Proposal

Building a Resilient Control Scheme over Unreliable Wireless Networks

Purpose

The purpose of this study is to develop a method of network communication and control of a remotely drivable research robot to enable effective network control. This can be thought of as a specific example of the general problem of establishing reliable communication over an unreliable network, such a last-mile internet connection over failing cable or a satellite internet connection, and will build off other efforts to minimize the effects of latency such as those applied in game development [2] and in other cases such as Voice over IP calling [3].

Methods

To increase network reliability, three methods will be used: operating system tuning, network hardware modification, and software mitigation of network error.

The first method, tuning of the operating system's network options to optimize for the type of connection used by the robot (where low latency is more important than intermittent packet loss) will be done by tuning the Linux kernel parameters on the robot. The Linux kernel provides several methods of doing such, and no kernel recompilation is required, so this method should be the simplest, and can quickly be tried with many different

combinations of parameters to determine the optimal values for each parameter.

The second method, network hardware modification, will add additional wireless hardware to the robot to introduce redundant network links. Known as multi-homing, this practice can increase reliability of network connections as when one connection drops, the second can resume the network connection with minimal downtime. Additionally, better antenna selection and placement (as currently the antenna is inside the metal shell of the robot, which likely greatly attenuates the signal) will allow for better range and longer intervals between connection drops.

The third method will be software mitigations of network errors. This will involve creating a watchdog, a timer that stops the robot when connection with the controller is lost. Additionally, software to accurately synchronize time between the controller and computer will be installed, and all requests will be timestamped, with requests older than a certain time being discarded. Collectively, this should minimize the negative effects felt when the robot jumps wireless networks.

Validation of the effectiveness of these methods will be done using by simulating a poor network connection using dummynet [2], where packet loss, periodic delays and disconnects, bandwidth limitations, and high latency can be effectively and repeatably simulated. A test suite will be developed that takes a reasonable range of conditions and subjects different versions of the hardware and software to those tests.

Outcome

This research will explore the problem of attempting to limit the effects of an inconsistent network [3] in a case where high responsiveness is essential. The resulting work should not only make the robot drive better but be extensible to other network-controlled devices, and allow smooth control over intermittent network connections. It will also provide quantitative data on which of the modifications made produce the best results.

References

1. https://www.technologyreview.com/s/408909/reducing-lag-time-in-online-games/
2. http://info.iet.unipi.it/~luigi/dummynet/
3. https://ieeexplore.ieee.org/abstract/document/6967689