Compact Representation of a Multi-dimensional Combustion Manifold Using Deep Neural Networks

Sushrut Bhalla[1], Matthew Yao[2], Jean-Pierre Hickey[2], ***Mark Crowley[1]***

1 - Electrical and Computer Engineering

2 - Mechanical and Mechatronics Engineering

WATERLOO.AI

WATERLOO ARTIFICIAL INTELLIGENCE INSTITUTE

UNIVERSITY OF WATERLOO

Mark Crowley

Assistant Professor

Electrical and Computer Engineering

University of Waterloo

Waterloo, Ontario, Canada

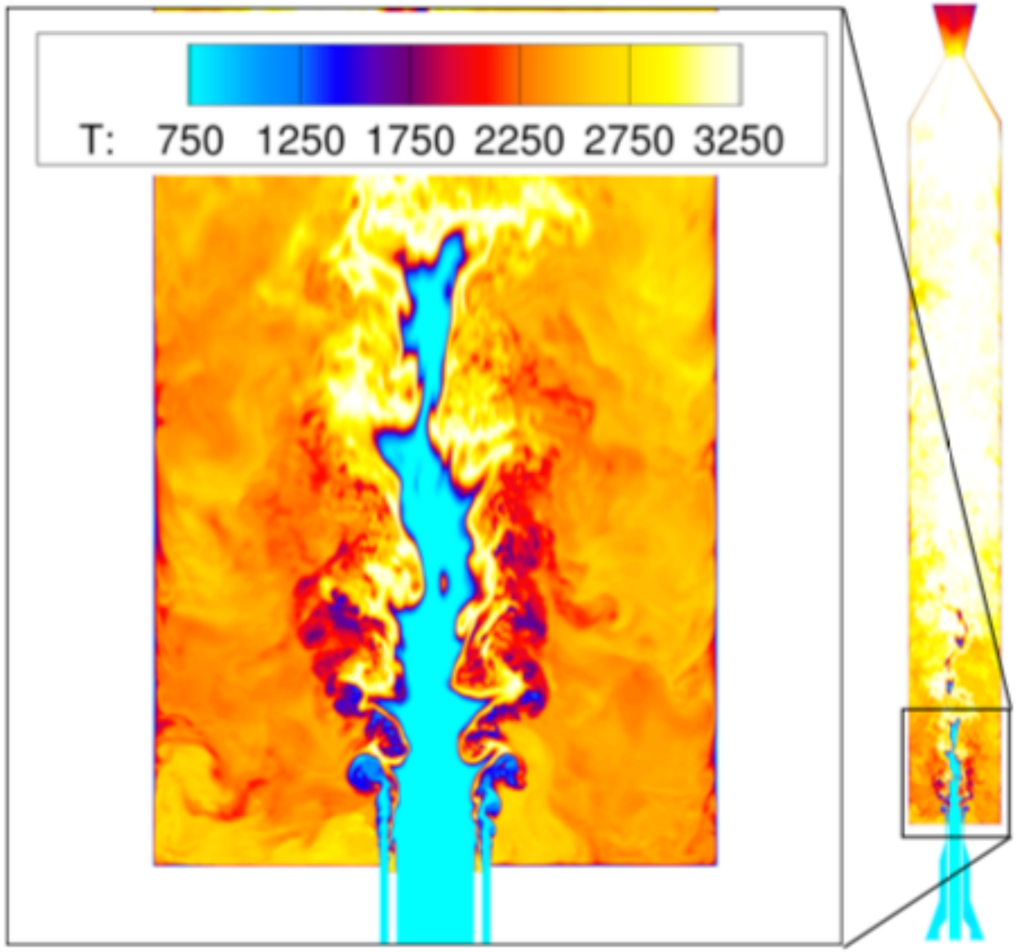
https://uwaterloo.ca/scholar/mcrowley

@compthink

Outline

- The Problem
  - Combustion Modelling
- Existing Work
  - Prior ML and non-ML approaches
- Our Approach
  - Deep Learning
  - Importance of Careful Regularization
- Results
  - Ground truth calculations
  - Simulator proof of concept
- Conclusions and Future Directions

The Problem : Combustion Modelling



Source: Ihme, AIAA SciTech Forum, 2019

- **Multi-physics problem:** chemical kinetics, turbulence, acoustics, heat transfer, phase change, radiation
- **Multi-scale challenges:** from the sub micro-meter to the meter scale with strong interactions among all scales

Modelling is required to render this problem numerically tractable.

