UiO : Faculty of Mathematics and Natural Sciences
University of Oslo

ifi

Estimating an Additive Path Cost with Explicit Congestion Notification*

Peyman Teymoori, David Hayes, Michael Welzl, Stein Gjessing

{peymant | michawe | steing}@ifi.uio.no, davidh@simula.no

IETF 104 Prague – ICCRG

* Teymoori, Peyman; Hayes, David Andrew; Welzl, Michael; Gjessing, Stein. *Estimating an Additive Path Cost with Explicit Congestion Notification*. Universitetet i Oslo 2019 (ISBN 978-82-7368-452-3).

Overview

- Explicit Congestion Notification (ECN)
- Network Utility Maximization (NUM)
- Solving NUM with ECN
- Model Validation & Simulation Results
- Applications
- Conclusion

Explicit Congestion Notification (ECN)

- Conveying at least one queue along the path is congested
- Inherently more reliable than delay-based signals
- Can be used as a multi-bit congestion signal
- Example: Datacenter TCP (DCTCP)
  - Instantaneous queue
  - Marking all the packets above the threshold
  - Interpret the number of ECN marks over a period of time (e.g. RTT)

ECN Issues

- As one-bit information: routers cannot re-mark CE
- Not a problem with “normal” RFC 3168 ECN
  - marking probability generally kept low
  - uses only 1 signal/RTT
- DCTCP’s use of ECN recognizes: low marking probability
  - diminishes the usefulness of ECN
  - only useful when dealing with packet drops
- It is not additive; it’s multiplicative
  - problems with the NUM theory!

Network Utility Maximization (NUM)

- Solves the rate allocation problem given the network's constraints
- Senders express their happiness wrt send rates by utility functions (usually concave)
- NUM maximizes the sum of utility functions subject to network constraints

$$\max_{x} \quad \sum_{r \in R} U_r(x_r)$$

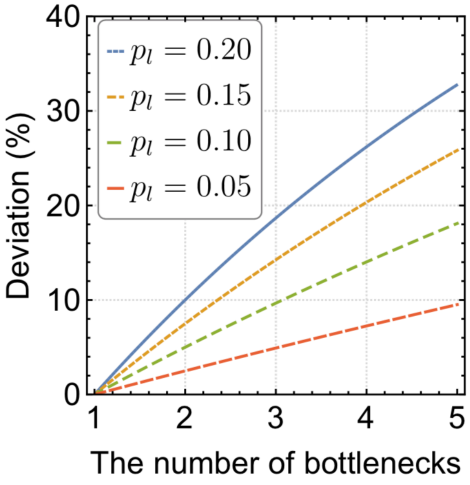
$$s.t. \quad y_l \leq c_l, \quad \forall l \in L$$

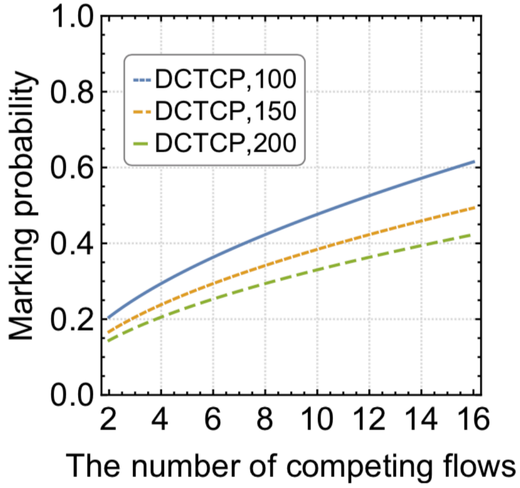
Using ECN as Cost in NUM

- Comparing:

$$p_r = 1 - \prod_{l \in L_r} (1 - p_l)$$

$$p_r = \sum_{l \in L_r} p_l$$





If $p_l \ll 1$, then $1 - \prod(1 - p_l) \approx \sum p_l$

Our Contributions

- Turning ECN into an additive signal and using it for NUM
- Marking can be attained by configuring RED (Random Early Detection) at routers
  - deployed with commodity switch hardware
- Benefit: faster convergence
- Could enable earlier feedback by marking below the capacity
  - virtual queues/phantom queues/proxy queues

KKT Theorem (with our extension)

