Statement of Work for Research Assistant (RA) Tony Liu for the period of October – December 2020:

Hyperthesis is a research project advancing the theory and application of hypergraphs as models for complex multiway data. The primary December deliverable for Hyperthesis is an updated release of PNNL’s open sourced Python library, HyperNetX (HNX), a collection of methods and classes for the analysis of data modeled as a hypergraph. The new version will include optimized methods written in C++ and bound to a Python namespace for seamless integration into the existing framework. The optimized framework will support the practical use of HNX on very large datasets, increasing its usefulness to our sponsor.

Deliverables for an optimized version of HNX with Tony’s deliverables highlighted as RA deliverable.

1. Introduce a more efficient data structure for static (immutable) hypergraphs with core functions supported by the C++ backend. Suggested data structures are adjacency dictionaries and/or COO format.

**RA deliverable** – Advise project on best data structure to use and support protocols for passing data structure between Python and C++

1. Core functions currently written in Python should will be optimized in C++ for the static hypergraphs. These include but may not be limited to:
   1. Collapsing nodes and/or edges
   2. Computation of k-cells and/or the associated Abstract Simplicial Complex
   3. Computation of toplexes
   4. Restriction to nodes and/or edges
   5. s-connected components
   6. s-diameter
   7. s-distance
   8. s-neighbors
   9. s-shortest paths

**RA deliverable –** Write the C++ software for the core functions of HNX using optimized NWGraph/Hypergraph libraries.

1. Replace the current NetworkX dependencies for static hypergraphs with optimized C++ methods. This should include but not be limited to:
   1. s-betweenness centrality
   2. s-harmonic closeness centrality
   3. s-eccentricity

**RA deliverable –** Write the C++ software for NetworkX replacement using optimized NWGraph/Hypergraph code.

1. Replace the current Python implementation for computing mod2 homology groups and Betti numbers with optimized C++ methods.

**RA deliverable –** Write optimized C++ software using existing algorithms currently implemented in HNX.

1. Introduce a PyBind module linking the C++ backend to a Python namespace. This should have a plug and play design so that expansion of the namespace for additional functions can be easily accomplished.

**RA deliverable –** Coordinate with Andrew Lumsdaine and Jesun Firoz to ensure seamless connections between backend and front end. RA is not responsible for writing the PyBind code.