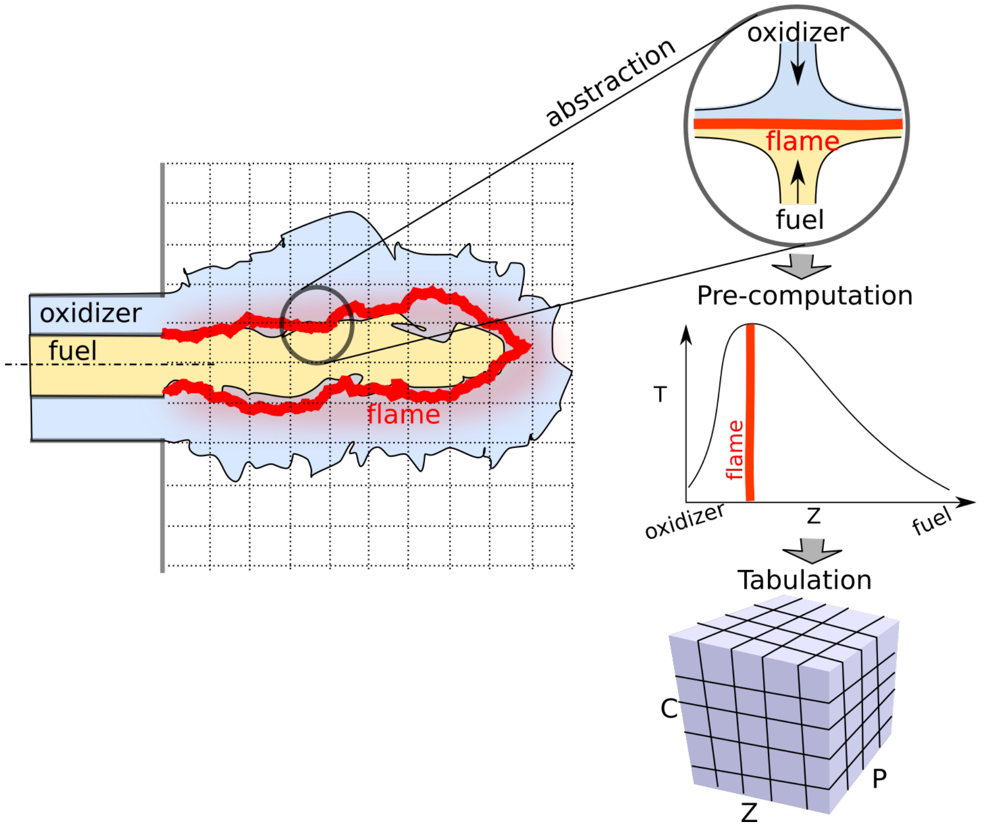
Two common physical models are:

- Turbulence modelling (not discussed today)
- **Combustion modelling**

WATERLOO.AI
WATERLOO ARTIFICIAL INTELLIGENCE INSTITUTE

A Widely Used Approach : FVPA

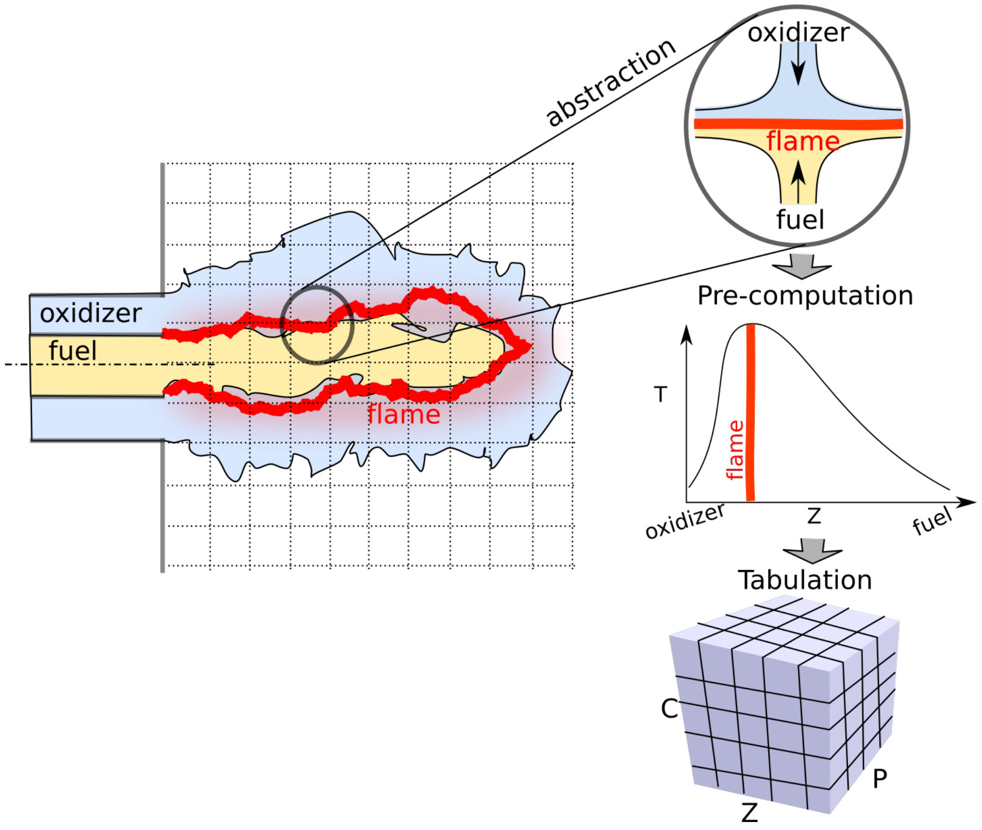
- **Flamelet :** to simplify calculation, imagine a corrugated, turbulent flame is made of a laminar flamelet coming from **directing fuel and oxidizer directly at each other**
- **Strain rate :** proportional to the velocity difference of the propellants in this setup
- **Flamelet-Progress Variable Approach (FPVA):** Flamelet can be completely characterized by the strain rate.
  - So, we can precompute a **combustion manifold** of all possible values as a **3D table**
- Strain rate depends on:
  - **Z: mixture fraction** (between 0: oxidizer and 1: fuel)
  - **C: progress variation** (defines the “evolution” of combustion at a given mixture fraction)
  - **P: pressure**
- **Combustion Simulations** use this table to provide the local chemical and thermodynamic state of the combustion



WATERLOO.AI
WATERLOO ARTIFICIAL INTELLIGENCE INSTITUTE

Increasing Demand for More Complex Simulations

- Modern combustion simulations require us to account for *additional* physical processes beyond (Z) mixture fraction, (C) progress and (P) pressure, such as:
  - Pressure variations (eg. rocket combustion)
  - Heat loss at the walls/ Radiation
  - Complex thermodynamics
  - Multi-fuel systems (multiple mixture fractions)
- Incorporating new physics into the flamelet combustion manifold requires *higher-dimensional tables* that quickly become intractable
- Example:
  - Standard 3D table will have:
    200 X 200 X 200 X 15 entries = **12^7 entries**
  - Extended table with pressure and heat loss:
    120^7 X 200 X 200 = **4.8^{11} entries**



