On Flow Completion Time Benchmarking in Datacenters

M. Gusat, C. Minkenberg and R. Luijten
IBM Zurich Research Lab
May 2007

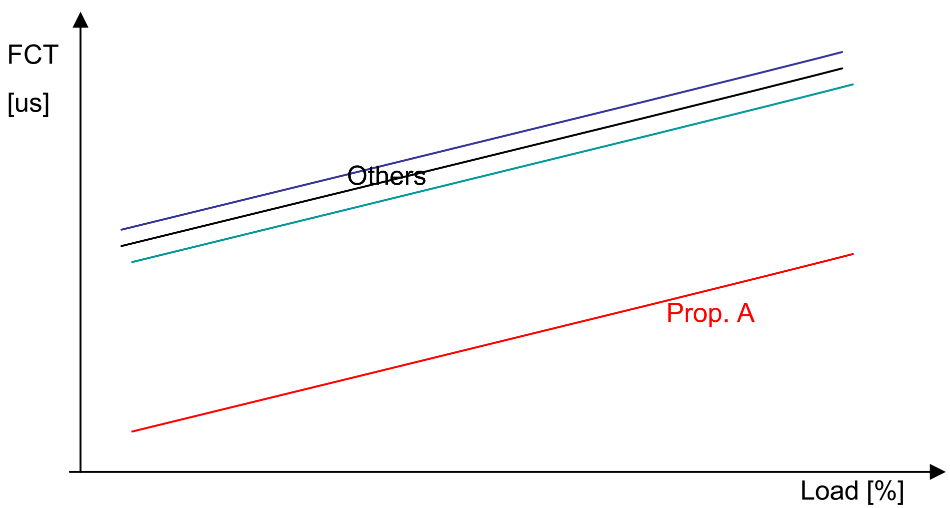
Outline

- Motivation
  - Desire for Bursty Benchmark for Ethernet CM evaluation
- Traffic generator
  - Bursty source w/ heavy-tailed (Pareto) distributions
- Metric: Flow Completion Time (FCT)
  - Definitions and methodology
- Topology
  - MINs
- Putting it all together
- Proposal for Bursty Benchmark
- Conclusions

Motivation

- I) Since Monterey Jan. '07 request for new metric, scenario and traffic
  1. Redo all sim runs without PAUSE (PAUSE=off)
  2. New metric: Flow Completion Time (FCT)
     1. As proxy for application-level latency
  3. More 'realistic' traffic: Bursty sources, heavy-tailed distribs (Pareto)
  4. More 'realistic' topos

- II) We need a consistent approach across all adhoc sim teams
  - Example plot: "Proposal A" (optimized) vs. "Others (B,C,D)" (basic versions)



=> Need to agree to a common "Bursty Benchmark"

Elements of a Benchmark

1. Work
2. Job
3. **Flow**
4. **Burst**
5. **Packet (Ethernet Frame)**

- Each has 2 random variables (Size and Interarrival), for which we must choose a distribution.
  - agree on parameter values
  - exponential or Pareto

