Learning Based Model Predictive Control

Saket Adhau
MIS: 121717001

Guide: Dr. Dayaram Sonawane

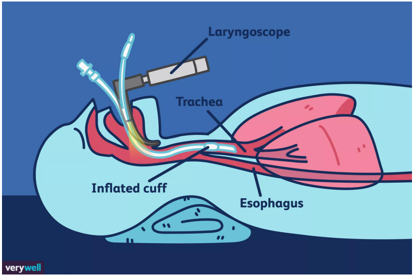
April 11, 2021

Problem Statement

Administration of Anesthesia to control pain during surgery

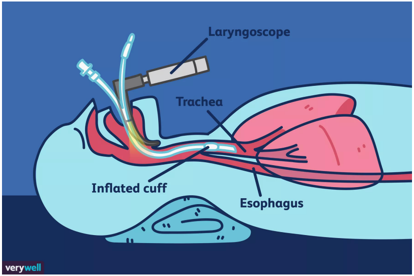
General Anesthesia

Breathing Mask/Intubation



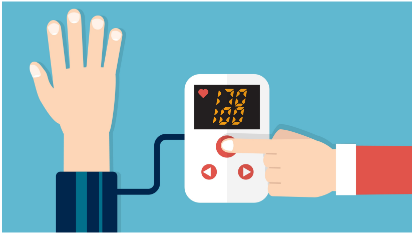
General Anesthesia

Breathing Mask/Intubation

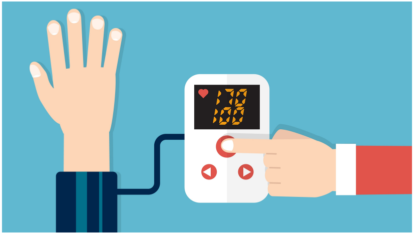


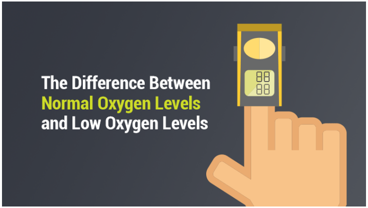
Intravenous (IV) line

Measuring the Depth of Anesthesia

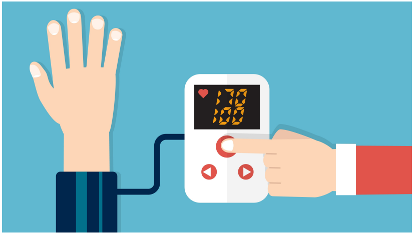


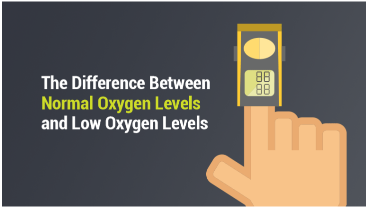
Measuring the Depth of Anesthesia



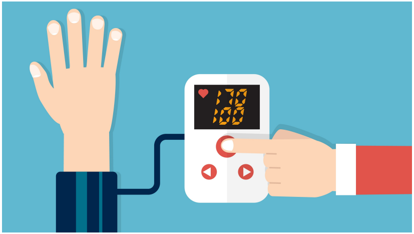


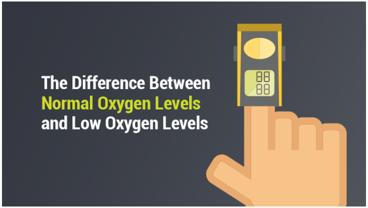
Measuring the Depth of Anesthesia

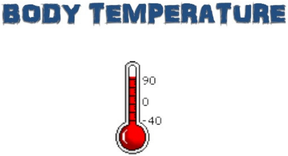




Measuring the Depth of Anesthesia

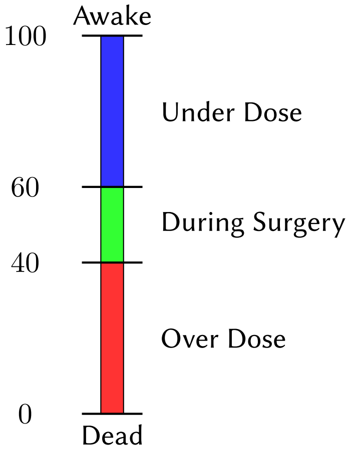






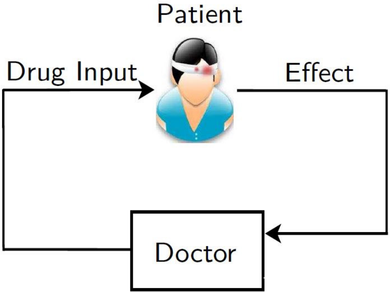
Bispectral Index (BIS)

