**Path Search of 5D Mill Parameters Using Dense Neural Networks as a Compact Polygonal-Approximation of Constraints**

Outline

* Abstract
  + Thesis: A ReLU unit neural network can approximate the constraints
* Background: Separators
  + Explain linear separators for single and multi-layer perceptron
  + Can encode relationships between many arbitrary constraints
  + Can do piece-wise approximation of constraints.
* Background: Milling
  + Explain 5D milling, paths, and milling parameters
  + Literature review on milling optimization.
* Problem Setup: Optimizing over a Dynamic System
  + We are going to optimize the feedrate while obeying all constraints on force and while the cutter engagement is dynamically changing with each state.
* Environment
  + Describe how events will be simulated.
  + List the specific functions used for each relationship/property of the environment.
* Proposed Methodology
  + Encode constraints on valid/invalid states with neural networks.
    - This allows for efficient storage of piece-wise linear separators as well as how those linear separators are inferenced over to create the decision boundary.
  + Search the space that the NN surface constrains so as to optimize feedrate while never entering an invalid state.
    - Thus we do not need advanced engineers to create complex control systems of interlinking differential equations. We can simply specify valid or invalid, and the NN automatically learns to approximate the constraints.
* Experiment
  + Get an autogenerated path from path planning software of some kind.
  + Optimize the feedrate of the path.
  + Compare completion time to the path planning software.
* Conclusion
  + Reaffirm the thesis of the abstract.
  + Reaffirm relevance of results.
  + Highlight applications of this method.
  + Highlight potential future work related to this method.
    - Improve training time by exploiting the properties of the NN (and with a tightest-positive-bound approach).
    - Account for linear collision checking with surface between two states.
    - Use continuous time instead of discrete time.