Buffer Management for Colored Packets with Deadlines

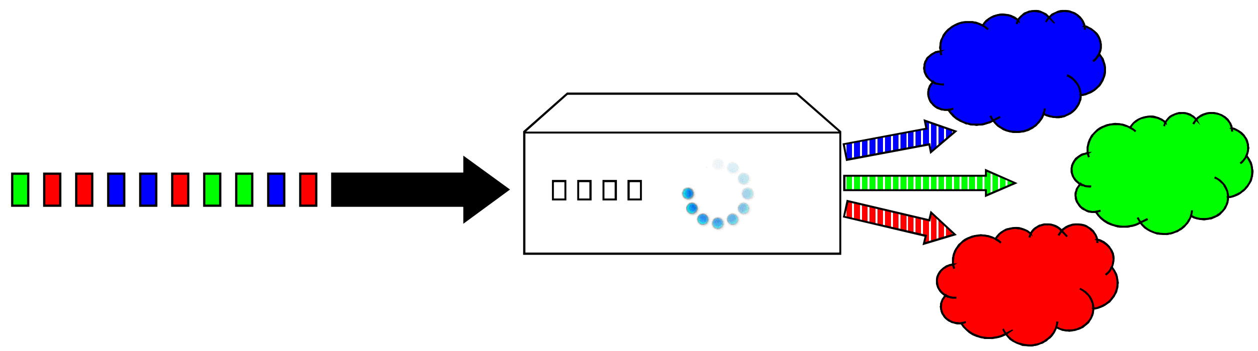
Yossi Azar

Tel Aviv University

Joint work with, Uriel Feige, Iftah Gamzu, Thomas Moscibroda and Prasad Raghavendra

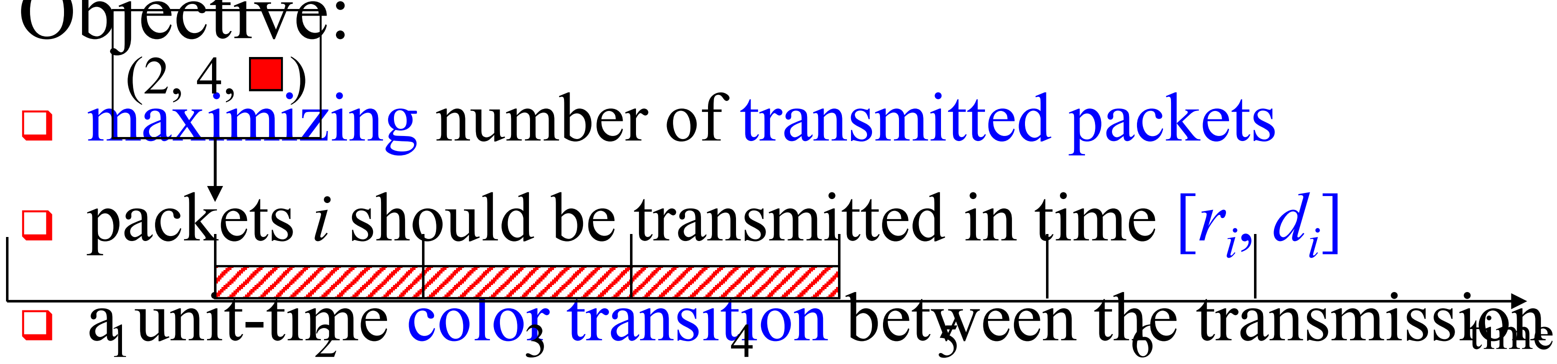
Motivation

- Packets scheduling in multi-port network devices:
  - packets arrive dynamically ⇔ release time
  - packets have limited delay time ⇔ deadline time
  - packets have outgoing port ⇔ color
  - device has reconfiguration overhead for switching ports ⇔ color transition