- Optimality Conditions in the KKT theorem:

$$\begin{aligned} \max_x \quad & U(x) & \\ s.t. \quad & g_i(x) \leq 0, & i = 1, \ldots, M_1 \\ & h_j(x) = 0, & j = 1, \ldots, M_2. \end{aligned}$$

- Necessary conditions:
  - if x* is a local optimum, then
    - stationarity:

$$\nabla U(x^*) = \sum_{i=1}^{M_1} \mu_i \nabla g_i(x^*) + \sum_{j=1}^{M_2} \lambda_j \nabla h_j(x^*),$$

  - our change to the theory:
    - stationarity:

$$\nabla U(x^*) = \sum_{i=1}^{M_1} f(p_i) \nabla g_i(x^*) + \sum_{j=1}^{M_2} k(v_j) \nabla h_j(x^*)$$

Optimization Using ECN

- The function multiplier: $f(p_l) = -\log_\phi(1 - p_l) \qquad p_l \in [0,1]$
-

  Lagrangian function:

$$\mathcal{L}(x,p) = \sum_{r\in R}\Big(U_r(x_r) - x_r\big(-\log_\phi(1-p_r)\big)\Big) + \sum_{l\in L}\Big(c_l\big(-\log_\phi(1-p_l)\big)\Big).$$

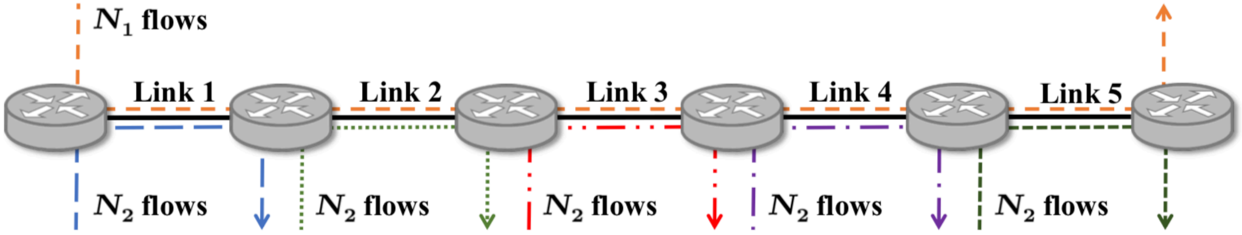
- Then, devising three types of algorithms:
  - Primal
  - Dual (RED)
  - Primal-Dual

Dual (RED)

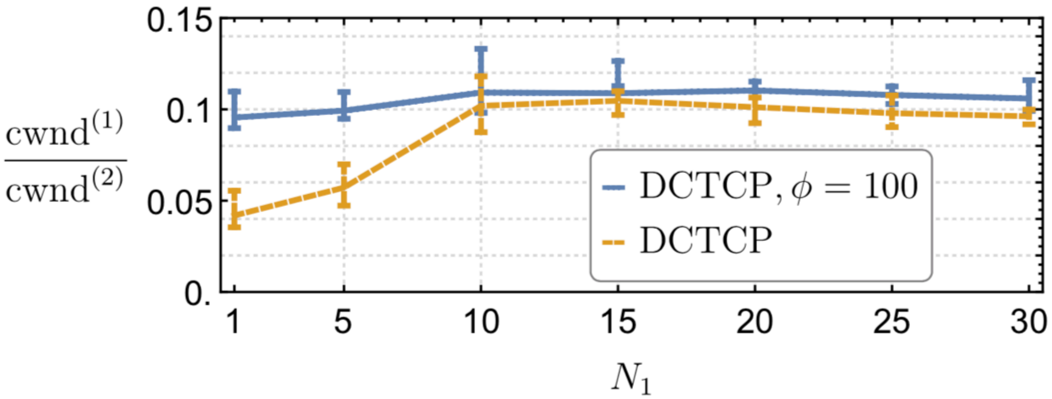
$$p_l[n] = \left[\frac{b_l[n]}{\text{max}_{\text{th}}} \right]_0^1$$

