Simulation of Adversarial Scenarios in OMNeT++

Putting Adversarial Queueing Theory from Its Head to Feet

Daniel Berger, Martin Karsten, and Jens Schmitt

OMNeT++ Workshop 2013 in Cannes, French Riviera
March 5th, 2013

Introduction

Overview

Adversarial Queueing Theory (AQT) studies hypothetical worst-case queueing scenarios.

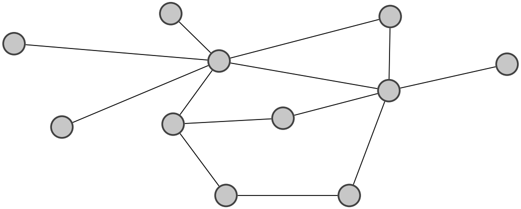
- What we did → simulation of AQT scenarios
- Why simulation → study, new insights, and communicate
- Results → e.g. on bound tightness, robustness

Goal of this talk

1. raise interest for AQT in the simulation community
2. establish simulation as complementary method to analysis

Scenario

- imagine a network with the following requirement: guaranteed delay $\leq$ 20ms



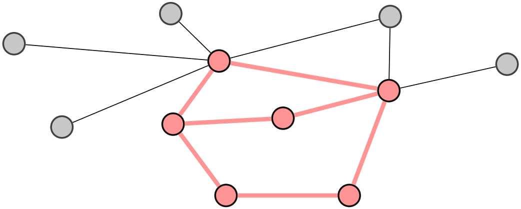
- worst-case delay? $\Rightarrow$ two steps:
  1. determine: is a delay guarantee possible
  2. if yes: find tight bound (employ network calculus/...)

rough definition

a network is stable $\Leftrightarrow$ a finite bound on delay exists

Scenario

- imagine a network with the following requirement: guaranteed delay $\leq$ 20ms

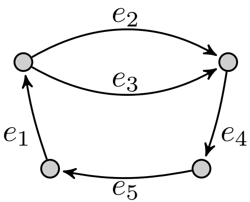


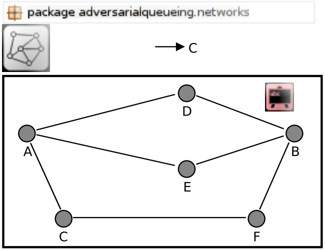
- worst-case delay? $\Rightarrow$ two steps:
  1. determine: is a delay guarantee possible
  2. if yes: find tight bound (employ network calculus/...)

rough definition

a network is stable $\Leftrightarrow$ a finite bound on delay exists

Contribution





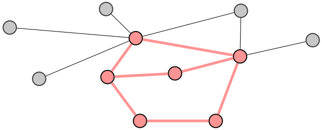
- framework to capture adversarial traffic description
  - in short: packet trajectories, and adversarial strategies
- implementation of some classical AQT scenarios
- interpretation of some of the results
  - modeling assumptions, bound tightness, parameter interaction

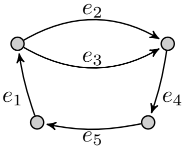
An open simulator for the Adversarial Queueing framework
`http://disco.informatik.uni-kl.de/content/Aqtmodel`

Adversarial Queueing

Network Instability

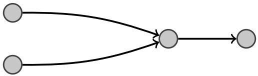
- network instability is based on inductive constructions
  - recall: mentioned cyclic graphs



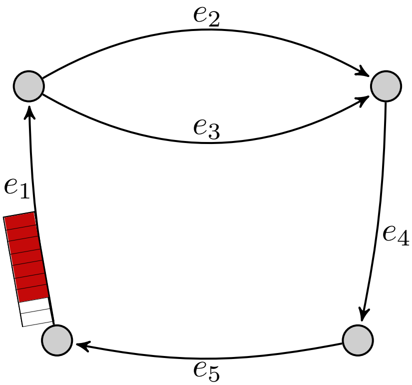


- hypothetical adversary (cf. online algorithms)
  - can inject anywhere and can choose packet trajectory
  - adversary's goal: maximize stress on network
- injections are subject to a leaky-bucket constraint (e.g.): for any edge: sum of injections in $[t_1, t_2] \leq r(t_2 - t_1) + b$
- $r <$ edge capacity

⇒ e.g. this topology is stable:

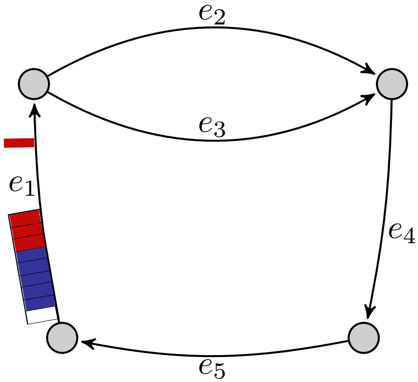


Example of Inductive Injection Scheme



1. assume a set of red packets queued at e_1

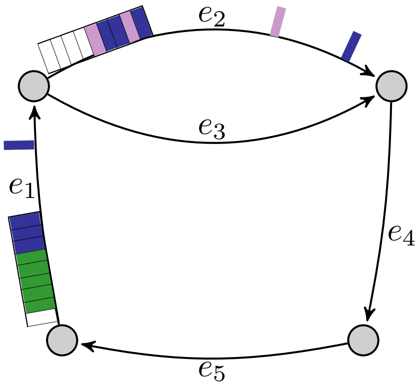
Example of Inductive Injection Scheme



1. assume a set of red packets queued at e_1
2. inject blue packets into e_1 with path (e_1, e_2, e_4)

Example of Inductive Injection Scheme

3.b inject packets into e_2
$\Rightarrow$ slow down blue packets

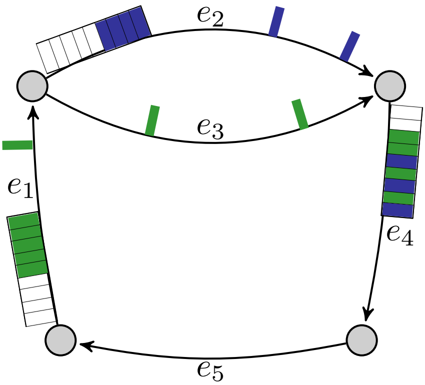


1. assume a set of red packets queued at e_1
2. inject blue packets into e_1 with path (e_1, e_2, e_4)

3.a inject green packets into e_1 with path (e_1, e_3, e_4).

Example of Inductive Injection Scheme

3.b inject packets into e_2
$\Rightarrow$ slow down blue packets



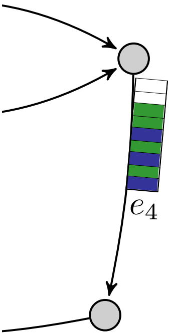
4. fill queue at e_4, more injections, transfer to e_1, repeat ...

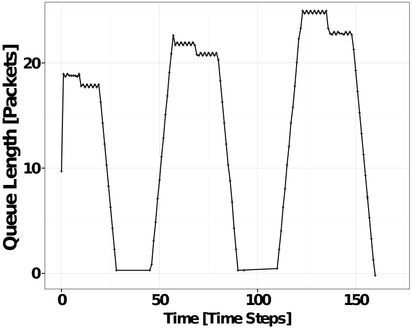
1. assume a set of red packets queued at e_1
2. inject blue packets into e_1 with path (e_1, e_2, e_4)

3.a inject green packets into e_1 with path (e_1, e_3, e_4).

Effect of Adversarial Injections

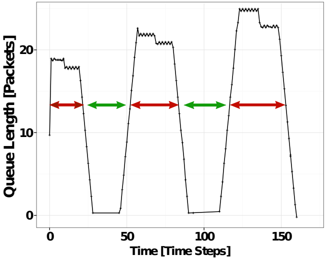
- queue of e_4 (and others) experiences repeated bursts
- burst size grows without bound over time
- no upper bound on queue length or delay





- theory seeks exact conditions of stability
- as an engineer: open questions may be different!?

Open Questions



in general:

- how real is the threat?
- so what does it mean?

more specific:

- how long does it take to induce some particular delay?
- countermeasures?
- how realistic are modeling assumptions?
  - perfect time synchronization
  - infinite buffers
  - errors in discretization, rough calculations, bounds, ...
- assumption of initial network state (so-called initial sets)

Simulation Results

Simulation Results

Case study with four examples

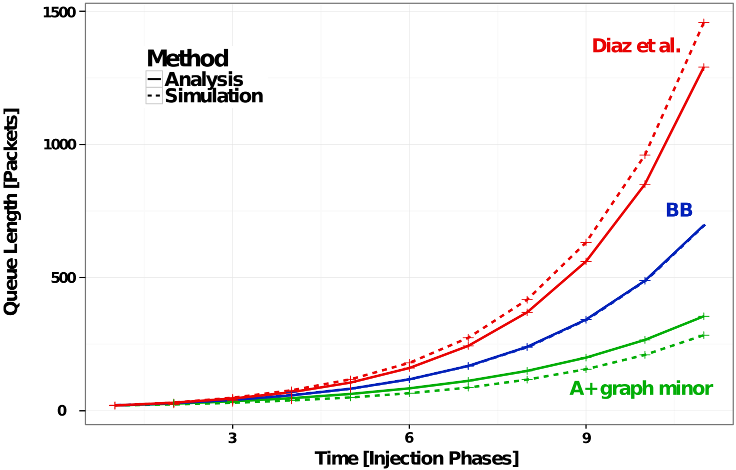
- Baseball (BB), due to Andrews et al., 2001
- Diaz et al., 2001
- graph minor $\mathcal{A}^+$, due to Weinard, 2006
- gadget chains, Lotker et al., 2004

Disclaimer

Only exemplary study - may not be universally valid.

