Universal Scheduling for Networks with Arbitrary Traffic, Channels, and Mobility

Scott Rager, Thomas La Porta

Introduction

Recent work by Michael Neely has developed a framework for an algorithm that attempts to maximize utility in networks with arbitrary channels, mobility, and traffic. In order to accomplish this goal, this *universal scheduling* algorithm examines random events occurring in the network, such as mobility or channel fading, and chooses corresponding control actions that define policies across several network layers including flow control, resource allocation, and routing. Using such an algorithm, we hope to improve performance of unpredictable networks by not relying on stochastic assumptions. For example, in figure 1, if node B fails, we would like a fast, reliable algorithm that automatically uses the alternate paths available.

The goal of this project is to implement this universal scheduling algorithm in a simulation environment to test its performance capabilities and to develop a fully distributed implementation suitable for unpredictable mobile ad hoc networks.

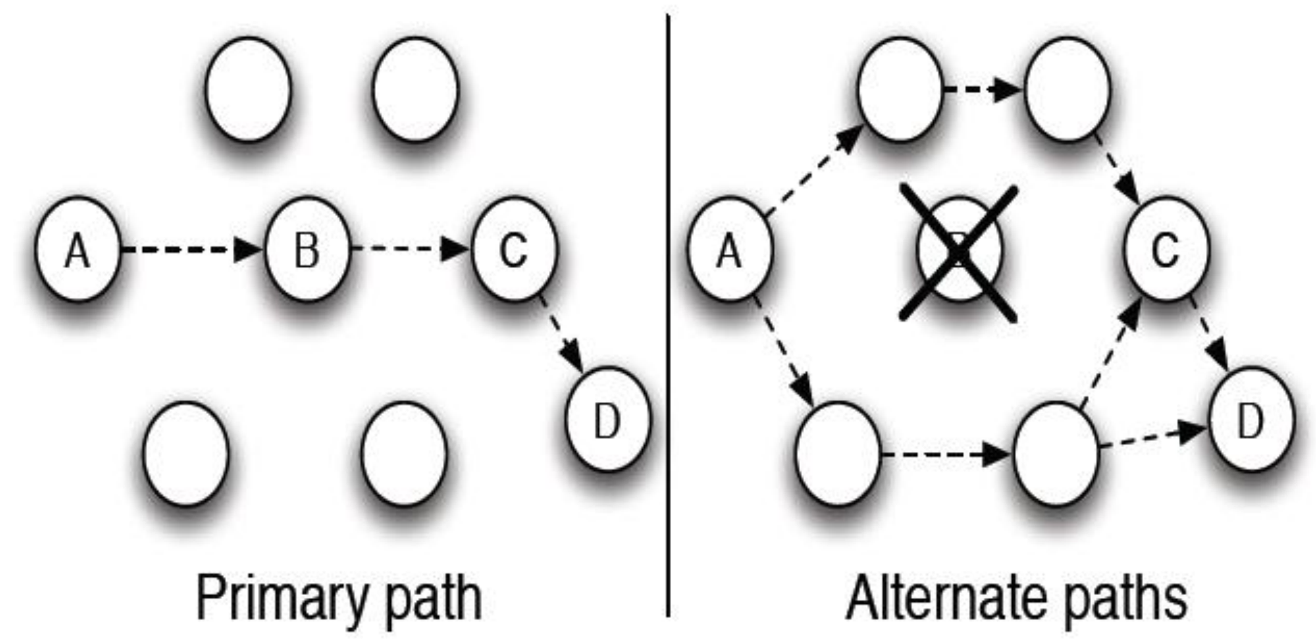


Fig. 1. A primary path from A to D, with alternative paths shown in the event of a failure at node B.

Algorithm Overview

The algorithm works by having the network controller observe the state of the network at each time slot and make the following decisions across several network layers:

1. **Fairness** –

   Choose $\gamma(t)$ to maximize: $V\phi_m(\gamma_m(t)) - H_m(t)\gamma_m(t)$

   Subject to: $0 \le \gamma_m(t) \le A_{Max}$

2. **Flow Control** -

   Choose $x_m(t)$ to maximize: $H_m(t)x_m(t) - Q_{n_m}^{(c_m)}(t)x_m(t)$

   Subject to: $0 \le x_m(t) \le A_{Max}$

3. **Routing and Resource Allocation** -

   Choose $I(t)$ to maximize: $\sum_{i=1}^{N}\sum_{j=1}^{N} C_{ij}(I(t), S(t))W_{ij}(t)$

   Subject to: Valid Resource Allocation Scheme

Where $\phi(\cdot)$ is the utility function, $x_m(t)$ is admitted traffic, $H_m(t)$ is a virtual queue, and $Q_n^c(t)$ represents actual queue backlogs.

