# Zero Load Predictive Model

*Axel Jantsch*

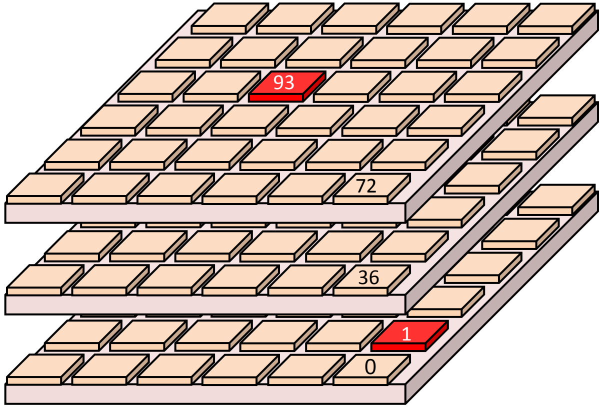
*Awet Y. Weldezion, Matt Grange, Roshan Weerasekera,*
*Hannu Tenhunen, Dinesh Pamunuwa,*

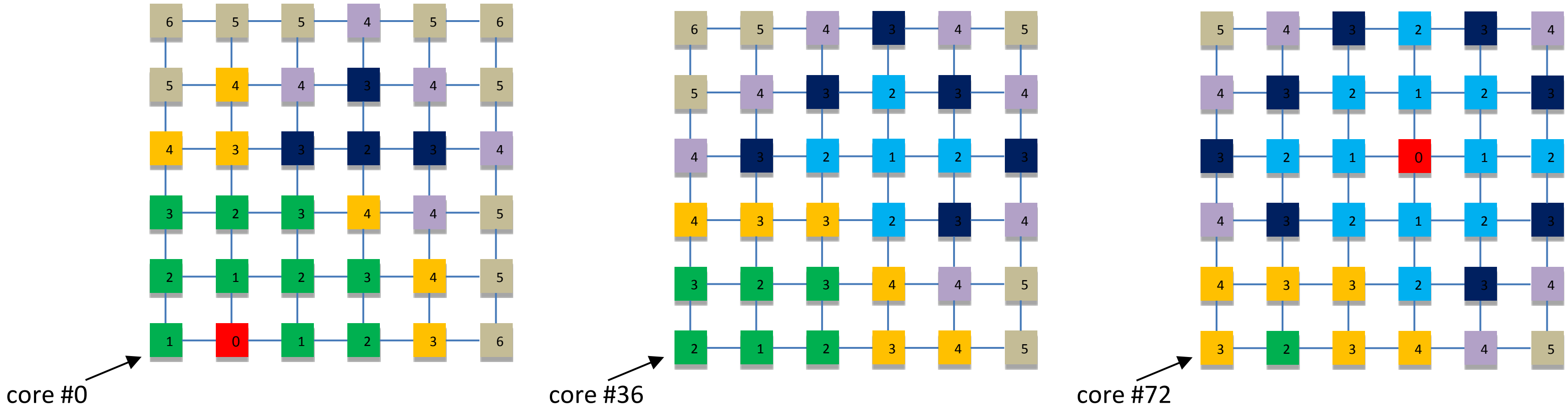
KTH VETENSKAP OCH KONST
ROYAL INSTITUTE OF TECHNOLOGY

*June 2013*

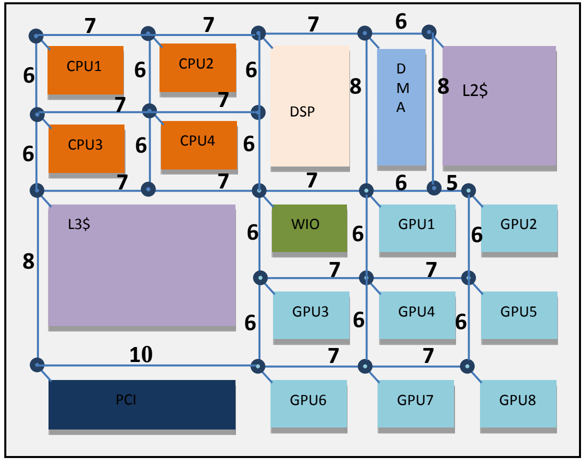
# Many Possible Configuration

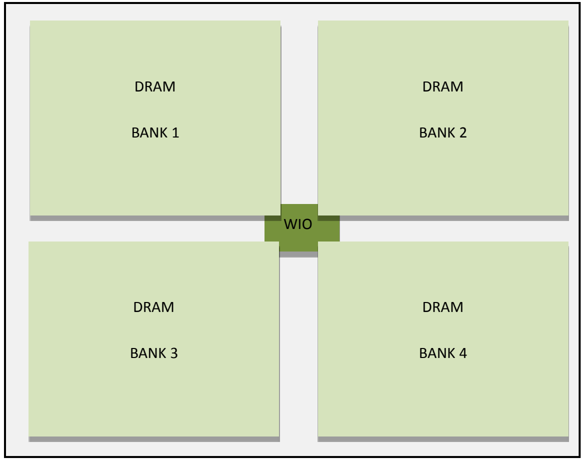
- 3 Stacked Layers with 6x6 nodes
- Two hotspots with placement constraints
- 5 types of processing units/nodes
- ~ $5 \cdot 10^{80}$ possibilities

