

Progress Report: Congestion Control Implementation

9th NDN Retreat

Klaus Schneider, Eric Newberry, Chavoosh Ghasemi, Beichuan Zhang

December 14, 2017

The University of Arizona

Motivation: Why Congestion Control Is Important

Crucial part of **Application Performance**

Motivation: Why Congestion Control Is Important

Crucial part of **Application Performance**

Examples:

- Hadoop on NDN

Motivation: Why Congestion Control Is Important

Crucial part of **Application Performance**

Examples:

- Hadoop on NDN
- Frequent questions about **ndnchunks** performance (mailing list)

Motivation: Why Congestion Control Is Important

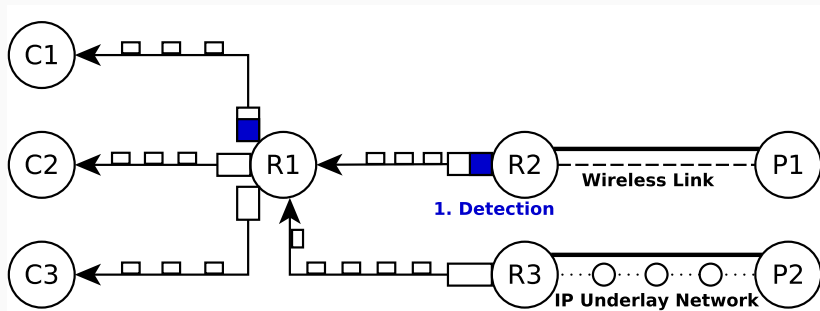
Crucial part of **Application Performance**

Examples:

- Hadoop on NDN
- Frequent questions about **ndnchunks** performance (mailing list)
- Also: **rate-adaptive video, wireless networks**, etc.

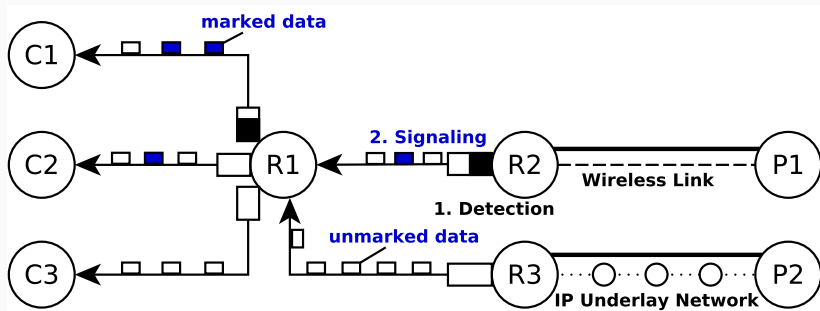
Implementation Overview

From PCON paper [1]:



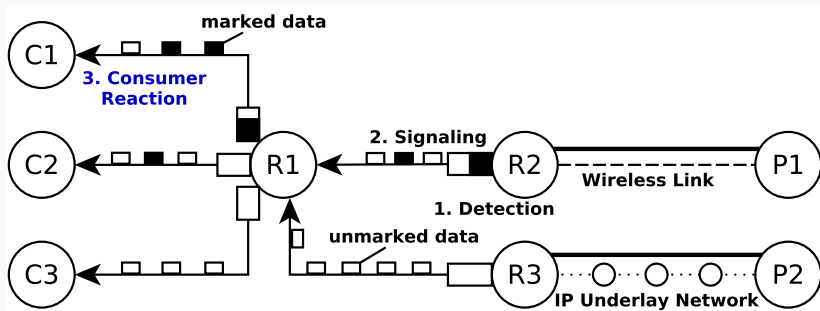
Implementation Overview

From PCON paper [1]:



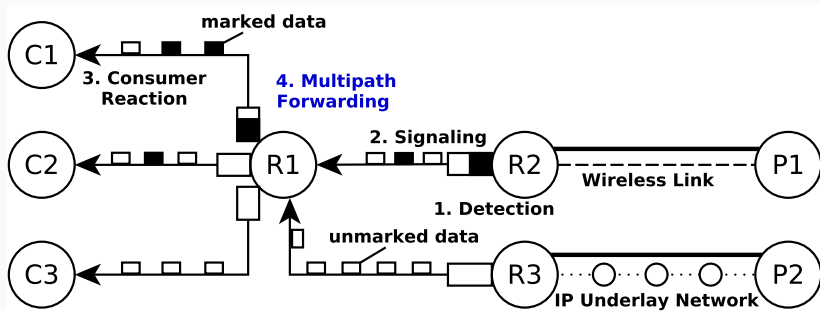
Implementation Overview

From PCON paper [1]:



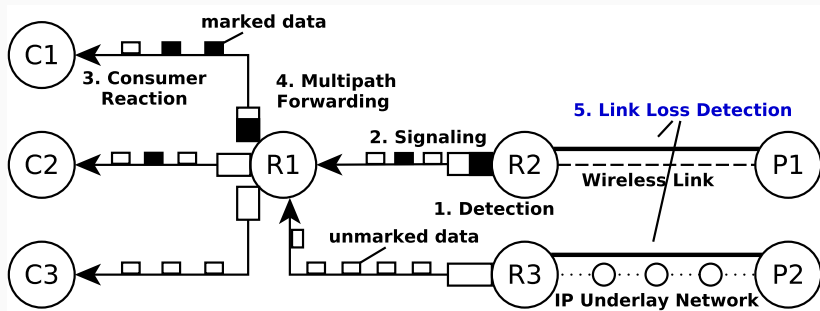
Implementation Overview

From PCON paper [1]:



Implementation Overview

From PCON paper [1]:



Implementation Overview: Redmine

Design and Implement Congestion Control

Added by Junxiao Shi over 3 years ago. Updated 9 days ago.

