CSME 2022 MLC Workshop

Machine Learning Control for Engineering Applications

C. R. Koch / M. Shahbakhti / A. Norouzi

2022-06-06

Workshop Presenters

1. Bob Koch, Professor Mechanical Engineering - ML Overview
   - bob.koch@ualberta.ca
2. Mahdi Shahbakhti, Associate Professor Mechanical Engineering - MPC Overview
   - mahdi@ualberta.ca
3. Armin Norouzi, PhD Candidate Mechanical Engineering - MLC Implementation
   - norouzi@ualberta.ca

C. R. (Bob) Koch

Mahdi Shahbakhti

Armin Norouzi

Workshop Overview [1]

Section 1 – Koch

Introduction to Machine Learning for Control – see also [1]

Section 2 – Shahbakhti

Model Predictive Control and Machine Learning Integration

Section 3 – Norouzi

Examples of Machine Learning Control Implementation

[1] References listed at the end

Workshop - Subject Area

Machine Learning Control for Engineering Applications

- Development of control-oriented dynamic models using Machine Learning (ML) techniques.
- Optimal, adaptive and model predictive control techniques that are
- Solved in integration with methods of machine learning including
  - neural networks, deep networks, and reinforcement learning
- Applications in broad linear and nonlinear engineering systems.

Graduate Course - Fall 2022

- MEC E 610 Machine Learning Control for Engineering Applications
- For University of Alberta Graduate students

General Background needed for MLC

Workshop

Only a very brief introduction to ML and MPC

- Motivation to combining MPC and ML
  - Briefly discuss ML, MPC and combining
  - Analysis and design: ideas so try it yourself in Matlab or Python
  - Some further resources for additional information

Helpful Background for Machine Learning Control

1. Undergraduate controls course (or 2) in classical control
2. Modern control
3. Linear algebra - vector spaces, SVD, eigenvalues, etc
4. Undergraduate course in statistics

Workshop Outline

Main Sections

1. Introduction to Machine Learning for Control (Koch)
   1. ML comparison to controls
   2. Methods of combining ML with Model Predictive Control (MPC)
   3. Overview of ML methods needed: ANN, RNN and LSTM, RL
2. MPC and ML integration (Shahbakhti)
3. Applications - implementation of MLC - some tools (Norouzi)
4. Further information and resources for ML

Machine Learning (ML)

- Machine learning is a method of data analysis that automates analytical model building.
- It is a branch of artificial intelligence based on the idea that systems can
  - Learn from data
  - Identify patterns and make decisions with minimal human intervention

Definitions [2]

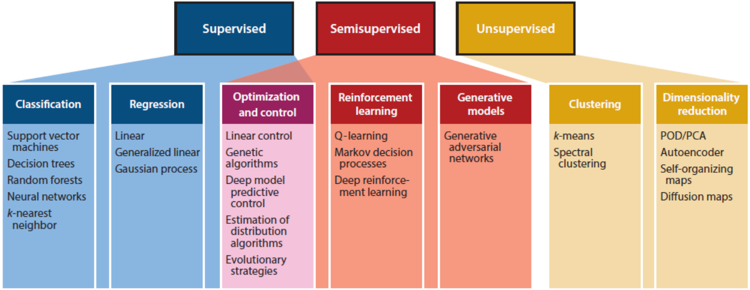
Machine learning algorithms that process and extract information from data; they facilitate automation of tasks and augment human domain knowledge

Supervised learning learning from data labeled with expert knowledge, providing corrective information to the algorithm

Semisupervised learning learning with partially labeled data or by interactions of the machine with its environment (reinforcement learning)

Unsupervised learning learning without labeled training data

Supervised,semi-supervised,unsupervised [2]



PCA - principle component Analysis, POD - proper orthogonal decomposition

Overview of Machine Learning for control

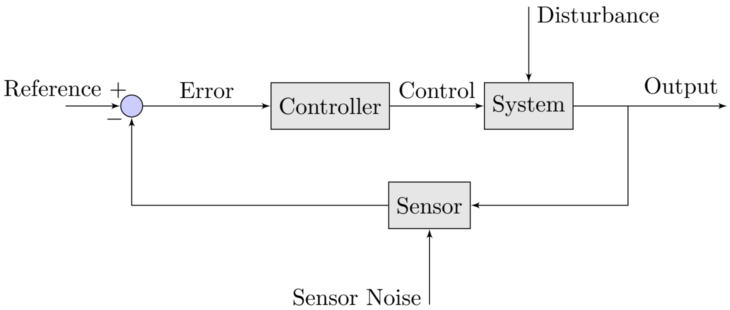
1. Artificial Neural Network (ANN) - models
2. Recurrent Neural Network (RNN) and Long Short Term Memory (LSTM) for sequential inputs
3. Reinforcement Learning (RL) - data driven (model free) control

Workshop Outline Part 1

Main Sections

1. Introduction to Machine Learning for Control (Koch)
   1. ML comparison to controls
   2. Methods of combining ML with Model Predictive Control (MPC)
   3. Overview of ML methods needed: ANN, RNN and LSTM, RL
2. MPC and ML Integration (Shahbakhti)
3. Applications - implementation of MLC - some tools (Norouzi)
4. Further information and resources for ML

Feedback Control



Advantages

- More robust to **disturbances**
- More robust to **parameter variation**
- Improve **transient response** (stabilize unstable systems)

Disadvantages

- Require a sensor
- System can go unstable

Workshop Outline Part 1

Main Sections

1. Introduction to Machine Learning for Control (Koch)
    1. ML comparison to controls
    2. Methods of combining ML with Model Predictive Control (MPC)
    3. Overview of ML methods needed: ANN, RNN and LSTM, RL
2. MPC and ML Integration (Shahbakhti)
3. Applications - implementation of MLC - some tools (Norouzi)
4. Further information and resources for ML

Why combine MPC and ML for "ML Control"

- Use a model for model based control
  - ML can help MPC
  - MPC can help ML
  - ML can replace the physical model to speed-up calculations for realtime
- Use the large amount of MPC knowledge and combine with ML to improve control
- Physical model and understanding can be used to incorporate "domain knowledge" into the control
- 5 categories below

Machine Learning (ML) and Model Predictive Control (MPC)

1. ML in model structure of MPC:
   - A machine learning-based model, is used to develop a model which is used directly to design MPC or implement optimization based on it
   - Depends on the structure of the learning algorithm, modeling can be done online or adaptive
2. ML in control structure of MPC: ML is combined in the controller structure
   - To enhance MPC structure in terms of stability, optimization accuracy, and computational cost
   - ML can be augmented inside the MPC or as an add-on controller
3. ML in imitation of MPC: In imitation of MPC, ML is used to mimic the MPC controller's behavior in a realtime
   - Can decrease the computation time decreases significantly
   - All optimization is done in the prototype system
   - An approximate function is deployed for the MPC in real-time
   - Depends on the structure of the learning algorithm, modeling can be done online or online.

Machine Learning (ML) and Model Predictive Control (MPC)

4. <u>ML in optimization of MPC</u>: ML methods are used to improve optimization accuracy and computational time
   - This optimization method can be employed in MPC.
5. <u>MPC for safe ML controller</u>: an MPC-based filter is used to guarantee the constraints satisfactory of ML-based controller
   - Pure learning-based algorithms, such as reinforcement learning, do not consider hard systems constraints

Workshop Outline Part 1

Main Sections

1. Introduction to Machine Learning for Control (Koch)
   1. ML comparison to controls
   2. Methods of combining ML with Model Predictive Control (MPC)
   3. Overview of ML methods needed: ANN, RNN and LSTM, RL
2. MPC and ML Integration (Shahbakhti)
3. Applications - implementation of MLC - some tools (Norouzi)
4. Further information and resources for ML

Motivation: ML for dynamic models

Describe when ML can be used for dynamic models

1. Offline learning modeling for MPC
2. Adaptive learning modeling for MPC
3. Disturbance and model mismatch

Main Modeling Methods

1. Artificial Neural Networks
    1. Shallow Neural Networks (SNN) and Nonlinear AutoRegressive eXogenous (NARX)
    2. Recurrent Neural Network (RNN) modeling, Long Short-Term Memory (LSTM)
    3. Deep Neural Networks (DNN)
2. Other ML approaches for System Identification (SI)
    1. Support Vector Machine (SVM)
    2. Regression Trees (RT) and Random Forest (RF)
    3. Gaussian Process Regression (GPRs)

Artificial Neural Network (ANN)

- A set of algorithms that try to distinguish the correlation between a set of data using rules thought to mimic human brain operation [3].
- An ANN includes simulated neurons where each node connects to other nodes in neurons through connections that match biological axon-synapse-dendrite joints.