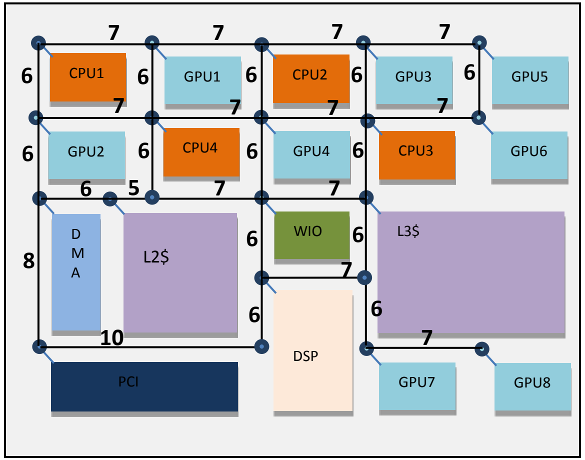


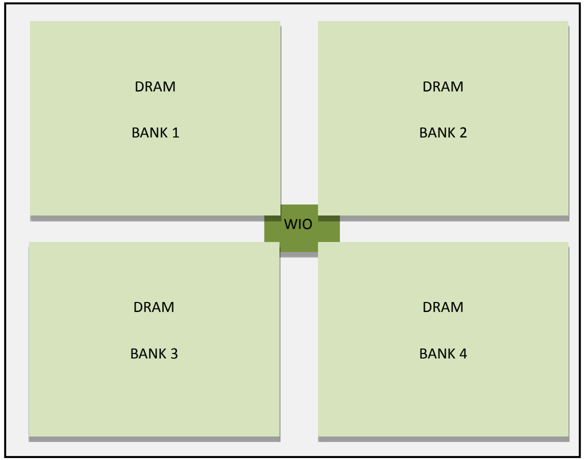



# Heterogeneous Systems

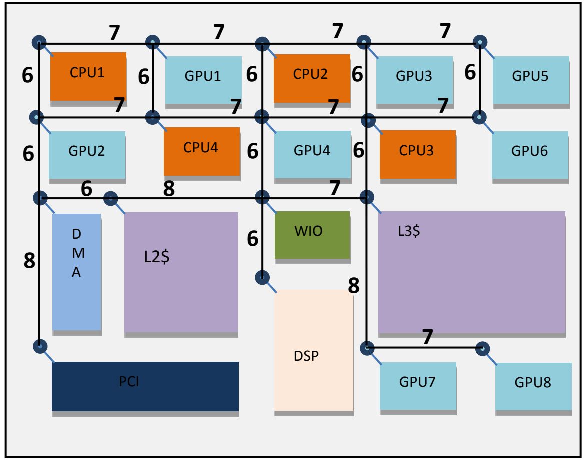


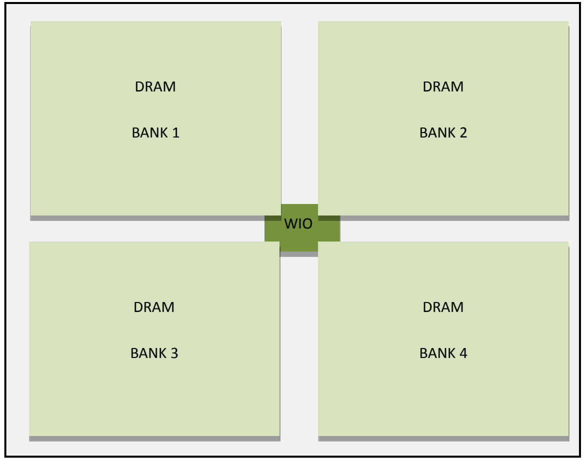



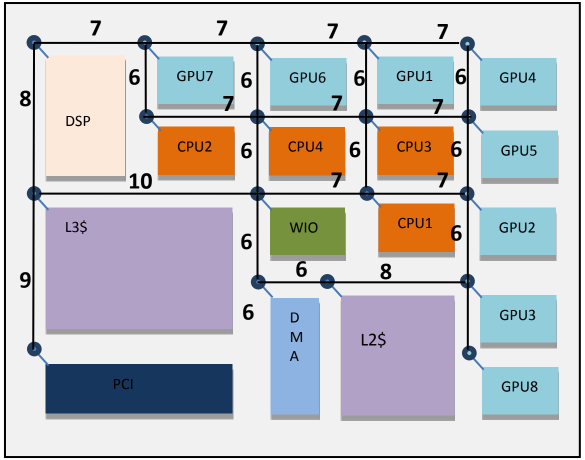



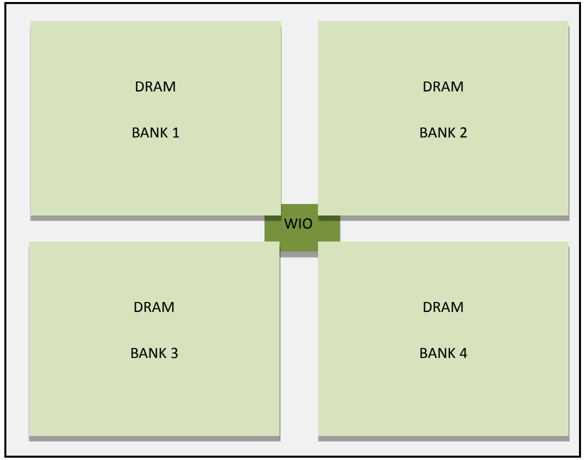



# Heterogeneous Systems



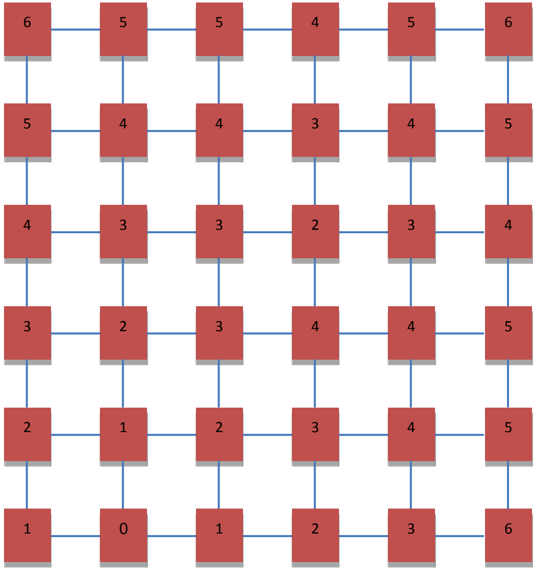



# Can we find the best configuration without simulation ?

# Zero Load Predictive Model

**is a static model and predicts, which configuration has better performance under load.**

# Average distance

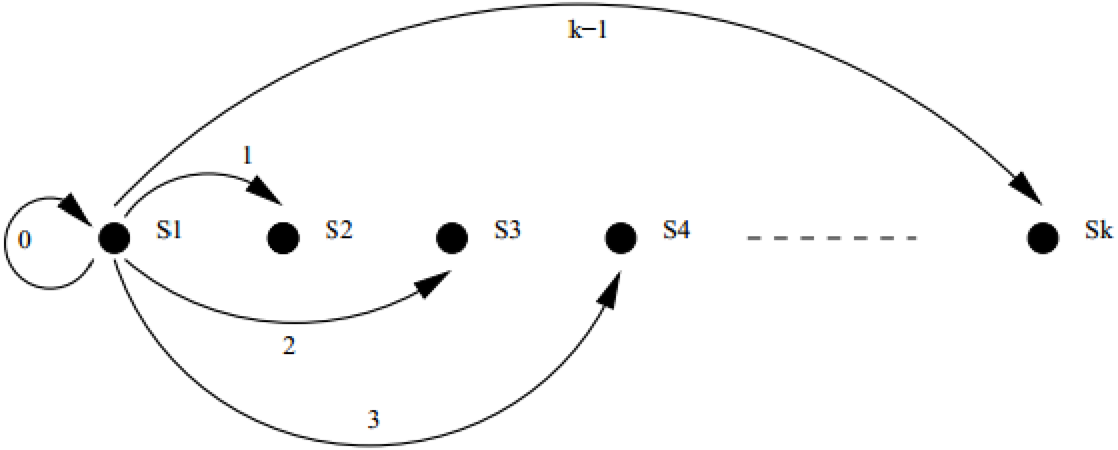



n x n Mesh for uniform random traffic:

$$\overline{H} = \frac{2}{3}n$$

# Average distance

- Average distance depends on the probability, p(i,j), of a packet to be sent to a destination and the actual source-destination Manhattan distance in terms of hops.




$$\bar{H}_{1\times k} = \frac{\sum_{i=1}^{k}\sum_{j=1}^{k} p_{i,j} \times |i-j|}{\sum_{i=1}^{k}\sum_{j=1}^{k} p_{i,j}}$$

# Average Distance for Uniform Random Traffic

**With self-traffic**

**With no self-traffic:**

- **n-dimensional mesh with radix k**

$$\overline{H} = \frac{n}{3}\left(k - \frac{1}{k}\right)$$

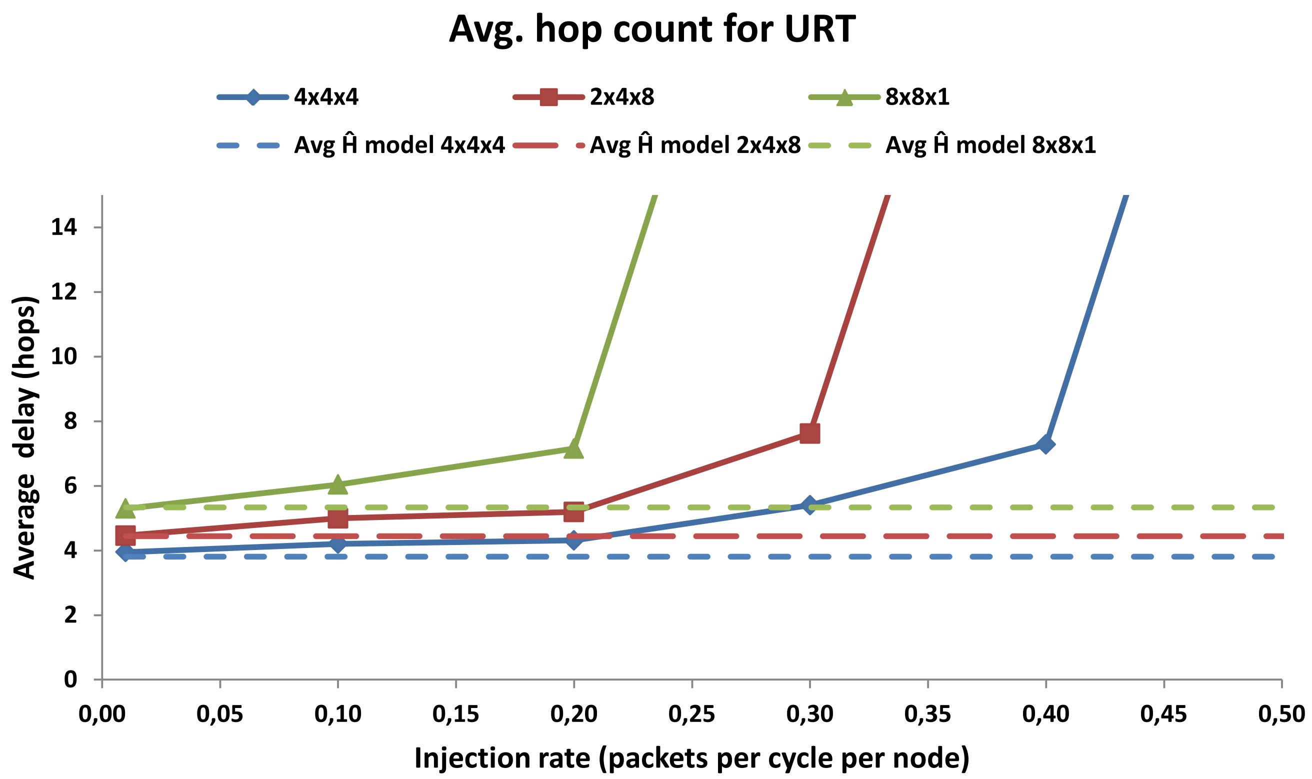
$$\overline{H} = \frac{n}{3}k$$

**n=3 with different k in each dimension:**

$$\overline{H} = \frac{1}{3}\left(k_1 - \frac{1}{k_1} + k_2 - \frac{1}{k_2} + k_3 - \frac{1}{k_3}\right)$$

$$\overline{H} = \frac{1}{3}(k_1 + k_2 + k_3)$$

# Uniform Random Traffic URT



# Spatial Distribution

- Each source node injects a packet that is sent to at least one destination based on a pre-defined patterns such as URT, Bit-reverse, Bit-complement….

- The source address (*S*) is in the form of bits (*b*) expressed as follows

$$\hat{S} = \boldsymbol{b}_m \, \boldsymbol{b}_{m-1} \, \boldsymbol{b}_{m-2} \dots\dots \, \boldsymbol{b}_3 \, \boldsymbol{b}_2 \, \boldsymbol{b}_1$$

# Bit-Reverse

- The destination is a mirror image of the source address.
- The average distance is the average of the sum of individual source-destination distances in all directions.

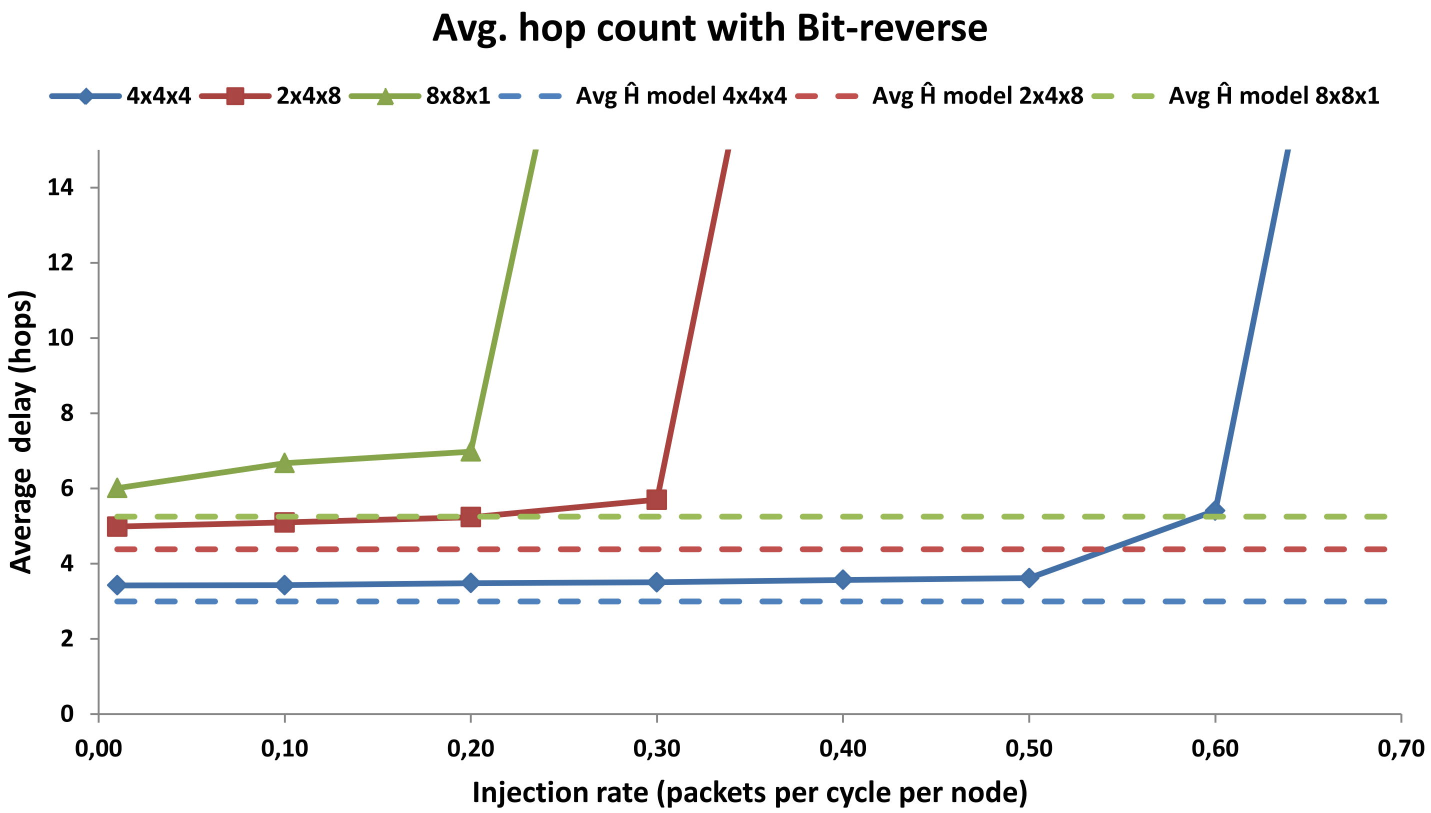
$$\hat{\mathrm{H}}_{\boldsymbol{br,xyz}} = |\text{Ƨ}_x - \hat{\mathrm{S}}_x| + |\text{Ƨ}_y - \hat{\mathrm{S}}_y| + |\text{Ƨ}_z - \hat{\mathrm{S}}_z|$$

$$\hat{\mathrm{H}}_{\boldsymbol{br}}(\boldsymbol{N}) = \frac{\mathbf{1}}{\boldsymbol{N}} \sum_{x \in N} \sum_{y \in N} \sum_{z \in N} \hat{\mathrm{H}}_{\boldsymbol{br,xyz}}$$

