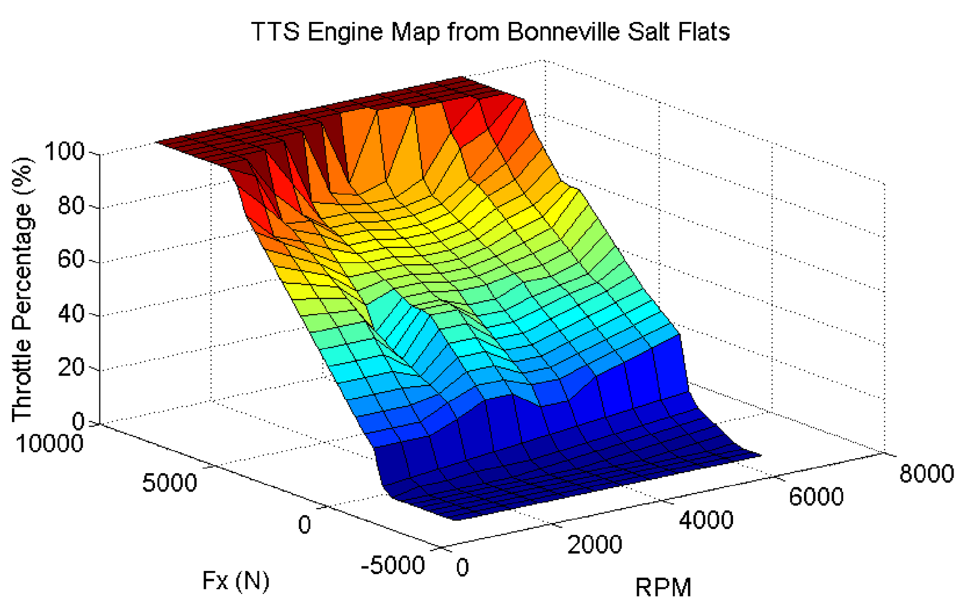
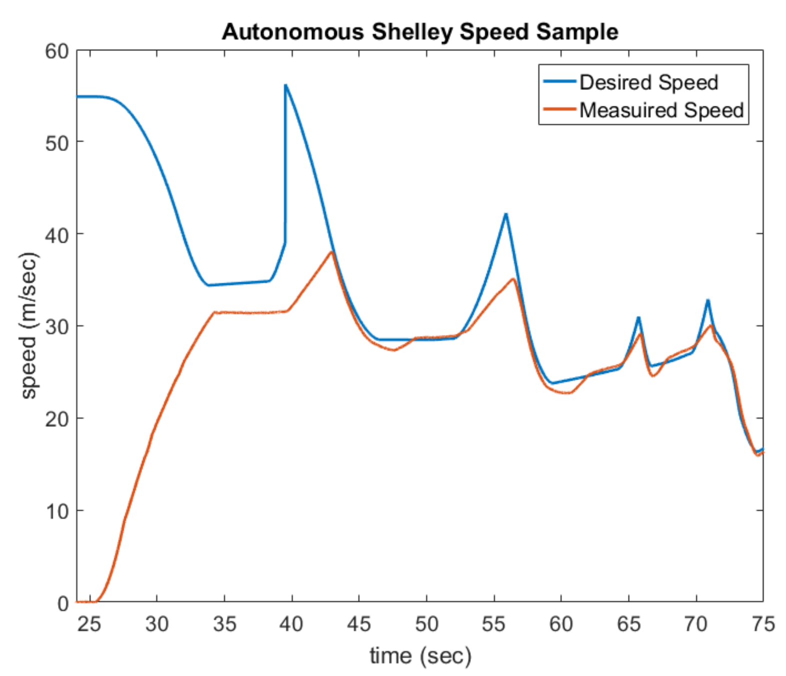
Dynamic Throttle Estimation by Machine Learning from Professionals