Artificial Neural Network (ANN) Structure

- Each link has a weight, which manages the strength of one node's influence on another.
- The neurons are usually organized into multiple layers.
- The nodes that receive external data as input are the input layer;
  - the output layer produces the predicted output data.
  - In the middle, hidden layers exist between the input layer and output layer that can be varied in size and structure.

Artificial Neural Network (ANN) Schematic

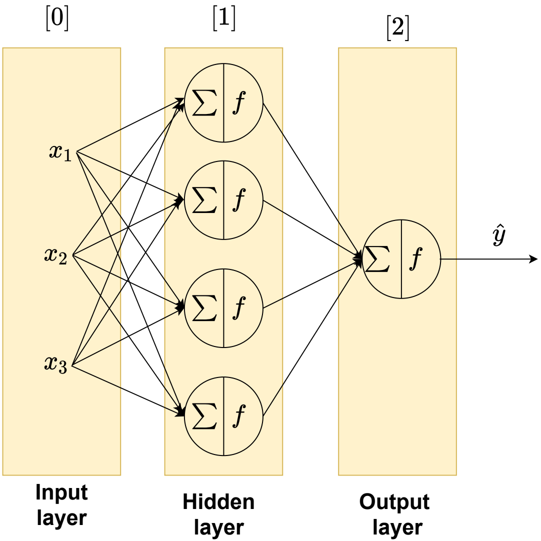


Figure: ANN Schematic of two layer ANN (one-hidden layer ANN) including three main layer

ANN Activation function $f(z)$

- for ANN, the activation function of a node defines the output of that node given an input or set of inputs
- However, only nonlinear activation functions allow ANN to give complex outputs using only a small number of nodes

Common examples

- Sigmoid activation function: $\sigma(z) = \frac{1}{1+e^{-z}}$
- Hyperbolic tangent: $\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$
- Rectified linear unit: ReLU: 0 if $z \leq 0, \quad z$ if $z > 0$

Gradient Decent (GD) for Neural Network

- the cost function is defined as

$$J(W^{[1]}, b^{[1]}, W^{[2]}, b^{[2]}) = \frac{1}{m}\sum_{i=1}^{m} L(\hat{Y}, Y)$$

- where $L(\hat{y}, y)$ is the loss function

  regression usually Mean Squared Error
  classification usually logistic regression

- the cost function can also have regularization

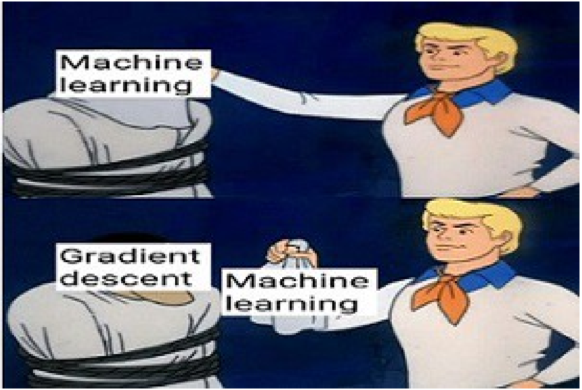
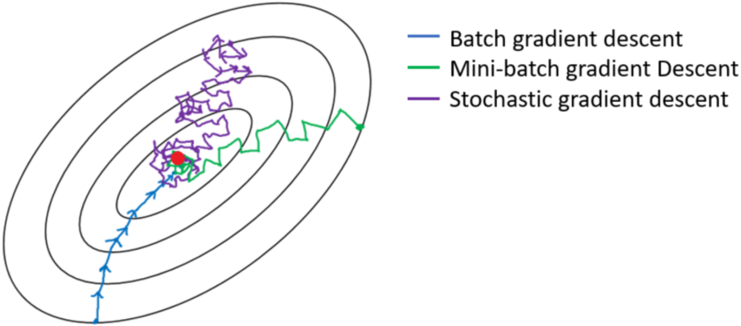


Figure: Machine Learning behind the scenes [4]

Gradient Descent Comparison: Batch, Stochastic and mini-Batch

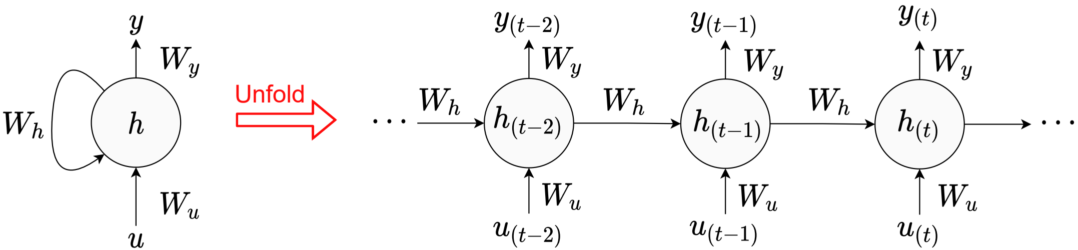
Gradient descent is used to train our model through back propagation

1. Batch Gradient Descent - all the data per step - local minimum
2. Mini Batch Gradient Descent - part of the data
3. Stochastic Gradient Descent - one data point at a time but slow



The Recurrent Neural Network (RNN)

- Has a similar structure to a feedforward ANN
- Has backward connections that are used to handle sequential inputs [6].
- The simplest RNN for time step t is shown in the Figure
- This recurrent neuron receives both inputs $u(t)$ and output from the previous time step, $y(t-1)$.
- The output of the first step is generally initialized as zero.
- The structure of RNN can be revealed by unrolling it in time, see Figure [6].
- The output of the recurrent neuron at the current time step t is a function of past inputs, so a recurrent neuron can be considered a memory [6].



RNN disadvantages are:

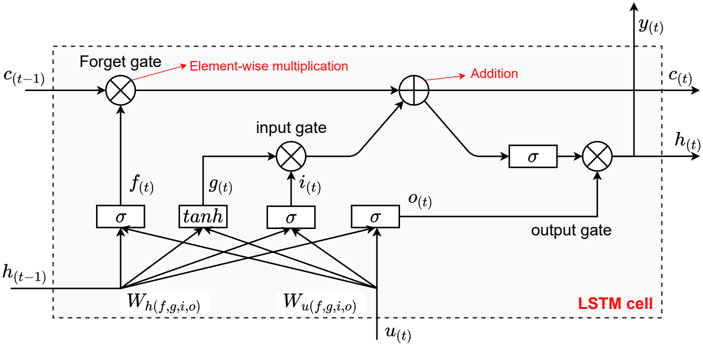
- Cannot capture long-term dependencies since RNN prediction is based on recent steps.
- RNN also suffers from "vanishing gradient", which means that
  - The contribution of earlier steps becomes increasingly small in the RNN gradient.
  - This ignores long-term dependencies during training.
- So various types of cells with long-term memory were introduced to tackle this problem.
  - The most popular and well-known of these long-term memory cells is the Long Short-Term Memory cell (LSTM).

Long Short-Term Memory (LSTM)

- when LSTM is compared to RNN, in LSTM the hidden state is split into two main part:
  - $h_{(t)}$ the short-term state, and $c_{(t)}$ the long-term state.