Comparison of Exponential vs. Pareto Distributions

CDF: $F(t)=1-e^{-\lambda t}$

RELIABILITY: $R(t)=e^{-\lambda t}$

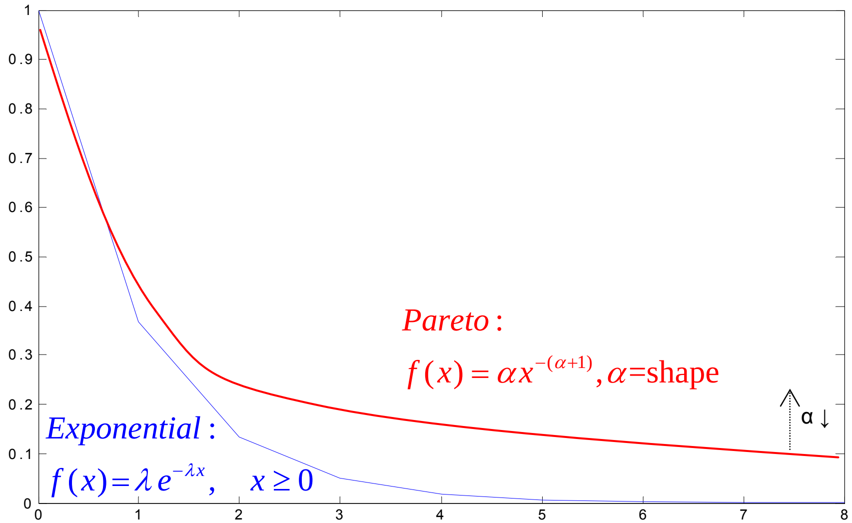
PDF: $f(t)=\lambda e^{-\lambda t}$

MEAN: $\frac{1}{\lambda}$

MEDIAN: $\frac{\ln 2}{\lambda} \cong \frac{.693}{\lambda}$

VARIANCE: $\frac{1}{\lambda^2}$

FAILURE RATE: $h(t)=\lambda$



$$\Pr[X > x] \approx x^{-\alpha},\ 0<\alpha<2$$

$$\Pr[X > x] = \frac{1}{x}, \alpha = 1$$

Exponential

1) Memoryless: probability of Bsize > b is the <u>same</u> regardless of how long the burst already is:

$$P(X > a+b \mid X > a) = P(X > b)$$

2) Mean and variance are finite (and simple):

$$E(X)=\frac{1}{\lambda}, resp.\ V(X)=\frac{1}{\lambda^2}$$

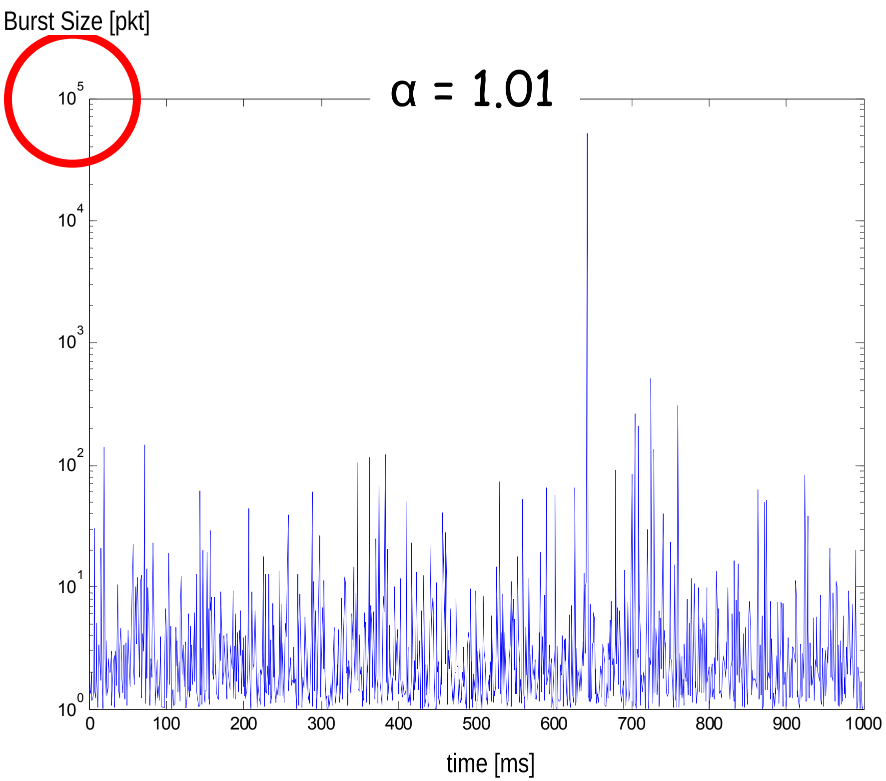
Pareto:

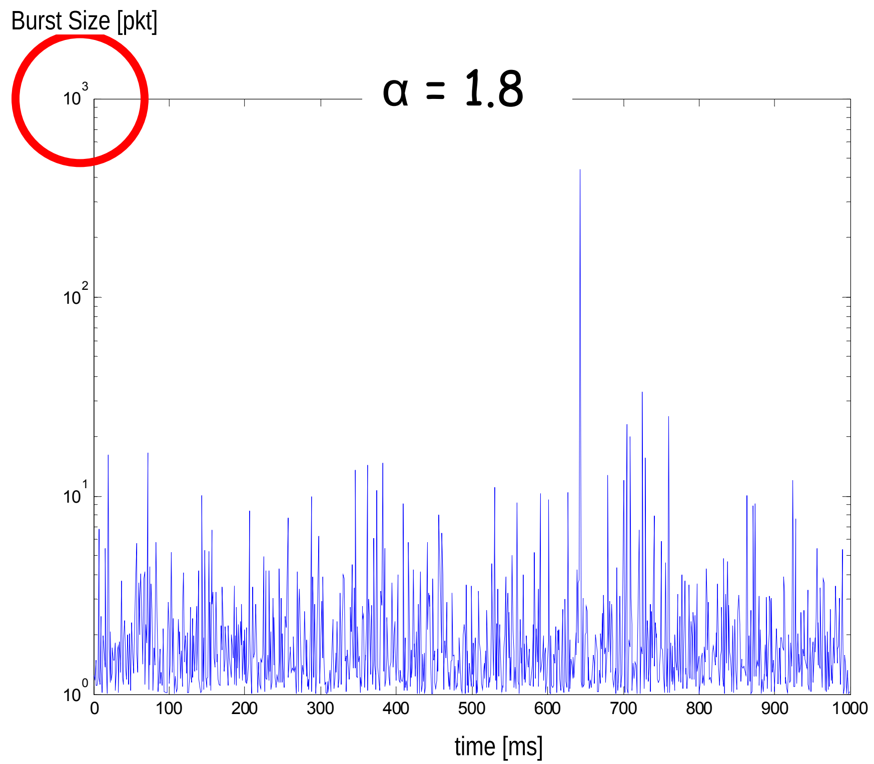
1. Mean and Variance unbounded
2. Heavy tail (α =1):
   1. For any burst length, the chance that it will double in size is 50%.
   2. Ca. 1% of the flows carry 50% of the volume (Bytes)
   3. For α >= 1 the expected burst size is bounded.

3. Central Limit Theorem does not apply
4. For 1< α < 2, despite bounded expected value, still

$$E(X)=?, resp.\ V(X)=\infty$$

Effect of Pareto Shape on Burstiness (generated w/ same seed)





- We should sweep the range between 1 and 2
  - Heaviest tail is in vicinity of 1
  - Less interesting around 2 and above
  - Most IP traffic studies found $1.1 < \alpha < 1.5$
- Range in datacenter traffic is unknown: $?? < \alpha < ??$

Elements of a Benchmark, Continued

- work
- job
- flow
- burst
- packet
- each has 2 random variables, for which we must choose a distribution.
  - agree on parameter values
  - exponential or Pareto

- If we choose Pareto for one or more of {flow, burst or packet} we reduce the use of analytical tools, with neither proof nor a clear benefit

  - A) No evidence of Pareto distribution for datacenter traffic
  - B) Will the original L4 distribution remain the same at injection time (at L2) ?