Nathan Spielberg, John Alsterda, J. C. Gerdes

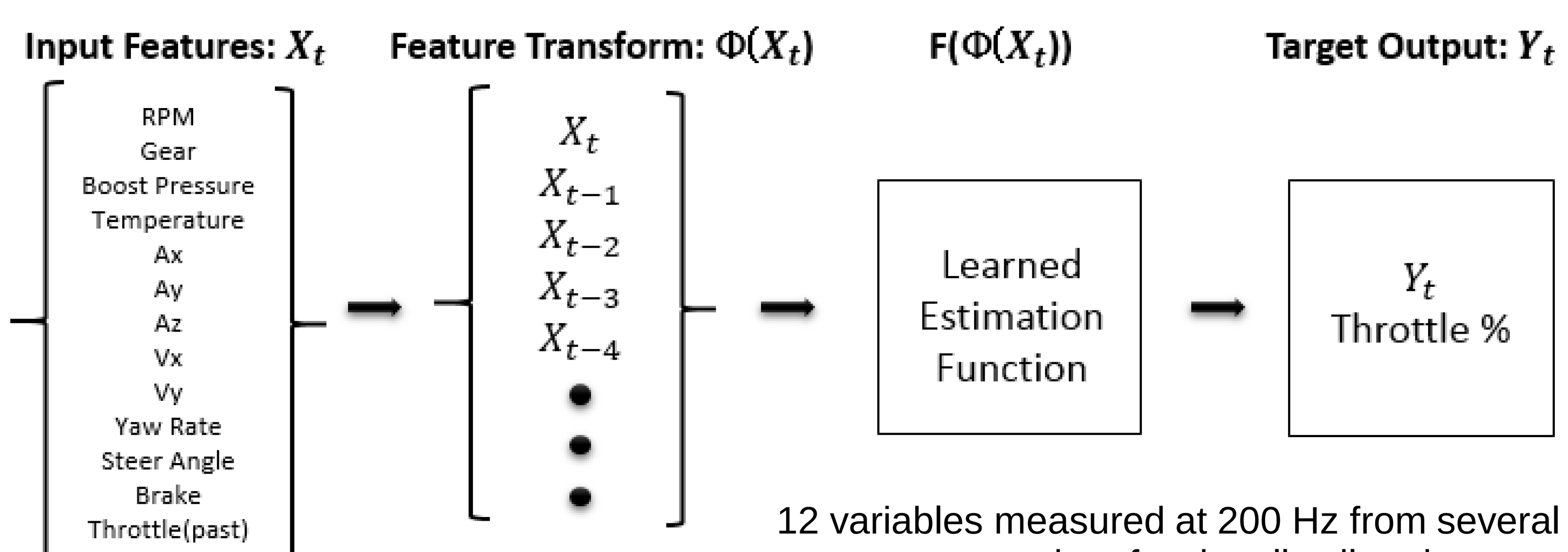
Motivation

Shelley, the Dynamic Design Lab's automated TTS racecar, uses an empirical map (below) to select throttle commands necessary to follow a desired speed profile (below right). We aim to improve this map in hopes to follow desired speed more closely and race faster laps! [1]