Long Short-Term Memory (LSTM)

- $h_{(t)}$ the short-term state, and $c_{(t)}$ the long-term state.
- The long-term state $c_{(t-1)}$ traverses the network from left to right.
  1. this state first enters the forget gate where past values are dropped
  2. then some values (memories) are added to the input gate to get c_t
  3. so at each time step, some memory is added, and some are dropped, and
  4. the current long-term state $c_{(t)}$ is output without further change.



Deep Neural Network (DNN)

- DNN using Fully Connected (FC) network often use a shallow neural network for modeling
- RNN to model dynamic of system for MPC application

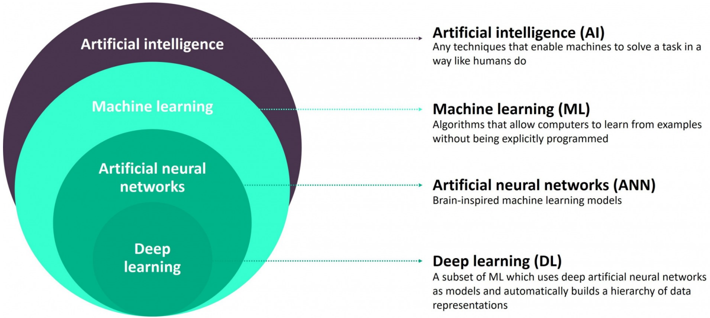
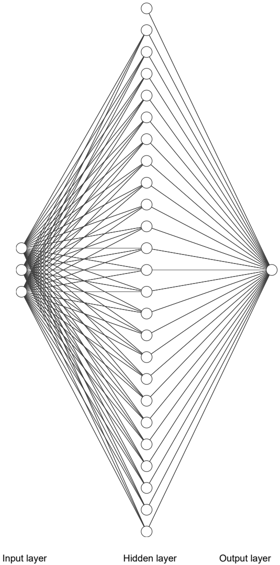


Figure: AI hierarchy with deep learning [7]

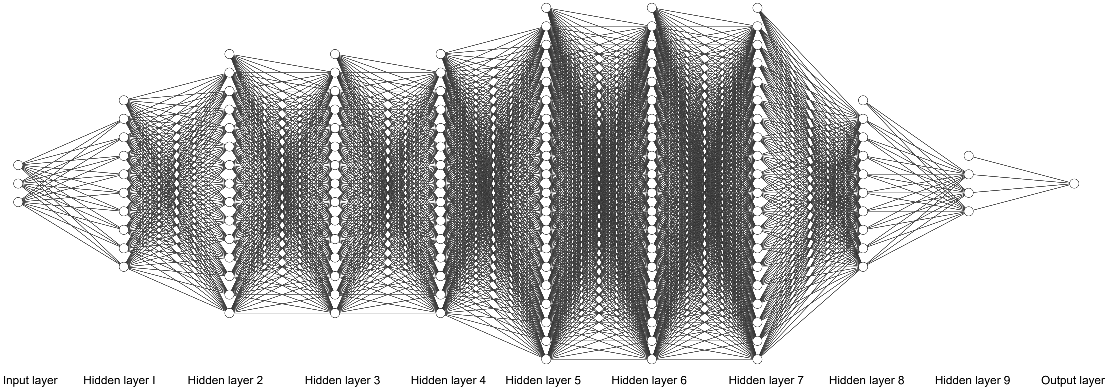
Deep Learning (DL) is a subset of ML

- DL uses a deep ANN as model (DNN) to automatically build a hierarchy of data representation – see Figure below
- So far the neural networks considered had one or two hidden layer
  - these are called shallow neural network
- To model complex systems, we need more learn parameter in our model. To do this we have two main options:
  1. A shallow network that has a low number of hidden layers.
     - While there are studies that a shallow network can fit any function [6]
     - it will need to be really tall - this causes a large increase in the number of parameters
  2. A deep network (wide) has been found to fit functions better with less parameters than a shallow network (tall) [6]

Example of Tall (shallow) Deep Neural Network (DNN)



Example of Wide Deep Neural Network (DNN)



Deep Neural Network (DNN) and CNN

- For example a deep network with 5 layers and 10 neurons in each layer may work more efficient than a single hidden layer network (shallow NN) with 1000 neurons.
- Broad application of DNN using Recurrent Neural Networks (RNNs) in which data can flow in any direction, such as language modeling and deep Convolutional Neural Networks (CNNs) which are used in computer vision.

DNN Advantages and Challenges

- DNN has advantages discussed above

Challenges of DNN:

1. related to training a much deeper ANN - many layers can have hundreds of neurons linked by hundreds of thousands of connections.
2. Vanishing and exploding gradients:
   - This is when the gradients grow smaller and smaller or larger and larger
   - when flowing backward through the DNN during training
   - Both vanishing and exploding gradients make lower layers very hard to train.
3. Not enough training data
4. Training may be extremely slow
5. A model with millions of parameters
   - at risk of severely overfitting the training set
   - especially if there are not enough training instances or if they are too noisy.

DNN Training Techniques

Vanishing/Exploding Gradients Solutions

- The signal must flow properly in both directions:
  - **Initialize** the variance of the outputs of each layer needs to be equal to the variance of its inputs,
  - in addition, need the gradients to have equal variance before and after flowing through a layer in the reverse direction.
- **activation function** needs to be a non-saturating function
- **Batch normalization** - scale and shift for each layer

Faster Optimizers

- can speed up with: Initialization strategy, proper activation function, Batch normalization

To significantly boost training speed use:

a faster optimizer than the regular Gradient Descent optimizer. Such as:

1. Gradient Descent With Momentum
2. Root Mean Squared Propagation (RMSProp)
3. Adam optimizer are used

Gradient Descent With Momentum

Gradient descent problems

- bounces around the search space when
  - optimization problems that have large amounts of curvature or noisy gradients,
- can get stuck in flat spots in the search space that have no gradient.

Momentum is an extension to GD optimization algorithm to allow

- the search to build inertia in a direction in the search space
- overcome the oscillations of noisy gradients
- "coast" across flat spots of the search space.

The Algorithm

- accumulates moving average of past gradients and move in that direction,
- while exponentially decaying.

The Momentum method accelerates learning, when:

- Facing high curvature
- Small but consistent gradients
- Noisy gradients
- Examples of optimizers come later

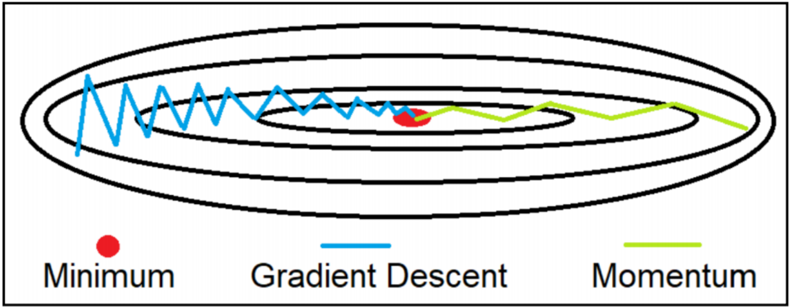


Figure: Gradient Descent vs Gradient Descent With Momentum [8]

Root Mean Squared Propagation, or RMSProp

RMSProp, is an extension of gradient descent that

- uses a decaying average of partial gradients in the adaptation of the step size for each parameter
- uses a decaying moving average allows the algorithm to forget early gradients
- focuses on the most recently observed partial gradients seen during the progress of the search.

Adam Optimization (adaptive moment estimation)

Adam combines the ideas of momentum optimization and RMSProp

- just like momentum optimization, it keeps track of an exponentially decaying average of past gradients
- just like RMSProp, it keeps track of an exponentially decaying average of past squared gradients.

