An Index Policy for Congestion Control in Routers with Future-Path Information

Peter Jacko
Joint work with José Niño-Mora & Brunilde Sansò

BCAM Seminar
June 2, 2009
Congestion Control in Router
Congestion Avoidance Mechanisms

- 1990’s: **Reactive** mechanisms: queue tail drop
- 2000’s: **Preventive** mechanisms: RED, BLUE, etc.
  - based on router-based measures
    - queue length, packet loss, bandwidth utilization
  - fairness from the sender’s point of view
    - each packet equally important
- 2010+ (?): **Anticipative** mechanisms:
  - fairness from the **receiver**’s point of view
    - packet’s importance depends on the future path
Motivation

- Network scarce resources: bandwidth and buffer space
- Dropping a packet on its route implies:
  - all scarce resources it has consumed so far are wasted
- Anticipating the loss of a packet
  - helps in economical allocation of scarce resources
  - avoids unnecessary packet losses
  - increases network throughput
- How to anticipate? ECN, plain loss estimation, etc.
Outline

- Transmission Control Protocol (TCP)
- MDP formulation of the problem
  - Network-capability fairness
- Relaxation and decomposition into subproblems
- Optimal solution to the subproblems
  - Indexability and transmission index
- Optimal solution to the relaxed problem
- Heuristics for the original problem
Transmission Control Protocol (TCP)

- Implemented at the two ends of a connection
- A way of end-to-end congestion control
- Provides reliable, ordered delivery of a stream of packets
- Fully-sensitive to packet losses
- Examples: web browsers, e-mail, file transfer (FTP)
- Must be distinguished from congestion control in routers
- An alternative (UDP) is used for VoIP, streaming, etc.
TCP End-to-End Connection

- Sender sends an initial packet and waits
- Receiver sends acknowledgment of each received packet, including information about congestion warnings, if any
- Sender sends subsequent packet(s) only after receiving acknowledgement(s)
TCP Dynamics

- We consider restarting-on-warning TCP in discrete time.

- **actualWindow** — actual packet sending rate.

- **Slow Start phase:**
  - `actualWindow` starts at 1 packet per RTT (period).
  - Doubled every RTT with no congestion warning.
  - Until reaching **congestionThreshold**.

- **Congestion Avoidance phase:**
  - Added 1 packet every RTT with no congestion warning.
  - Until reaching **advertisedWindow**.
TCP Dynamics as Markov Chain

- **States:** $n \in \{0, 1, \ldots, N - 1\} = \text{sending rate level}$
  - $n = 0$: sending rate $W^\text{sent}_n$ of 1 packet/RTT
  - $n = N - 1$: maximum (advertisedWindow) rate

- **Transitions:** OK (no warning), NO (congestion warning)
Congestion Control of TCP in Router

- Time epochs $t = 0, 1, 2, \ldots$
- State $X(t)$ given by TCP dynamics
- Two possible control actions $a(t)$:
  - transmit the flow packets as they arrive
  - warn the flow sender by dropping/marking packets
- If $a(t)$ is applied in state $X(t)$, then
  - $W_{X(t)}^{a(t)} \leq W_{X(t)}^{\text{sent}}$ amount of resources is used
  - goodput (reward) $R_{X(t)}^{a(t)}$ is earned
  - the sender sets $X(t + 1)$
Flow Dynamics as Seen in Router

- **Finite-length** flows: flow terminates before the next time epoch with probability $1 - \beta$, due to
  - finalizing file transmission, or
  - sender’s impatience, or
  - external factors such as broken connections

- Congestion warning can be sent also by routers on future path, therefore
  - if transmitted, flow restarts with probability $1 - p_X(t)$
  - if warned, flow restarts for sure
Router Resources

- Bandwidth $W$, i.e., deterministic “server capacity”
- Target time-average router throughput $\bar{W} < W$, i.e., “virtual capacity”
- Buffer size $B \geq W$
- Backlog process $B(t)$ at epochs $t$
  ▶ number of packets buffered for more than one period
- To be allocated to randomly appearing and disappearing flows
Congestion Control in Router

\[ X_1(t), X_2(t), X_3(t), \ldots, X_M(t) \]

\[ W_{\text{sent}}^{1,X_1(t)}, W_{\text{sent}}^{2,X_2(t)}, W_{\text{sent}}^{3,X_3(t)}, \ldots, W_{\text{sent}}^{M,X_M(t)} \]

\[ a_1(t), a_2(t), a_3(t), \ldots, a_M(t) \]

\[ W_{1,X_1(t)}^{a_1(t)}, W_{2,X_2(t)}^{a_2(t)}, W_{3,X_3(t)}^{a_3(t)}, \ldots, W_{M,X_M(t)}^{a_M(t)} \]

\[ R_{1,X_1(t)}^{a_1(t)}, R_{2,X_2(t)}^{a_2(t)}, R_{3,X_3(t)}^{a_3(t)}, \ldots, R_{M,X_M(t)}^{a_M(t)} \]
Congestion Control in Router

- Maximizing time-average expected goodput

\[
\max_{\pi \in \Pi} \lim_{T \to \infty} \frac{1}{T} \mathbb{E}_{\pi, B_0} \left[ \sum_{t=0}^{T-1} \sum_{m \in M(t)} R_{m,X_m(t)}^{a_m(t)} \right]
\]

- Subject to limited bandwidth and buffer space

\[
\lim_{T \to \infty} \frac{1}{T} \mathbb{E}_{\pi, B_0} \left[ \sum_{t=0}^{T-1} \sum_{m \in M(t)} W_{m,X_m(t)}^{a_m(t)} \right] \leq W
\]

\[
B(t) + \sum_{m \in M(t)} W_{m,X_m(t)}^{a_m(t)} \leq B, \text{ for all } t
\]