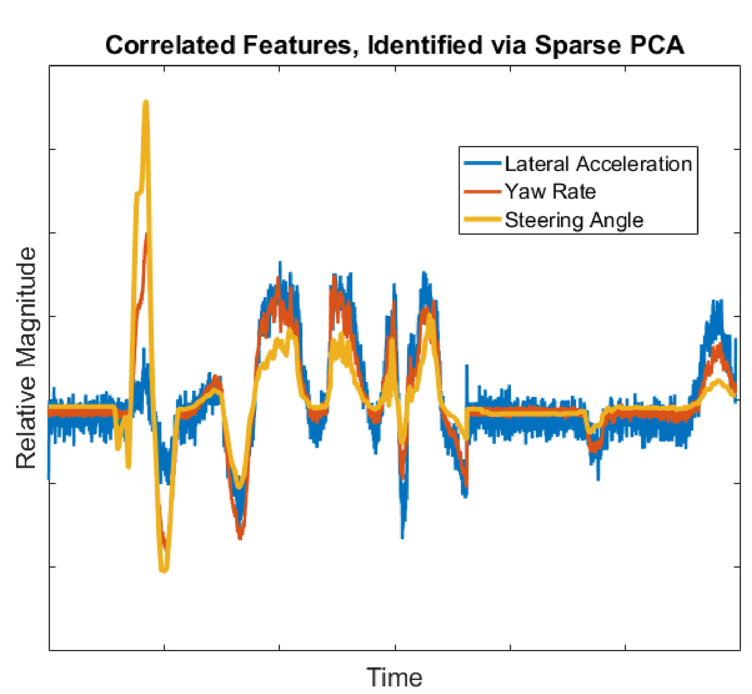




Feature Selection



12 variables measured at 200 Hz from several autonomous and professionally piloted test runs compose our feature set. Throttle commands form our targets. The feature set was also augmented with delay states from previous time steps, as illustrated above.

