

Adapting TFRC to Mobile Networks with Frequent Disconnections

Olivier Mehani

National ICT Australia, and
University of New South Wales, Australia, and
Mines ParisTech, CAOR — Centre de
Robotique, Mathématiques et Systèmes, France
olivier.mehani@nicta.com.au

Roksana Boreli

National ICT Australia
Locked Bag 9013
Alexandria NSW 1435, Australia, and
University of New South Wales, Australia
roksana.boreli@nicta.com.au

ABSTRACT

In the context of mobile and pervasive networking, it is not uncommon to experience frequent loss of connectivity. Congestion control algorithms usually mistake resulting packets losses for congestion events and unnecessarily reduce their network usage. We propose an enhancement to the TCP-Friendly Rate Control (TFRC) protocol to overcome this issue. Our contribution proposes to suspend the data transfer, in a way similar to Freeze-TCP, when an upcoming disconnection is predicted. Furthermore, a probing mechanism is introduced to enable fast adaptation to new network conditions. We present a description of the additional freezing and resuming mechanisms. This proposal has been implemented in ns-2. Simulation results are thus compared to the regular behavior and show encouraging improvements.

Keywords

TFRC, congestion control, MANET, mobility, DCCP

1. INTRODUCTION

Recent years have seen the emergence of an increasing number of portable devices. Due to their relative mobility, though, it is not uncommon that a network path changes over the lifetime of a data session. Routing methods can be used to take care of nodes moving inside a Mobile Ad-hoc Network (MANET) [6]. Similarly, IP mobility support schemes can handle horizontal or vertical handoffs [5].

These approaches however introduce disconnected periods when changing network paths or updating bindings. Packets sent during these periods are usually lost.

The most common response of congestion control mechanisms to these losses is to reduce the packet-sending pace. Hence, these mobility enhancements have a disruptive impact on transport protocols.

This contribution introduces a mobility-aware feature for the TCP-Friendly Rate Control [2] protocol. Packet losses are prevented by “freezing” the connection before a handoff occurs. When connectivity is back, the data transfer is restored at its previous rate. The network is then probed for higher capacity. This allows for quick adjustments to better connections (*e.g.* faster technology or less congested network).

We purposefully focus on the transport layer only, assuming abstract information about (dis)connections is available. Additionally, no assumption is made about the mobility scheme in use, as most of those generate temporary disconnections of variable duration.

Our preliminary simulation results show that this enhancement of TFRC enables better use of the network.

2. MOTIVATION

Freeze-TCP [3] has been proposed to suspend a TCP connection in the event of an upcoming wireless disconnection. When in use, no packets are lost during the disconnected period. As soon as connectivity becomes available again, the data communication can resume at the same rate as before.

Additionally, it is argued in [1] that rate-based flow control is well suited to MANETs. Mobility schemes at the network layer exhibit the same type of behavior as MANETs, with short but possibly frequent disconnections. Thus, it is reasonable to assume that such a mechanism would perform similarly in these contexts.

Hence, we propose to enhance TFRC [2], a rate-based congestion control mechanism. TFRC mimics the behavior of TCP based on receiver-observed parameters, including the packet loss rate p . Moreover, it can be used by DCCP [4] as its congestion control algorithm. DCCP itself provides for an unreliable but congestion-aware datagram transport protocol. This is highly de-

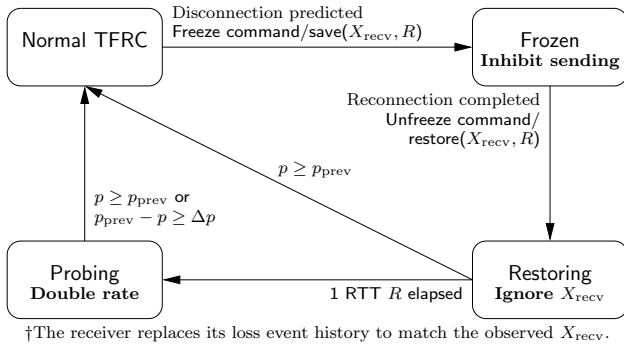


Figure 1: Additional states required for a TFRC sender to freeze and unfreeze its connection.

sirable for some of the key applications of mobile and pervasive networking such as media streaming or VoIP.