RT implementation of LSTM-NMPC: Model in/outputs [2]

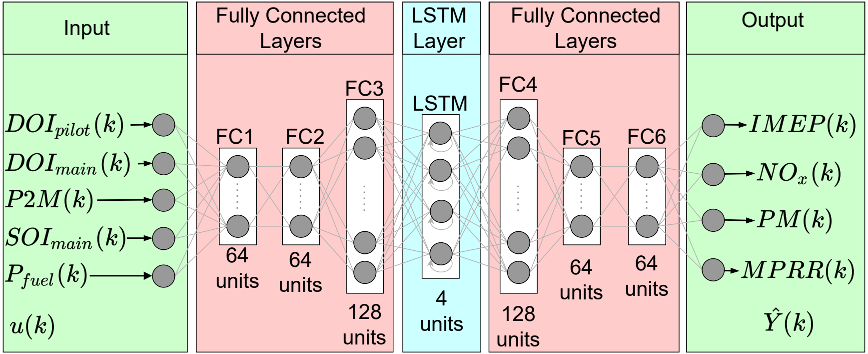


Figure: Structure of proposed deep neural network model for engine performance and emission modeling. LSTM: Long-short term memory, SOI: start of injection, DOI: duration of injection, P_{fuel}: fuel rail pressure, IMEP: indicated mean effective pressure, MPRR: Maximum pressure rise rate, PM: particle matter, t_{P2M}: duration between end of pilot injection and start of main injection

[2] Deep Learning and Nonlinear Model Predictive Control Integration for Compression Ignition Engine Emission Reduction, Norouzi,Shahpouri, Gordon, (Winkler,Nuss Andert RWTH), Shahbakhti, Koch: Monday 12:00

RT implementation of LSTM-NMPC- schematic

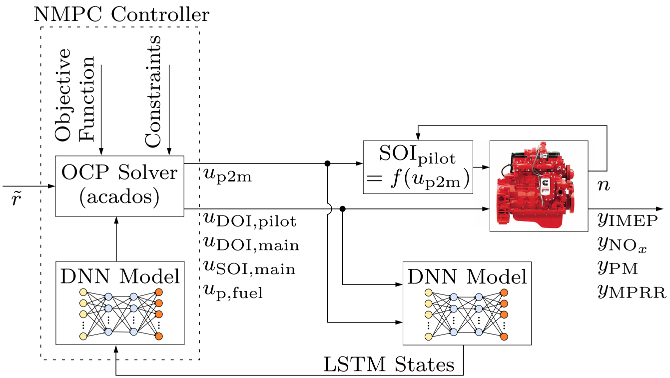
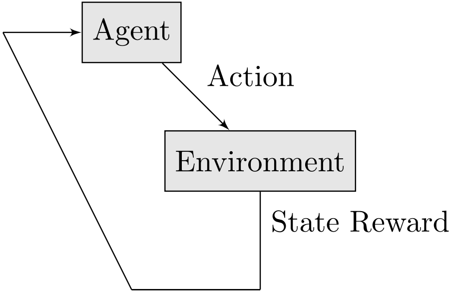


Figure: Block diagram of LSTM-NMPC structure– IMEP: Indicated mean effective pressure, NO_x: Nitrogen Oxide, PM: Particle matter, MPRR: Maximum pressure rise rate, *n*:Engine speed, DOI: Duration of injection, SOI: Start of injection

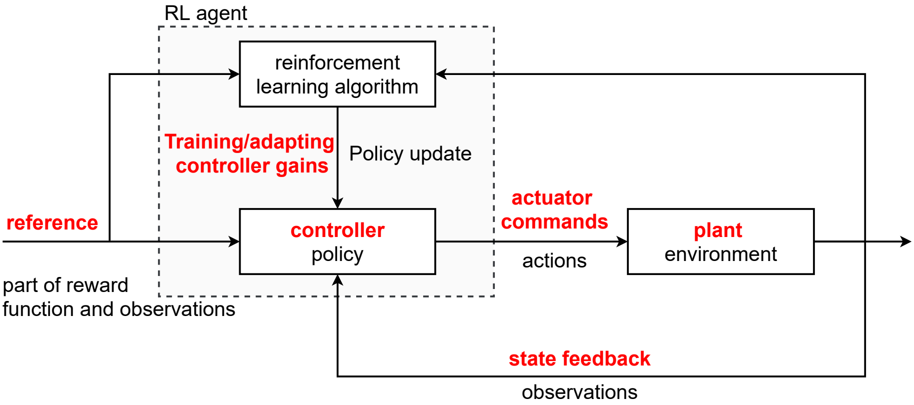
Reinforcement Learning

Reinforcement Learning (RL) is

- one of the most exciting fields of Machine Learning today
- also one of the oldest since it has been around since the 1950s
- some interesting applications over the years
  - particularly in games (e.g., TD-Gammon, a Backgammon-playing program)
  - in machine control, but not as publicized in the common press



Reinforcement learning compared to controls



.

RL Concept

- Often, many trials needed. RL for highly constrained engineering systems?[9] [3]

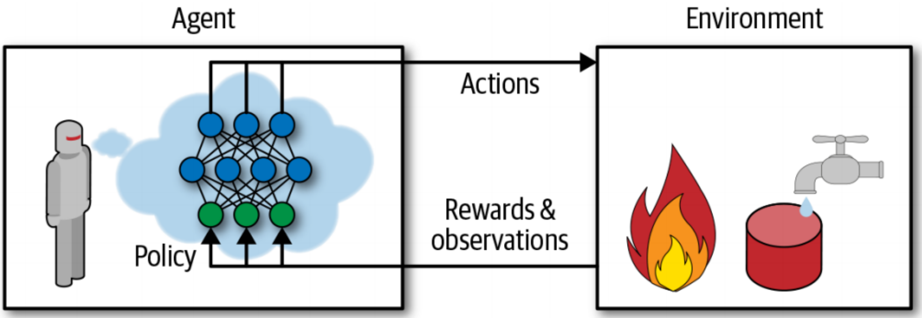
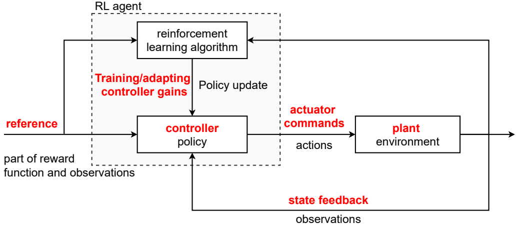
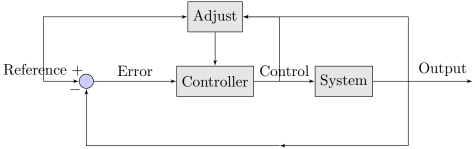


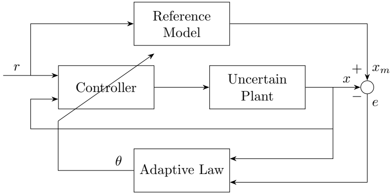
Figure: Reinforcement Learning using a neural network policy

[3] Sutton, RL book Chap 9, One of the most pressing areas for future reinforcement learning research is to adapt and extend methods developed in control engineering with the goal of making it acceptably safe to fully embed reinforcement learning agents into physical environments

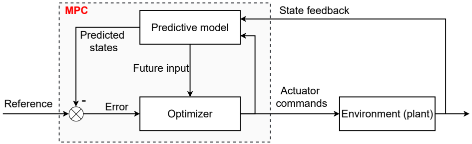
RL compared to feedback control - similar structure



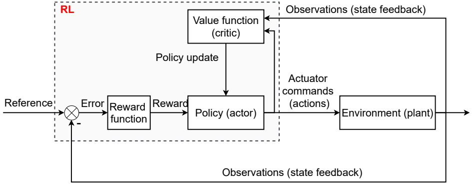




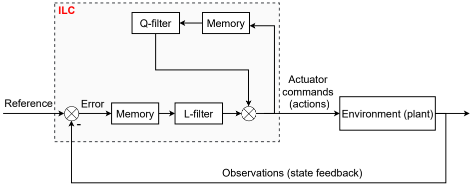
(a) Model Predictive Control (MPC)



(b) Reinforcement Learning (RL)



(c) Iterative Learning Control (ILC)