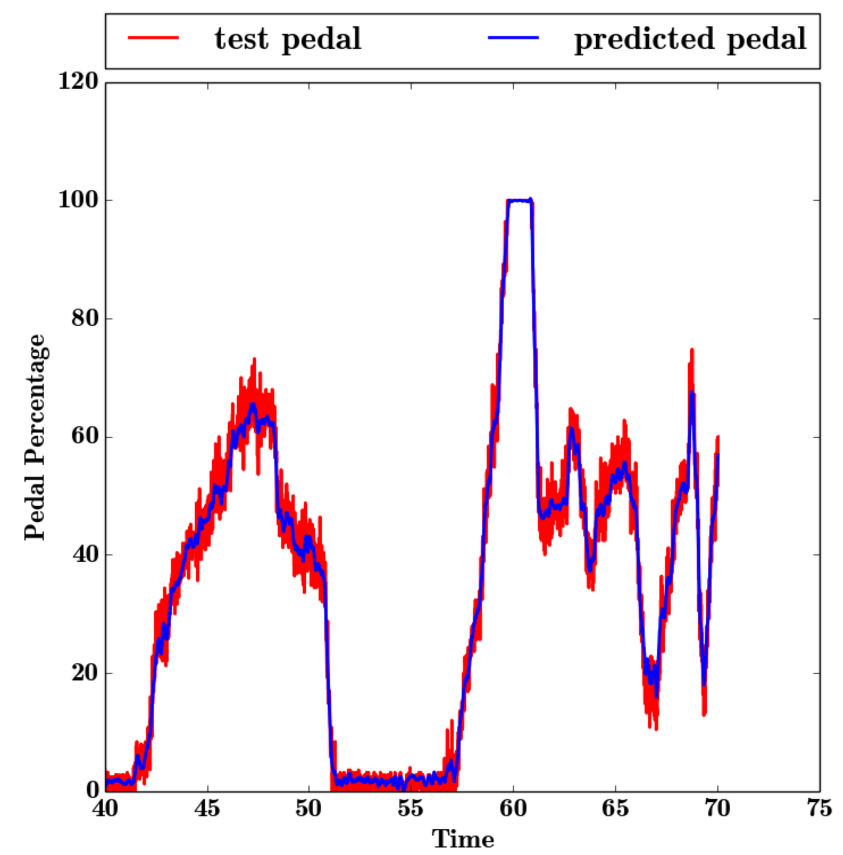


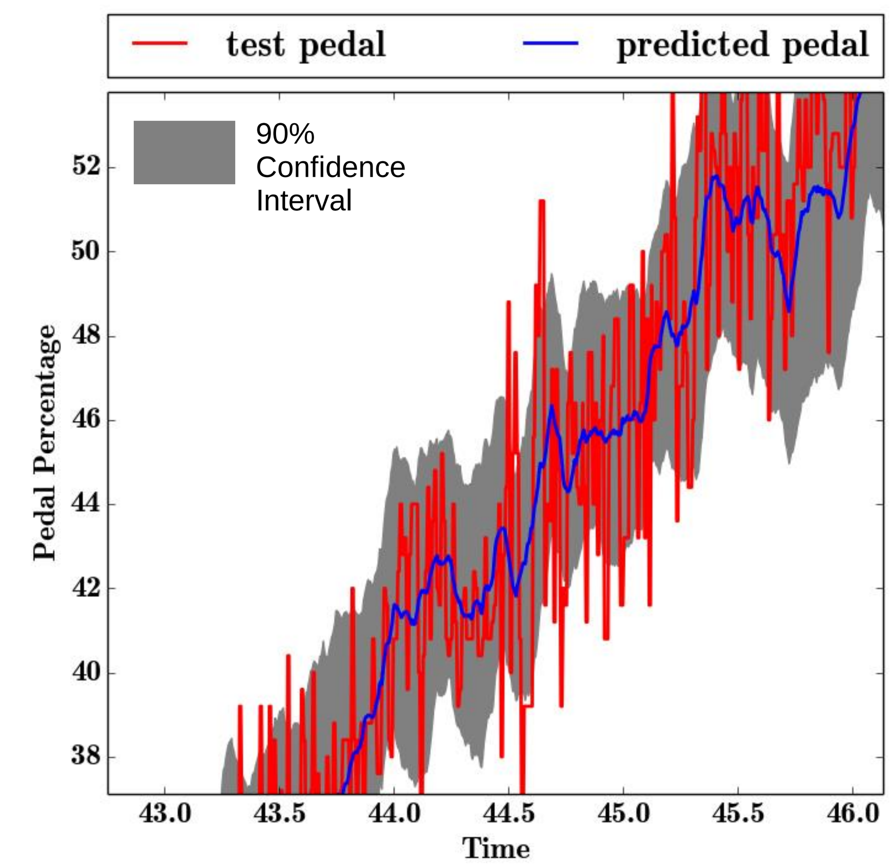
Sparse Principal Component Analysis was then performed to identify features with redundant information, as shown on the left. Steering Angle and Yaw Rate were subsequently removed from our features with negligible effect on performance. [4]

SPCA: $X = U\Sigma V^T$

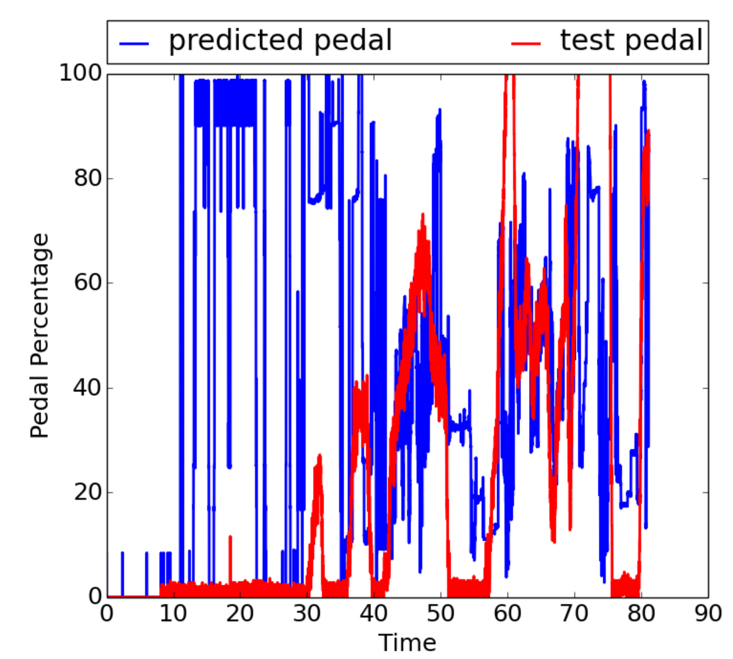
X = Feature Set Matrix U = Non-Unitary Basis
Σ = Singular Values V = Loading Matrix

