Statistical Approach to NoC Design

Itamar Cohen, Ori Rottenstreich and Isaac Keslassy

Technion (Israel)
The traffic matrix in NoCs is often-changing and unpredictable

⇒ makes NoCs hard to design
Goal: Design road capacities between a city and its suburbs

Let’s model the traffic matrices...
Road Capacities

- **Morning peak**: most traffic towards city
Road Capacities

- **Morning peak**: most traffic towards city

- **Afternoon peak**: most traffic leaving city
Solution (1): Average-Case

- Solution (1): plan for average-case
  - i.e. allocate capacity of ~5 for each link.
  - $\lambda < \mu$
- Problem: traffic jam during many hours, every day.
Solution (2): Worst-Case

- Solution (2): plan for worst-case
  - i.e. allocate capacity of 10 for each link.
Problem: traffic burst in the holidays ⇒ excessive resources

Solution (3): statistical approach
- Enough capacity for 99% of the time
- Allow for occasional congestion
Back to the NoC world

- Similar problems in NoC design process
  - City → Shared cache
  - Suburbs → Cores
  - Many possible traffic matrices: writing, reading, etc.
Motivation

Tradeoff:
more resources vs. better guarantee

“Easy” solution: Worst-Case Guarantee
- E.g., no-congestion guarantee for 100% of the time

Objective: Statistical Guarantee
- E.g., no-congestion guarantee for 99.99% of the time
Statistical Approach to NoC Design

Given:

1. Traffic matrix distribution
   - Topology
   - Routing
   - Link capacities

2. Compute congestion guarantee
   - “95% of traffic matrices will receive enough capacity”

3. Optimal Capacity Allocation
How can we model the future traffic matrix distribution?
Traffic Matrix Distribution

- **Measure** on real large CMP networks?
- **Problems**
  - Their future applications are unknown
  - Their future traffic types are unknown: Between processors? Cache accesses? Control traffic?
  - (Chicken-and-egg problem: traffic might depend on what architecture offers)
General model: any core can communicate with any core - but with a bounded core input/output rate.

Example: each core may send/receive up to 1 Gbps.

Used in CMPs [Murali et al. ’07] – but also backbone networks [Dukkipati et al. ’05], interconnection networks [Towles and Dally ’02], VPN networks [Duffield et al. ’99], routers [McKeown et al. ‘96]

Core 1 sends

\[ T = \begin{pmatrix} 
0 & 0.5 & 0.1 & 0.2 \\
0 & 0 & 0 & 0.6 \\
0.2 & 0.1 & 0 & 0 \\
0.1 & 0.1 & 0.3 & 0 \\
\end{pmatrix} \leq 1 \]

Reminder: just one model among many…
Given:

1. Traffic matrix distribution
2. Topology
3. Routing
4. Link capacities

Compute congestion guarantee

“95% of traffic matrices will receive enough capacity”
Show that “95% of all traffic matrices will receive enough capacity on link $l$”

$\Leftrightarrow$

1. Compute *Traffic-load distribution Plot* (or *T-Plot*) for link $l$
2. Show that the load on link $l$ is less than its capacity for 95% of traffic matrices.
We want to provide statistical guarantees on the whole network i.e. on all links

**Global T-Plot:** for each traffic matrix $T$, measure maximum load among all links
Local T-Plots in NoCs

Given:
- Traffic matrices $T \in S$
- 3x4 mesh topology
- XY routing
- Link $l$

Find T-Plot on $l$
Close-up view:

- **Gaussian?**

- **Worst-case traffic load = 2**
  [Towles and Dally, ’02; Towles, Dally and Boyd, ’03]

- **99.99% of traffic matrices bring load under 1.6**

- **20% gain**
Are T-Plots Gaussian?

- Some T-Plots may be well approximated as Gaussian.

- **Intuition (Central Limit Theorem):** when $N$ grows, the sum of $N$ i.i.d. (independent and identically distributed) random variables converges to a Gaussian distribution.
Gaussian T-Plot Example

- Example: $n \times n$ mesh with XY routing.
- Assume i.i.d traffic from $i$ to $j$
- **Theorem:** As $n$ grows, T-Plot for $l$ converges to Gaussian.
Computing the T-Plot

Theorem: for an arbitrary graph and routing, computing the T-Plot is \#P-complete.

#P-complete problems are at least as hard as NP-complete problems.

NP: “Is there a solution?”

#P: “How many solutions?”
Computing the T-Plot

- Idea: use Monte Carlo simulations to approximate T-Plots
  - Plot many points, and count number of points to approximate density.
**Link T-Plot Vs Global T-Plot**

- **Global T-Plot**: for each traffic matrix $T$, measure maximum load among **all** links.

![Diagram](diagram.png)

- **PDF**: Higher average
- **Gaussian?**
- **Model? Assume independent links...**
Assume:

If $l_1$ and $l_2$ independent:

Global Load:

\[
\begin{pmatrix}
0 & w.p. \frac{1}{4} \\
1 & w.p. \frac{3}{4}
\end{pmatrix}
\]
Independent-Gaussian Model

CDF

Global Load

Simulations
Independent-Gaussian Model (Intuitive Lower Bound)
Statistical Approach to NoC Design

Given:
1. Traffic matrix distribution
2. Topology
3. Routing
4. Link capacities

Compute congestion guarantee
- “95% of traffic matrices will receive enough capacity”
Capacity Allocation Intuition

- Idea: given a total capacity, distribute it so that the overflow probability on each link is the same.

\[
\begin{align*}
\mu_2 &= 0.24 \\
\sigma_2 &= 0.11 \\
\mu_1 &= 1.96 \\
\sigma_1 &= 0.25
\end{align*}
\]
Capacity Allocation Intuition

- **Idea:** given a total capacity, distribute it so that the overflow probability on each link is **the same**.

- **Theorem:** if Gaussian-independent model holds, with same standard-deviation $\sigma$ on all links, then **this scheme is optimal**.
Capacity Allocation Performances

- Total needed capacity for various Capacity Allocation (CA) targets

Worst-case capacity saving with different link capacities

- Worst-case approach: 41%
- Statistical approach: 37%

Statistical gain with 90% guarantee
Average Flow Delay Model

- Assume M/M/1 delay model
- Average flow delay distribution over all traffic matrices, for two different Capacity Allocation (CA) schemes:
Given:

1. Traffic matrix distribution
   - Topology
   - **Routing**
   - Link capacities

2. Compute congestion guarantee
   - “95% of traffic matrices will receive enough capacity”

3. Optimal Capacity Allocation
Comparing routing algorithms

DOR (XY) and O1TURN (½ XY, ½ YX) on 3x4 mesh:

PDF

Global Load

DOR Simulations
DOR Fitted Gaussian
O1TURN Simulations
O1TURN Fitted Gaussian

O1TURN: 95% cutoff
DOR: 95% cutoff
More complex CMP architectures show similar results

- **NUCA** (Non-Uniform Cache Architecture) with sharing degree 4.
- Traffic model: each core (cache) may only send/receive traffic to/from caches (cores) in its sub-network.
Total capacity required for various Capacity Allocation (CA) targets.

Gain of statistical approach 48%
Summary

- Statistical approach to NoC design
  - Deals with several traffic matrices
  - Can optimize capacity allocation and routing

- Can be applied to any
  - Traffic matrix distribution
  - Topology
  - Oblivious routing
  - Link capacities
Thank you.