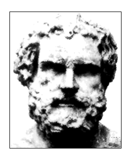
Modelling and controlling the future?

Peter Key
MSR Cambridge
peterkey@microsoft.com
http://www.research.microsoft.com/~peterkey/

A 5th Century world view (Democritus)



- All matter is composed of *atoma*
- Differ in shape size and weight
- Too small to be seen
- Appearances caused by higher-layer effects (post-modern interpretation?)
- *Indestructible*

21st C Networking

- Everything composed of packets
- Packet level simulations
- Packet level QoS (none!)
  - Packets seem to get lost …
- Science studies macro level (eg thermodynamics etc) as well as micro-level
- But network Modelling & control still largely atomic!

Missing the point …

"More generally, there are almost no interesting problems in networking for which an M/M/1 model is appropriate unless the result is negative (i.e., if you can show that even with an M/M/1 queue, the proposed algorithm or traffic management scheme fails woefully then the algorithm/scheme is clearly worthless)."
(Name withheld to protect the …)

Moving up the stack

- *Application* performance matters to users
- Too much emphasis on Transport layer
- Can have application level reactions
- MAC matters for wireless

Process	Application Presentation
Host to host	Session Transport
Internet	Network
Network I/f	Data Link PHY

Modelling applications

- Need stochastic demand models
- What are the right timescales?
  – eg traffic may be LRD
  – but locally Poisson (infinitely divisible limit)
  – Interacting feedback control loops
- Right level of detail?
- Want *Network* level models & not just particle (queuing) models

Modelling large systems

- Flows $N_r(t)$, stochastic arrivals rate ν_r
- *Fixed* expected duration m_r variable b/w x_r
- Packet level measures, $p(N_r(t)\ x_r\ (t),\ldots)$
- Scale system by $C\ \nu_r{}^C ® C\nu_r$ & $n_r(t)\ ® N_r(t)/C$

then have a FLLN for network equilibrium behaviour

$$\dot{n}_r = n_r - n_r m_r, \quad p = p\left({n_r}/{m_r}, \ldots\right)$$

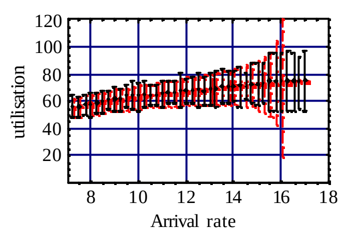
Modelling large systems -II

- Diffusion limit (OU) for *second-order* properties

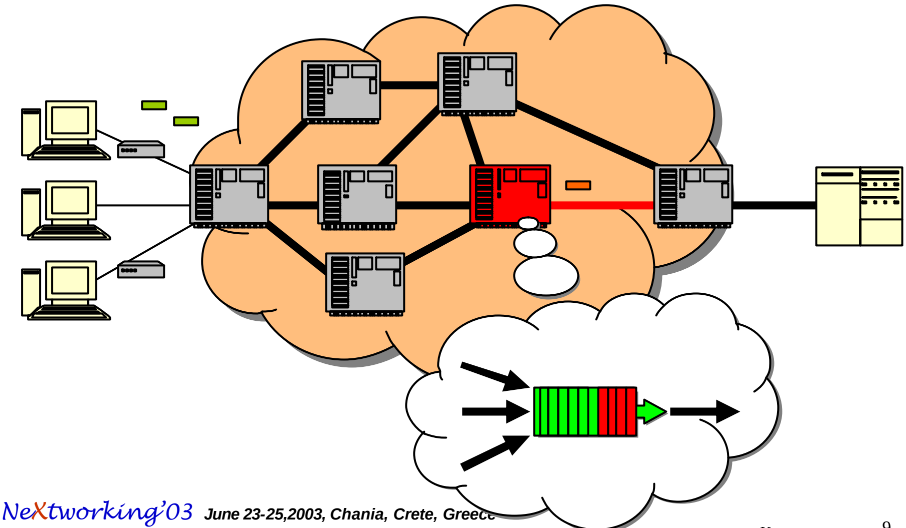
$$Z \overset{\text{def}}{=} \frac{1}{\sqrt{C}}\left(n_r - C\bar{n}_r\right)$$