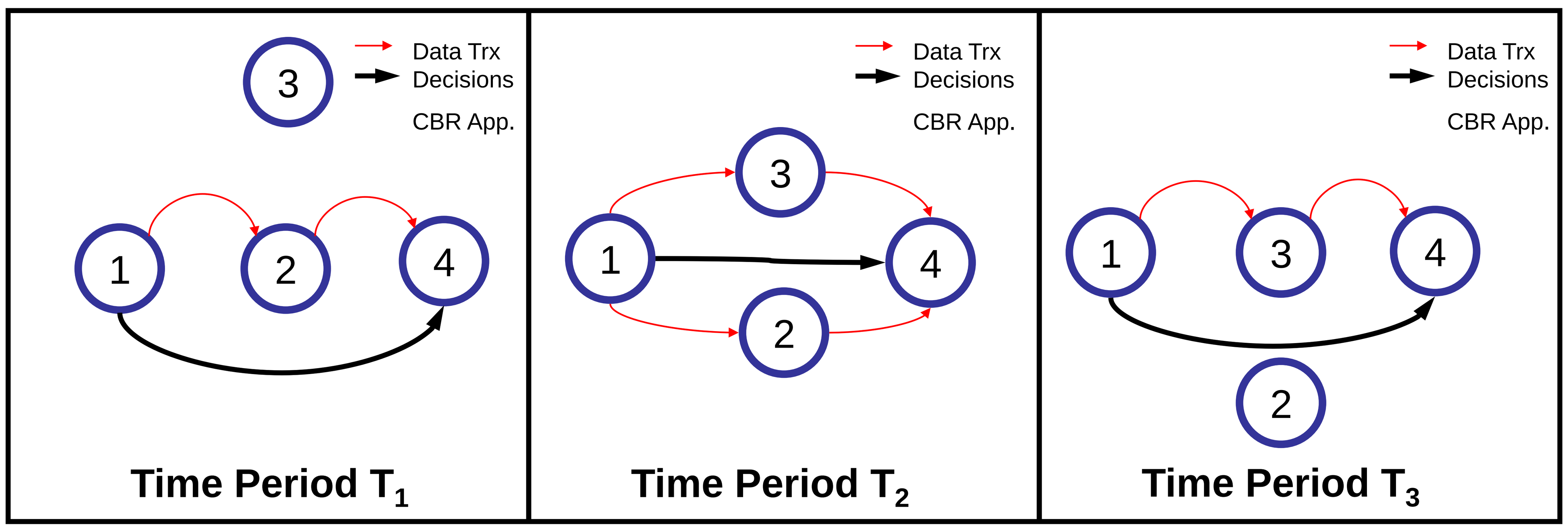
Future Work

To date, we have implemented the universal scheduling algorithm using a central network manager that is assumed to have global knowledge of network states and conditions. In working toward a more practical application for MANETs, we will explore several directions:

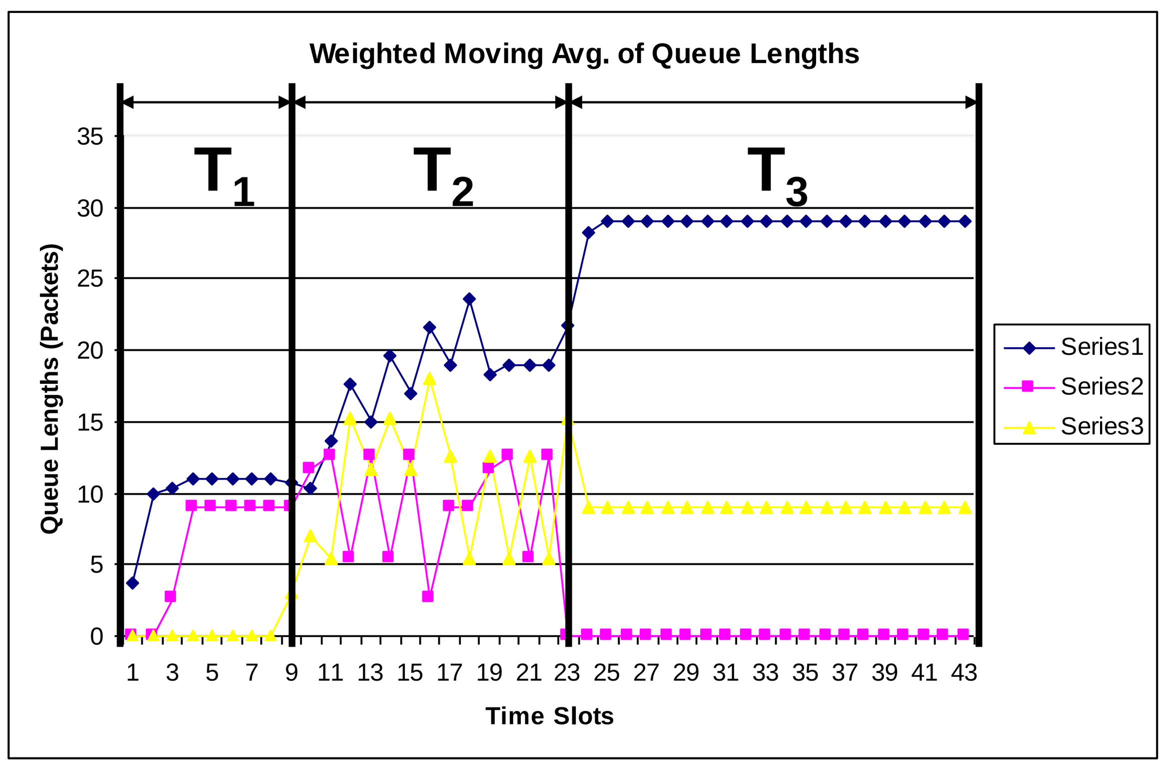
- Develop protocol for universal scheduling in a completely distributed setting, taking advantage of eavesdropping nodes and control packets to disseminate network information and make decisions using local knowledge.
- Incorporate multiple commodities in determining utility of data, instead of just throughput, e.g., power, QoI, or delay/staleness of data.
- Extend algorithm to take advantage of a priori knowledge of network, e.g., known mobility patterns, planned node shutdowns, etc.