- The algorithm at routers:
- Can be configured by
  - $\text{min}_{\text{th}} = 1$
  - $\text{max}_{\text{th}} >= \alpha\, L\, S$ (stability condition)
    - example: for a scenario in which BDP=1.7 MByte, max_{th} >= 1.3 Mbyte, but we set max_{th} = 220 Kbyte (without observing instability). Average queue size was around 75 Kbyte.
  - $\text{max}_p = 1$
  - $w_q = 1$ (instantaneous queue)

Validation with DCTCP



Instead of p_r (last RTT's marking prob.),
senders use $[-\log_\phi(1-p_r)]_0^1$



Other Optimization Algorithms

- Primal: $$\dot{x}_r = \varrho_r(x_r)\Big(U_r'(x_r) + \log_\phi(1 - p_r)\Big)$$

- Dual1: $$\dot{p}_l = \sigma_l(p_l)\frac{1}{1 - p_l}\Big[y_l - c_l\Big]_{p_l}^{+}$$

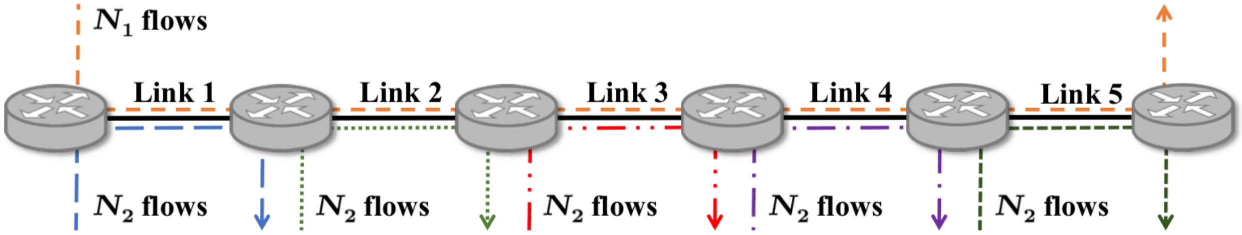
$$x_r = U_r'^{-1}\big(-\log_\phi(1 - p_r)\big)$$

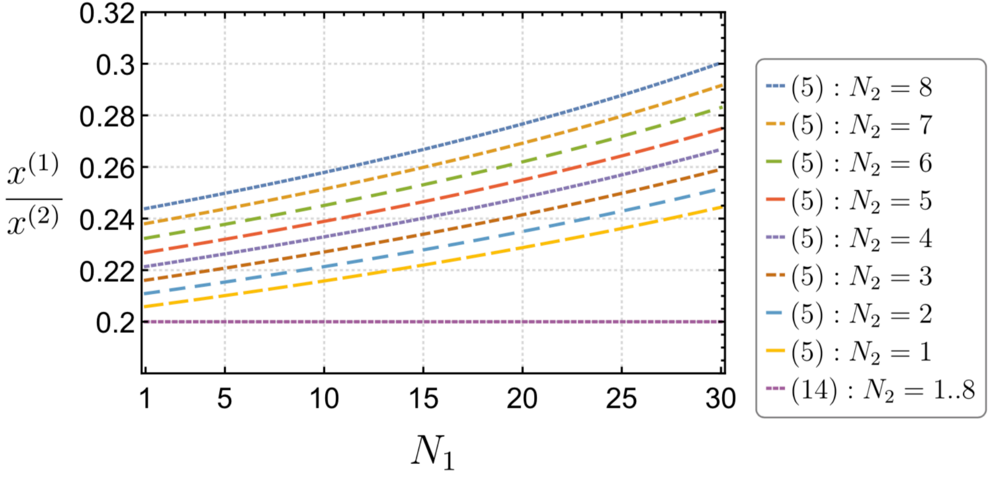
- Dual2: $$\dot{p}_l = \sigma_l(p_l)\Big[y_l - c_l\Big]_{p_l}^{+}$$