$\hat{\mathrm{S}}$ – **Source adress**

Ƨ – **reversed destination**

# Bit-Reverse



Avg. hop count with Bit-reverse
4x4x4
2x4x8
8x8x1
Avg Ĥ model 4x4x4
Avg Ĥ model 2x4x8
Avg Ĥ model 8x8x1
Average delay (hops)
0
2
4
6
8
10
12
14
0,00
0,10
0,20
0,30
0,40
0,50
0,60
0,70
Injection rate (packets per cycle per node)

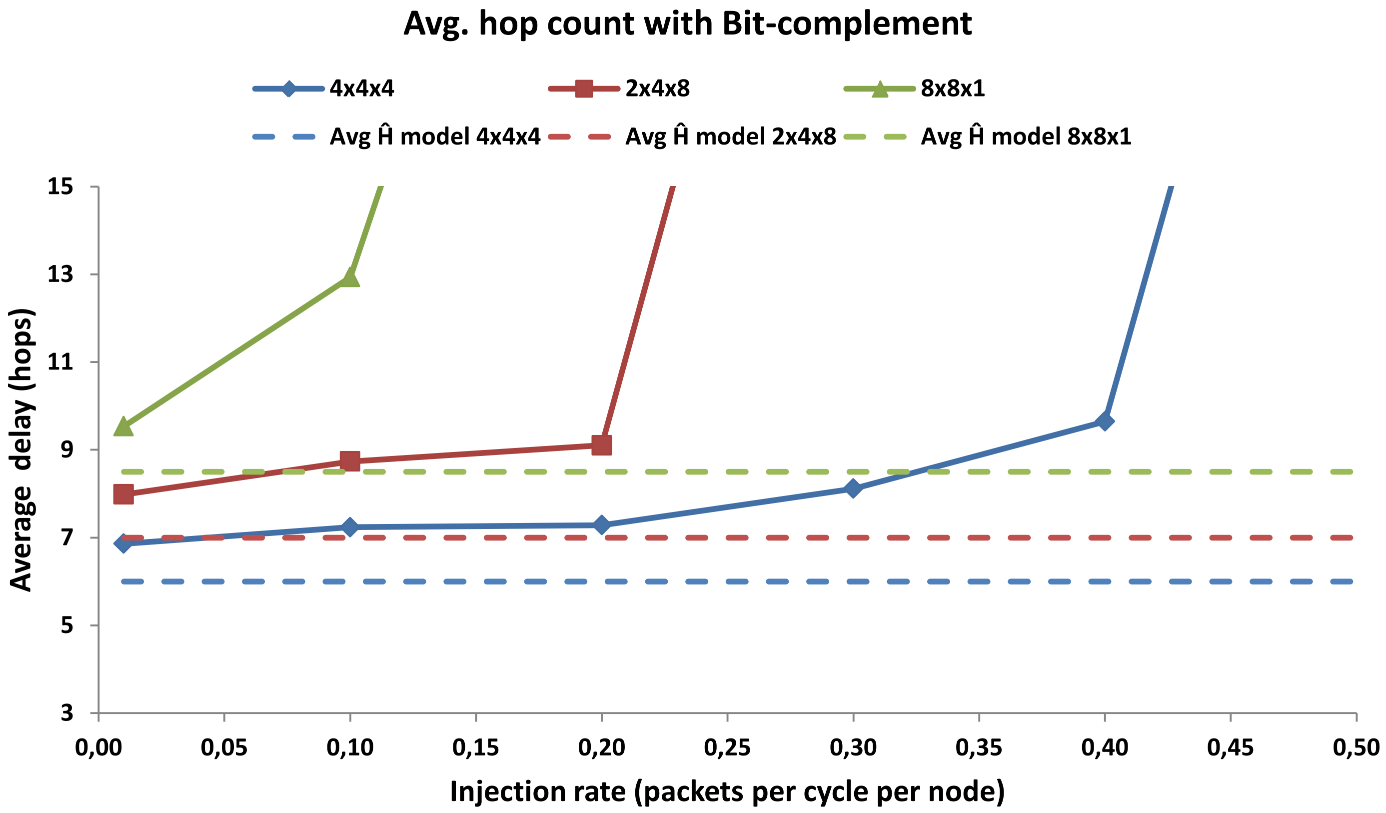
# Bit-Complement

- A destination address is set as the 1's complement of each bit of the source address
- For a given 2D and 3D network the average distance is expressed as follows respectively.

$$\hat{H}_{bc,2D}(N) = \frac{x + y}{2}$$

$$\hat{H}_{bc,3D}(N) = \frac{x + y + z}{2}$$

# Bit-Complement



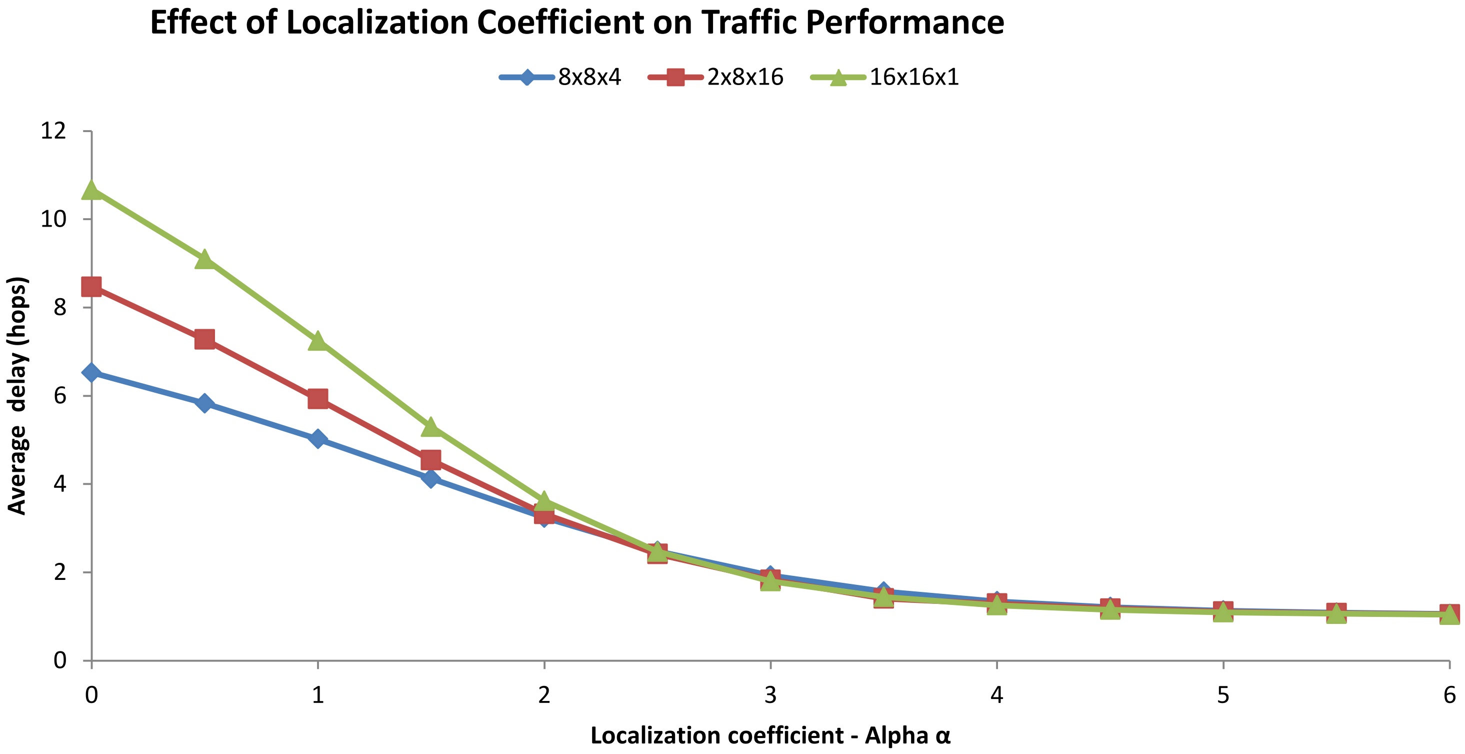
# α-Model Localized Pattern

- Under a local traffic pattern, packet destinations are preferred for destinations (*D*) close to the source.

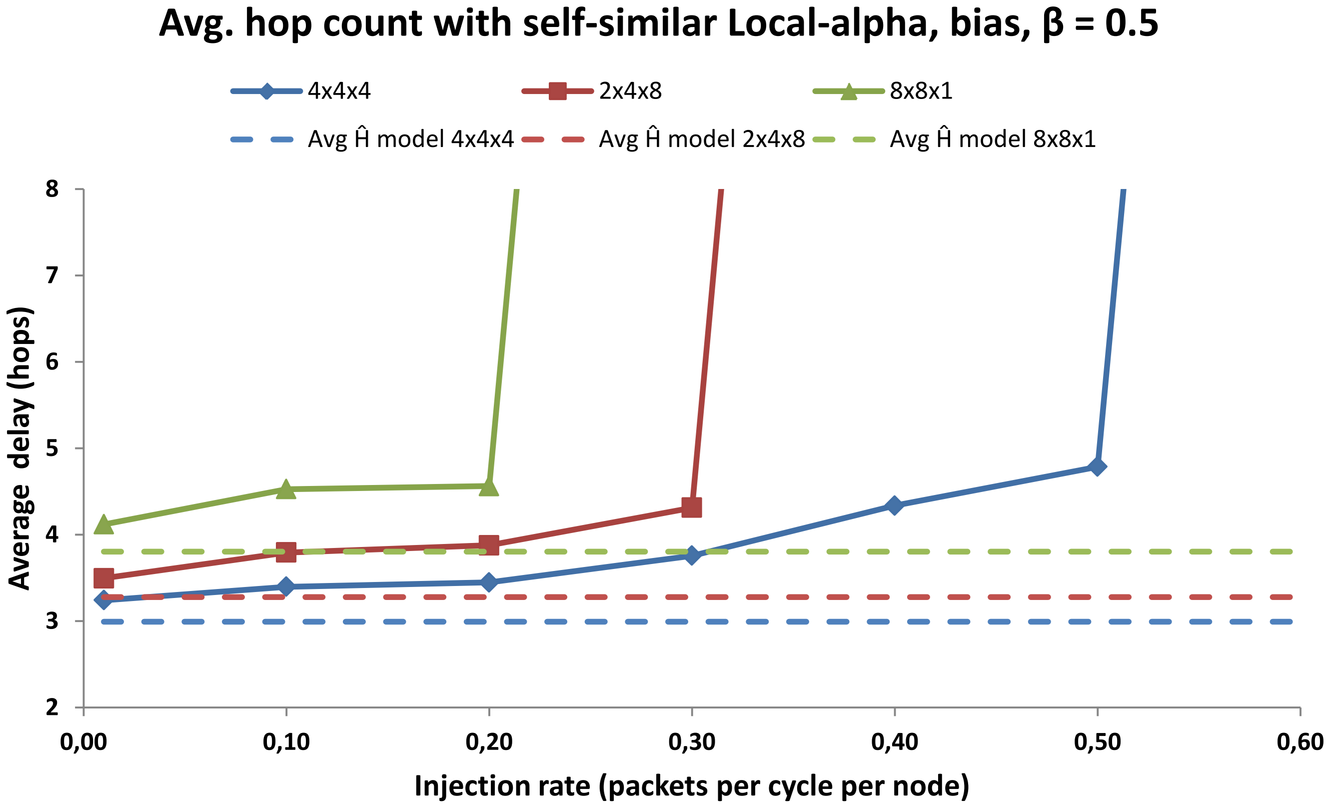
- The variation in level of closeness (localization) is represented with locality coefficient alpha, (α). When α = 0 it means the level of localization is low (in this case localization does not exist)Output

$$\hat{\mathrm{H}}_{\alpha,xyz} = \frac{1}{N-1} \sum_{\hat{\mathrm{S}}=0}^{N} \sum_{\check{\mathrm{D}}=0}^{N} \frac{\left|\hat{\mathrm{S}} - \check{\mathrm{D}}\right|^{\alpha+1}}{\sum_{\check{\mathrm{D}}=0}^{N} \left|\hat{\mathrm{S}} - \check{\mathrm{D}}\right|^{-\alpha}}$$