Example Scenario

As an example of how the universal scheduling protocol works, we can examine this simple example of one network flow in a four node network with arbitrary mobility.

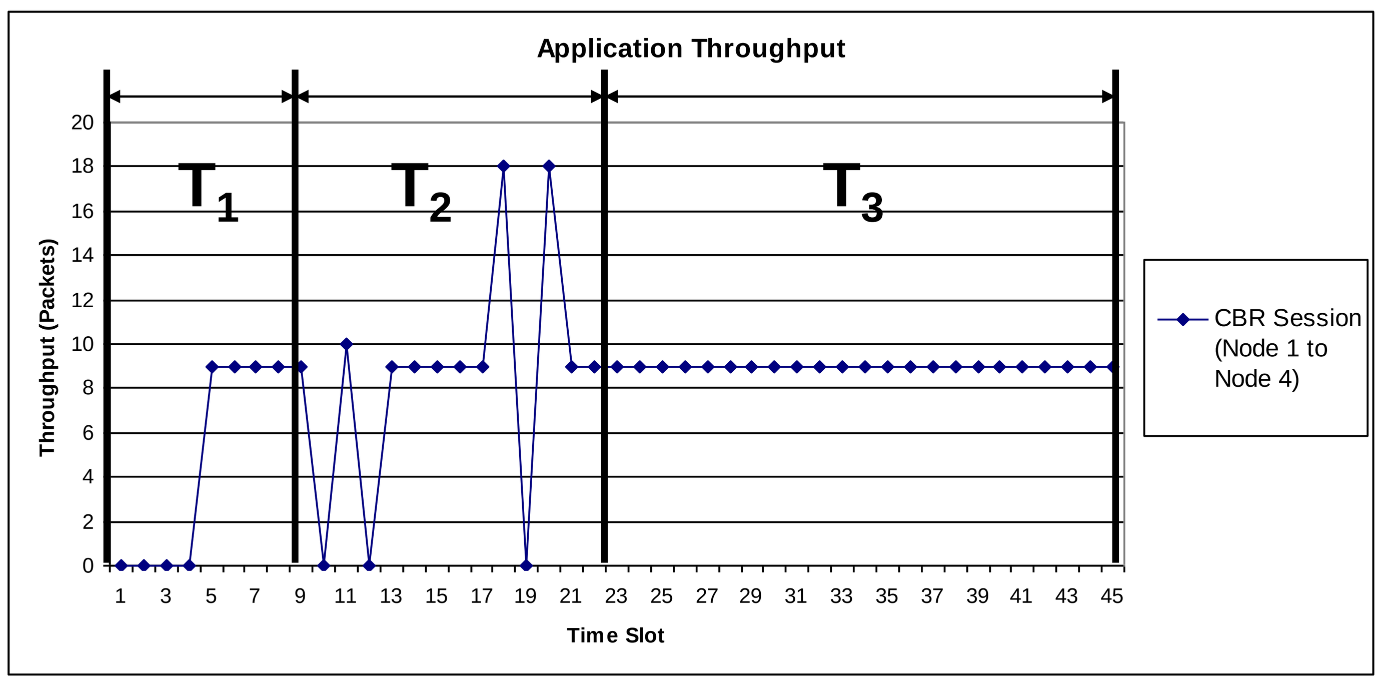


At T_1, node 2 is within range of nodes 1 and 4. In T_2, node 3 moves into range, such that both 2 and 3 can route session traffic. Finally, in T_3, node 2 moves out of range, leaving only node 3 to forward session data.



Looking at the lengths of each queue as time passes, we can see that node 1 utilizes node 2 while possible, and then transitions to using both nodes 2 and 3 before node 2 moves out of range, leaving 3 to forward all traffic.

By examining the throughput at each time slot, we can see the advantage of using the universal scheduling algorithm. It is able to transition smoothly between intermediate nodes, taking advantage of when both are able to forward traffic.



Work Sponsored by US-ARL Communications Technology Alliance (NS-CTA)