- Depth of Anesthesia
- Scale $0 - 100$

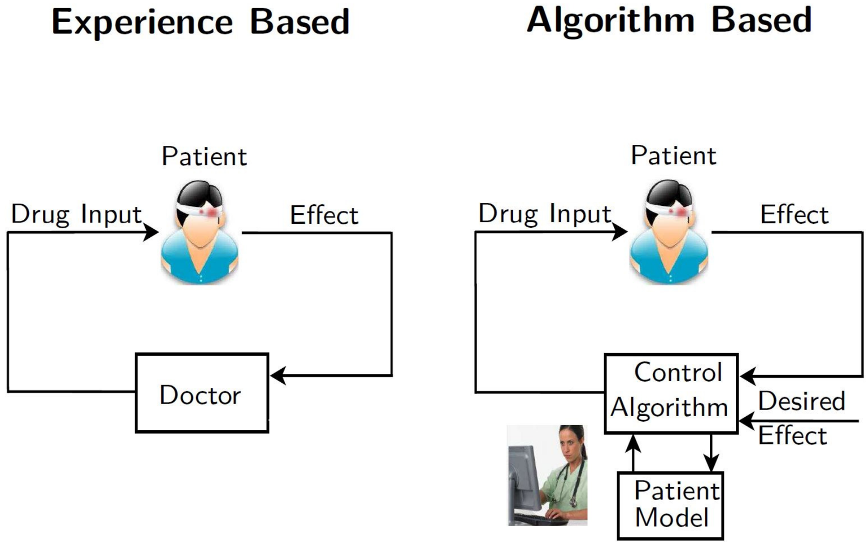


Drug Administration Strategies

Experience Based



Drug Administration Strategies



Available Control Schemes

- Proportional-Integral-Derivative (PID)
  1. Simplest Controller
- Model Predictive Control (MPC)
  1. Model based control
  2. Constraint handling
- Deep Neural Network based MPC
  1. Robustness to uncertainties in model
  2. Less Computational Complexity as compared to MPC

Available Control Schemes

- Proportional-Integral-Derivative (PID)
  1. Simplest Controller
- Model Predictive Control (MPC)
  1. Model based control
  2. Constraint handling
- Deep Neural Network based MPC
  1. Robustness to uncertainties in model
  2. Less Computational Complexity as compared to MPC

Available Control Schemes

- Proportional-Integral-Derivative (PID)
  1. Simplest Controller
- Model Predictive Control (MPC)
  1. Model based control
  2. Constraint handling
- Deep Neural Network based MPC
  1. Robustness to uncertainties in model
  2. Less Computational Complexity as compared to MPC

Available Control Schemes

- Proportional-Integral-Derivative (PID)
    1. Simplest Controller
- Model Predictive Control (MPC)
    1. Model based control
    2. Constraint handling
- Deep Neural Network based MPC
    1. Robustness to uncertainties in model
    2. Less Computational Complexity as compared to MPC

Available Control Schemes

- Proportional-Integral-Derivative (PID)
    1. Simplest Controller
- Model Predictive Control (MPC)
    1. Model based control
    2. Constraint handling
- Deep Neural Network based MPC
    1. Robustness to uncertainties in model
    2. Less Computational Complexity as compared to MPC

Available Control Schemes

- Proportional-Integral-Derivative (PID)
  1. Simplest Controller
- Model Predictive Control (MPC)
  1. Model based control
  2. Constraint handling
- Deep Neural Network based MPC
  1. Robustness to uncertainties in model
  2. Less Computational Complexity as compared to MPC

