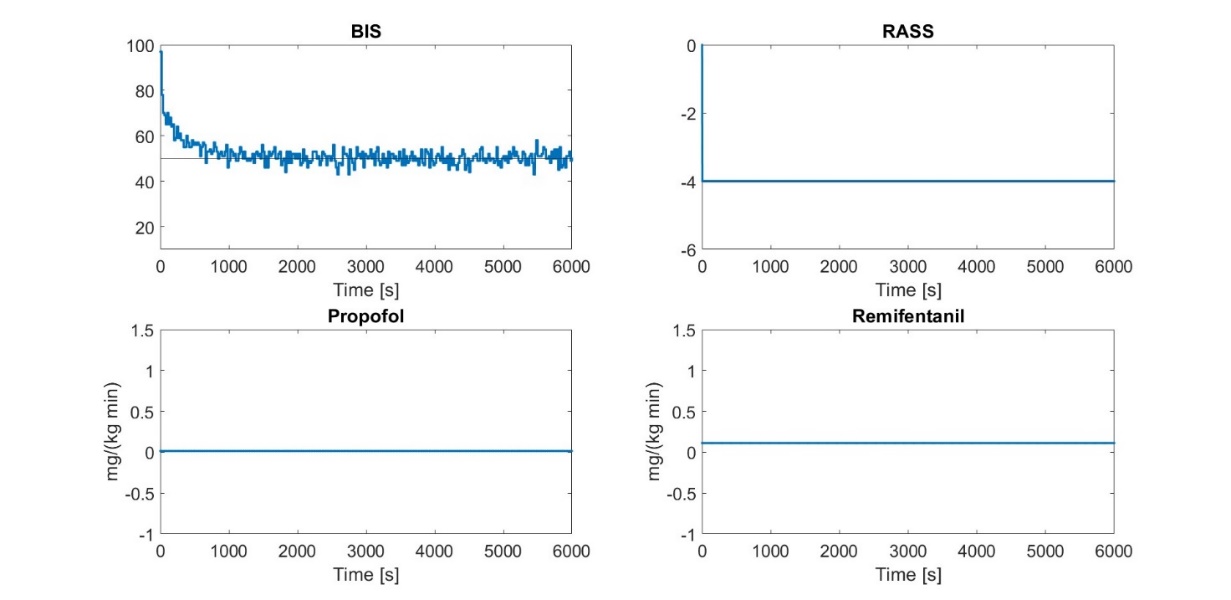


Figure : Inputs and outputs of the non-linear system.

**Task 1: Model Linearization Average Patient and Validation**

In order to linearize a non-linear system model, we need to calculate the equilibrium point around which we can do the linearization. We used *Simulink*'s *Model Linearizer* to obtain the operating point required by the linearization, knowing the outputs that we were aiming for. The first attempt, trying to trim the model knowing the desired outputs didn't yield any solution because the constraints made it unfeasible. Then, we tried trimming the model considering not the desired outputs, but the starting doses of the drugs, as given by the homework instructions. This yielded viable results, and the computed outputs were comparable to the desired outputs. We then used the states that were computed for this operating point to estimate a new operating point with unknown inputs and the desired outputs. This last attempt was successful, and we used the latest operating point as parameters for the *Model* *Linearizer* to linearize our system model.

The linearize model is described through four matrices A, B, C and D which can be used in a Simulink scheme using State Space block as can be seen in Figure 4.

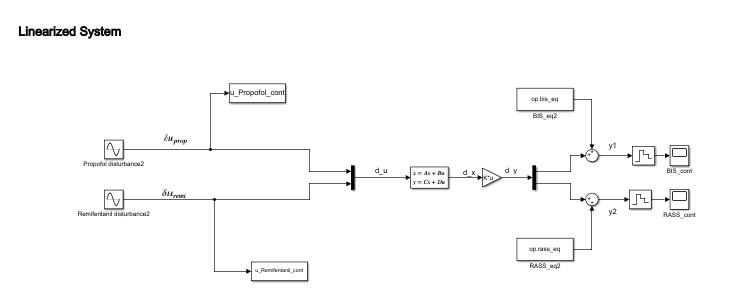


Figure : Linear model in Simulink.

Through a Matlab function (*c2d*) we can easily discretize the linearized model. The Simulink scheme for the discrete model is the same as the previous but uses a Discrete State Space block and sampling when needed.

The non-linear model works with actual values of input and output. On the other hand, the linearized model accepts in input the variation of the input from the equilibrium points and gives in output the variation of BIS and RASS from the desired output.