Workshop Outline Part 2 - Presentation - Shahbakhti

Main Sections

1. Introduction to Machine Learning for Control (Koch)
    1. ML comparison to controls
    2. Methods of combining ML with Model Predictive Control (MPC)
    3. Overview of ML methods needed: ANN, RNN and LSTM, RL
2. MPC and ML Integration (Shahbakhti)
3. Applications - implementation of MLC - some tools (Norouzi)
4. Further information and resources for ML

Workshop Outline Part 3 - Presentation - Norouzi

Main Sections

1. Introduction to Machine Learning for Control (Koch)
    1. ML comparison to controls
    2. Methods of combining ML with Model Predictive Control (MPC)
    3. Overview of ML methods needed: ANN, RNN and LSTM, RL
2. MPC and ML integration (Shahbakhti)
3. Applications - implementation of MLC - some tools (Norouzi)
4. Further information and resources for ML

Workshop Outline Part 4 - Presentation - Further Resources

Main Sections

1. Introduction to Machine Learning for Control (Koch)
   1. ML comparison to controls
   2. Methods of combining ML with Model Predictive Control (MPC)
   3. Overview of ML methods needed: ANN, RNN and LSTM, RL
2. MPC and ML integration (Shahbakhti)
3. Applications - implementation of MLC - some tools (Norouzi)
4. Further information and resources for ML

Computer tools

Matlab

- very powerful software for controls
- also good for ML

Python

- open source and very good for ML
- not as good for control
- can Jupyter Notebooks
  - install on your machine (Anaconda)
  - run in a Web Browser: Google Colab
  - many Universities have a Jupyter portal, for example UAlberta

Online Courses - ML - Intro

1. Machine Learning by Stanford University

- Link: Coursera
- Level: Machine learning mathematics included **Good for understanding insight of machine Learning**
- Language: Octave or Matlab

2. Python: 2020 Complete Python Bootcamp: From Zero to Hero in Python

- Link: Udemy
- Level: Elementary to advanced level
- Language: Python

Online Courses - ML - Implement and RL

3. Machine Learning with Python by IBM

- Link: Coursera
- Level: **The best and easiest course to start implementing machine learning** (mathematics behind the machine learning are not included)
- Language: Python

4. Reinforcement Learning Specialization by University of Alberta and Alberta Machine Intelligence Institute (AMII)

- Link: Coursera
  - Fundamentals of Reinforcement Learning
  - Sample-based Learning Methods
  - Prediction and Control with Function Approximation
  - A Complete Reinforcement Learning System (Capstone)
- Deep mathematics of algorithm have been covered based on RL reference Book [9] - **Good for understanding insight of RL**

Online Courses - Deep Learning

5. Deep Learning Specialization by deeplearning.ai

- Link: Coursera
  - Neural Networks and Deep Learning
  - Improving Deep Neural Networks: Hyperparameter tuning, Regularization and Optimization
  - Structuring Machine Learning Projects
  - Convolutional Neural Networks
  - Sequence Models
- Language: Python and in TensorFlow

Online Courses - RL - Matlab and Data Science

6. Reinforcement Learning Toolbox User's Guide by Matlab

- Link: MathWorks
- Level: Includes some examples in Matlab- requirements: 1 and 4
- Language: Matlab

7. IBM Data Science Professional Certificate

- Link: Coursera
  - What is Data Science?
  - Tools for Data Science
  - Data Science Methodology
  - Python for Data Science and AI
  - Databases and SQL for Data Science
  - Data Analysis with Python
  - Data Visualization with Python
  - Machine Learning with Python (4 above)
  - Applied Data Science Capstone

Possible online resources - summary

Extra Information about ML - Possible Study Plan

- If Machine Learning and Python are new for you, you should start from 1 and 2
- If you know machine learning but want to implement ML in Python, you should start from 3
- If your are interested in RL and Deep Learning, you should take 4 and 5
- If you know about RL and you want to implement RL Matlab, you should check out 6
- If your are interested in Data science, you should take 7

[1] Steven L. Brunton and J. Nathan Kutz. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press, 2019. DOI: 10.1017/9781108380690.

[2] Steven L. Brunton, Bernd R. Noack, and Petros Koumoutsakos. "Machine Learning for Fluid Mechanics". In: *Annual Review of Fluid Mechanics* 52.1 (Jan. 2020), pp. 477–508. DOI: 10.1146/annurev-fluid-010719-060214.

[3] Mohamad H Hassoun et al. *Fundamentals of artificial neural networks*. MIT press, 1995.

[4] meme. *Machine Learning behind the scenes*. URL: https://me.me/i/machine-learning-gradient-descent-machine-learning-machine-learning-behind-the-ea8fe9fc64054eda89232d7ffc9ba60e. (accessed: 08.31.2021).

[5] Analytics Vidhya. *Gradient Descent vs Stochastic GD vs Mini-Batch SGD*. URL: https://medium.com/analytics-vidhya/gradient-descent-vs-stochastic-gd-vs-mini-batch-sgd-fbd3a2cb4ba4. (accessed: 08.31.2021).

[6] Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, 2019.

[7] BBN times. *Artificial Intelligence*. URL: https://www.bbntimes.com/science/artificial-intelligence-vs-machine-learning-vs-artificial-neural-networks-vs-deep-learning. (accessed: 08.31.2021).

[8] Cedar buffalo. *Gradient Descent vs Gradient Descent with Momentum*. URL: \https://cedar.buffalo.edu/~srihari/CSE676/8.3\%20BasicOptimizn.pdf. (accessed: 08.31.2021).

[9] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction. Chapter 9*. MIT press, 2018.

[10] Changyeob Shin et al. "Autonomous tissue manipulation via surgical robot using learning based model predictive control". In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE. 2019, pp. 3875–3881.

[11] Yunduan Cui, Shigeki Osaki, and Takamitsu Matsubara. "Reinforcement Learning Boat Autopilot: A Sample-efficient and Model Predictive Control based Approach.". In: *IROS*. 2019, pp. 2868–2875.

[12] Farbod Farshidian, David Hoeller, and Marco Hutter. "Deep Value Model Predictive Control". In: *arXiv preprint arXiv:1910.03358* (2019).

[13] Olov Andersson et al. "Model-predictive control with stochastic collision avoidance using bayesian policy optimization". In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2016, pp. 4597–4604.

[14] Colin Greatwood and Arthur G Richards. "Reinforcement learning and model predictive control for robust embedded quadrotor guidance and control". In: *Autonomous Robots* 43.7 (2019), pp. 1681–1693.

[15] David D Fan, Ali-akbar Agha-mohammadi, and Evangelos A Theodorou. "Deep learning tubes for tube MPC". In: *arXiv preprint arXiv:2002.01587* (2020).

[16] Karnchanachari Napat et al. "Practical Reinforcement Learning For MPC: Learning from sparse objectives in under an hour on a real robot". In: *2nd Annual Conference on Learning for Dynamics and Control (L4DC 2020)*. 2020.

[17] Hitesh Shah and M Gopal. "Model-free predictive control of nonlinear processes based on reinforcement learning". In: *IFAC-PapersOnLine* 49.1 (2016), pp. 89–94.

[18] Mohammad Babaie et al. "Supervised Learning Model Predictive Control Trained by ABC Algorithm for Common Mode Voltage Suppression in NPC Inverter". In: *IEEE Journal of Emerging and Selected Topics in Power Electronics* (2020).

[19] Muhammad Saleh Murtaza Gardezi and Ammar Hasan. "Machine learning based adaptive prediction horizon in finite control set model predictive control". In: *IEEE Access* 6 (2018), pp. 32392–32400.

[20] Xuechao Wang et al. "Hierarchical model predictive control via deep learning vehicle speed predictions for oxygen stoichiometry regulation of fuel cells". In: *Applied Energy* 276 (2020), p. 115460.

[21] Alex S Ira et al. "A machine learning approach for tuning model predictive controllers". In: *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. IEEE. 2018, pp. 2003–2008.