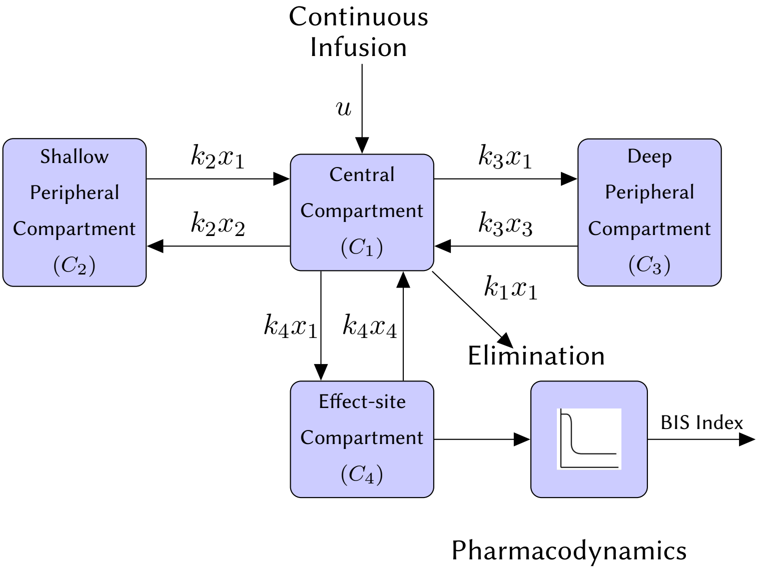
Available Control Schemes

- Proportional-Integral-Derivative (PID)
  1. Simplest Controller
- Model Predictive Control (MPC)
  1. Model based control
  2. Constraint handling
- Deep Neural Network based MPC
  1. Robustness to uncertainties in model
  2. Less Computational Complexity as compared to MPC

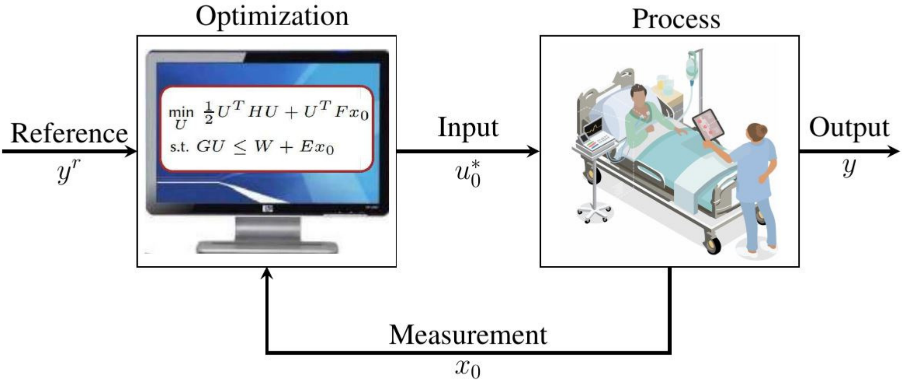
Available Control Schemes

- Proportional-Integral-Derivative (PID)
    1. Simplest Controller
- Model Predictive Control (MPC)
    1. Model based control
    2. Constraint handling
- Deep Neural Network based MPC
    1. Robustness to uncertainties in model
    2. Less Computational Complexity as compared to MPC

PK-PD Model of the Patient



Model Predictive Control (MPC)



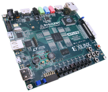
Use a dynamical model of the process to predict its future evolution and choose the "best" control action

Embedded Control Systems: Main Requirements

- Model-based, optimal performance, constraints

- Control law simple enough for software certification

- Require simple hardware (e.g., cheap and fast FPGA)



- Speed: fast dynamics $\Rightarrow$short sample time ($< 1\ \mu s$)

- Well defined worst-case execution time, to certify hard real-time system properties

Main Drawbacks of On-line Optimization

- ☺ Excellent QP solvers available today

but ...

- ☹ Computation time may be too long
- ☹ Requires relatively expensive hardware
- ☹ Cannot handle uncertainties in the model
- ☹ Not suitable of safety critical applications

Deep Neural Networks

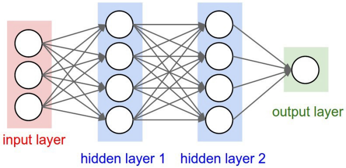
How it works ?

A neural network consists of highly connected networks of neurons that relate the inputs to the desired outputs. The network is trained by iteratively modifying the strengths of the connections so that given inputs map to the correct response.