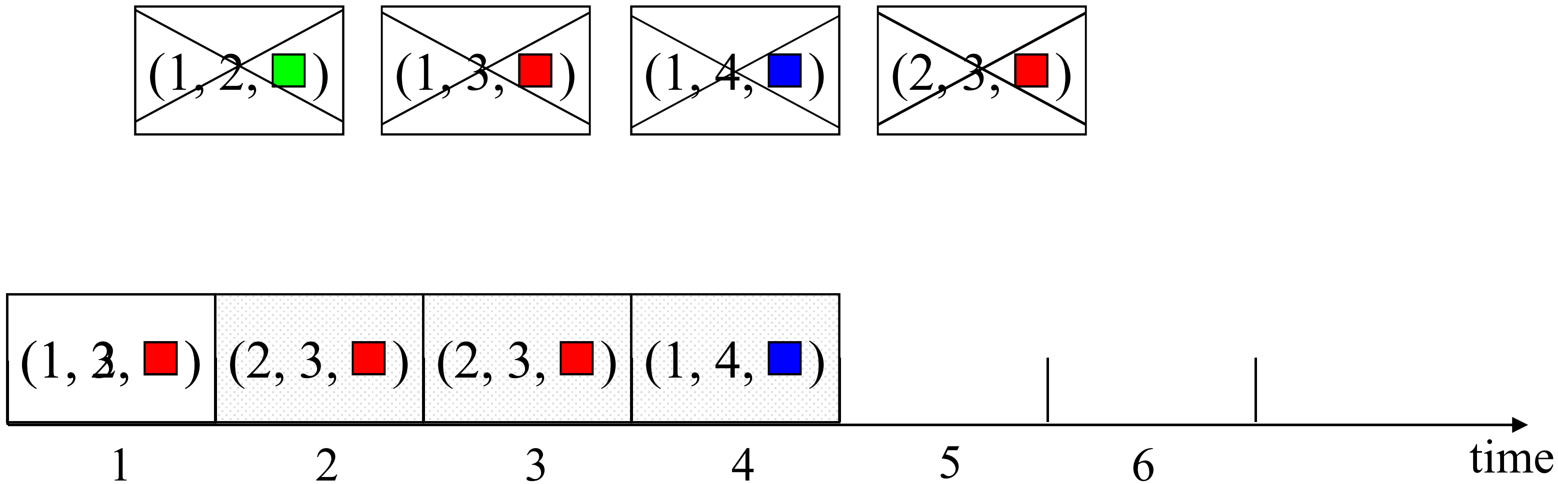


Problem Definition

- Input:
  - an online stream of packets
  - each packet i is characterized by (r_i, d_i, c_i)
    a release time, a deadline, and a color
- Objective:
  - maximizing number of transmitted packets
  - packets i should be transmitted in time $[r_i, d_i]$
  - a unit-time color transition between the transmission of packets with different colors



Example



- **Optimal solution:**

a throughput of 3

a throughput of 2

Observations

- Single color (no color or no configuration cost):
  - packets with release time, deadlines
  - schedule to maximize throughput
- **EDF-style strategy:** *earliest-deadline-first* schedules an active packet with minimal deadline
- EDF is optimal 1-competitive

Observations: Multiple colors

- Modified EDF
  - apply EDF while adding transition slots when needed
- Modified EDF is at least 1/2-competitive
  - at most one transition slot for each transmitted packet
- Unfortunately it is not better than 1/2-competitive
  - flipping between two color
- Moreover, best possible for any deterministic online algorithm

What can we do ?

- ➔ 50% performance loss is unacceptable in real-life scenarios… what can we do?
- consider the practical setting in which $C << L$
  - C – number of colors
  - L – minimum laxity (deadline – release time)
- modified EDF is still not better than 1/2-competitive
- Can we do better ?