- We need to define α for one or more of {flow, burst or packet} distributions, but no guidelines exist for useful values of α in datacenters

FCT as Congestion Metric

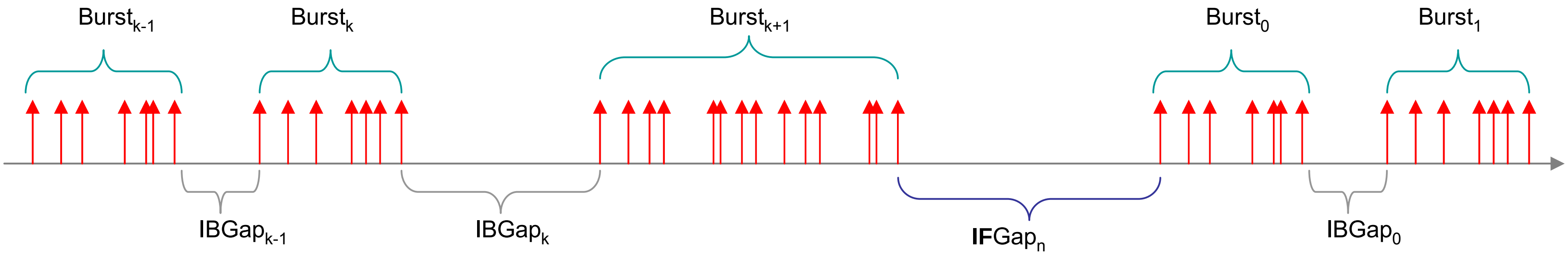
- FCT was recently proposed by Stanford Univ. for CM [refs]
  - "*FCT is an important – arguably the most important - performance metric for the user*" [N. Dukkipati, N. McKeown "Why Flow-Completion Time is the Right metric for Congestion Control and why this means we need new algorithms"]
  - FCT is being de-facto adopted also in .1au simulation results from Stanford, Cisco and ZRL
  - Characterizes CM performance from an User's perspective

- FCT: intriguing, yet difficult metric... It elicits precise
  1. Flow definition
  2. Completion definition
  3. Benchmarking measurement method

...none of which trivial !

You get what you measure...

- I) Assuming precise definition of "flow", measuring FCT results with PAUSE=On is un-ambiguous according to Case #1



- II) However, with PAUSE = Off, FCT *also* requires definition of "completion"
  - flows entirely received w/o any loss
  - flows entirely received w/ some loss
  - flows partially received
  - flows not arrived yet at destination...

- How do we count for these?
- Traffic-driven
  - to get good Tput, just drop all small flows (mice)
  - to get good latency, just drop all large flows (elephants)
- We need an agreed upon FCT approach to fully capture the relevant statistics

Difficulty of the FCT Metric

- Components of $FCT = \Sigma\,(t_{queue,i} + t_{inject,i} + t_{flight,i} + t_{RTX})$, for i = SRC to DST

- Q: Can these (complex) components be characterized by a single L_{e2e} variable?
- A: Depends on their distributions.

- Except $t_{flight,i}$ all other t's are independent random variables

  - if one or more of their PDFs are from Pareto distributions, the sum can NOT be represented by a single random variable L_{e2e} with the same expected value, mean and variance.

$$\boldsymbol{FCT = \Sigma\,(t_{queue,i} + t_{inject,i} + t_{flight,i} + t_{RTX}) \neq L_{e2e}(X)}\text{ , } \textit{i.e. CLT doesn't apply.}$$

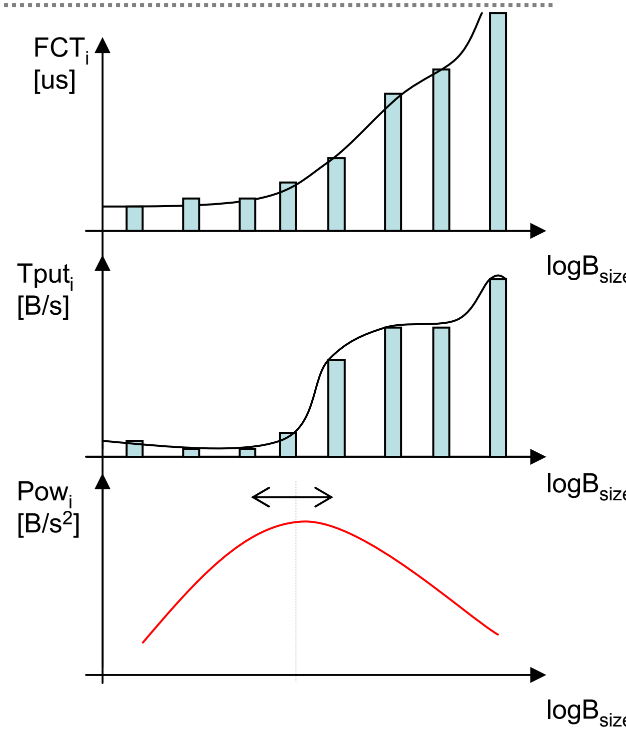
→ Each term of the sum above (except t_{flight}) must be independently analysed and reported. A global FCT is not meaningful w/o a detailed breakdown.