Deep Neural Networks

Best used for:

- For modeling highly nonlinear systems
- When data is available incrementally and you wish to constantly update the model
- When there could be unexpected changes in your input data



Deep Neural Networks based MPC

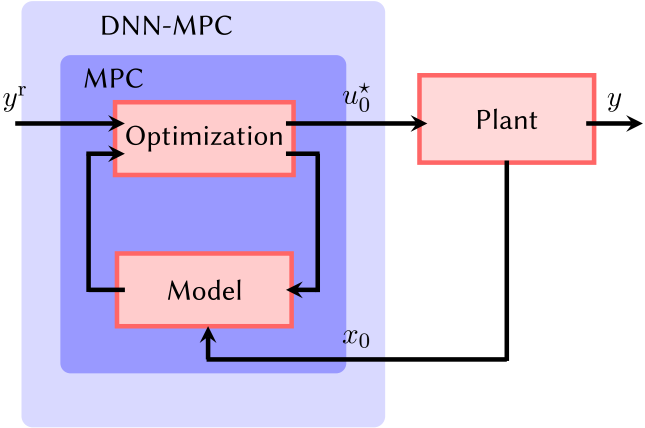
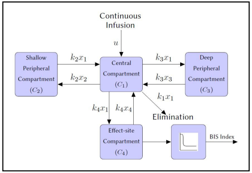
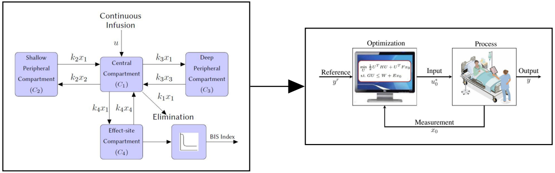


Figure: Replacing MPC with deep neural network trained model.

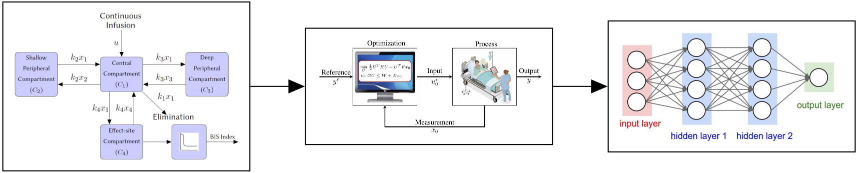
Implementation of a Real-time Explicit MPC

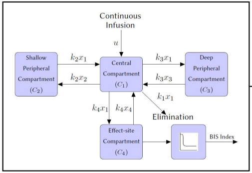


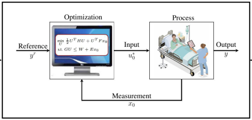
Implementation of a Real-time Explicit MPC

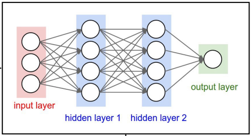


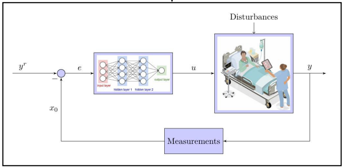
Implementation of a Real-time Explicit MPC




Continuous
Infusion
u
Shallow
Peripheral
Compartment
(C_2)
k_2x_1
k_2x_2
Central
Compartment
(C_1)
k_3x_1
k_3x_3
Deep
Peripheral
Compartment
(C_3)
k_1x_1
k_4x_1
k_4x_4
Elimination
Effect-site
Compartment
(C_4)
BIS Index