The Main Result

$1 - O\left(\sqrt{C/L}\right)$-competitive algorithm

- almost optimal $1\text{-}o(1)$-competitive when $C = o(L)$
- strikes a balance between the natural greedy approaches w.r.t. deadlines and colors

Idea – Relax EDF

- Packets with similar (not necessarily identical) deadlines– transmit by colors

- Question: how to do it correctly

- First we need to analyze perturbation on EDF with no colors

Problem definition reminder (no colors)

- Input:
  - an online stream of packets
  - each packet i is characterized by (r_i, d_i)
    a release time and a deadline
- Objective:
  - maximizing number of transmitted packets
  - packets i should be transmitted in time $[r_i, d_i]$
- EDF is optimal

EDF – perturbation with no colors

Perturbation theorem on classical EDF

- small perturbation in release/deadline times cannot induce major degradation in optimal throughput

➔ "EDF is robust": its throughput is close to optimum even when deadlines are not precisely known

Idea – Perturb EDF – no colors

- Perturbation time:
  - decrease deadline of each packet by at most x
  - increase release time of each packet by at most x

- Throughput should drop
  - by how much it drops ?

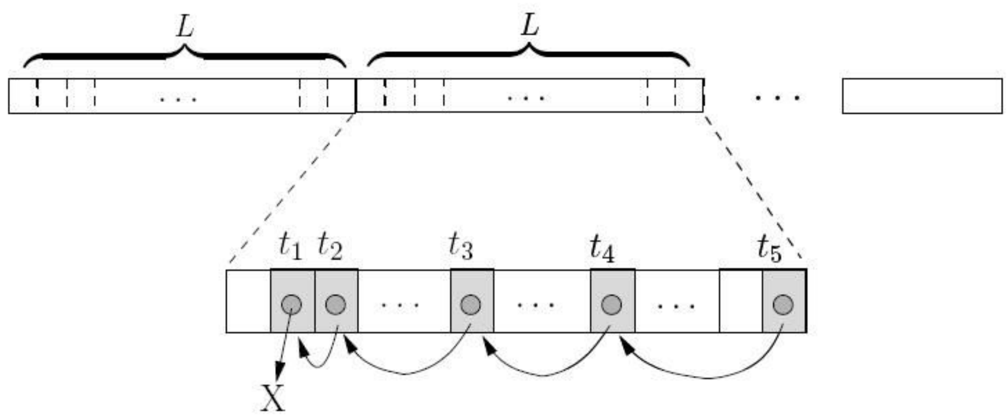
Perturbation Theorem on EDF

The optimal throughput does not degrade by more than 1-δ (multiplicatively) where δ = perturbation time / min laxity

proof sketch for perturbation of at most 1 in deadlines:

“old” schedule:

“new” schedule:



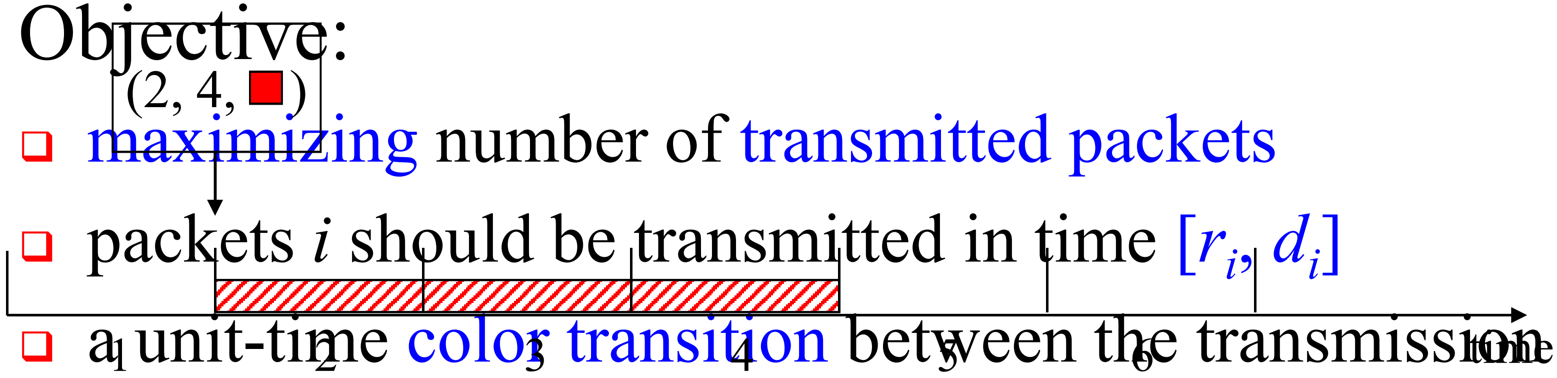
➔ the throughput degraded by no more than 1 - 1/L