[22] Stéphanie Lefevre, Ashwin Carvalho, and Francesco Borrelli. "Autonomous car following: A learning-based approach". In: *2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. 2015, pp. 920–926.

[23] Stéphanie Lefevre, Ashwin Carvalho, and Francesco Borrelli. "A learning-based framework for velocity control in autonomous driving". In: *IEEE Transactions on Automation Science and Engineering* 13.1 (2015), pp. 32–42.

[24] Martina Joševski and Dirk Abel. "Tube-based MPC for the energy management of hybrid electric vehicles with non-parametric driving profile prediction". In: *2016 American Control Conference (ACC)*. IEEE. 2016, pp. 623–630.

[25] Vivek Mahalingam and Abhishek Agrawal. "Learning agents based intelligent transport and routing systems for autonomous vehicles and their respective vehicle control systems based on model predictive control (MPC)". In: *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*. IEEE. 2016, pp. 284–290.

[26] Mohammad Rokonuzzaman et al. "Learning-based Model Predictive Control for Path Tracking Control of Autonomous Vehicle". In: *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2020, pp. 2913–2918.

[27] Chaoyang Jiang et al. "Learning based Predictive Error Estimation and Compensator Design for Autonomous Vehicle Path Tracking". In: *2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA)*. IEEE. 2020, pp. 1496–1500.

[28] Bingzhao Gao et al. "Personalized Adaptive Cruise Control Based on Online Driving Style Recognition Technology and Model Predictive Control". In: *IEEE Transactions on Vehicular Technology* (2020).

[29] Árpád Fehér, Szilárd Aradi, and Tamás Bécsi. "Hierarchical Evasive Path Planning Using Reinforcement Learning and Model Predictive Control". In: *IEEE Access* 8 (2020), pp. 187470–187482.

[30] Maximilian Brunner et al. "Repetitive learning model predictive control: An autonomous racing example". In: *2017 IEEE 56th annual conference on decision and control (CDC)*. IEEE. 2017, pp. 2545–2550.

[31] Cheng-Yu Kuo, Yunduan Cui, and Takamitsu Matsubara. "Sample-and-computation-efficient Probabilistic Model Predictive Control with Random Features". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2020, pp. 307–313.

[32] Yunduan Cui, Shigeki Osaki, and Takamitsu Matsubara. "Autonomous boat driving system using sample-efficient model predictive control-based reinforcement learning approach". In: *Journal of Field Robotics* (2020).

[33] Björn Lautenschlager and Gerwald Lichtenberg. "Data-driven iterative learning for model predictive control of heating systems". In: *IFAC-PapersOnLine* 49.13 (2016), pp. 175–180.

[34] Andrea Carron et al. "Data-driven model predictive control for trajectory tracking with a robotic arm". In: *IEEE Robotics and Automation Letters* 4.4 (2019), pp. 3758–3765.

[35] Lukas Hewing, Juraj Kabzan, and Melanie N Zeilinger. "Cautious model predictive control using Gaussian process regression". In: *IEEE Transactions on Control Systems Technology* (2019).

[36] Juraj Kabzan et al. "Learning-based model predictive control for autonomous racing". In: *IEEE Robotics and Automation Letters* 4.4 (2019), pp. 3363–3370.

[37] Defeng He and Binbin Peng. "Gaussian learning-based fuzzy predictive cruise control for improving safety and economy of connected vehicles". In: *IET Intelligent Transport Systems* 14.5 (2020), pp. 346–355.

[38] Chris J Ostafew et al. "Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking". In: *Journal of Field Robotics* 33.1 (2016), pp. 133–152.

[39] Chao Shang and Fengqi You. "A data-driven robust optimization approach to scenario-based stochastic model predictive control". In: *Journal of Process Control* 75 (2019), pp. 24–39.

[40] Xue-Cheng Xi, Aun-Neow Poo, and Siaw-Kiang Chou. "Support vector regression model predictive control on a HVAC plant". In: *Control engineering practice* 15.8 (2007), pp. 897–908.

[41] Francesco Smarra et al. "Data-driven model predictive control using random forests for building energy optimization and climate control". In: *Applied energy* 226 (2018), pp. 1252–1272.

[42] Achin Jain et al. "Data-driven model predictive control with regression trees—An application to building energy management". In: *ACM Transactions on Cyber-Physical Systems* 2.1 (2018), pp. 1–21.

[43] Yujiao Chen et al. “Transfer learning with deep neural networks for model predictive control of HVAC and natural ventilation in smart buildings”. In: *Journal of Cleaner Production* 254 (2020), p. 119866.

[44] Jiangyu Wang et al. “Data-driven model predictive control for building climate control: Three case studies on different buildings”. In: *Building and Environment* 160 (2019), p. 106204.

[45] Ka In Wong et al. “Modelling of diesel engine performance using advanced machine learning methods under scarce and exponential data set”. In: *Applied Soft Computing* 13.11 (2013), pp. 4428 –4441. ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2013.06.006. URL: http://www.sciencedirect.com/science/article/pii/S1568494613001956.

[46] Francesco Smarra et al. “Data-driven switched affine modeling for model predictive control”. In: *IFAC-PapersOnLine* 51.16 (2018), pp. 199–204.

[47] M Germin Nisha and GN Pillai. "Nonlinear model predictive control with relevance vector regression and particle swarm optimization". In: *Journal of control theory and applications* 11.4 (2013), pp. 563–569.

[48] Zhe Wu et al. "Machine learning-based predictive control of nonlinear processes. Part I: Theory". In: *AIChE Journal* 65.11 (2019), e16729.

[49] Zhe Wu et al. "Model predictive control of phthalic anhydride synthesis in a fixed-bed catalytic reactor via machine learning modeling". In: *Chemical Engineering Research and Design* 145 (2019), pp. 173–183.

[50] BAO Zhejing, PI Daoying, and SUN Youxian. "Nonlinear model predictive control based on support vector machine with multi-kernel". In: *Chinese Journal of Chemical Engineering* 15.5 (2007), pp. 691–697.

[51] Wee Chin Wong et al. "Recurrent neural network-based model predictive control for continuous pharmaceutical manufacturing". In: *Mathematics* 6.11 (2018), p. 242.

[52] Zhe Wu and Panagiotis D Christofides. "Economic machine-learning-based predictive control of nonlinear systems". In: *Mathematics* 7.6 (2019), p. 494.

[53] Scarlett Chen, Zhe Wu, and Panagiotis D Christofides. "Decentralized machine-learning-based predictive control of nonlinear processes". In: *Chemical Engineering Research and Design* 162 (2020), pp. 45–60.

[54] Scarlett Chen et al. "Machine learning-based distributed model predictive control of nonlinear processes". In: *AIChE Journal* 66.11 (2020), e17013.

[55] Gongming Wang et al. "Deep Learning-Based Model Predictive Control for Continuous Stirred-Tank Reactor System". In: *IEEE Transactions on Neural Networks and Learning Systems* (2020).

[56] Maciej ŁAwryńCzuk and Piotr Tatjewski. "Nonlinear predictive control based on neural multi-models". In: *International Journal of Applied Mathematics and Computer Science* 20.1 (2010), pp. 7–21.

[57] Neha Sharma and Kailash Singh. "Neural network and support vector machine predictive control of tert-amyl methyl ether reactive distillation column". In: *Systems Science & Control Engineering: An Open Access Journal* 2.1 (2014), pp. 512–526.

[58] Haralambos Sarimveis and George Bafas. "Fuzzy model predictive control of non-linear processes using genetic algorithms". In: *Fuzzy sets and systems* 139.1 (2003), pp. 59–80.

[59] JM Manzano et al. "Robust data-based model predictive control for nonlinear constrained systems". In: *IFAC-PapersOnLine* 51.20 (2018), pp. 505–510.

[60] Eric Bradford et al. "Stochastic data-driven model predictive control using Gaussian processes". In: *Computers & Chemical Engineering* 139 (2020), p. 106844.