Optimization
Process
Reference
y^r
Input
u_0^*
Output
y
Measurement
x_0


input layer
hidden layer 1
hidden layer 2
output layer

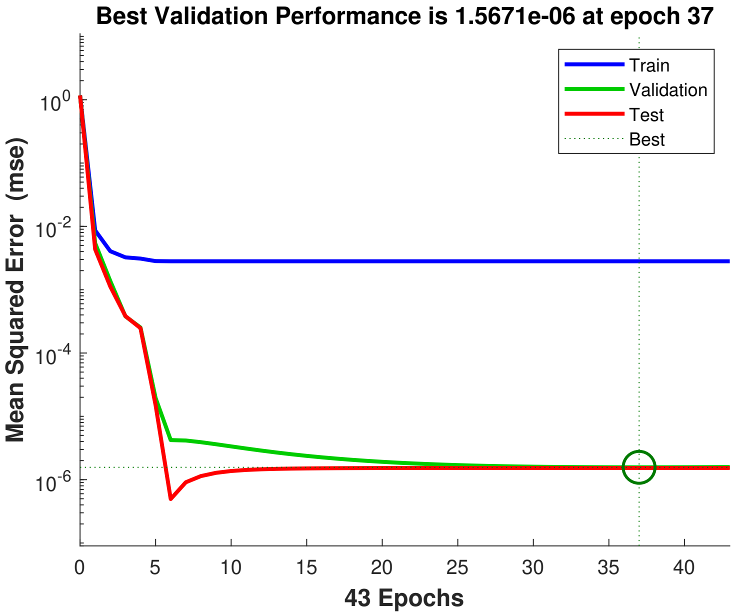

Disturbances
y^r
e
u
y
x_0
Measurements

Training of the Controller

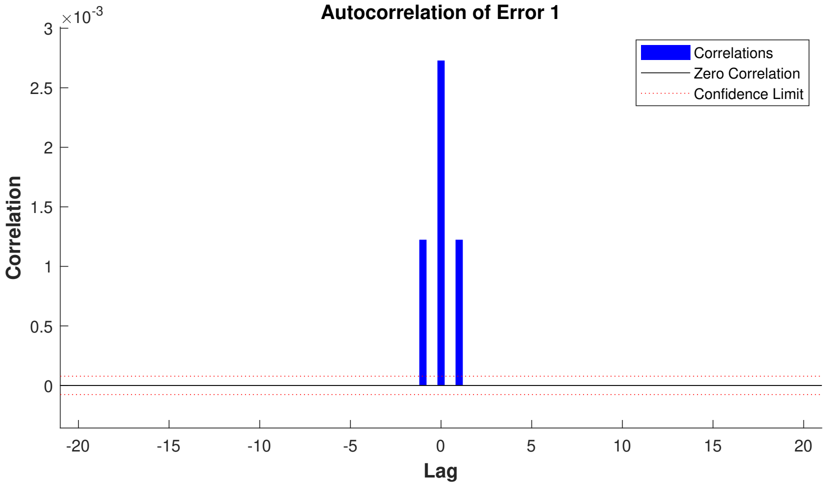
Scenario	Target Values	MSE
Training	35000	6.28241×10^{-9}
Validation	7500	1.14839×10^{-6}
Testing	7500	1.40538×1^{-8}

Table: Training statistics with 50000 data-points in which 70% was allocated for training while 15% each for testing and validation

Training Performance

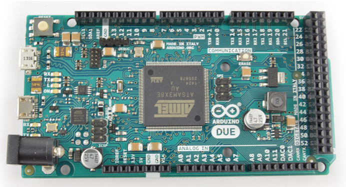


Training Auto-Correlation

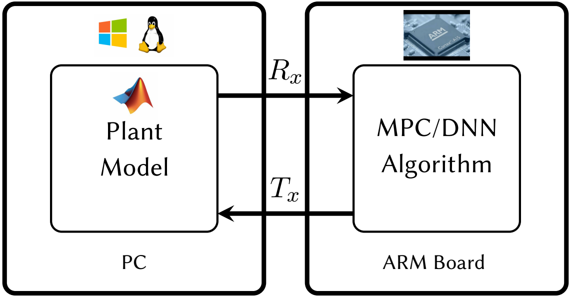


Atmel's ARM Cortex-M3:

- 32-bit microcontroller running at 84MHz.
- 512kB of program memory and 96kB of SRAM.
- Mostly prefer in embedded automotive applications.