3. PROPOSED ENHANCEMENTS

Freezing and restoring a TFRC connection mostly require modifications on the sender’s side. Figure 1 shows the additional states proposed to handle a temporary loss of connectivity and a possible change of network.

Some parameters may locally evolve due to various timers. When a disconnection is predicted, the sender first has to save these parameters. It then enters the *frozen state* and refuses to send any more packets.

Upon an unfreeze instruction, the sender enters the *restoration phase*. It reinstates the saved parameters and resets its timers. Packet sending then resumes at the previous rate. In standard TFRC, the receiver must measure the rate X_{recv} , and report it to the sender. After the reconnection, the initial values of this estimate will not be valid as the measurement spans over the disconnected period. To prevent oscillations in the sending rate, X_{recv} has to be ignored during the first RTT.

If no losses were reported by the end of the restoring phase, the *probing phase* can be entered. By doubling its sending rate every RTT, the sender tries to quickly discover whether the new network can support a higher data rate. The receiver has to check for an increase in the error rate. When this happens, it computes a loss rate matching the observed receive rate and reports it to the sender. The sender, in turn, notices an unexpected change in the error rate and exits the probing phase to revert back to regular TFRC.

4. EVALUATION BY SIMULATION

The proposed additions have been implemented in the DCCP/TFRC agent within ns-2. In the simulations, a node disconnects for two seconds then reconnects to networks with either similar, smaller or larger capacities.

Figure 2 shows that the freeze-enabled version performs better: data transfers restart earlier, and adapt

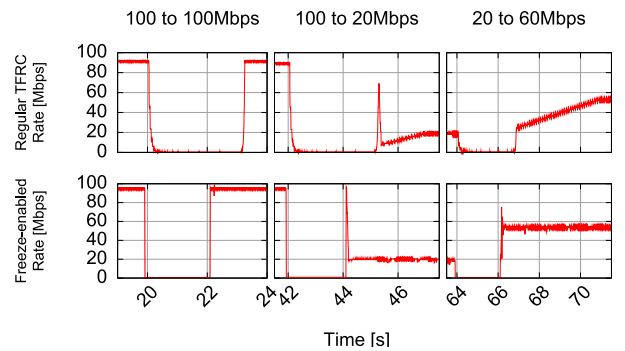


Figure 2: The proposed enhanced TFRC restarts earlier (up to 2 s) and immediately adapts its throughput to the available bandwidth (instead of taking up to 3 s).

more quickly to the new network capacity.

5. CONCLUSION AND FUTURE WORK

We have introduced the idea of temporarily suspending a TFRC connection before a predicted disconnection occurs. Preliminary simulation results show that this contribution allows for a quicker and better network usage upon reconnection.

Besides refining the simulation scenarii, future work will focus on a real implementation and live experiments to validate the simulation results. More attention will be paid to protocol exchanges and how to inform the transport of predicted disconnections. Additionally, it is important to verify that these modifications do not badly impact on TFRC’s fairness with TCP flows.

6. REFERENCES

- [1] Kai Chen and Klara Nahrstedt. The utility of explicit rate-based flow control in mobile ad hoc networks. In *IEEE WCNC '04*, 2004.
- [2] S. Floyd, M. Handley, J. Padhye, and J. Widmer. Equation-based congestion control for unicast applications. In *ACM SIGCOMM '00*, Stockholm, Sweden, September 2000.
- [3] T. Goff, J. Moronski, D. S. Phatak, and V. Gupta. Freeze-TCP: A true end-to-end TCP enhancement mechanism for mobile environments. In *IEEE INFOCOM '00*, 2000.
- [4] E. Kohler, M. Handley, and S. Floyd. Designing DCCP: Congestion control without reliability. In *ACM SIGCOMM '06*, September 2006.
- [5] E. Perera, V. Sivaraman, and A. Seneviratne. Survey on network mobility support. *ACM SIGMOBILE: Mobile Computing Communications Review*, 8(2), 2004.
- [6] L. Qin and T. Kunz. Survey on mobile ad hoc network routing protocols and cross-layer design. Technical report, Carleton University, 2004.