# KBAC: Knowledge-Based Admission Control

*Doreid Ammar *, Thomas Begin *, Isabelle Guérin-Lassous * and Ludovic Noirie ***

* *UCB Lyon* 1 / *LIP* (*UMR ENS Lyon - INRIA - CNRS - UCBL*)
** *Alcatel-Lucent Bell Labs*, *Nozay*, *France*

*Semantic Networking project*
*Common laboratory INRIA - Alcatel Lucent-Bell Labs*

# Motivation

Growing traffic volume

- *e.g.*, streaming, live video watching, P2P, video games

Satisfy QoS ? ⇒ Admission control

Accept or Reject a new incoming flow

- too many acceptances ⇒ performance collapse
- too many rejections ⇒ low level of resource utilization

Existing admission control solutions

- measurement algorithm & decision algorithm
- difficult to calibrate

Objective

- avoid the critical step of calibration
- Knowledge-Based Admission Control solution (KBAC)

# Plan

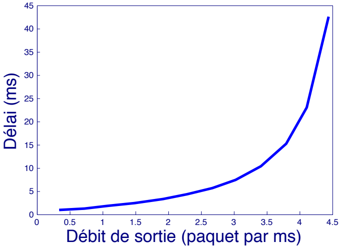
KBAC

Performance evaluation

Conclusion

## Our approach

Link behavior $\Longleftrightarrow$ mono-server queue




Which queue to chose ?

- *G/G/1* queue, but how to solve it ?
- *M/G/1* queue, simple solution for its steady-state but general enough ?

How to chose the queue parameter values ?

- in a dynamic and automatic way

# Major steps

## Measurement algorithm

- on short timescale, collect couples of $(X,P)$
- $X$ = actual throughput
- $P$ = packet delay or packet loss rate
- $(X,P)$ = *measurement point*

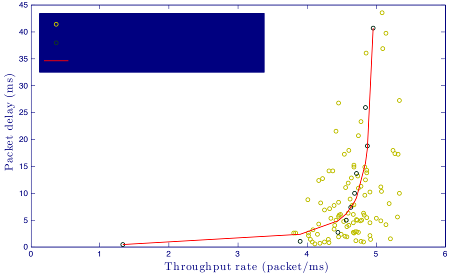
## Knowledge Plane

- too many data $\Rightarrow$ *K-means* clustering method
- partition *measurement points* into $k$ *centroid points*
- model the behavior by a mono-server queue (*M/G/1* queue or *M/G/1/K* queue)
- $f_P$ = the discovered queue

# Illustration

## Example

- here, $P$ = packet delay
- the found *M/G/1* queue ($\mu = 5.01$, $CV = 2.02$ and $off = 0.08$)




Example of a Knowledge Plane

## Avoid the flood of information

Too many *measurement points* $(X,P)$

- keep only 1000 points

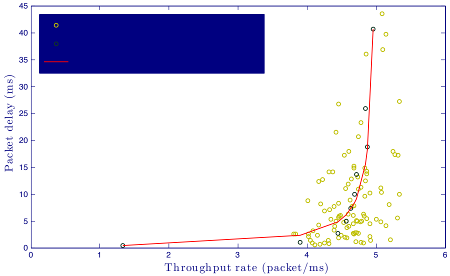
But which of them ?

- the 1000 last points are the most recent
- but very likely to all look alike $\Rightarrow$ loss of information
- instead, we split the possible range of throughput in 10 intervals
- and we enforce the existence of 20 $(X,P)$ in each throughput interval

## Illustration

### Example

- here, $P$ = packet delay
- the found *M/G/1* queue ($\mu = 5.01$, $CV = 2.02$ and $off = 0.08$)




Example of a Knowledge Plane

# Decision algorithm

## Performance prediction

- new flow requesting admission, with a peak rate $r$
- the expected performance $\Rightarrow \widehat{P} = f_P(X + r)$

## A new flow is accepted if

- $\widehat{P} + \alpha\widehat{\sigma}_p < P^*$
- $P^*$ = target performance
- $\widehat{\sigma}_p$ = standard deviation of $\widehat{P}$
- $\alpha$ = tuning parameter (*Chebyshev's inequality* $Q = 75\% \Rightarrow \alpha = 1.7$)

## Scenario

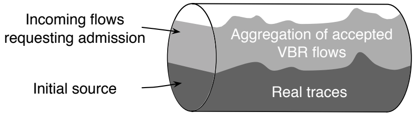
### Initial source

- real traffic trace
- Average rate = 2.5 Mbps

### Incoming flows (VBR)

- Sending rate = 64 Kbps
- Packet size = 190 bytes
- CV = 2.5
- Poisson arrivals

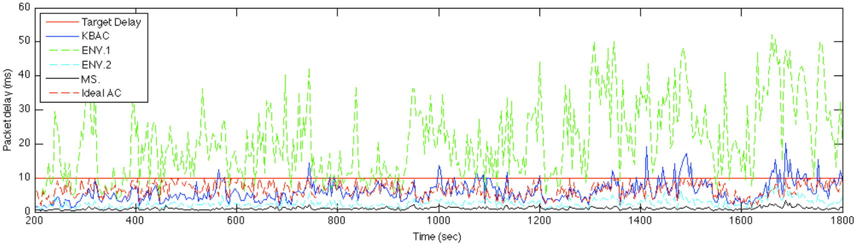
### Single communication link




- Capacity, $C$ = 10 Mbps
- Queue size = 60 ms
- Queueing discipline
  FIFO (First In First Out)
- Queue management algorithm
  Drop-Tail

$P^* = 10\text{ms}$

| | KBAC | MS | ENV.1 | ENV.2 | Ideal |
|---|---|---|---|---|---|
| % accepted flows | 28% | 19% | 32% | 24% | 30% |
| % QoS violation | 2% | 0% | 55% | 0% | 0% |




Performance of admission control solutions

# Conclusion

Knowledge-Based Admission Control solution (KBAC)

Data-driven and evolutive solution

- dynamic environement
- no assumption on the traffic

Performance

- avoids the critical step of precisely calibrating key parameters
- good trade-off between flow performance and resource utilization
- automatic adjustment of admission policy according to the actual variations on the traffic conditions

Futur work

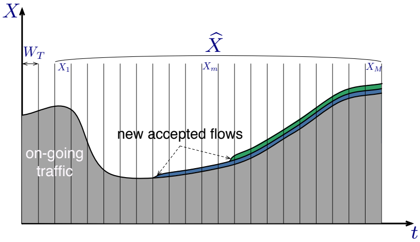
- extend to packet loss rate

## *Questions ...*

## $\widehat{X}$ = the adjusted throughput

### Why ?

- to avoid the erratic behavior of $X$




Estimation of the adjusted throughput $\widehat{X}$

- computed over the last $M$ measurement windows
- ponderation of the peak rate + average value of the throughput