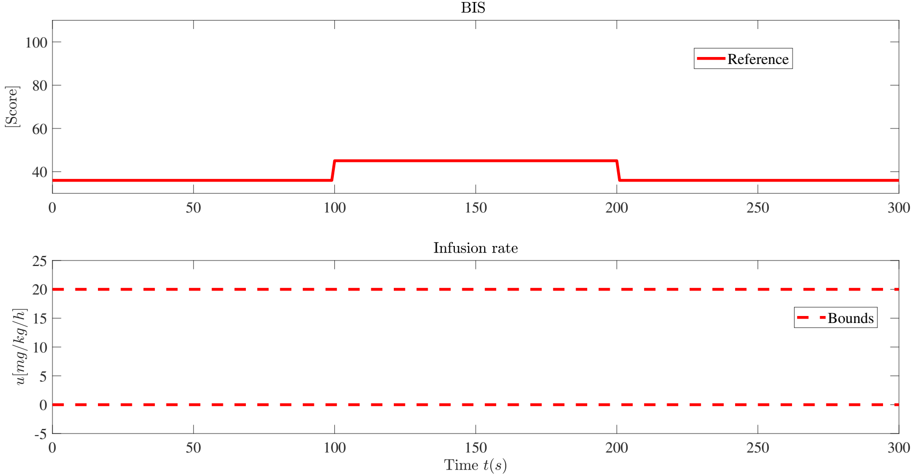
Hardware-in-loop (HIL) Work Flow



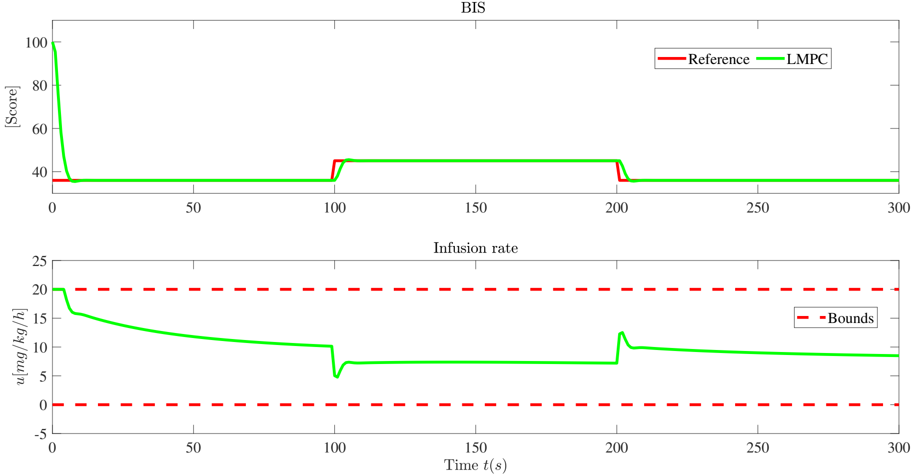
Case Study

- Patient of age 25 years and 63 kg weight
- BIS between $40 - 60$
- Input constraints $0 - 20$ mg/kg/h

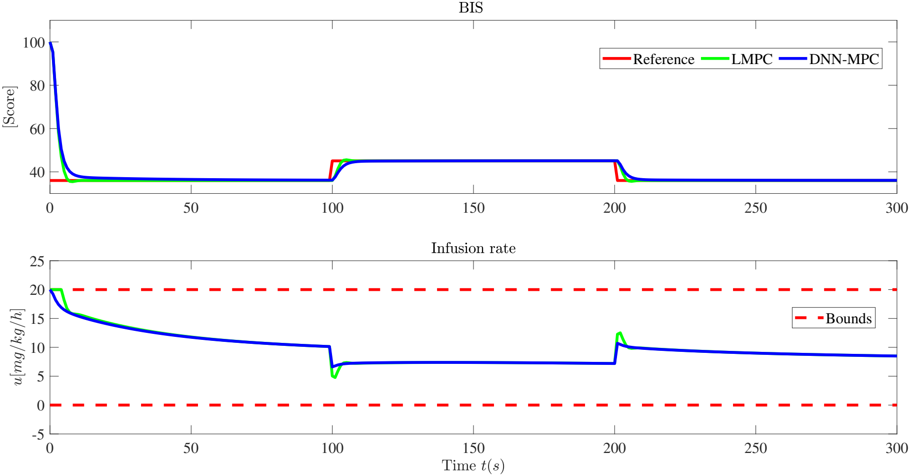
HIL Simulations Results



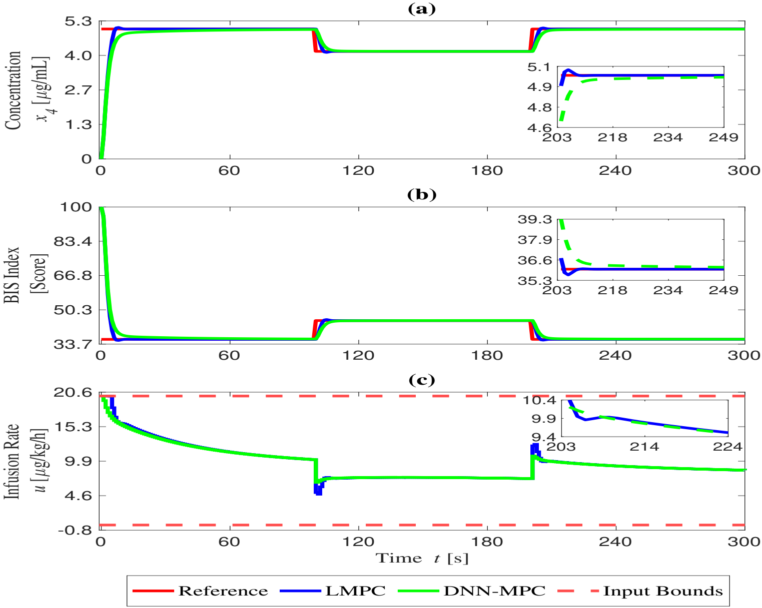
HIL Simulations Results



HIL Simulations Results



HIL Simulations Results



Computational Time and Memory Calculations

Controller	Memory [%]		Run-time
	% Data Usage	**% Program Usage**	**[ms]**
Linear MPC	3.50	4.90	11.354
DNN MPC	3.01	3.92	2.999

Table: Memory footprints and run-time results of proposed DNN-MPC and LMPC.

