Title of Research Project: A Novel Framework for Model Checking UDP Network Interactions

1.) Reflectively describe your literature research process. Tell us how you used library resources or services of any kind (from ILL to online databases to archives collections to meeting with your liaison librarian). *

My research involved a developing field of computer science known as software model checking. Since its inception in the 1990s, automated model checking systems have remained largely within the parlay of academic research. They are invaluable resources: they are used to develop automatically a mathematical model of a software program and mathematically prove that the model is valid. Because they can prove that software does not fail, they are used to evaluate complicated, safety-critical systems such as airplane computers and NASA spacecraft.

The first model checking system developed, SPIN, is virtually undocumented but for two books published by its developers. My library research began in September of last year and involved scoping out virtually all tutorials and resources available for SPIN and making frequent interlibrary loan requests for the two available books on SPIN. The project changed directions and we decided to use a different system called Java Pathfinder developed by NASA; it, too, is relatively undocumented.

My project involved applying Java Pathfinder software to a networking program, but it required an extension to Pathfinder since the software does not natively support networking. At this point, library research became the central guide for my project. After a definitive search of the literature, I identified several papers focusing on networking for Java Pathfinder. The library’s collection of published dissertations and thesis proposals provided a valuable but underutilized research for broad reviews of the field. The library’s Association for Computing Machinery Digital Library was invaluable for identifying relevant publications. I identified Cyrille Artho, an expert developer of networking extensions, but his were not applicable to our project. After further searching the Digital Library and extensively on Google Scholar for cross-referenced and forward-cited works, I identified one paper providing a Pathfinder extension called Netstub for our type of networking application known as UDP. Unfortunately, this paper failed to completely model the UDP network making its use in reliably proving models extremely limited. The paper proposed developing a full UDP model as future work.

I determined that there presently existed no complete model of UDP networks for use with Java Pathfinder. My library research led me to write my own model using the protocols presented in the Netstub paper and the paper by Artho. Based on my research, I am confident that my model is the first system to completely model the UDP networking protocol for use with Java Pathfinder. Without the library’s search system, I would not have realized the necessity for my research to the field of computer science model checking. Moreover, I discovered that the developers of Java Pathfinder recently announced that they are actively seeking development of UDP model checking extensions for Pathfinder.

Computer science research is unique in that it requires knowledge of both academic publications and technical documentation. In addition to articles, I read NASA and SPIN’s technical documentation, and I conducted searches on online open source software repositories.
2.) Tell us about a challenge you faced while doing literature research and how you overcame it. *

One central challenge to my research was the general absence of reliable information on model checking networking systems within the scholarly community. Model checking is a commonplace technique for verifying that large, complicated software systems will not fail; naturally, developers want to use model checking to verify software networks, which are large and complicated systems. Unfortunately, this task is far from trivial. Java Pathfinder, developed by NASA, is among the most popular model checking systems but it lacks support for networking operations, a significant limitation.

Supporting networks is relatively straightforward for Java Pathfinder. Two common networking protocols exist: TCP/IP and UDP networks. Support for both is required to make Java Pathfinder useful for network model checking. While a large amount of literature exists on TCP/IP networking, virtually none exists for UDP networking with Pathfinder.

Determining how to apply model checking to UDP networks was a significant challenge. Papers outlining approaches to modeling TCP/IP networks present methods vastly simpler than those needed for modeling UDP. While information in these papers and others on model checking was useful, the papers did not provide a guide for overcoming the challenges involved in UDP modeling. The only paper offering a system for UDP model checking elected to simplify the system in order to avoid encountering these challenges. I needed a complete rather than simplified UDP model but had little guidance on how to develop one.

I turned to a more general literature review for guidance. After finding all papers citing the UDP paper, I had found no viable methods for model checking UDP. Instead, I had to survey papers creatively on simplifying and increasing the efficiency of mathematical models in order to make a complete UDP networking model tractable. Fortunately, there were multiple dissertations and papers on these topics. After a significant literature search, I could combine these with the UDP paper to develop an approach to crafting my own model checking system.

Lastly, much of the literature on model checking is published outside of the United States. Some papers required conducting searches for foreign language articles and I also had to research scientists working abroad. The library’s electronic resources made finding international articles and resources as simple as those available domestically.

3.) How does your research contribute to the scholarly conversation in your field? What is the significance of your research, in layperson’s terms? *

Developers of massive and complicated networking systems such as Facebook, Dropbox, or Google need ways to verify that their computer systems will not fail because even brief failures could have far-reaching consequences. Model checking is a mathematical method for verifying a mathematical model of a complicated system to ensure that it will not fail and that it will abide by various desired rules; for example, Dropbox may want to verify that two users will not access one file simultaneously.
While model checking is the ideal solution for verifying these types of systems, model checking computer networks is a challenging problem first addressed in the early 2000s, and little progress has been made since. Two types of networking protocols, TCP/IP and UDP, are used commonly in networks. Support for modeling UDP networks is not available in one of the most popular model checking systems, NASA’s Java Pathfinder.

My research developed a software library that extends Java Pathfinder to support UDP networks. It is the first system to completely model the UDP network; a project developed at the University of California Santa Barbara called Netstub provide a UDP networking model for Pathfinder, but it is incomplete, and they cite developing a complete model as an area of future work. Because UDP networks are unreliable and Netstub assumes the network works perfectly, the model checking system is insufficient for testing most applications.

My work provides the first system to realistically model UDP networks for Java Pathfinder. NASA cites this as an extension they would like to use for Java Pathfinder. (As far as I can tell, my work also provides the first literature review documenting the current status of networking extensions for Java Pathfinder.)

After developing the extension, I used it to prove that a UDP networking simulator developed by Dr. Brad Richards did not fail. Dr. Richards uses the system in his networking class for every student project, and it is imperative that the system does not fail. My work ensures that his system behaves reliably.