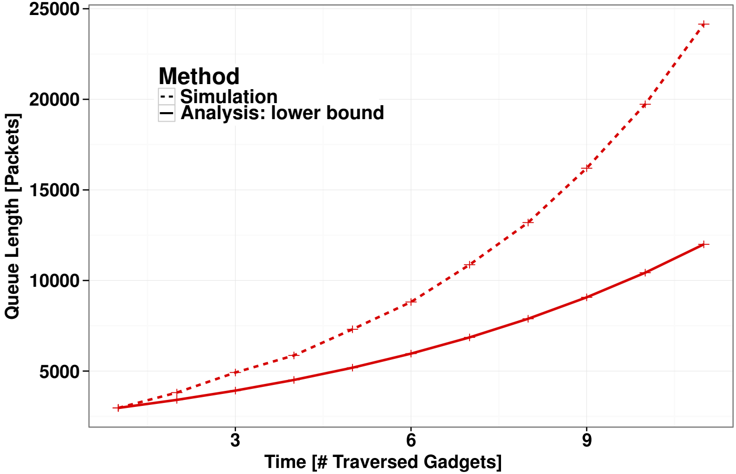
- yet, we consider the selection of scenarios representative

Analytical Prediction vs. Simulative Result



Bound Tightness

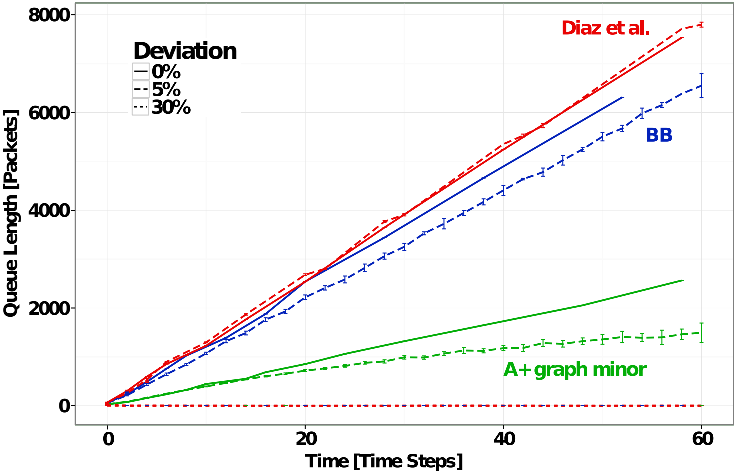
Gadget Chains, Lotker et al.



Rerandomization

- adversary relies on strictly deterministic synchronization
  - events take place on distinct nodes with exact timing
  - no noise: fixed channel capacity
- we introduce "rerandomization"
  - channel with variable delay
  - every traversal delay is sampled from a Normal distribution
- channel mean delay = deterministic delay
- delay standard deviation e.g. of 5% or 30%

Rerandomization



Challenges and Future Work

Challenges and Future Work

- high number of nodes, events → approximation
- long execution time → parallelism
- simplifying abstraction, modeling error → step-wise convergence towards realism

- classical challenges in simulation
- we have great interest to learn from simulation community

- future work
  - utilize as part of our own theoretic work in AQT
  - develop towards more realistic scenarios
    e.g. assess the role of cross traffic in the adversary's operation
  - enhance visualization

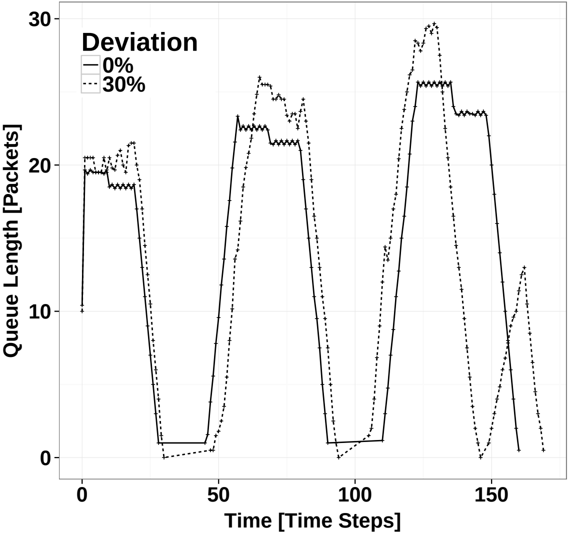
Simulation of Adversarial Scenarios in OMNeT++

Putting Adversarial Queueing Theory from Its Head to Feet

Daniel Berger, Martin Karsten, and Jens Schmitt

Slides will be available:
`http://disco.informatik.uni-kl.de/software/AQTmodel/SlidesOMNeT2013.pdf`

Rerandomization Detailed Operation



Interaction of Initial Sets with Injection Rate