# α-Model Localized Pattern

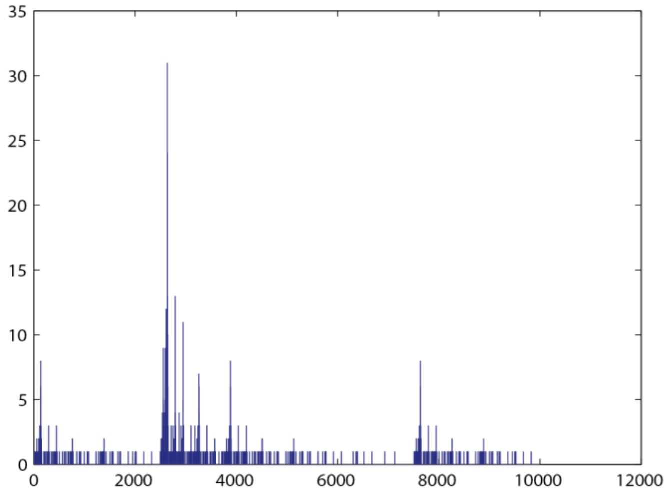


# α-Model Localized Pattern



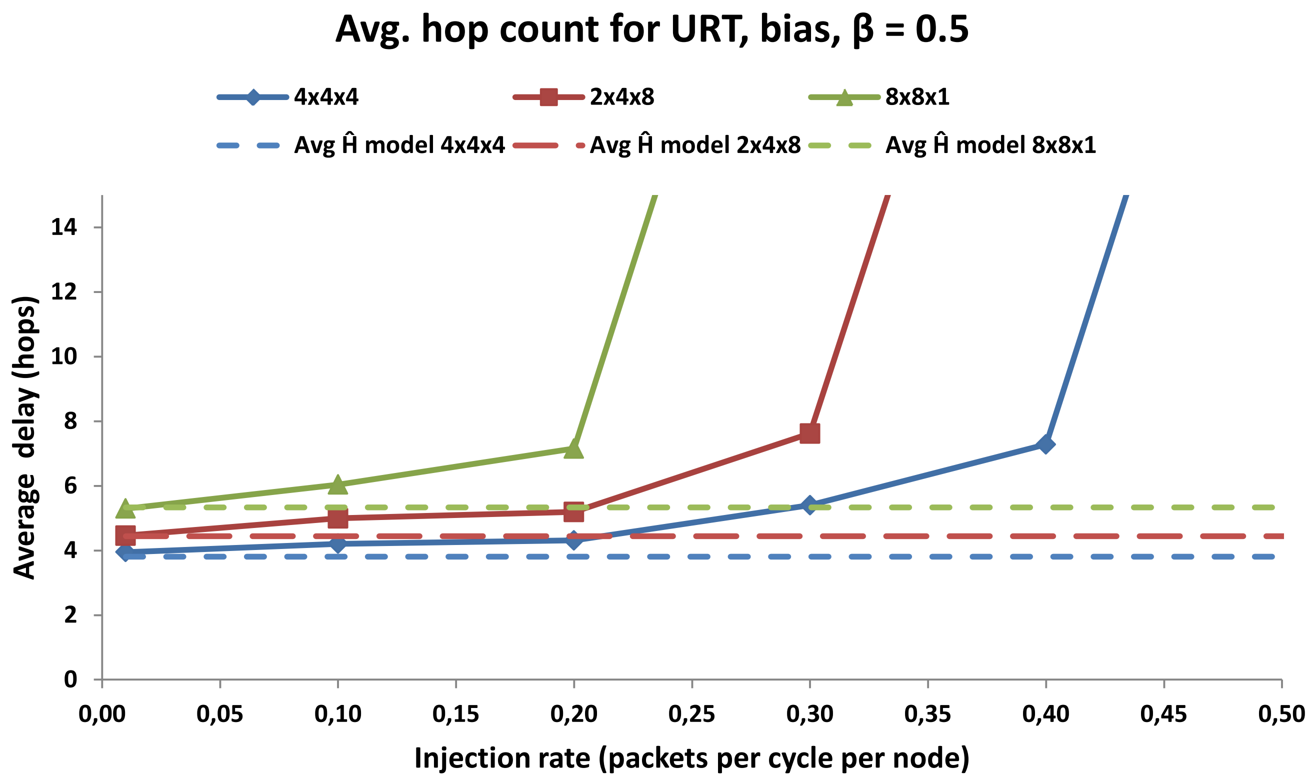
# Temporal Distribution (B-Model)

- Networks show a pattern of self-similarity in real traffic scenario.
- In a Bursty model, at any point, x(i*n/2d), in a time series, the number of packets that a node injects to the network is expressed as a function of the bias, β, the division depth, d, and the injection rate, γ.



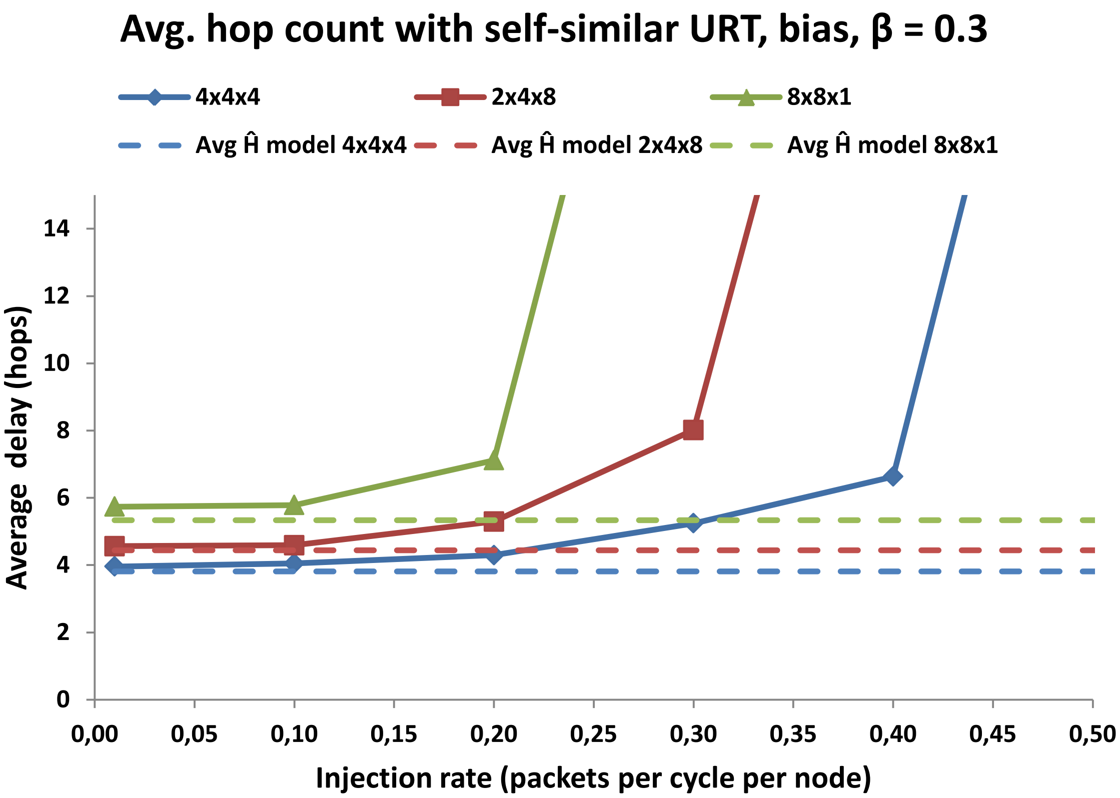

$$\mathrm{x}\left(i\frac{n}{2^{\mathrm{d}}}\right) = (\{\beta, 1-\beta\})^{\mathrm{d}}\left(\gamma n - \sum_{j=0}^{i-1} x(j\frac{n}{2^{\mathrm{d}}})\right)$$

# URT

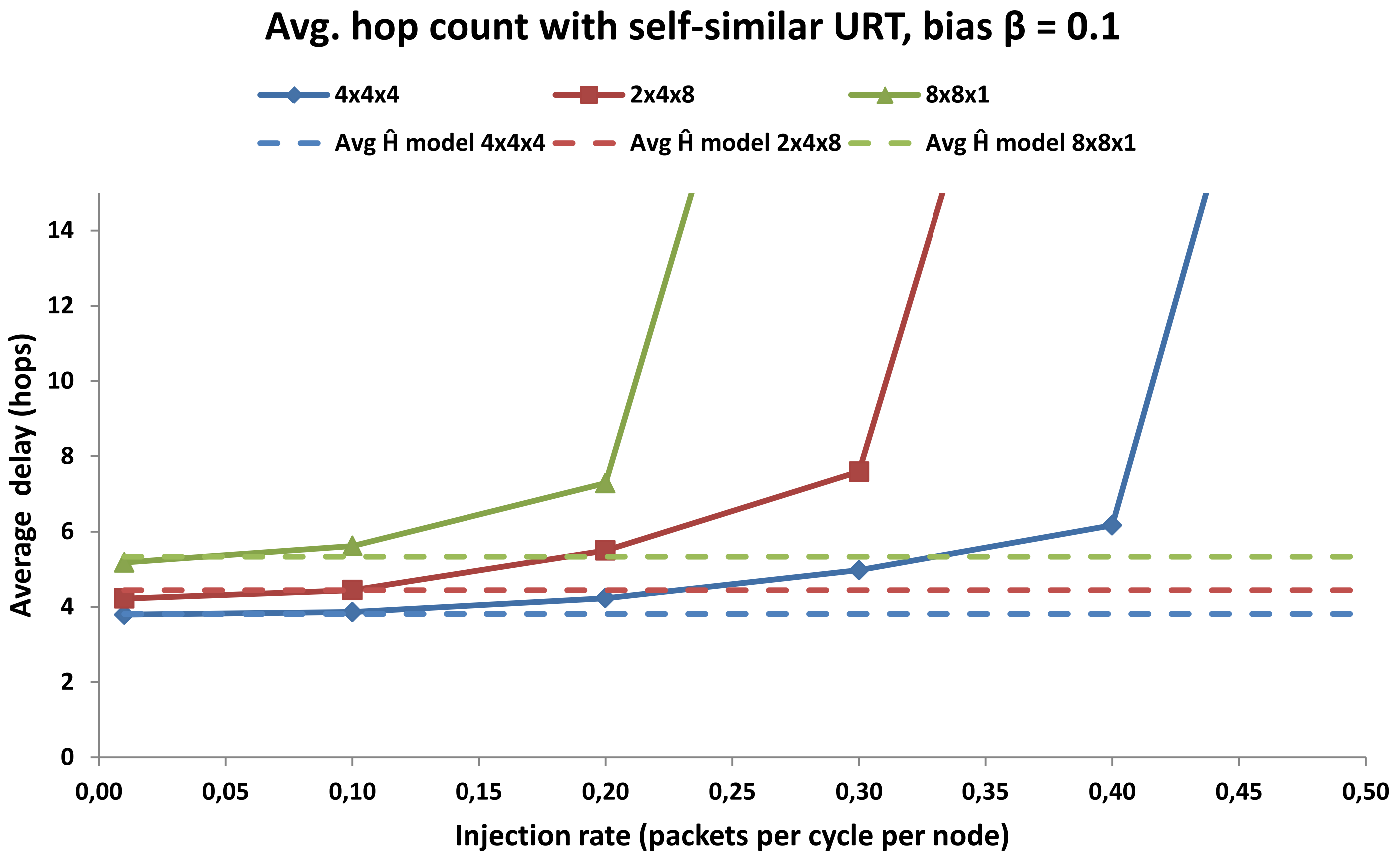


Avg. hop count for URT, bias, β = 0.5
4x4x4
2x4x8
8x8x1
Avg Ĥ model 4x4x4
Avg Ĥ model 2x4x8
Avg Ĥ model 8x8x1
Average delay (hops)
0
2
4
6
8
10
12
14
0,00
0,05
0,10
0,15
0,20
0,25
0,30
0,35
0,40
0,45
0,50
Injection rate (packets per cycle per node)