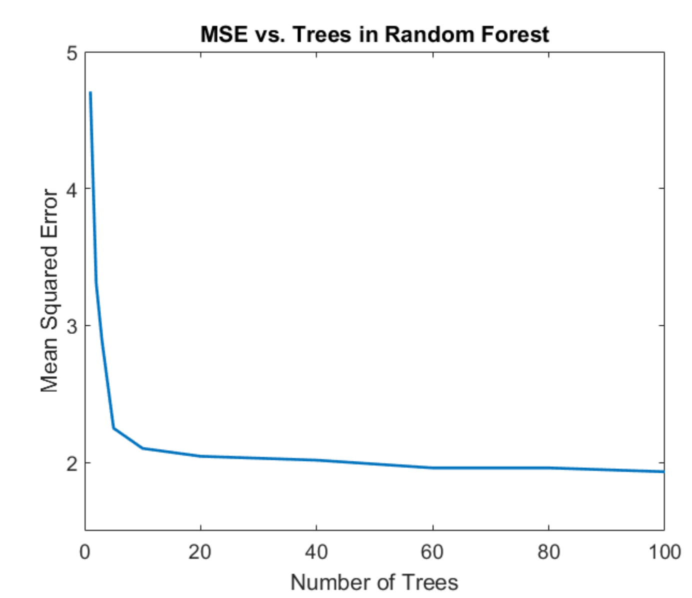
Random Forest





Using the Random Forest algorithm, predicted throttle is shown (left) to follow actual throttle through a test trial with 1.8% mean squared error (MSE). A shorter segment is shown (right), illustrating that 84% of test throttle values fell within our model's gray confidence region. Our model was confident that 90% of test throttles would fall within ± 2.1% of predicted values. [2]