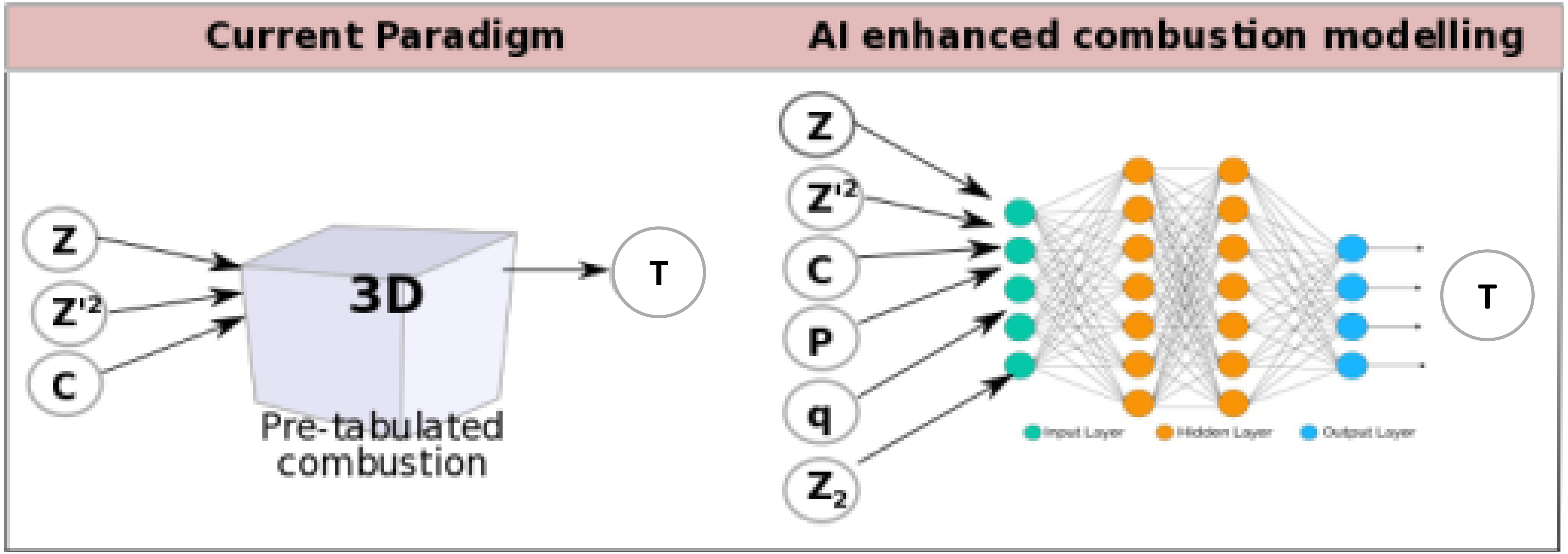
WATERLOO.AI
WATERLOO ARTIFICIAL INTELLIGENCE INSTITUTE

Prior non-ML Approaches

- Increasing the discretization to deal with size → less accurate
- Functional Representations → piecemeal approaches
  - **Polynomial basis functions** [Weisse, 2018]
  - **Bézier patches** to fit parts of the energy curves [Yao, 2018]
    → These are compact and fast once they are set up, but still require nonlinear increase in parameters as dimensions grow

- Use **PCA** and other Dimensionality Reduction method to project back down to 3D [Najafi-Yazdi, 2012]
  → not sustainable for more complex demands

So Why not use Neural Networks?



Benefits

- Perfect supervised training data can be made available (expensive PDE computations)
- Interpolation comes for free
- Model size does not increase as number of inputs increase

WATERLOO.AI
WATERLOO ARTIFICIAL INTELLIGENCE INSTITUTE

Prior ML Approaches

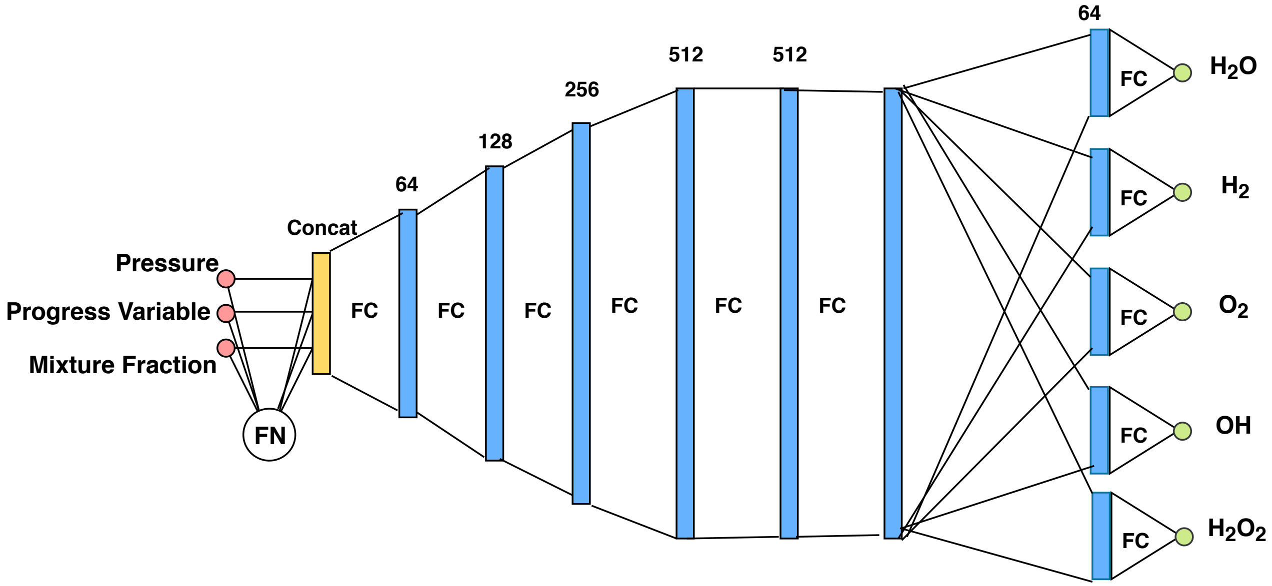
- [Ihme, 2009] Multi-Layer Perceptron
  - Used a simple MLP to replace the tabular representation
  - Used a fixed loss metric, focused on varying the number of nodes and layers to maximize that.
  - Used Sigmoid activation functions, standard backprop, no regularization

  → Their Conclusion: Found it was generalizable but much less accurate than table. Community moved on to other things.
- [Lapeyre, 2018] Autoencoder + CNN:
  - Extract structure for part of the flame of given length using a CNN
  - Train an autoencoder with a U-Net structure from CNN features on two consecutive time steps
  - Goal is to predict the third time step

  → Promising, but not directly applicable to flamelet-based approaches.