We compared the performance of the three models when the inputs present a sinusoidal 30% perturbation of the operating point.

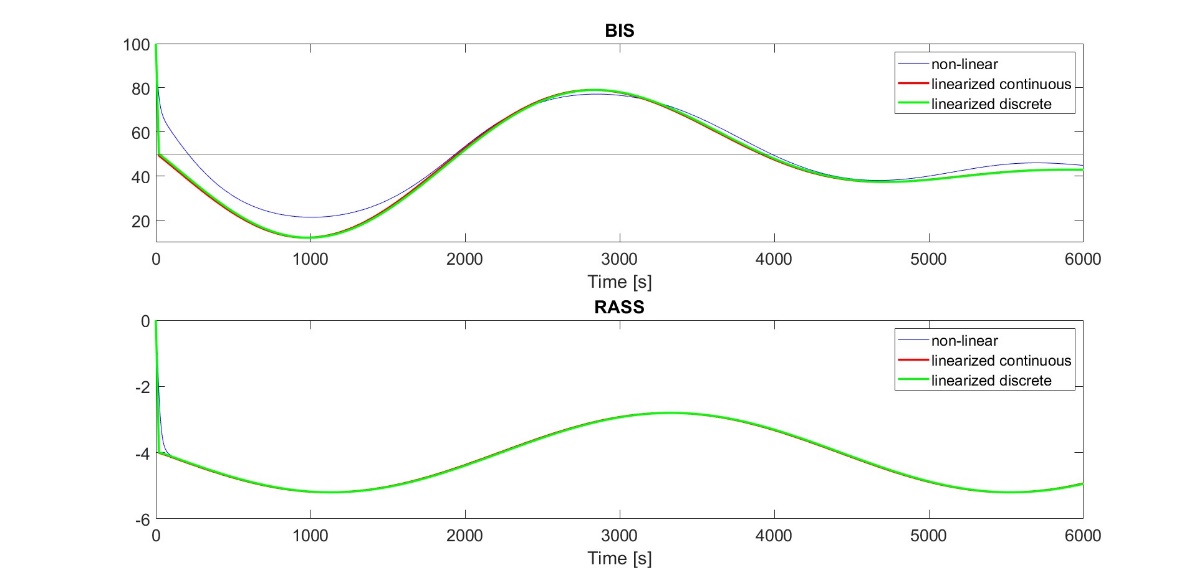


Figure : Comparison between Non-Linear, Linear continuous and Linear discrete models.

From the comparison of the three models, we can see that the linearized system faithfully follows the non-linear system aside from a slight difference at the beginning.

**Task 2: MPC for the Linearized Average Patient, Accessible**

We try now to control our linearized system to obtain the performance described before, by applying the Model Predictive Control strategy. In this case, the MPC requires in input the current state of the system. We consider the state as accessible, and we employed a simple escamotage to access the state as the output of the linear system. The actual outputs are computed at a later step using a gain block where the gain is the matrix C computed in the linearization.

Since this is a biological system we couldn’t withdraw injected drugs, therefor we set the constraints that input as computed by MPC couldn’t be negative.

We tune the MPC to value more the output rather than the input by setting low values of R matrix and by regulating the aggressiveness of the controller with the matrix Q.

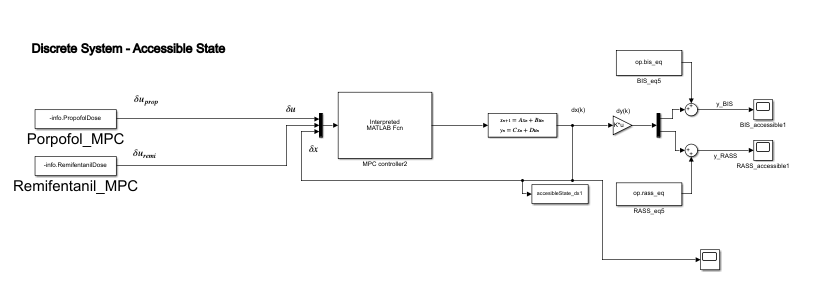


Figure : Simulink Model with the MPC.

We notice that there’s a perfect match and response from the controlled system with MPC, this is expected since we are applying an MPC build around the linear system that it is controlling, therefore there's an exact match with the plant.