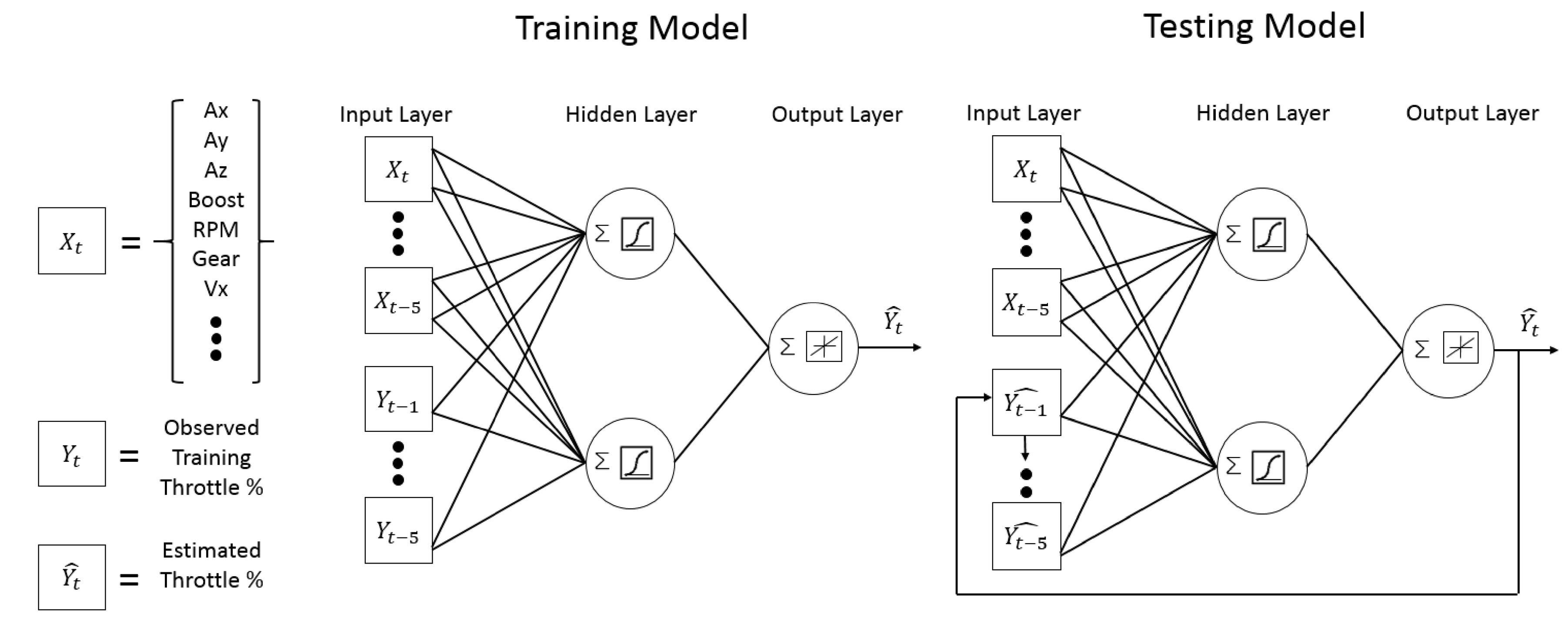


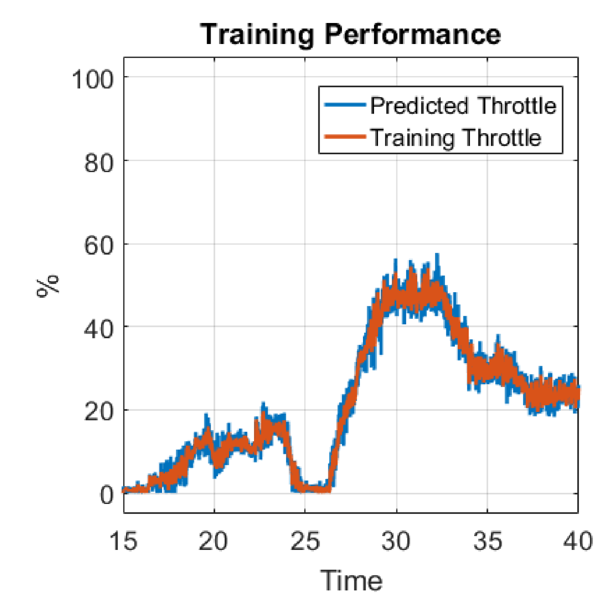


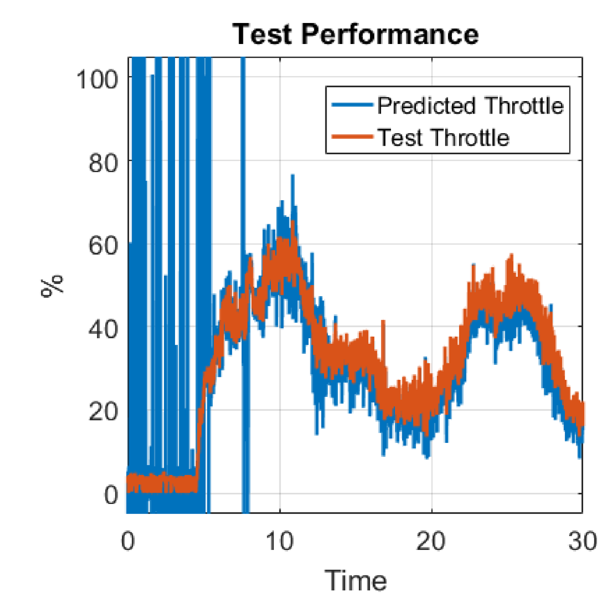
When delay states of past inputs are not included, the average MSE is 30% for a forest of ten trees. When delay states are included, the average MSE reduces to 2.5%, showing the system's high dependency on time series feature data.