Perturb EDF – no colors

- Perturbation time:
  - decease deadline of each packet by at most x – apply iteratively the previous theorem x times: the throughput decreases by $1-x/L$
  - increase release time of each packet by at most x - apply the mirror theorem by $1-x/L$

Problem definition reminder (with colors)

- Input:
  - an online stream of packets
  - each packet i is characterized by (r_i, d_i, c_i)
    a release time, a deadline, and a color
- Objective:
  - maximizing number of transmitted packets
  - packets i should be transmitted in time $[r_i, d_i]$
  - a unit-time color transition between the transmission of packets with different colors

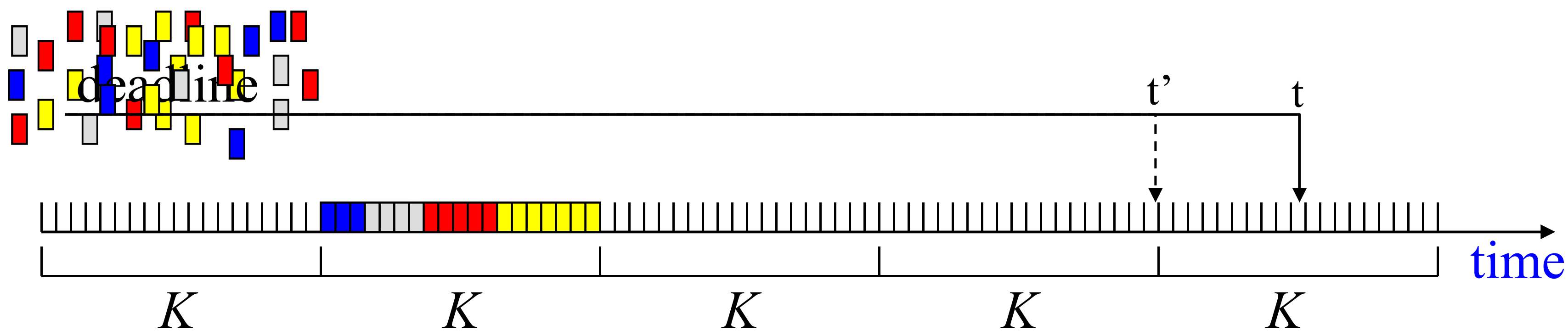


Algorithm Balanced Greedy

Works in phases, each spans a time frame of $K = (CL)^{1/2}$ time-slots, and consists of two steps:

1. a collection step
2. a scheduling step

- accumulate packets arriving during the time frame
  consider a K-length prefix of EDF schedule packets
- order by the deadline of packets
  modify color (drop packets for color transitions)



Idea of proof

- We can view it as perturb EDF
  - We loose $1-K/L$ in throughput by EDF perturbation
  - We loose at most $1-C/K$ because of color transition
- We choose $K = (CL)^{1/2}$ to optimize
- The competitive ratio is

$$1 - O\left(\sqrt{C / L}\right)$$

Lower Bound

- To achieve $1\text{-}o(1)$-competitive we need $C = o(L)$
- Specifically $1\text{-}C/L$ lower bound
- Lower bound – not tight

Conclusion

- Contributions
  - $1\text{-}o(1)$-competitive algorithm when $C << L$
  - perturbation theorem on classical EDF
  - constant bound for any online algorithm when $C \approx L$
  - NP-hardness of the offline version
- Open questions
  - offline setting?
  - valued packets?
  - other natural extensions…

Thank You!