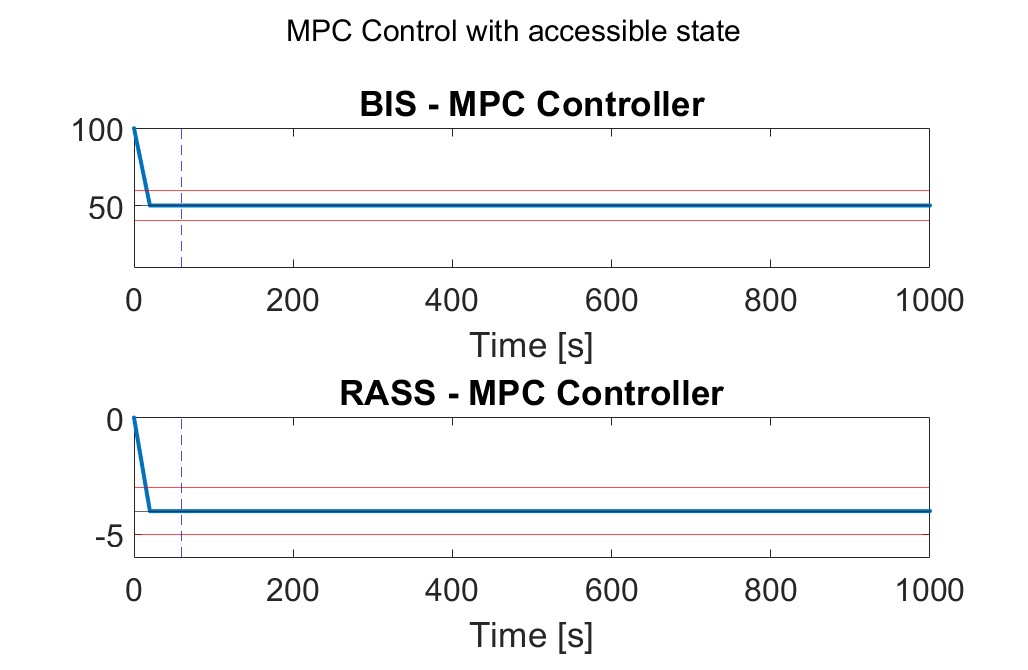


Figure : Output from the MPC controlled system.

**Task 3: MPC for the Linearized Average Patient, State Estimated with Kalman Filter**

**3.1 State Estimation**

For this task, we introduced a Kalman filter for state estimation instead of considering the state accessible.

When defining the parameters to build a Kalman filter, we selected as system model the discrete linearized model that we computed earlier. The initial state is set as the same as the initial state of the plant.

The matrices and have been tuned as follows:

* For , we had a diagonal matrix such that for each element of the diagonal

Where

* While for , since it’s linked to the measurement error of the sensor, which we don’t know, we assumed a very good sensor with little uncertainty and set

Our goal is that the estimated states are equal to the states obtained with the linearized system. Since the linearized system is the same system that’s used to define the Kalman filter, we expect a near perfect match.