# URT

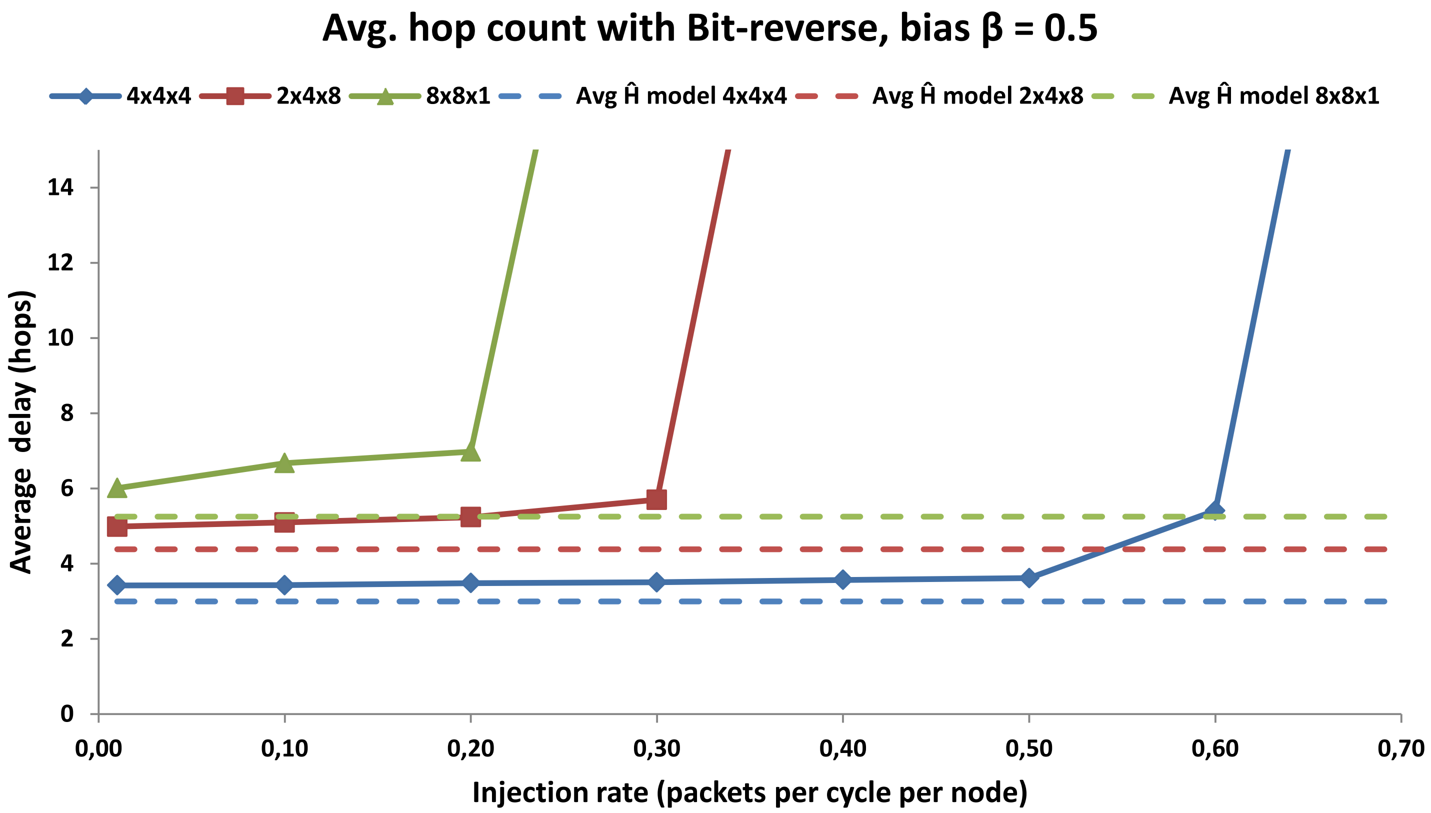

Avg. hop count with self-similar URT, bias, β = 0.3
4x4x4
2x4x8
8x8x1
Avg Ĥ model 4x4x4
Avg Ĥ model 2x4x8
Avg Ĥ model 8x8x1
Average delay (hops)
0
2
4
6
8
10
12
14
0,00
0,05
0,10
0,15
0,20
0,25
0,30
0,35
0,40
0,45
0,50
Injection rate (packets per cycle per node)

# URT



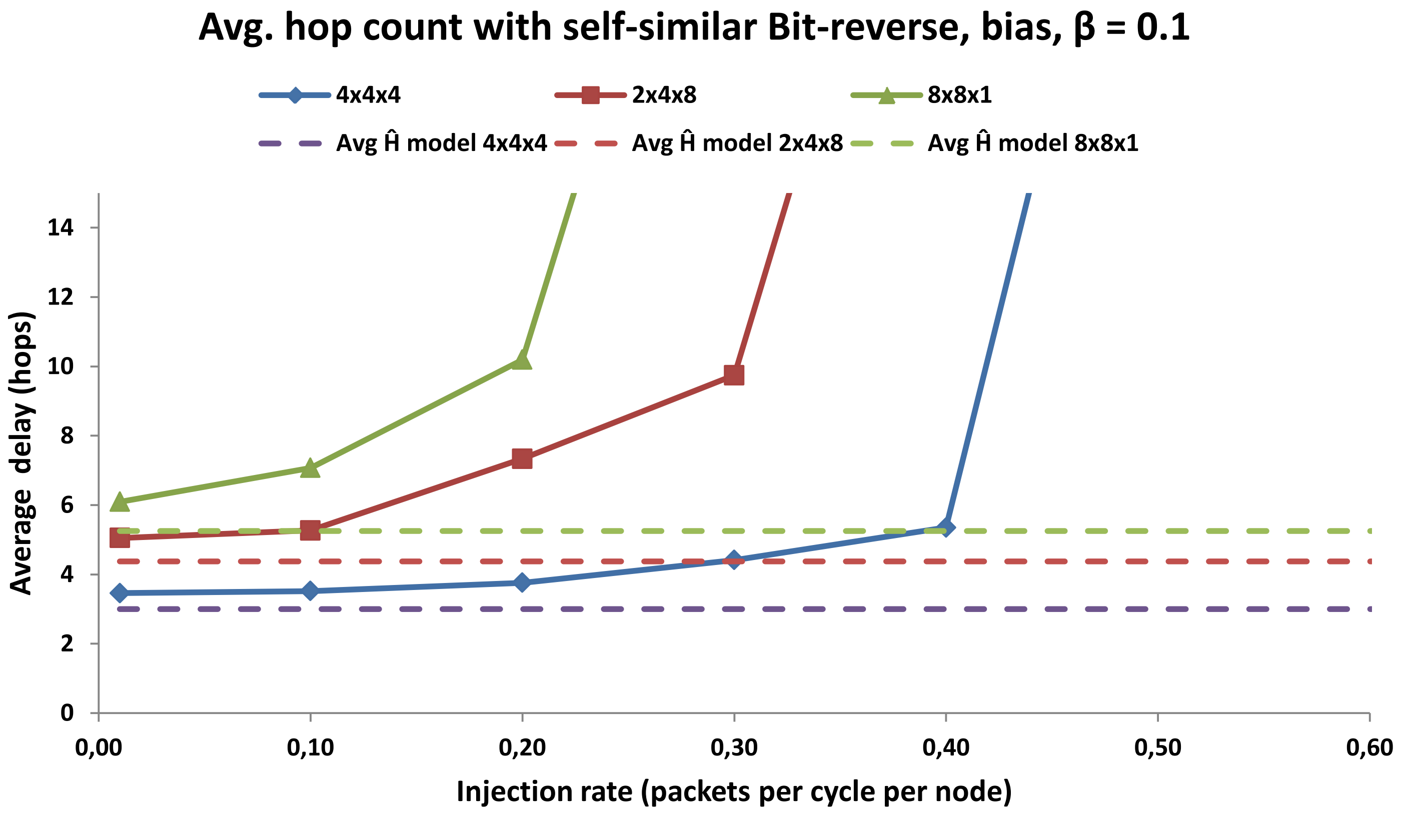
Avg. hop count with self-similar URT, bias β = 0.1
4x4x4
2x4x8
8x8x1
Avg Ĥ model 4x4x4
Avg Ĥ model 2x4x8
Avg Ĥ model 8x8x1
Average delay (hops)
0
2
4
6
8
10
12
14
0,00
0,05
0,10
0,15
0,20
0,25
0,30
0,35
0,40
0,45
0,50
Injection rate (packets per cycle per node)