Performance improved with additional trees in the forest, but increased less that 1% after 80 trees. Computational time also increased with number of trees in the forest.

NARX Network



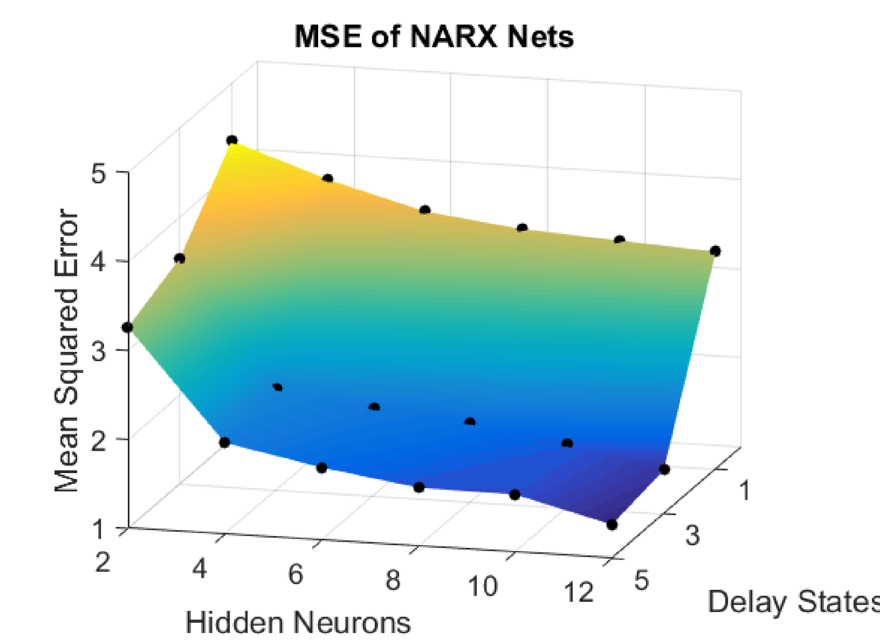




We implemented a Nonlinear Autoregressive Neural Net with External Input (NARX) due to its design for time series prediction. Weights and bias values were computed via Bayesian Regularization. The net is trained in open loop and closed for testing, as shown above. [3].

Training performance of 1.5% MSE was achieved (above left), but when testing provided novel data, the NARX net would sometimes yield physically impossible and unnatural throttle predictions (above right), indicating high generalization error.

Performance improved with additional delay states and neurons (right). Increasing the delay and neuron parameters to achieve even lower MSEs became prohibitively time consuming.



Conclusions and Future Work

This work shows strong potential to provide a throttle function which may outperform the vehicle's current empirical map. The Random Forest algorithm, supplied with vehicle and engine states which are available in real time, demonstrates capability to accurately predict the throttle necessary to achieve desired accelerations within a MSE of 1.8% and within a 90% confidence bound of ± 2.1%.

In the near future, we will integrate this model onto Shelley for experimental validation. To do so, the model must be optimized for real time performance and the existing control architecture.

Thanks to: Prof. John Duchi, Prof. Chris Gerdes, the NSF, DDL, VW and ERL

[1] Kritayakirana, Krisada, and J. Christian Gerdes. Controlling an Autonomous Racing Vehicle. ASME Dynamic Systems and Control Conference (2009)
[2] L. Breiman. Random forests. Machine Learning, 45(1):5–32, (2001)
[3] Neural Network Toolbox™ Reference. Vers. R2016a. Natick, MA: The MathWorks, Inc.
[4] Zou, Hui, Trevor Hastie, and Robert Tibshirani. Sparse Principal Component Analysis. Journal of Computational and Graphical Statistics 15(2): 265-286, (2006)