Status:	New	Start date:	
Priority:	High	Due date:	
Assignee:	Klaus Schneider	% Done:	<div><div></div></div> 38%
Category:	Forwarding	Estimated time:	(Total: 6.00 h)
Target version:	-		

Description

Quote

Design a congestion control scheme.

design-20160229.pdf (138 KB)	Jeff Burke, 03/06/2016 09:37 AM		
draft_own_cong_marks.pdf (816 KB) data for note 5	Jeff Burke, 03/13/2016 06:37 PM		
draft_multi_for_bic3.pdf (1.06 MB) TCP BIC Performance	Klaus Schneider, 03/29/2016 11:22 PM		
draftmulti_for_aimd3.pdf (1.07 MB) TCP Reno Performance	Klaus Schneider, 03/29/2016 11:22 PM		

Subtasks

[Add](#)

Feature #3797: Congestion Control: generic congestion marks	Closed	Eric Newberry	<div></div>
Feature #3823: Congestion Control: design Local Link Loss Detection	Closed	Eric Newberry	<div></div>
ndn-tools - Feature #4289: ndncatchunks: React to congestion marks	Closed	Chavoosh Ghasemi	<div></div>
Feature #4327: Congestion mark integration test case	In Progress	Eric Newberry	<div></div>
Feature #4362: Congestion Detection by observing router queues	New		<div></div>
Task #4363: Adjust Multipath Forwarding Strategy based on Congestion Marks	New		<div></div>
Feature #4364: Create Consumer API for Congestion Control	New		<div></div>
Task #4391: Congestion Control: Test Local Link Loss Detection	Feedback	Eric Newberry	<div></div>

What We Accomplished since last Retreat

Thanks to Eric, Davide, Chavoosh, Junxiao, and others.

What We Accomplished since last Retreat

Thanks to Eric, Davide, Chavoosh, Junxiao, and others.

1. **Generic Congestion Marks**

- Defined in NDNLP
- Simple API with getters & setters

What We Accomplished since last Retreat

Thanks to Eric, Davide, Chavoosh, Junxiao, and others.

1. **Generic Congestion Marks**

- Defined in NDNLP
- Simple API with getters & setters

2. **Consumer Congestion Adaptation**

- Catchunks: AIMD, react to congestion marks

What We Accomplished since last Retreat

Thanks to Eric, Davide, Chavoosh, Junxiao, and others.

1. **Generic Congestion Marks**

- Defined in NDNLP
- Simple API with getters & setters

2. **Consumer Congestion Adaptation**

- Catchunks: AIMD, react to congestion marks

3. **Local Link Loss Detection (NDNLP)**

- Detect lost packets (via gap in SeqNr or ACK Timeout)
- Signal to forwarding strategy **onLostInterest()**.

Future Work and Timeline

1. **Integration tests** (2 months)

- For congestion marks & link loss detection
- Check if current functionality works as expected

Future Work and Timeline

1. **Integration tests** (2 months)
 - For congestion marks & link loss detection
 - Check if current functionality works as expected
2. **Cong. Detection** based on **queue backlog** (6 months)
 - To work on TCP/UDP Tunnels, Ethernet, Wireless
 - \Rightarrow See our **Hackathon Project!**

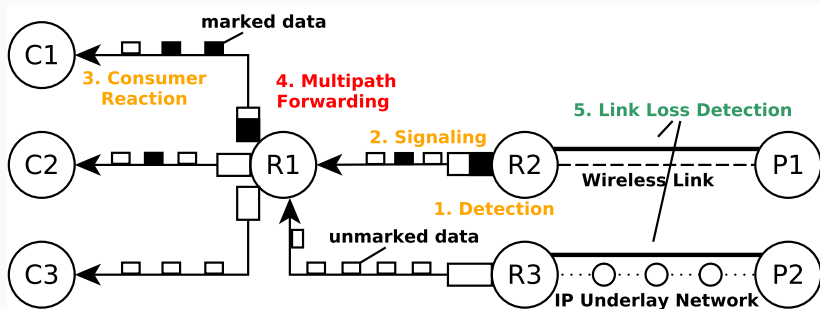
Future Work and Timeline

1. **Integration tests** (2 months)
 - For congestion marks & link loss detection
 - Check if current functionality works as expected
2. **Cong. Detection** based on **queue backlog** (6 months)
 - To work on TCP/UDP Tunnels, Ethernet, Wireless
 - \Rightarrow See our **Hackathon Project!**
3. **Consumer/Producer API** (9 months)
 - Look at Ilya's work and Cisco's libcnet API

Future Work and Timeline

1. **Integration tests** (2 months)
 - For congestion marks & link loss detection
 - Check if current functionality works as expected
2. **Cong. Detection** based on **queue backlog** (6 months)
 - To work on TCP/UDP Tunnels, Ethernet, Wireless
 - \Rightarrow See our **Hackathon Project!**
3. **Consumer/Producer API** (9 months)
 - Look at Ilya's work and Cisco's libcnet API
4. **Multipath Forwarding** (needs more design)

Progress in one Picture



Any Questions?

Klaus Schneider

`klaus@cs.arizona.edu`

`https://www.cs.arizona.edu/~klaus/`

- [1] Klaus Schneider, Cheng Yi, Beichuan Zhang, and Lixia Zhang.
A practical congestion control scheme for named data networking.
In *Proceedings of the 2016 conference on 3rd ACM Conference on Information-Centric Networking*, pages 21–30. ACM, 2016.