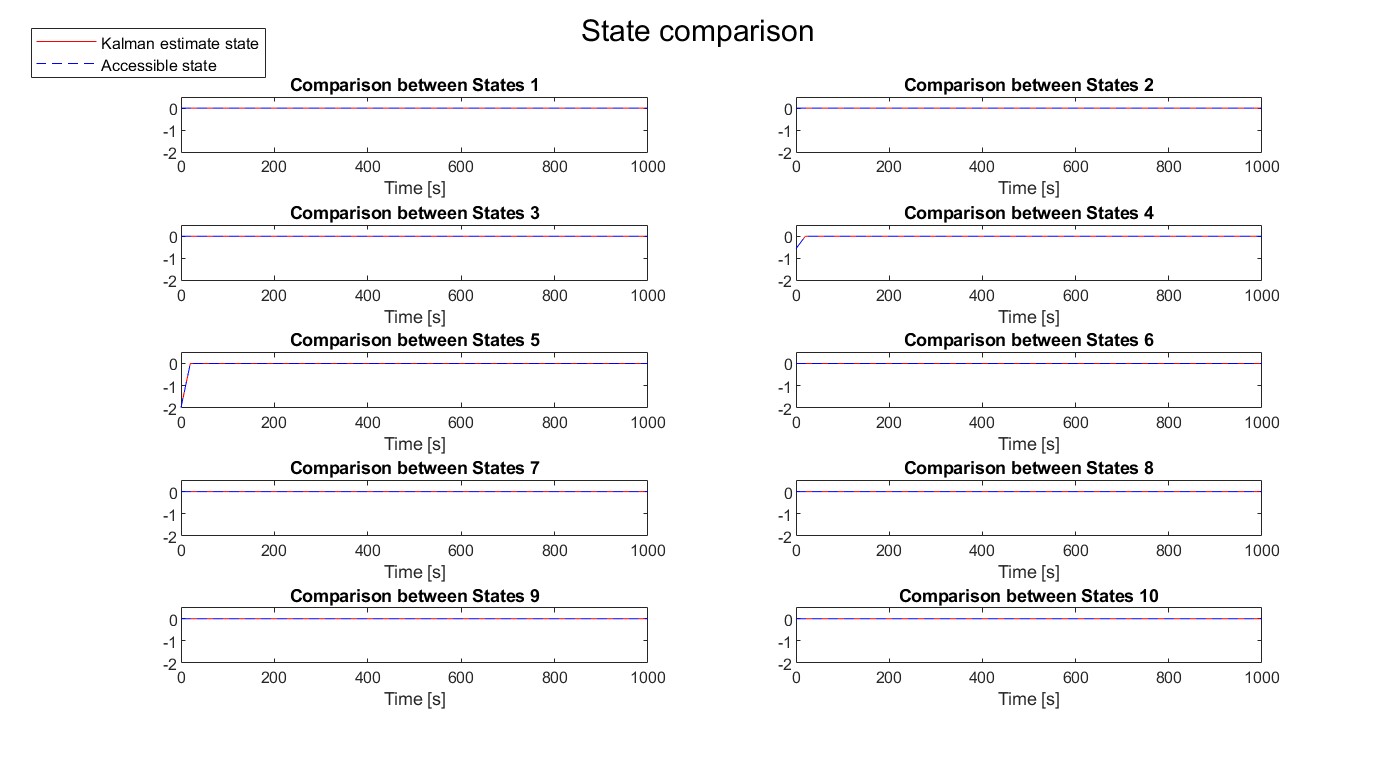


Figure : State comparison between accessible states (dotted line) and Kalman estimated states (red line).

**3.2 MPC for Linearized Average Patient, inaccessible state**

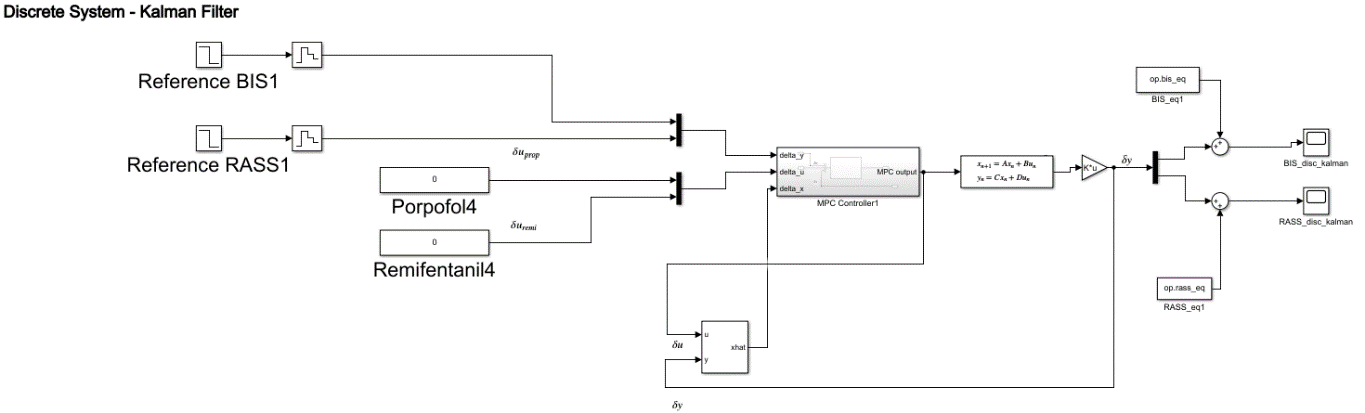
We added the state estimator to the Simulink scheme with the MPC.

Figure : Simulink scheme with Kalman filter and MPC

Since the linearized system, the MPC controller and the Kalman filter use as inputs and outputs the variations of the signals, the Kalman filter is fed directly with the outputs of the MPC and the linearized system. We add back the equilibrium values of the BIS and RASS signals when feeding the signals to the scopes for visualization.