- **PSPACE-hard** due to the last constraint
Network-Capability Fairness

- Flows treated fairly according to what the network can deliver, not what the users want to send.
- Arises by making routers maximize expected network goodput, which relies on flows’ future-path information.
- We assume that this information is gathered in certain time intervals that are bigger than flows’ lifetimes.
  - each flow finds congestion warning probabilities constant
Relaxation and Decomposition

1: Relax (omit) the hard sample-path buffer constraint

2: Dualize the bandwidth constraint using Lagrangian multiplier

\[
\max_{\pi \in \Pi} \lim_{T \to \infty} \frac{1}{T} \mathbb{E}_{n}^{\pi} \left[ T - 1 \sum_{t=0}^{T-1} \sum_{m \in M(t)} \left( R_{m,X_m(t)}^{a_m(t)} - \nu W_{m,X_m(t)}^{a_m(t)} \right) \right] + \nu \overline{W}
\]

3: Decompose the Lagrangian relaxation due to flow independence into single-flow parametric subproblems

\[
\max_{\pi_m \in \Pi_m} \mathbb{E}_{n_m}^{\pi_m} \left[ \sum_{t=0}^{\infty} \beta_m^t \left( R_{m,X_m(t)}^{a_m(t)} - \nu W_{m,X_m(t)}^{a_m(t)} \right) \right]
\]
Optimal Solution to Subproblems

- For finite-state finite-action MDPs there exists an optimal policy that is deterministic, stationary, and independent of the initial state
  - we narrow our focus to those policies
  - represent them via transmission state-sets $S \subseteq \mathcal{N}$
  - policy $S$ prescribes to transmit in states in $S$ and warn in states in $S^C := \mathcal{N} \setminus S$

- Combinatorial $\nu$-wage problem $\max_{S \subseteq \mathcal{N}} R_n^S - \nu W_n^S$, where

$$R_n^S := \mathbb{E}_n^S \left[ \sum_{t=0}^{\infty} \beta^t R_{a(t)}^X(t) \right], \quad W_n^S := \mathbb{E}_n^S \left[ \sum_{t=0}^{\infty} \beta^t W_{a(t)}^X(t) \right]$$
Optimal Solution to Subproblems

- **Deteriorating Quality-of-Service assumption:**
  - the higher the sending rate, (approximately) the higher the probability of congestion

- **Concave Adjusted Goodput assumption:**
  - $R_n$ “sort of” concave in $W_n$

- **A threshold policy is optimal**
  - transmit if the flow’s sending rate level is **below** a threshold
  - warn if the flow’s sending rate level is **above** the threshold
Indexability and Transmission Index

- The \( \nu \)-wage problem is known as a restless bandit
- We use the restless bandit indexation methodology
  - if the problem is “indexable,” we identify break-even values of wage \( \nu \), where the optimal policy changes
  - we call such values transmission indices (prices)
  - they measure marginal productivity of transmission
- Geometric interpretation:
  - optimal policies are extreme points of the upper boundary of the performance region
  - transmission indices are slopes of the upper boundary
Performance Region
Performance Region
Performance Region
Performance Region
Performance Region

\[ RS \]

\[ WS \]

\[ N \]
Optimal Solution to Subproblems

- One transmission index is assigned to each state
  - state $n$ is in the optimal transmission state-set for all $\nu$ below its transmission index $\nu_n$

- In terms of transmission indices, the optimal policy is:
  - warn at flow’s sending rate level $n$ iff transmission index $\nu_n$ is lower than wage $\nu$

- Sort of a dual concept to threshold policies
  - but not equivalent!
Optimal Solution to the Relaxed Problem

- For the multi-flow Lagrangian relaxation the \textit{optimal policy} is: For each flow,
  \begin{itemize}
    \item warn at flow’s sending rate level $n$ iff transmission index $\nu_n$ is lower than wage $\nu$
  \end{itemize}
- For the multi-flow problem without the buffer constraint, for each target $\overline{W}$ there exists $\nu(\overline{W})$ such that if $\nu(\overline{W}) \neq 0$, then the \textit{optimal policy} is: For each flow,
  \begin{itemize}
    \item warn at flow’s sending rate level $n$ iff transmission index $\nu_n$ is lower than wage $\nu$
  \end{itemize}
Heuristics for the Original Problem

- For the original multi-flow problem, apply the following heuristical policy: For each flow,
  - warn at flow’s sending rate level \( n \) iff transmission index \( \nu_n \) is lower than wage \( \nu \)

- Where \( \nu \) is a parameter dynamically tuned so that
  - \( \nu \) is high enough to assure that bandwidth is not left idle (do not warn flows if virtual capacity is not used fully)
  - \( \nu \) is low enough to assure no buffer overflow (do not transmit flows if buffer is to be overflowed)
Heuristics for the Original Problem

- Also, transmission indices can be used
  - to “weight” the warning probabilities in existing congestion avoidance mechanisms
  - to prioritize in scheduling, i.e., to modify the order of service (instead of FIFO)

- Note, in general,
  - flows with higher transmission index should have higher transmission priority

- These heuristics are in general tractable and well-performing
Summary

- Optimal policy for a wide range of flows
- Possible practical implementation
- Limitations
  - real TCP is more complicated
  - restless bandit indexation: only two possible actions (we would like to use 3: “transmit,” “drop,” “mark”)
  - practice: not applicable into the Internet of today due to missing or delayed future-path information
- Though, a good starting point
- Extensions under development
Thank you for your attention!