# Bit-Reverse



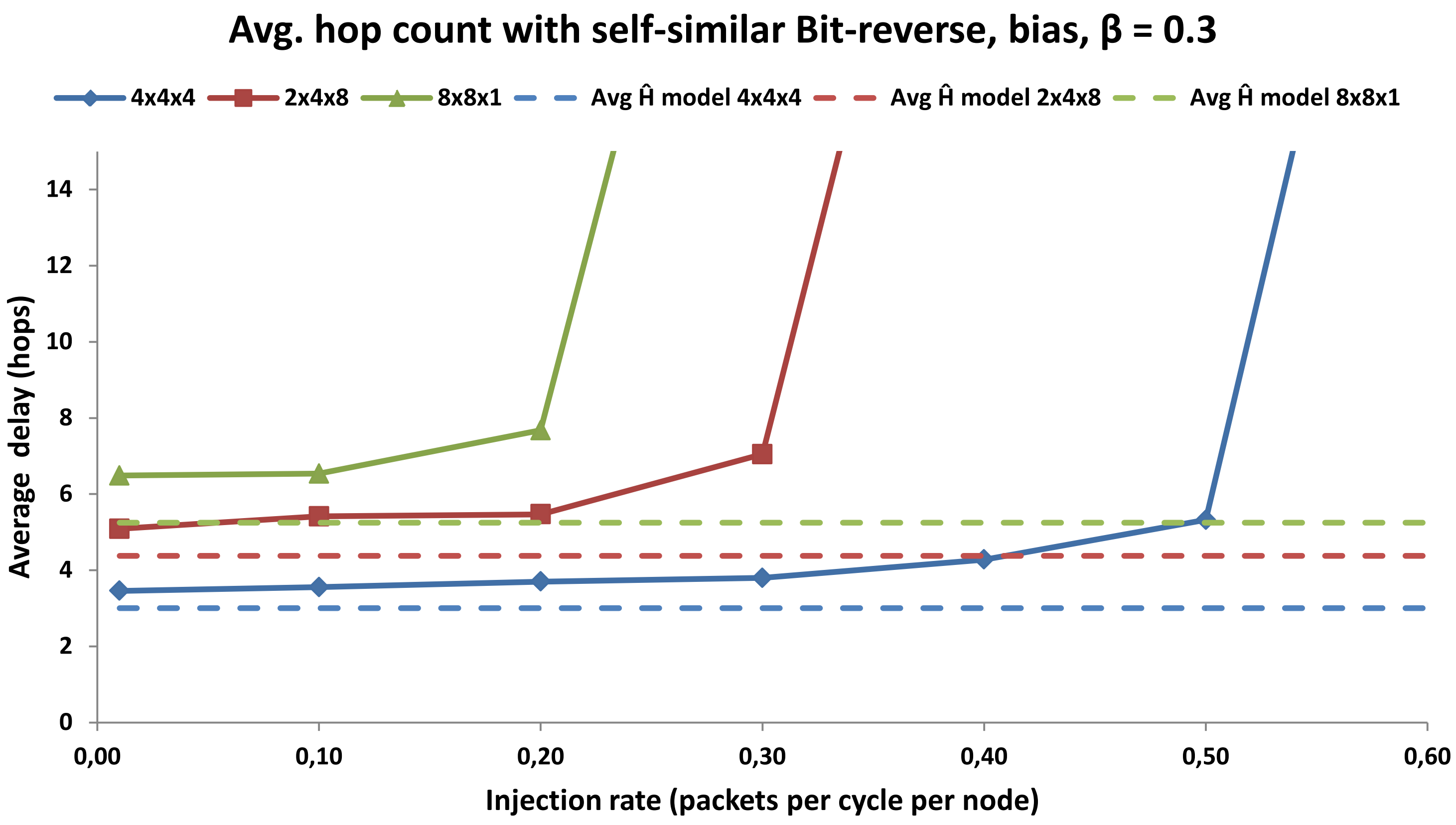
Avg. hop count with Bit-reverse, bias β = 0.5
4x4x4
2x4x8
8x8x1
Avg Ĥ model 4x4x4
Avg Ĥ model 2x4x8
Avg Ĥ model 8x8x1
Average delay (hops)
0
2
4
6
8
10
12
14
0,00
0,10
0,20
0,30
0,40
0,50
0,60
0,70
Injection rate (packets per cycle per node)

# Bit-Reverse



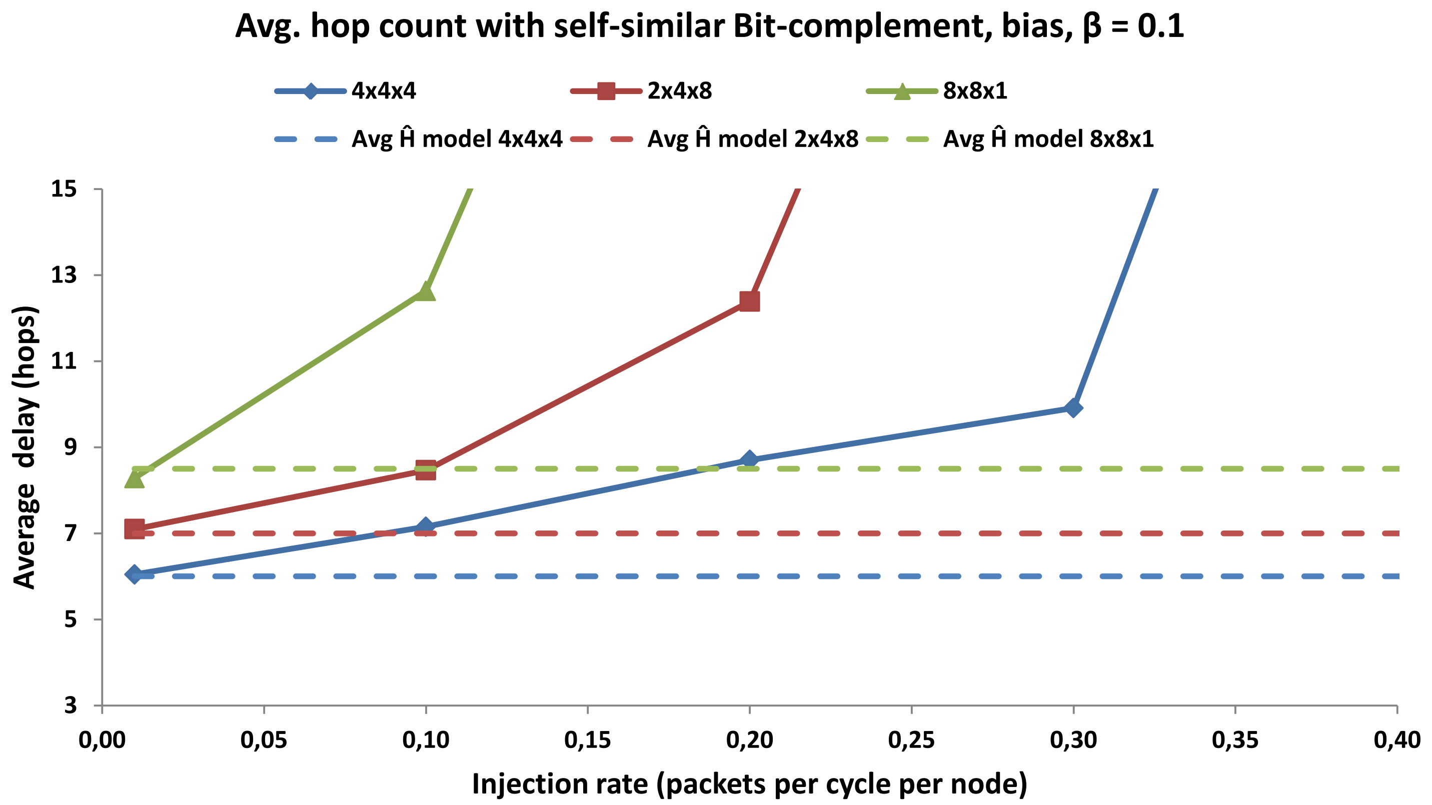
Avg. hop count with self-similar Bit-reverse, bias, β = 0.1
4x4x4
2x4x8
8x8x1
Avg Ĥ model 4x4x4
Avg Ĥ model 2x4x8
Avg Ĥ model 8x8x1
Average delay (hops)
0
2
4
6
8
10
12
14
0,00
0,10
0,20
0,30
0,40
0,50
0,60
Injection rate (packets per cycle per node)

# Bit-Reverse

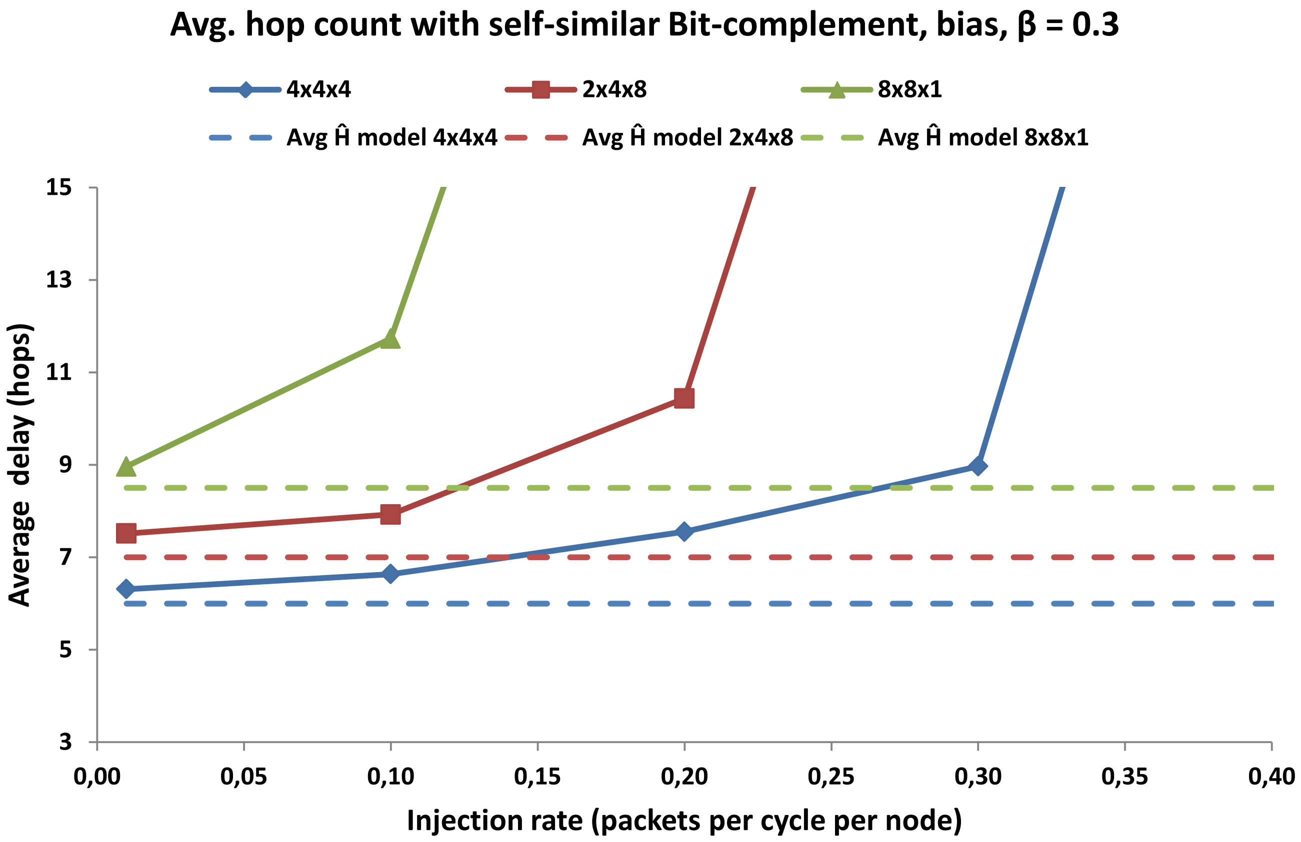


Avg. hop count with self-similar Bit-reverse, bias, β = 0.3
4x4x4
2x4x8
8x8x1
Avg Ĥ model 4x4x4
Avg Ĥ model 2x4x8
Avg Ĥ model 8x8x1
Average delay (hops)
0
2
4
6
8
10
12
14
0,00
0,10
0,20
0,30
0,40
0,50
0,60
Injection rate (packets per cycle per node)