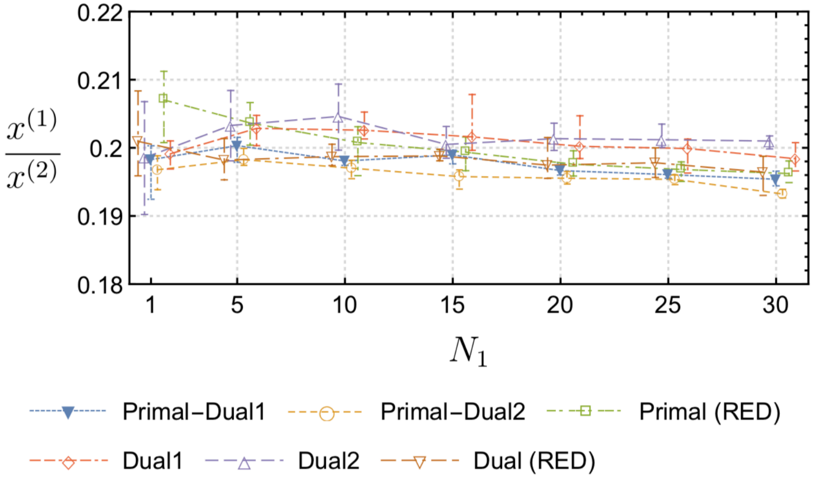
Primal-Dual

Validation with a Logarithmic Utility

$$U_r(x_r) = \mathrm{Log}(x_r)$$







Applications

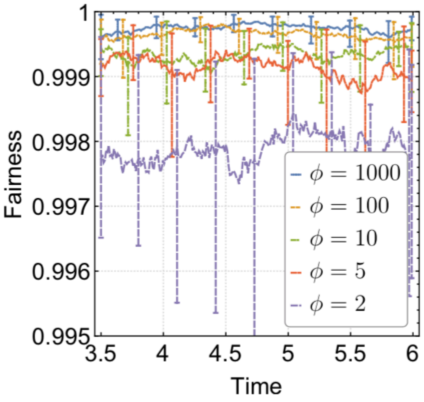
- Obtaining utility function when the marking probability is high, e.g. DCTCP and LGC
- Deflating/inflating marking probability
  - playing with the base of log

$$f(p_l) = -\log_{\phi}(1 - p_l)$$

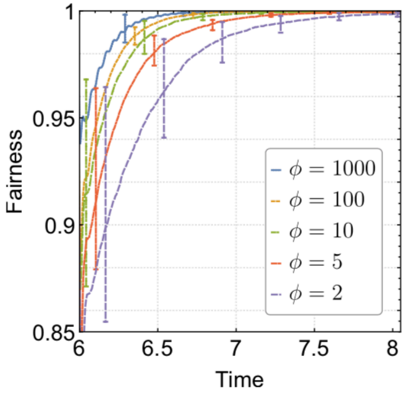
- And the potential of dealing with virtual queues
  - producing an early signal before the physical queue grows

Simulation Results

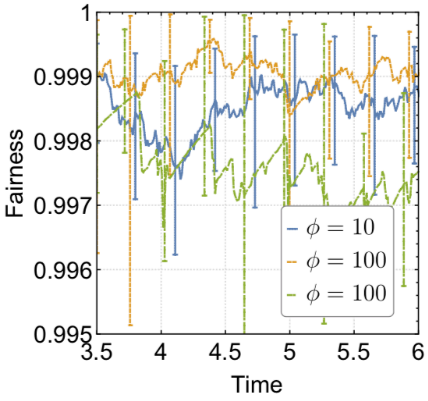
Effects of marking probability on convergence:



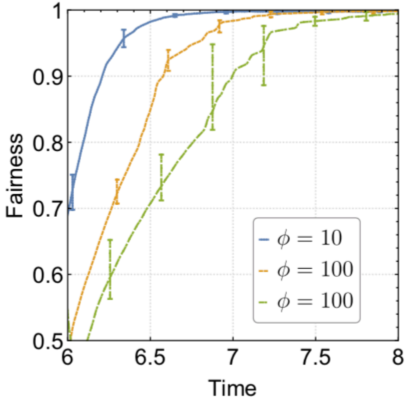
(a) Steady-state short-term fairness



(b) A new flow joins



(a) Steady-state short-term fairness



(b) A new flow joins

$C_l = 1$ Gbps

$C_l = 40$ Mbps

Marking probabilities: 0.031, 0.066, 0.087, 0.149, 0.211, $\cdots$, 0.84, 0.97, 0.99

Conclusion

- Improving ECN – widening its range!
  - faster convergence
  - smoother behavior
  - possibility of marking before queue grows
- Next steps:
  - experiments
  - deployment

Thank you!