So we tried it again…

Straightforward Fully Connected Deep NN Architecture



Prediction of proportions of output components (ie. Species)

Activation Function: Leaky ReLU
Optimizer: Adam

Other Fully Connected Networks for each Task

Prediction Output	Hidden Layers
Temperature (T)	(**64,128,512,512,1024**, 1024)
Source Term (W)	(**64,128,512,512,512**, 512)
Heat Release (HR)	(**64,128,512,512,1024,2048**, 2048)

Why not just one network?

We could, but numerical scales are very different, so it didn't work as well, would need much more data.

WATERLOO.AI
WATERLOO ARTIFICIAL INTELLIGENCE INSTITUTE

Regularization Methods

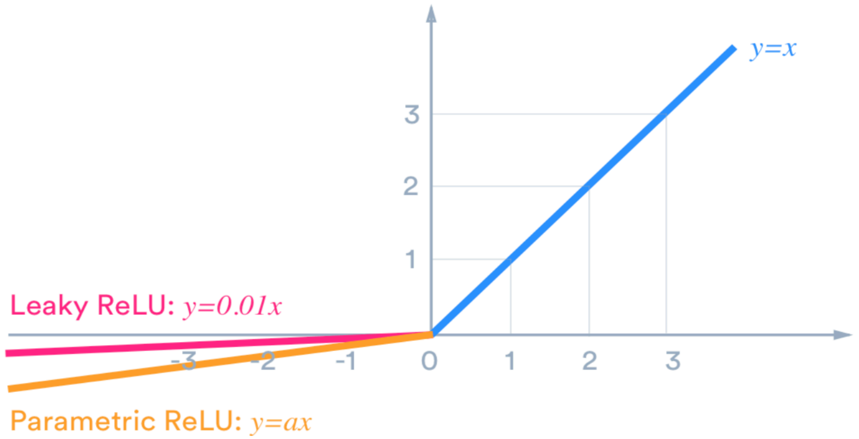
- Problem: Oscillating output for the temperature
- Solution: Experimented with different regularization techniques to reduce overfitting to noisy data. We found that L1+L2 regularization worked best.
- We also tried
  - Batch normalization – some improvements
  - Dropout (p=.5) – did quite badly
  - Layer Normalization – almost as good as L1+L2 for this domain

- Problem: Classification Errors
- Solution: Use an ensemble of models.
  - This is achieved by training 5 networks with different learning rates and random seeds
  - Prediction is average of top four networks based on variance estimate

Regularization Methods

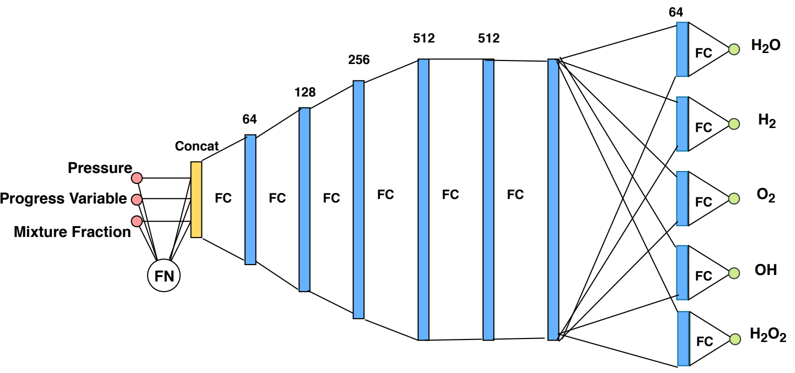
- Problem: Error on Areas with Large Variation
- Solution: Over-sample “hard examples” during training [Lin, 2017]
  - Defined as examples having error larger than median error of the batch.
  - The training batches are created by sampling with replacement 75% hard and 25% easy examples from the sampling batch.
  - Also used **gradient clipping** here to increase stability

- Problem: “Dying ReLU” – losing too much information on the negative gradient cases
- Solution: Use Leaky ReLU [Glorot, 2011]