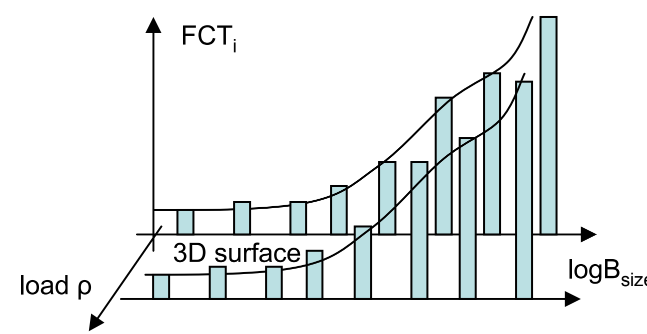
Case #1: Lossless ICTN FCT Measurement

- Iff
  1. workload defined as in our Bursty Benchmark "Trace File" proposal, and,
  2. PAUSE is enabled

⇒ <u>Measurement method</u>:

1. Conduct N no. runs for 95% confid. interv.
2. Collect flow stats in K=8 histogram bins
   1. Collect aggregate Job and Work stats
   2. Work Completion Time (WCT): Full drain.
3. Display on log axes (see ex. plot)
   1. FCT_i
   2. $Tput_i$
   3. $Power_i = Tput_i / FCT_i$
4. Repeat (1-3) for different loads / HSV
   1. Optional, 3D surfaces of 3.1..3
5. Calculate mean aggregate Tput
   1. per Workload = WKLD_Size [B] / WCT
   2. per burst size $Tput_{size} = \Sigma Tput_i / K$