[61] Peyman Sindareh-Esfahani. "Machine Learning Modeling and Robust Model Predictive Control of a Wind Turbine". In: (2019).

[62] Xiuxing Yin et al. "Reliability aware multi-objective predictive control for wind farm based on machine learning and heuristic optimizations". In: *Energy* (2020), p. 117739.

[63] Paul D Sclavounos and Yu Ma. "Wave energy conversion using machine learning forecasts and model predictive control". In: *33rd International Workshop on Water Waves and Floating Bodies*. 2018, pp. 4–7.

[64] Morgan T Gillespie et al. "Learning nonlinear dynamic models of soft robots for model predictive control with neural networks". In: *2018 IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE. 2018, pp. 39–45.

[65] Gang Cao, Edmund M-K Lai, and Fakhrul Alam. "Gaussian process model predictive control of an unmanned quadrotor". In: *Journal of Intelligent & Robotic Systems* 88.1 (2017), pp. 147–162.

[66] A Bernardelli et al. "Real-time model predictive control of a wastewater treatment plant based on machine learning". In: *Water Science and Technology* 81.11 (2020), pp. 2391–2400.

[67] B. Khoshbakht Irdmousa et al. "Data-driven Modeling and Predictive Control of Combustion Phasing for RCCI Engines". In: *American Control Conference* (2019), pp. 1–6.

[68] LN Aditya Basina et al. "Data-driven Modeling and Predictive Control of Maximum Pressure Rise Rate in RCCI Engines". In: *2020 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE. 2020, pp. 94–99.

[69] M. Iwadare, M. Ueno, and S. Adachi. "Multi-variable air-path management for a clean diesel engine using model predictive control". In: *SAE International Journal of Engines* 2.1 (2009), pp. 764–773.

[70] SW Wang et al. "Adaptive neural network model based predictive control for air–fuel ratio of SI engines". In: *Engineering Applications of Artificial Intelligence* 19.2 (2006), pp. 189–200.

[71] Vijay Manikandan Janakiraman, XuanLong Nguyen, and Dennis Assanis. "An ELM based predictive control method for HCCI engines". In: *Engineering Applications of Artificial Intelligence* 48 (2016), pp. 106–118.

[72] Vijay Manikandan Janakiraman, XuanLong Nguyen, and Dennis Assanis. "Nonlinear model predictive control of a gasoline HCCI engine using extreme learning machines". In: *arXiv preprint arXiv:1501.03969* (2015).

[73] J. Macek et al. *Transient Engine Model as a Tool for Predictive Control.* Tech. rep. SAE Technical Paper, 2006.

[74] Yunfeng Hu et al. "Nonlinear model predictive controller design based on learning model for turbocharged gasoline engine of passenger vehicle". In: *Mechanical Systems and Signal Processing* 109 (2018), pp. 74–88.

[75] Jun Lu et al. "Model predictive engine control using support vector machine". In: *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*. IEEE. 2015, pp. 1569–1573.

[76] Shaobo Xie et al. "Pontryagin's minimum principle based model predictive control of energy management for a plug-in hybrid electric bus". In: *Applied energy* 236 (2019), pp. 893–905.

[77] Matthias Marx, Xi Shen, and Dirk Soffker. "A data-driven online identification and control optimization approach applied to a hybrid electric powertrain system". In: *IFAC Proceedings Volumes* 45.2 (2012), pp. 153–158.

[78] Shaobo Xie et al. "Time-efficient stochastic model predictive energy management for a plug-in hybrid electric bus with an adaptive reference state-of-charge advisory". In: *IEEE Transactions on Vehicular Technology* 67.7 (2018), pp. 5671–5682.

[79] Xiaolin Tang et al. "Naturalistic Data-Driven Predictive Energy Management for Plug-in Hybrid Electric Vehicles". In: *IEEE Transactions on Transportation Electrification* (2020).

[80] Chao Huang et al. "Multistructure radial basis function neural-networks-based extended model predictive control: Application to clutch control". In: *IEEE/ASME Transactions on Mechatronics* 24.6 (2019), pp. 2519–2530.

[81] Paul Drews et al. "Aggressive deep driving: Combining convolutional neural networks and model predictive control". In: *Conference on Robot Learning*. 2017, pp. 133–142.

[82] Hadi Kazemi et al. "A learning-based stochastic MPC design for cooperative adaptive cruise control to handle interfering vehicles". In: *IEEE Transactions on Intelligent Vehicles* 3.3 (2018), pp. 266–275.

[83] Lei Lin, Siyuan Gong, and Tao Li. "Deep learning-based human-driven vehicle trajectory prediction and its application for platoon control of connected and autonomous vehicles". In.

[84] OA Dahunsi, JO Pedro, and OT Nyandoro. "Neural network-based model predictive control of a servo-hydraulic vehicle suspension system". In: *AFRICON 2009*. IEEE. 2009, pp. 1–6.

[85] Eugenio Alcalá et al. "TS-MPC for Autonomous Vehicle using a Learning Approach". In: *arXiv preprint arXiv:2004.14362* (2020).

[86] Fernando Henrique Morais Da Rocha et al. "Model predictive control of a heavy-duty truck based on Gaussian process". In: *2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)*. IEEE. 2016, pp. 97–102.

[87] Gowtham Garimella et al. "Neural network modeling for steering control of an autonomous vehicle". In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2017, pp. 2609–2615.

[88] Aliasghar Arab and Jingang Yi. "Safety-Guaranteed Learning-Predictive Control for Aggressive Autonomous Vehicle Maneuvers". In: *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE. 2020, pp. 1036–1041.

[89] Yuan Jin et al. "A data-driven model predictive control for lighting system based on historical occupancy in an office building: Methodology development". In: *Building Simulation*. Springer. 2020.

[90] Jinghan Cui, Tianyou Chai, and Xiangjie Liu. "Deep-Neural-Network-Based Economic Model Predictive Control for Ultrasupercritical Power Plant". In: *IEEE Transactions on Industrial Informatics* 16.9 (2020), pp. 5905–5913.

[91] Shiyu Yang et al. "Model predictive control with adaptive machine-learning-based model for building energy efficiency and comfort optimization". In: *Applied Energy* 271 (2020), p. 115147.

[92] Jongho Shin et al. "Model predictive flight control using adaptive support vector regression". In: *Neurocomputing* 73.4-6 (2010), pp. 1031–1037.

[93] Konstantinos Dalamagkidis, Kimon P Valavanis, and Les A Piegl. "Nonlinear model predictive control with neural network optimization for autonomous autorotation of small unmanned helicopters". In: *IEEE Transactions on Control Systems Technology* 19.4 (2010), pp. 818–831.

[94] Ahmad Mozaffari, Mahyar Vajedi, and Nasser L Azad. "A robust safety-oriented autonomous cruise control scheme for electric vehicles based on model predictive control and online sequential extreme learning machine with a hyper-level fault tolerance-based supervisor". In: *Neurocomputing* 151 (2015), pp. 845–856.

[95] Stefano Di Cairano et al. "Stochastic MPC with learning for driver-predictive vehicle control and its application to HEV energy management". In: *IEEE Transactions on Control Systems Technology* 22.3 (2013), pp. 1018–1031.

[96] Grady Williams et al. "Information theoretic MPC for model-based reinforcement learning". In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2017, pp. 1714–1721.

[97] Y. Bao, J. Mohammadpour Velni, and M. Shahbakhti. "AN ONLINE TRANSFER LEARNING APPROACH FOR IDENTIFICATION AND PREDICTIVE CONTROL DESIGN WITH APPLICATION TO RCCI ENGINES". In: *ASME 2020 Dynamic Systems and Control Conference* (2020).

[98] Adam Vaughan. *Adaptive machine learning method to predict and control engine combustion*. US Patent 10,030,602. 2018.

[99] Sergio Saludes Rodil and Maria J Fuente. "Fault tolerance in the framework of support vector machines based model predictive control". In: *Engineering Applications of Artificial Intelligence* 23.7 (2010), pp. 1127–1139.