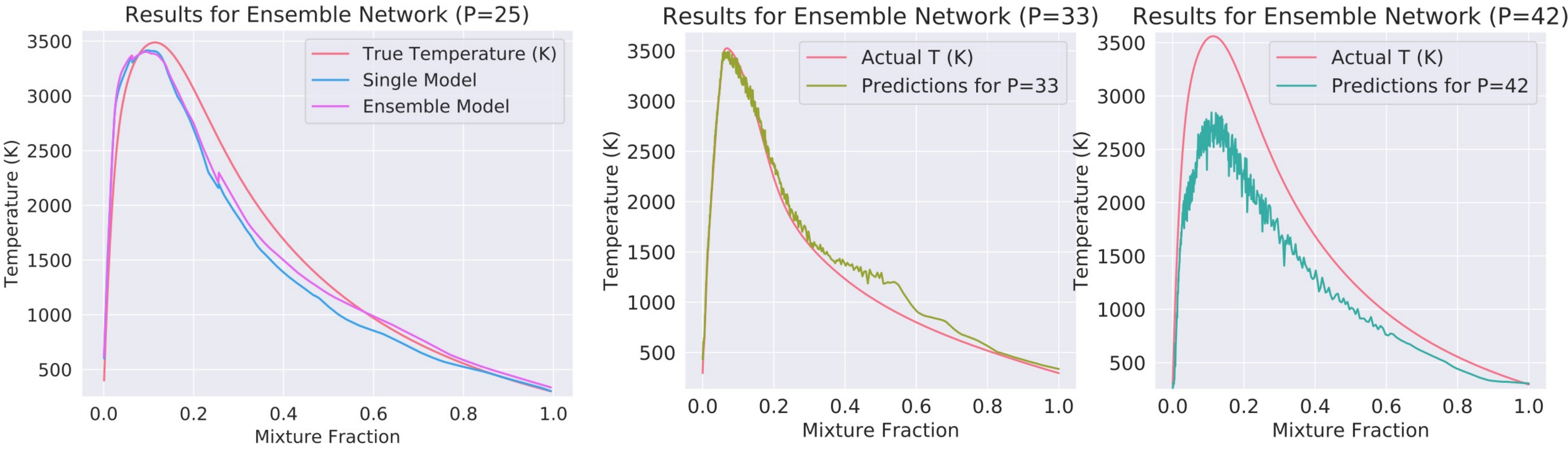
https://medium.com/@danqing/a-practical-guide-to-relu-b83ca804f1f7

Results: Prediction of Proportion of Chemical Outputs



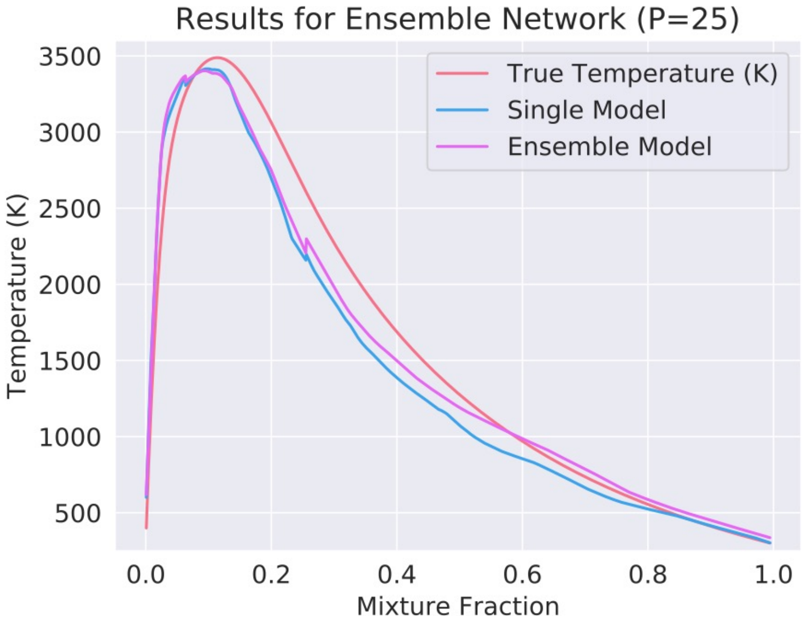
	Range $(O_{MAX} - O_{MIN})$	Standard Error Range (E_O)	Mean Error	Training Accuracy	Validation Accuracy
H	0.0219	0.0001	0.00431	29.23	21.44
O_2	1.0	0.005	0.3419	20.17	12.56
O	0.0665	0.00033	0.00892	55.79	37.43
OH	0.1279	0.00064	0.0345	56.91	43.34
H_2	1.0	0.005	0.2606	38.40	25.41
H_2O	0.8865	0.0044	0.6056	26.19	14.59
HO_2	0.0142	$7.1e-05$	0.00137	60.36	41.42
H_2O_2	0.0091	$4.5e-05$	0.00259	55.92	36.05
$HR(J/m^3)$	$93.4e+81$	$6.4e+78$	$1.05e+76$	81.23	81.18
$T(K)$	3295	34.47	82.44	61.97	54.60
W	30	0.149	3.023	71.55	58.144