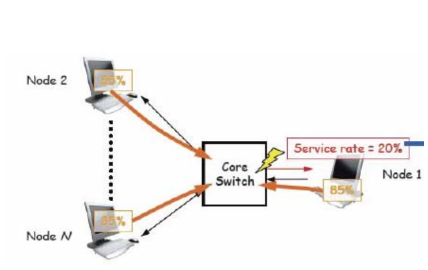


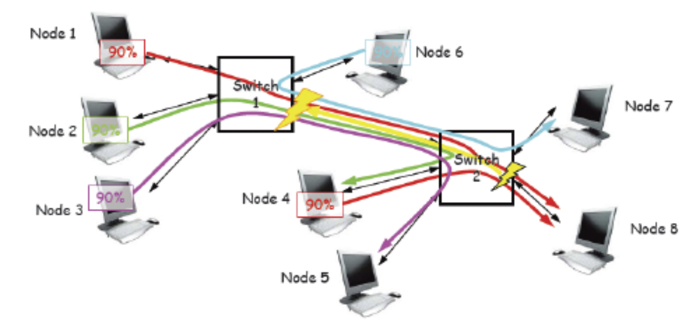
Case #2: Lossy ICTN FCT Measurement

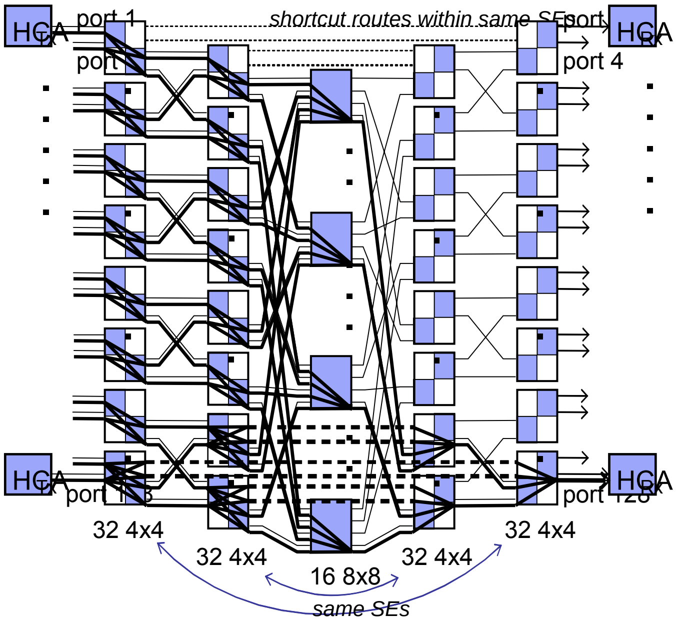
1. For PAUSE = Off (assuming some RTX method in place)
   1. we must qualify "completion" and distinctly count the Bytes per flows:
   2. Fully Completed w/o loss => Good-put
   3. Fully Completed with loss => Part-put
   4. Partially Completed => Part-put
   5. Dropped => Drop-put

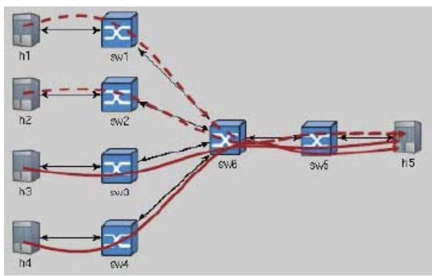
2. Goodput: Perform steps 1-5 as in Case #1