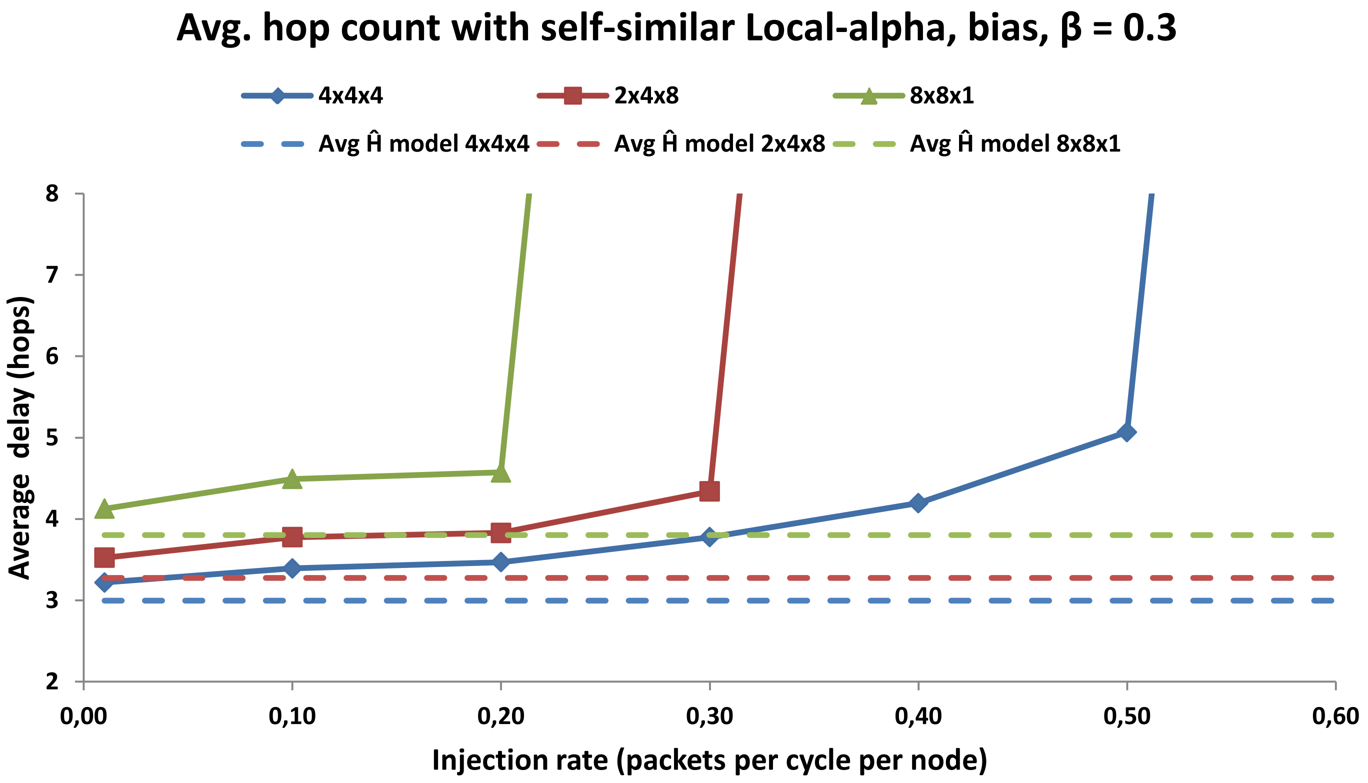
# Bit-Complement



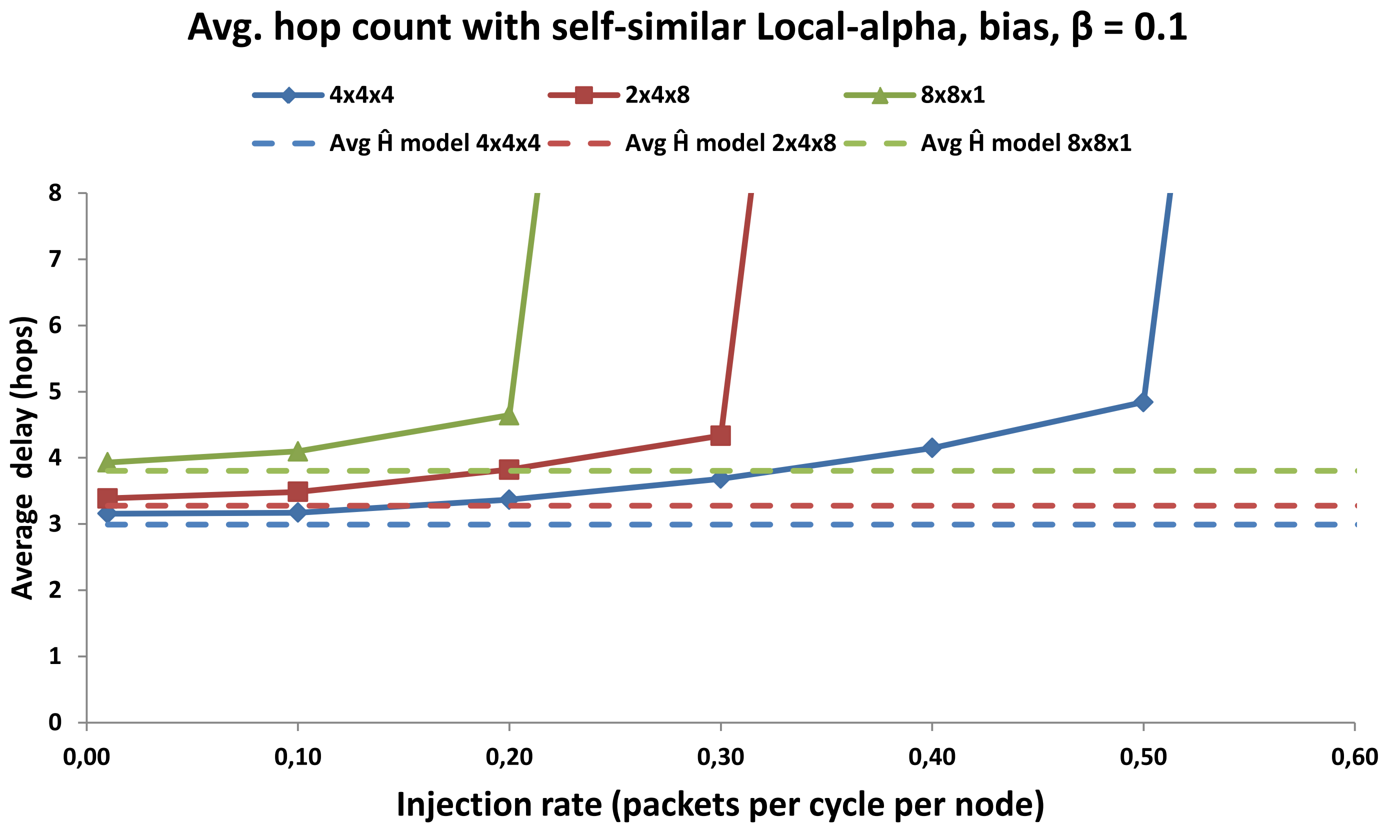
# Bit-Complement



# α-Model Localized Pattern

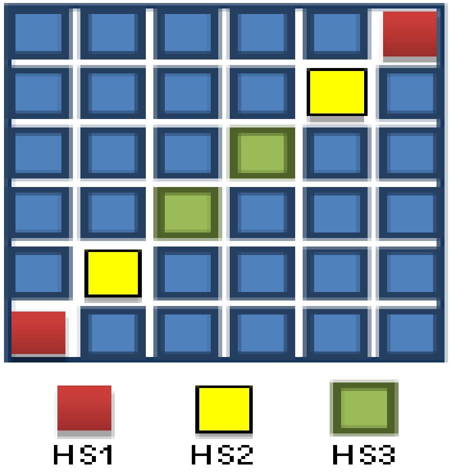


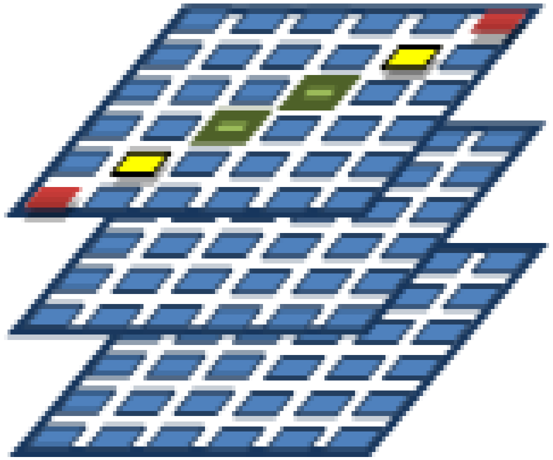
# α-Model Localized Pattern



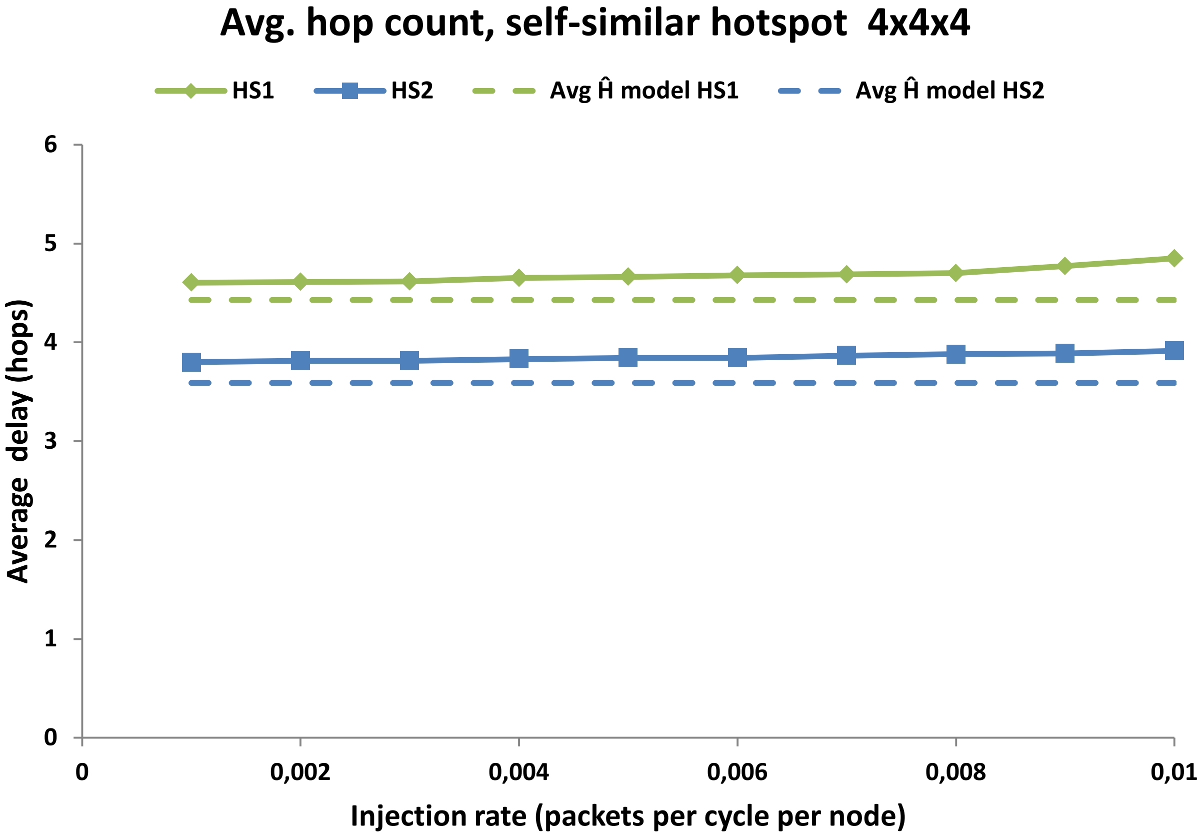
# Hotspots

➢ The placement of hotspot nodes in the network affects the overall network performance.