[100] Zhe Wu, David Rincon, and Panagiotis D Christofides. "Real-time adaptive machine-learning-based predictive control of nonlinear processes". In: *Industrial & Engineering Chemistry Research* 59.6 (2019), pp. 2275–2290.

[101] Katharina Bieker et al. "Deep model predictive flow control with limited sensor data and online learning". In: *Theoretical and Computational Fluid Dynamics* (2020), pp. 1–15.

[102] Hong-Gui Han et al. "Nonlinear model predictive control based on a self-organizing recurrent neural network". In: *IEEE transactions on neural networks and learning systems* 27.2 (2015), pp. 402–415.

[103] Ján Drgoňa et al. "Approximate model predictive building control via machine learning". In: *Applied Energy* 218 (2018), pp. 199–216.

[104] Benjamin Karg and Sergio Lucia. "Deep learning-based embedded mixed-integer model predictive control". In: *2018 European Control Conference (ECC)*. IEEE. 2018, pp. 2075–2080.

[105] Mohamed Toub et al. "MPC-trained ANFIS for Control of MicroCSP Integrated into a Building HVAC System". In: *2019 American Control Conference (ACC)*. IEEE. 2019, pp. 241–246.

[106] Sergio Lucia and Benjamin Karg. "A deep learning-based approach to robust nonlinear model predictive control". In: *IFAC-PapersOnLine* 51.20 (2018), pp. 511–516.

[107] Yannic Vaupel et al. "Accelerating nonlinear model predictive control through machine learning". In: *Journal of Process Control* 92 (2020), pp. 261–270.

[108] Keuntaek Lee, Kamil Saigol, and Evangelos A Theodorou. "Safe end-to-end imitation learning for model predictive control". In: *arXiv preprint arXiv:1803.10231* (2018).

[109] Liting Sun et al. "A fast integrated planning and control framework for autonomous driving via imitation learning". In: *Dynamic Systems and Control Conference*. Vol. 51913. American Society of Mechanical Engineers. 2018, V003T37A012.

[110] Xiaojing Zhang, Monimoy Bujarbaruah, and Francesco Borrelli. "Safe and near-optimal policy learning for model predictive control using primal-dual neural networks". In: *2019 American Control Conference (ACC)*. IEEE. 2019, pp. 354–359.

[111] Brian Cera and Alice M Agogino. "Multi-cable rolling locomotion with spherical tensegrities using model predictive control and deep learning". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2018, pp. 1–9.

[112] Ihab S Mohamed et al. “A neural-network-based model predictive control of three-phase inverter with an output LC filter”. In: *IEEE Access* 7 (2019), pp. 124737–124749.

[113] Mateja Novak and Tomislav Dragicevic. “Supervised imitation learning of finite set model predictive control systems for power electronics”. In: *IEEE Transactions on Industrial Electronics* (2020).

[114] Martin Klaučo, Martin Kalúz, and Michal Kvasnica. “Machine learning-based warm starting of active set methods in embedded model predictive control”. In: *Engineering Applications of Artificial Intelligence* 77 (2019), pp. 1–8.

[115] Kim P Wabersich et al. “Probabilistic model predictive safety certification for learning-based control”. In: *arXiv preprint arXiv:1906.10417* (2019).

[116] Mario Zanon and Sébastien Gros. “Safe reinforcement learning using robust MPC”. In: *IEEE Transactions on Automatic Control* (2020).

The momentum update rule (recursive) - details

The momentum update rule is, for $\ell = 1, \ldots, l$;

$$\begin{cases} v_{dW^{[\ell]}} := \beta v_{dW^{[\ell]}} + (1-\beta) dW^{[\ell]} \\ W^{[\ell]} := W^{[\ell]} - \alpha v_{dW^{[\ell]}} \end{cases}$$

$$\begin{cases} v_{db^{[\ell]}} := \beta v_{db^{[\ell]}} + (1-\beta) db^{[\ell]} \\ b^{[\ell]} := b^{[\ell]} - \alpha v_{db^{[\ell]}} \end{cases}$$

- $dW^{[\ell]}$ is $\frac{\partial L}{\partial W^{[\ell]}}$ and $db^{[\ell]}$ is $\frac{\partial L}{\partial b^{[\ell]}}$
- where l is the number of layers, β is the momentum and α is the learning rate.
- A new hyperparameter β, called the momentum
  - must be set between 0 (high friction) and 1 (no friction)
  - a typical momentum value is 0.9

RMSProp update rule (recursive) - details

- The parameter update rule for RMSProp is

$$\begin{cases} s_{dW^{[\ell]}} := \beta_2 s_{dW^{[\ell]}} + (1-\beta_2) dW^{[\ell]} \odot dW^{[\ell]} \\ W^{[\ell]} := W^{[\ell]} - \alpha \frac{dW^{[\ell]}}{\sqrt{s_{dW^{[\ell]}} + \epsilon}} \end{cases}$$

$$\begin{cases} s_{db^{[\ell]}} := \beta_2 s_{db^{[\ell]}} + (1-\beta_2) db^{[\ell]} \odot db^{[\ell]} \\ W^{[\ell]} := W^{[\ell]} - \alpha \frac{db^{[\ell]}}{\sqrt{s_{dW^{[\ell]}} + \epsilon}} \end{cases}$$

- ϵ is hyperparameter preventing division by zero
- β_2 Exponential decay hyperparameter for the moment estimates. β_2 is usually 0.9.
- Except on very simple problems
  - this optimizer almost always performs much better than GD
  - it was the preferred optimization algorithm of many researchers until Adam optimization became prevalent

Adam Optimization- details

The general update rule for Adam (recursive) is, for $\ell = 1, \ldots, l$:

$$\begin{cases} v_{db^{[\ell]}} := \beta_1 v_{db^{[\ell]}} + (1-\beta_1) db^{[\ell]} \\ v^{corrected}_{db^{[\ell]}} := \frac{v_{db^{[\ell]}}}{1-(\beta_1)^t} \\ s_{db^{[\ell]}} := \beta_2 s_{db^{[\ell]}} + (1-\beta_2)(db^{[\ell]} \odot db^{[\ell]}) \\ s^{corrected}_{db^{[\ell]}} := \frac{s_{db^{[\ell]}}}{1-(\beta_2)^t} \\ b^{[\ell]} := b^{[\ell]} - \alpha \frac{v^{corrected}_{db^{[\ell]}}}{\sqrt{s^{corrected}_{db^{[\ell]}}}+\varepsilon} \end{cases}$$

- β_1: exponential decay hyperparameter for the first moment estimates
- β_2: exponential decay hyperparameter for the second moment estimates,
- ϵ hyperparameter preventing division by zero in Adam updates.
- v: moving average of the first gradient in the Adam rule
- s: moving average of the squared gradient in the Adam rule
- for initial values can choose $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-7}$, and $\alpha = 0.001$.

Combined ML and MPC from applications

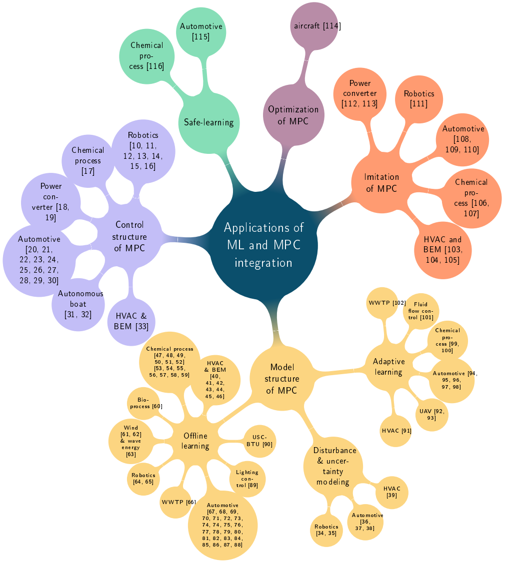


Figure: Applications of AI and MPC integration overview