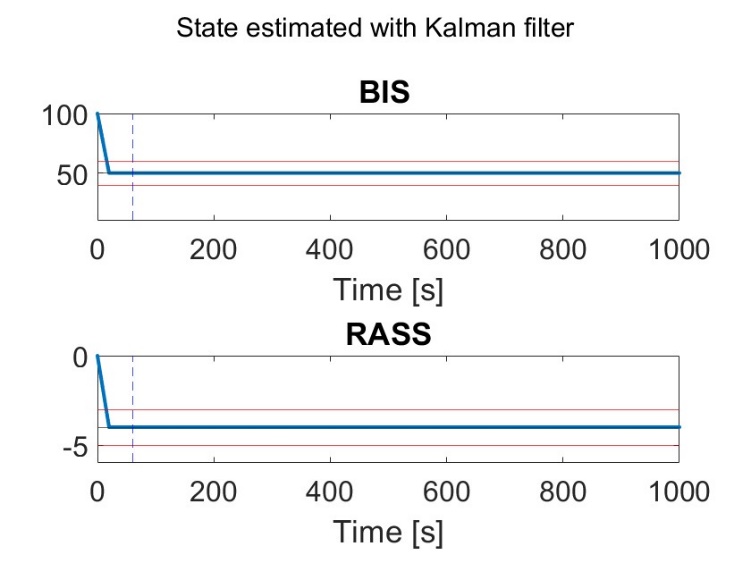


Figure : Output from the MPC controlled system, inaccessible state

The output is almost identical to the previous one. This may be explained by the perfect match in the linearized plant and the model used to create the Kalman filter.

**Task 4: MPC for nonlinear Average Patient and on the population**

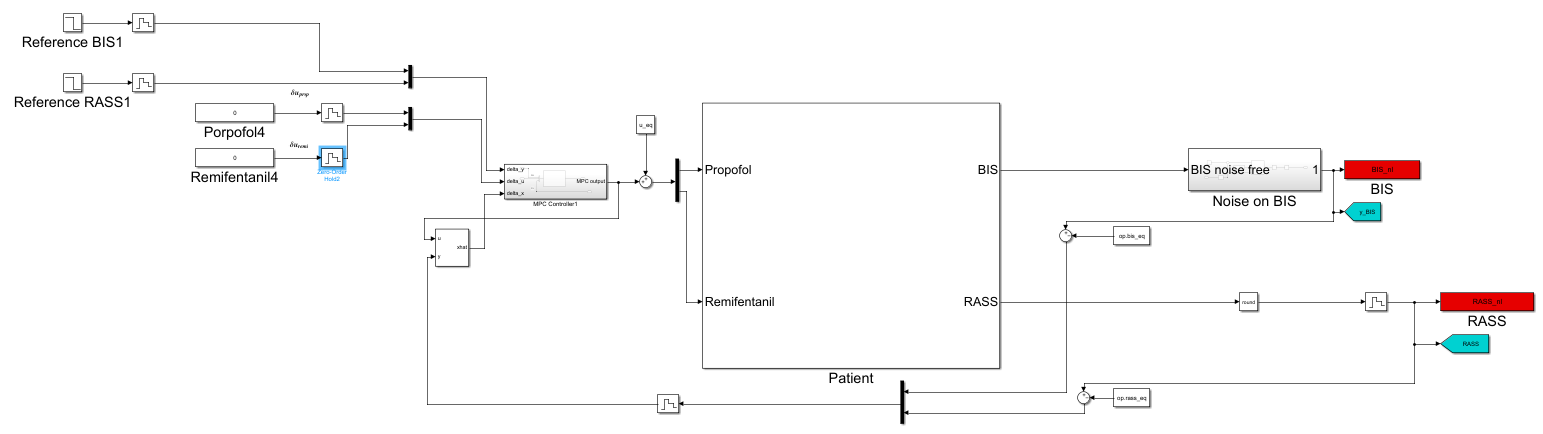


Figure : Simulink scheme for MPC applied to non linear patient

We apply the MPC controller, with inaccessible state, to the nonlinear system. In this case, we must add to the MPC output also the input equilibrium point, because while the MPC works with , the nonlinear system requires the full input . In the same way, when feeding to the Kalman filter, we must first compute that as the difference between the system output and the equilibrium point.

We evaluated this model for the average patient, and a random patient (subject 4) (Figure 12).

Then we evaluated the average behaviour of all 24 patients (Figure 13).

…

We couldn’t tune the system because any change to the MPC would not have reflections on the system output, regardless of our attempts. We could not find the problem, although we deem our scheme and our MPC function correct.

**Task 5: Integral Action**

LOL.

Conclusion

We are unable to make a good analysis of the problem, since for some unknown to us reason the MPC appears to be ineffectual, and the system always achieves perfect tracking regardless of the tuning. In the same way, we cannot reach our goal within 60s because our tuning parameters don’t have any effect on the system.

1. Final Project Anesthesia, prof. Del Favero [↑](#footnote-ref-1)