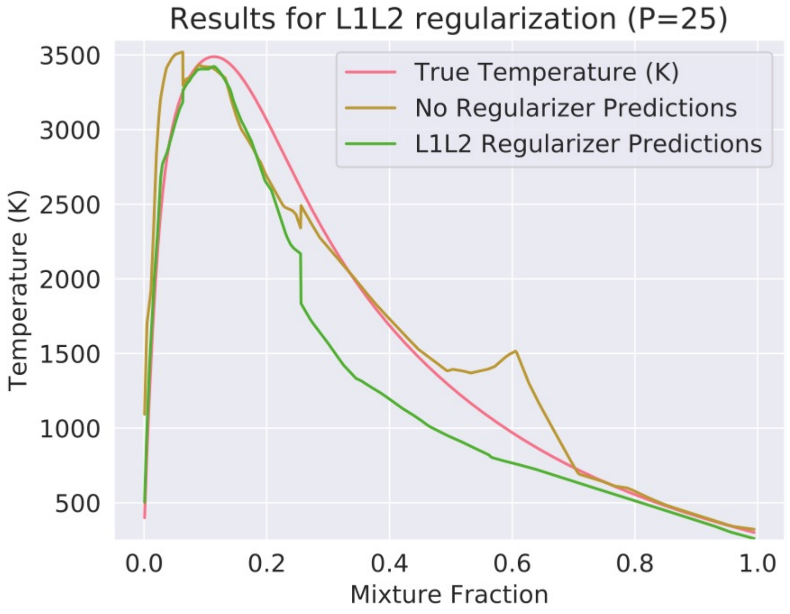
Results: Ensembles on Varying Pressure

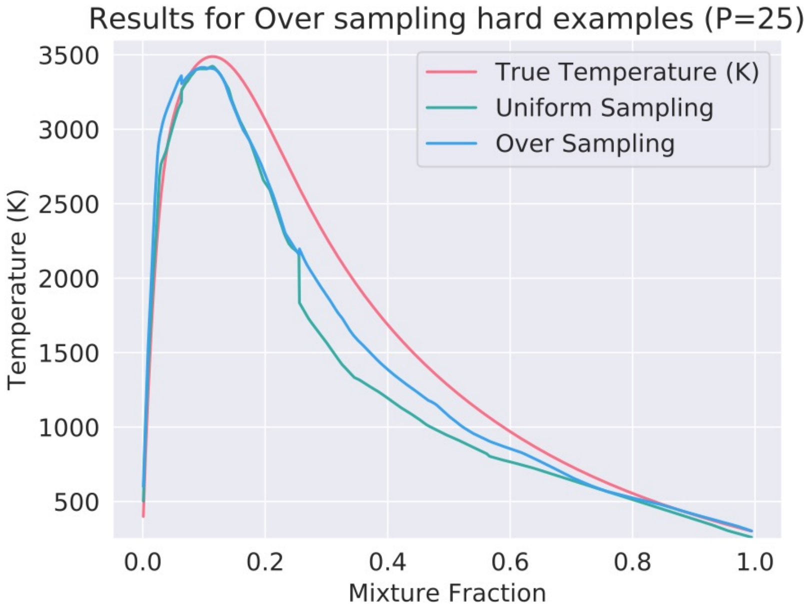


WATERLOO.AI
WATERLOO ARTIFICIAL INTELLIGENCE INSTITUTE

Results: Ablation Study on Three Regularization Approaches







Some Results

Advantage of L1L2

	No Regularization	L1L2 Regularization	Layer Normalization	Batch Normalization
Accuracy	31.44	39.83	23.63	12.28