$$dZ(t) = -BZ(t - t)dt + dW(t)$$

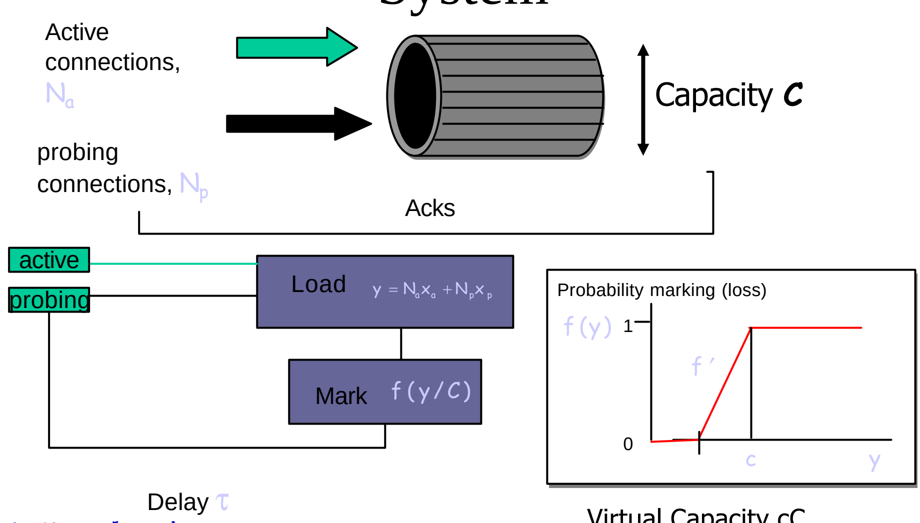
Theory vs simulation



Distributed admission control



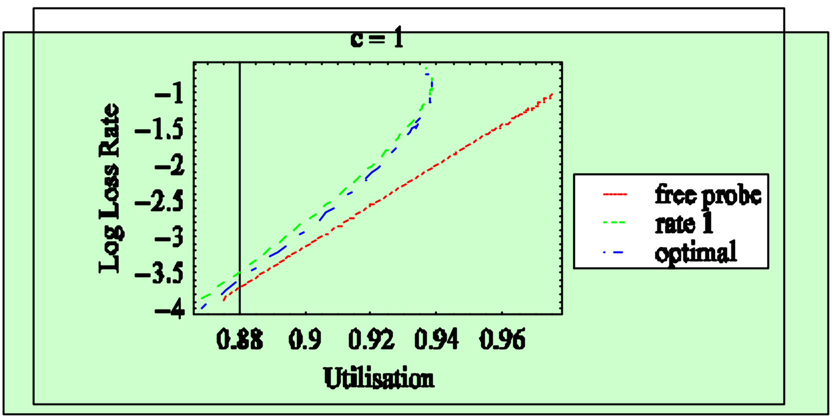
System



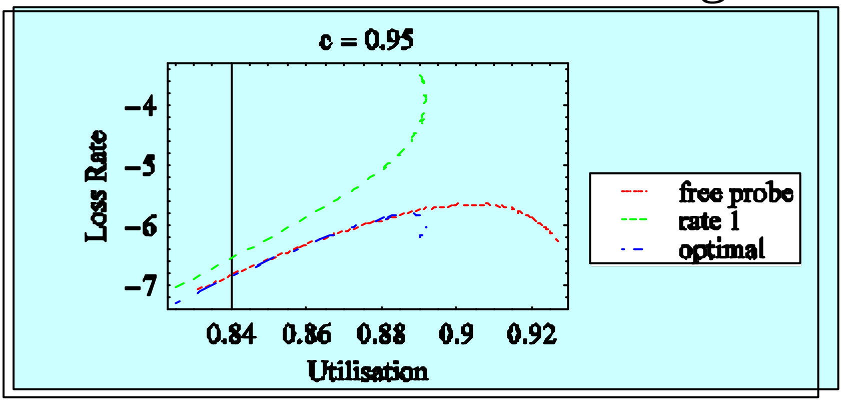
Example: Admission Control

- Q: How Long to probe for & stability?
  - Critical to ensure mean holding time / rtt *Large*
- Q: What rate?
  - Zero rate (“free probing”) or
  - “optimal” non-zero rate =half rate at which packets marked / lost
- Q: Feedback signal?
  - Tail marking (drop tail) cannot protect system
  - Early warning marking (apparently sacrificing5%) with probing gives a controllable system
- Q: Dynamic Rate Adjustment?
  - If “correct” at entry, can be “Infrequent”

Performance with Loss based feedback



Performance with Marking



- Give up a small amount of efficiency (eg 5-10%) to gain control

Small networks?

- Analysis breaks down for small networks
  - Ad-hoc / home networks (small diameter)
  - or where ratio of peak b/w /capacity large (small capacity)
- Variance not a good predictor of behaviour
- Timescale separation can break down
- If rtts are small, then end-system effects are no longer negligible

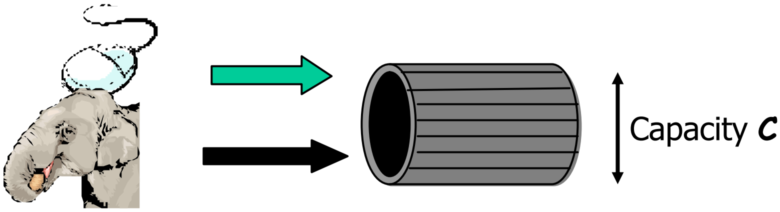
Mixing traffic?

- If scaling applied to fixed-volume traffic (eg file transfers), limit distribution unstable or has mean zero
- *BUT*, mixing with fixed duration traffic *stabilizes* the system
  - See forthcoming ITC paper!

User preferences?

- How should different types of traffic be treated?
  - Need to have *real* incentives somewhere (incentive compatibility implies real money)
  - or mandated behaviour
- Current TCP-friendliness definition inhibits evolution
- Distributing resources: Zero-sum game at network level , *not* at user level
- Sneaker-net currently cheaper than real-net, contention at access points, difference increasing

Example: Scheduling file transfers



- TCP allocates equal weights (or bias against small)
  - forces concave user utility function on *rate*
- But Shortest remaining processing time
  - "optimal", implementation, stability (& fairness?) questions
- We can design a distributed solution which gives greater weight to mice – everyone better off!

Modelling application level recap

- Have given some examples of application level issues
  - Flow level models / demand
  - User level preferences

Overlays – an alternative method of sharing?

- Overlays can provide both a addressing/routing substrate and a "community"
- Increase availability of certain resources (eg information, processing) not others (eg b/w)
- Can performance guarantees be given?
  - use of FEC can just be an arms race ..
- Asymmetric resources imply benefits for the "rich"! Eg Splitstream can save on sources costs

Some concluding remarks

- Modelling needs to guide design not follow …
- Current Internet provides connectivity but what else?
  - With better signals (eg early-warning, ECN) could give (soft) QoS guarantees
  - But why should users cooperate?
  - Learn from game theory & economics?
  - Incentives for content, connectivity, routing are interconnected