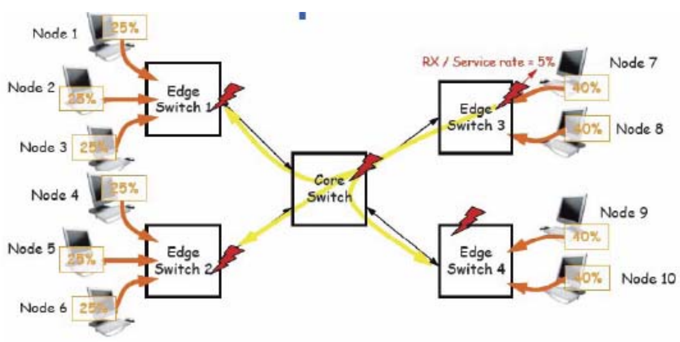
3. Report Drop- and Part-put





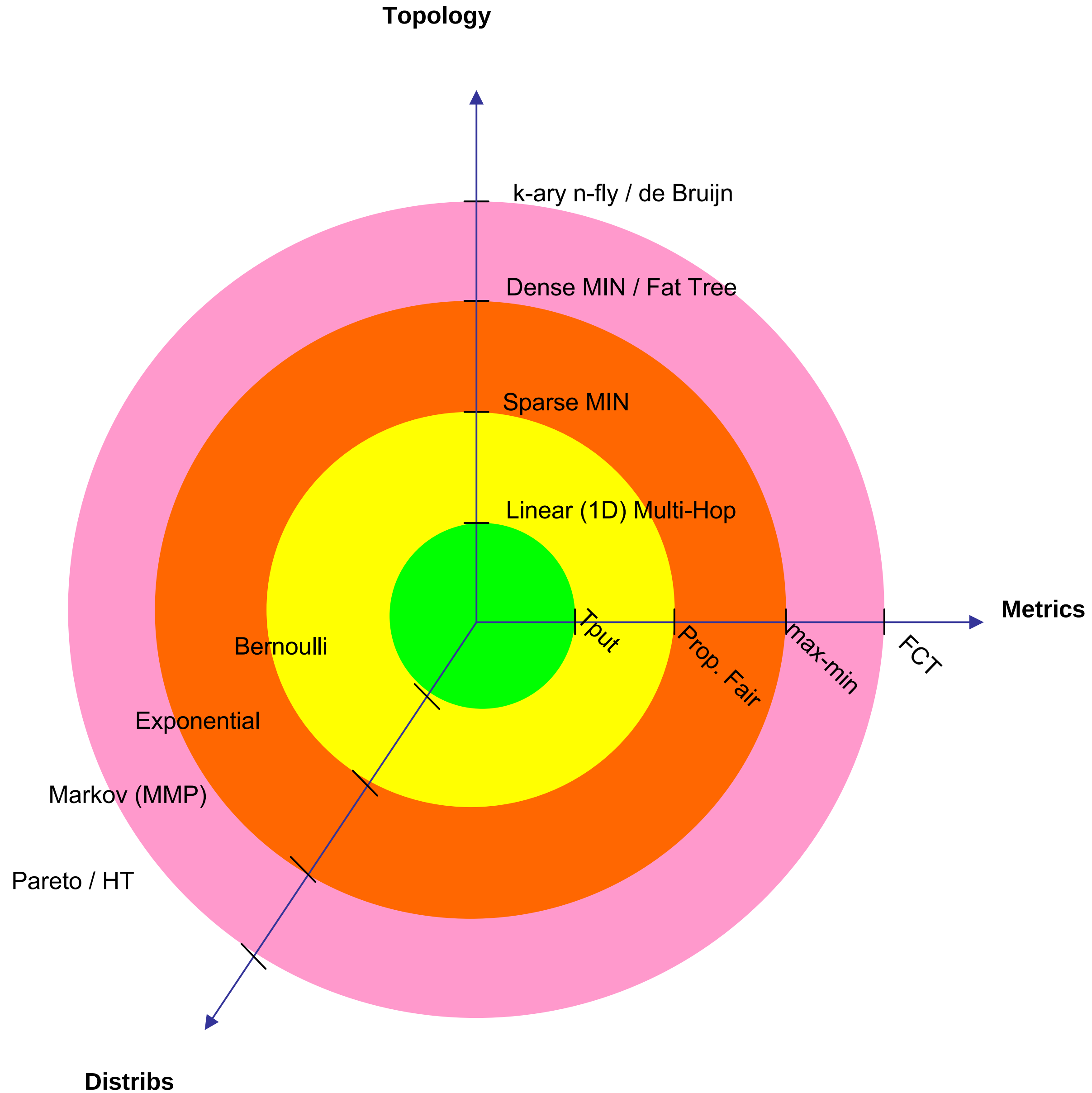






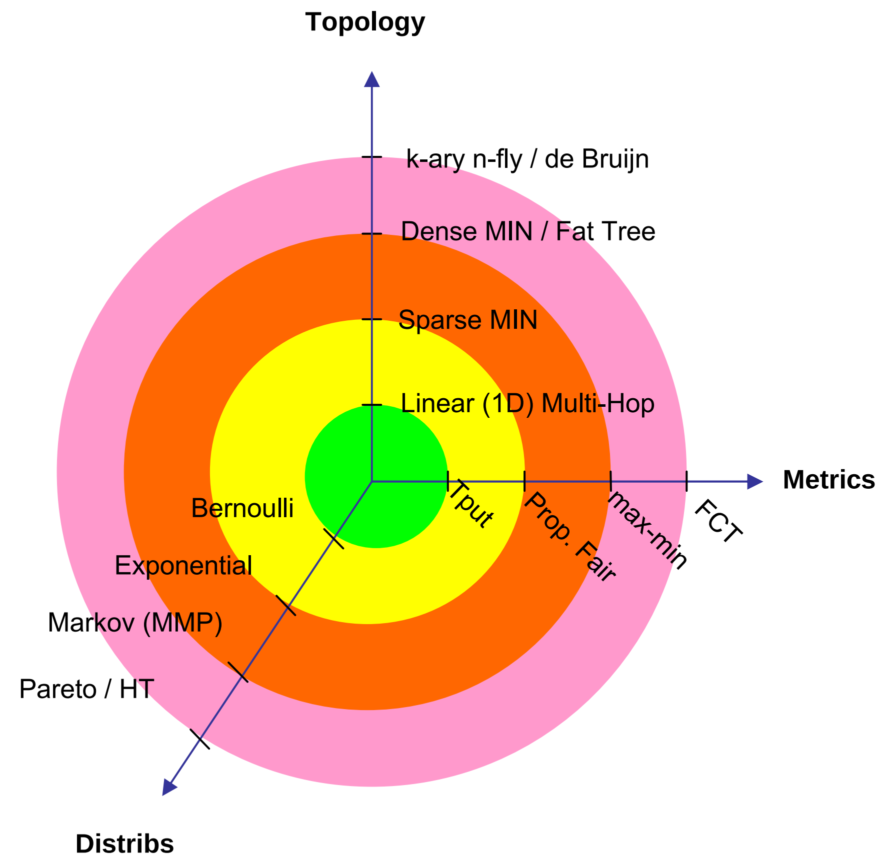
- From single stage and dumbbells (unidim. topo graphs) to 2D nets: a step up in realism (and complexity)
  - sim runtimes grow (super/sub)-linear: see ZRLs plots in [ref]

Putting It All Together: CM Benchmarking Sphere



Putting It All Together: CM Benchmarking Sphere

- Our benchmarking proposal
  - Method for reproducible results
  - Furthers the approach proposed in Orlando by [Cisco]
- Traffic Gen. code jointly developed w/ Cisco and Broadcom
  - preliminary results from Cisco and ZRL
- Next steps
  - discuss and improve Bursty Benchmark r1.0
  - adopt it
  - discuss the CM BMRK Sphere →



- CM Benchmarking Sphere:
  - concentrical layers => balance
  - natural expansion of layers => realism
  - avoid unidimensional explorations
    - ✓ we kept the topology simpler than the known DC reality, while speculatively exploring along the the other 2 axes.

Bursty Benchmark Proposal: Traffic Generator Details

- Fixed pkt size = 1.5KB MTU
  - generate a fixed size "Trace File" => WSize as system workload

- Trace format

```
| (1) Time | (2) SRC | (3) DST | (4) Prio | (5) BSize |
```

- For testing the Traffic Generator necessary to generate the above trace
  - Install and link the following distribution functions from Gnu Scientific Library (GSL):
  1. `gsl_ran_exponential (const gsl_rng * r, double mu)`
  2. `gsl_ran_pareto (const gsl_rng * r, double a, double b)`
  - use Pareto $1 < a < 2$ and scale $b = 1.0$

- Benefit of GSL: The IEEE environment settings (FP precision, rounding/truncation, ordering) are automatically taken care of...!
  - results are consistent across a wide range of machines, CPUs and OSes

ACKs: Contributions from D. Bergamasco and B. Kwan.

Conclusions

- Bursty Benchmark traffic generator and trace files will be available
  - use exponential first, possibly extended by bounded Pareto distribs
  - initially we recommend the trace file to calibrate our baseline sims
- FCT is an intriguing, yet time-intensive new metric
  - recently proposed in CM
  - can characterize performance from User's point of view
- However, in DC environments it can be confusing, even misleading…
  - requires large investment for little practical value
- Suggestions to .1au
  1. Adopt the Bursty Benchmark to achieve consistent and reproducible results
  2. Use the established metrics (Qlenght, Tput, fairness)
  3. Focus on real topologies instead of unproven metrics