Advantage of Over Sampling Hard Cases

	Uniform Sampling	Over Sampling	Ensemble Model
Accuracy	39.83	48.87	47.73

Computational Benefits of Combined Methods over Tabular

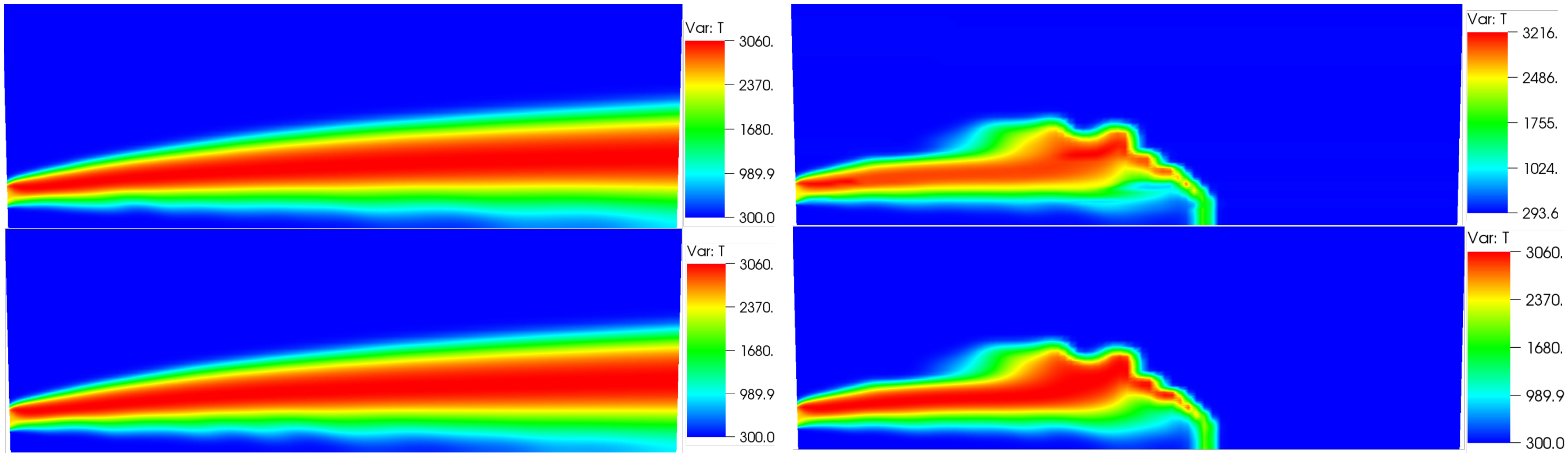
	Parallel Inference Time (in ms)	Serial Inference Time (in s)	Memory Requirements (in MB)
Tabulation Method	1.2×10^5	10.997	184.64
Deep Neural Network	13.92	55.27	24.158

Next Steps: Proof of Concept for Simulation Results

Using `OpenFOAM` simulator a single step of simulation was replace with the DNN predicted temperature and then run forward.

DNN:

Tabular:



Conclusions

- Main takeaway
  - Sometimes it’s worth trying an old idea again with new approaches
  - Deep Neural Networks can be use for this kind of complex lookup mapping
  - There are a huge number of regularization methods that need to be explored in these kind of applied problems, a thorough experimental methodology for comparing their effects is essential
- Next Steps
  - Potential for increased complexity of combustion simulations with more dimensions is massive
  - Connect the DNN models directly to a simulator and run entire dynamics through to compare results on 3D case
  - Demonstrate ability to do more
    - Try 5-8 input values with more supervised data

Thank you – See you at the Posters!

Compact Representation of a Multi-dimensional Combustion Manifold Using Deep Neural Networks

Sushrut Bhalla[1], *Matthew Yao*[2], *Jean-Pierre Hickey*[2], ***Mark Crowley***[1]

1 - Electrical and Computer Engineering

2 - Mechanical and Mechatronics Engineering

WATERLOO.AI
WATERLOO ARTIFICIAL INTELLIGENCE INSTITUTE

UNIVERSITY OF WATERLOO

Thanks to funding from :

NSERC CRSNG

My Research Vision

To augment human ***decision making*** *in complex domains and environments in a* ***dependable*** *and* ***transparent*** *way.*

Domains (where does the complexity come from?)

- Forest Fire, Invasive Species Spread
  - Catastrophy prediction, strategic management, decision aids
- Medical Imaging
  - Diffusion MRI, Digital Pathology
- Automotive
  - Human Driving Behaviour Learning
  - Autonomous Driving
  - Surroundings/Object classification and understanding
  - LIDAR pointcloud analysis

Domains

Tasks (what are we trying to do?)

- Decision Making Under Uncertainty
- Anomaly Detection
- Classification
- Prediction

Tasks

Methods (how do we solve it?)

- Reinforcement Learning
- Deep Learning
- Ensemble Methods
- Data/Dimensionality Reduction

Methods