Conclusion

- This work proposes the use of deep neural networks to approximate linear MPC control law, efficiently and effectively with minimal computational efforts for real-time embedded implementation.
- Computational time reduced to as much as $4\times$ times
- Promising approach for safety-critical applications running on embedded systems.
- Future work will include to make complex non-linear MPC solutions using deep neural networks to use them on embedded hardware.

Publications

[1] S. Adhau, S. Patil, D. Ingole, and D. Sonawane, "Embedded Implementation of Deep Learning-based Linear Model Predictive Control," *(Submitted)*.

[2] S. Adhau, S. Patil, D. Ingole, and D. Sonawane, “Implementation and Analysis of Nonlinear Model Predictive Controller on Embedded Systems for Real-Time Applications,” in *2019 European Control Conference (ECC)*, Naples, Italy. IEEE, 2019.

[3] S. Adhau, K. Phalke, A. Nalawade, D. Ingole, S. Patil and D. Sonawane, "Implementation and Analysis of Offset-Free Explicit Model Predictive Controller on FPGA," in *2019 Fifth Indian Control Conference (ICC)*, New Delhi, India, 2019, pp. 231-236.

[4] S. Adhau, S. Dani, D. Ingole, S. Patil and D. Sonawane, "Embedded Model Predictive Controller on Low-Cost Low-End Microcontroller for Electrical Drives," *in 2018 I2CT*, India, 2018.

References

[1] S. Lucia, D. Navarro, B. Karg, H. Sarnago, and O. Lucia, “Deep Learning-Based Model Predictive Control for Resonant Power Converters,” *arXiv preprint arXiv:1810.04872*, 2018..

[2] U. Rosolia and F. Borrelli, “Learning model predictive control for iterative tasks. a data-driven control framework,” *IEEE Transactions on Automatic Control*, pp. 1883–1896, 2017.

[3] Y. Sawaguchi, E. Furutani, G. Shirakami, M. Araki, and K. Fukuda, “A model-predictive hypnosis control system under total intravenous anesthesia,” *IEEE transactions on biomedical engineering*, pp. 874– 887, 2008.

[4] D. Ingole and M. Kvasnica, “FPGA implementation of explicit model predictive control for closed loop control of depth of anesthesia,” *IFAC-PapersOnLine*, pp. 483–488, 2015.

Back-up Slides

PK Model

$$A = \begin{bmatrix} -\dfrac{k_1 + k_2 + k_3 + k_4}{V_1} & \dfrac{k_2}{V_1} & \dfrac{k_3}{V_1} & \dfrac{k_4}{V_1} \\ \dfrac{k_2}{V_2} & -\dfrac{k_2}{V_2} & 0 & 0 \\ \dfrac{k_3}{V_3} & 0 & -\dfrac{k_3}{V_3} & 0 \\ \dfrac{k_4}{V_4} & 0 & 0 & -\dfrac{k_4}{V_4} \end{bmatrix},$$

$$B = \begin{bmatrix} \dfrac{1}{V_1} & 0 & 0 & 0 \end{bmatrix}^T, \; C = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T, \text{ and } D = \begin{bmatrix} 0 \end{bmatrix}.$$

Model Parameters

Parameter	Value
k_1	$0.0595^{0.75}$ L/min (age ≤ 60) $(0.0595\text{BW}^{0.75} - 0.45\text{AGE} + 2.7)$ L/min (age > 60)
k_2	$0.0969\text{BW}^{0.62}$ L/min
k_3	$0.0889\text{BW}^{0.55}$ L/min
k_4	0.12 L/min
V_1	$1.72\text{BW}^{0.71}\text{AGE}^{-0.39}$ L
V_2	$3.32\text{BW}^{0.61}$ L
V_3	266 L
V_4	$0.01V_1$ L