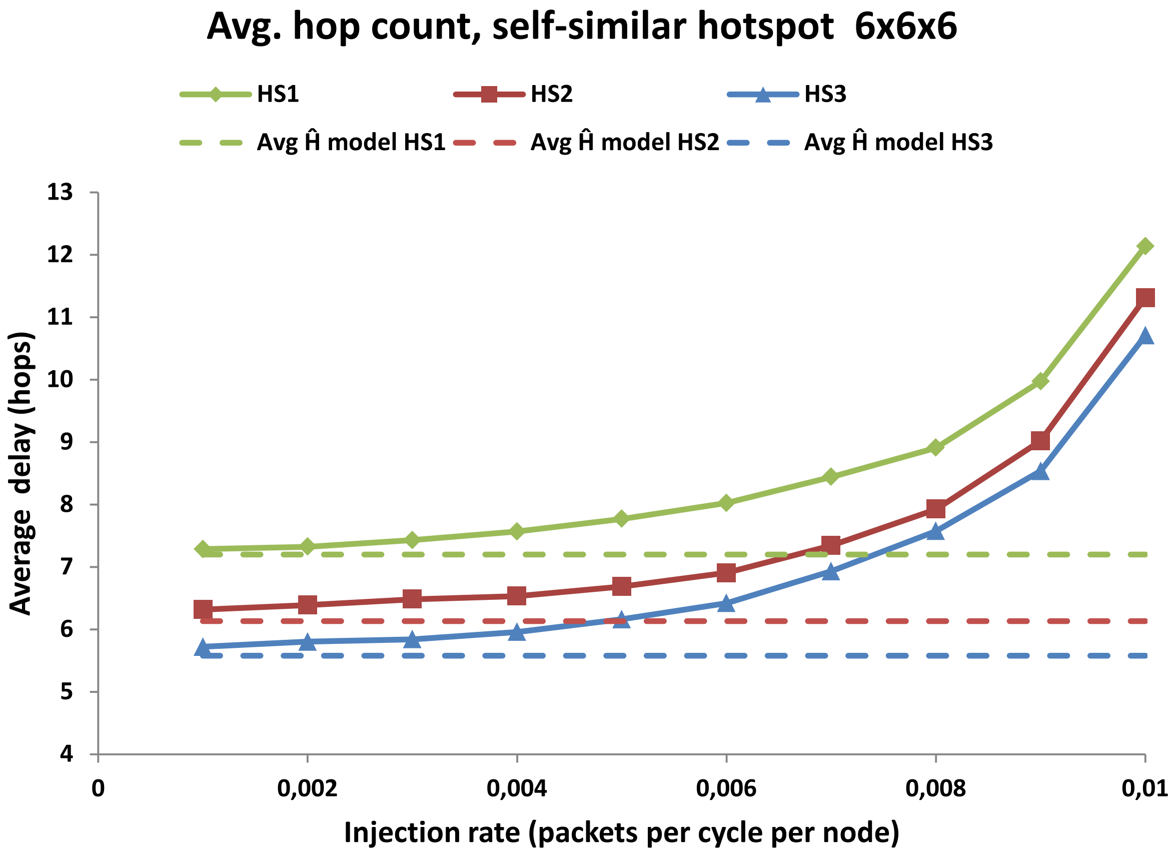


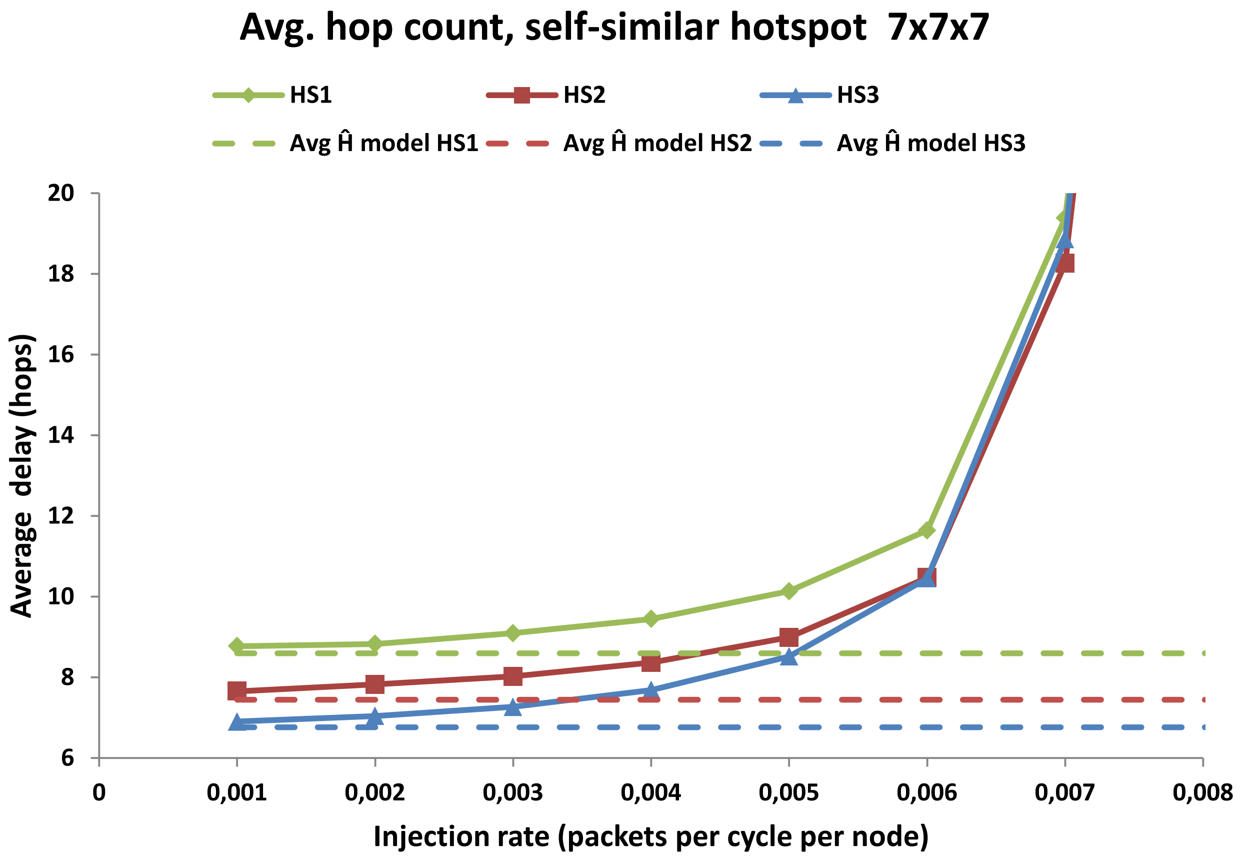
# Hotspots


Avg. hop count, self-similar hotspot 4x4x4
HS1
HS2
Avg Ĥ model HS1
Avg Ĥ model HS2
Average delay (hops)
6
5
4
3
2
1
0
0
0,002
0,004
0,006
0,008
0,01
Injection rate (packets per cycle per node)

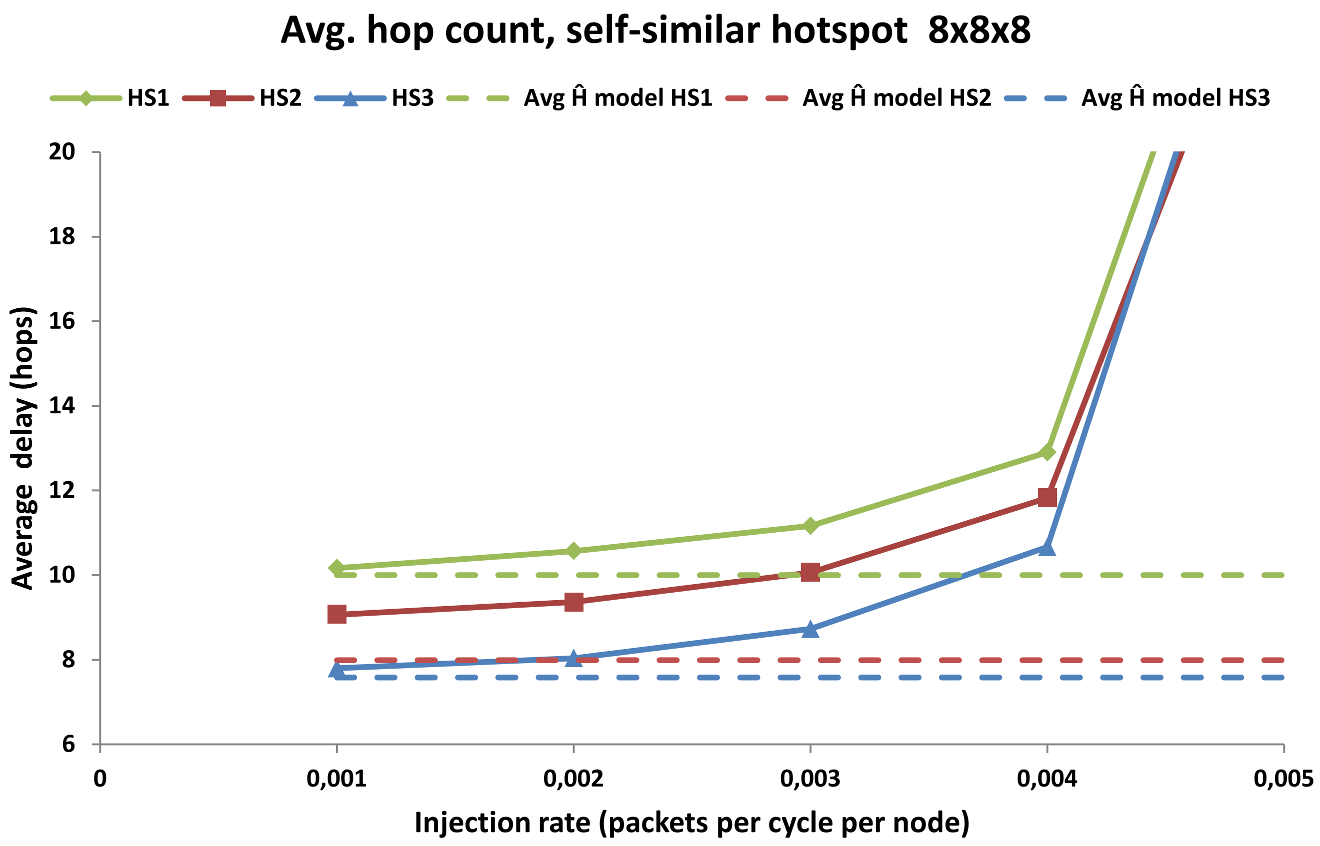
# Hotspots


Avg. hop count, self-similar hotspot 6x6x6
HS1
HS2
HS3
Avg Ĥ model HS1
Avg Ĥ model HS2
Avg Ĥ model HS3
Average delay (hops)
13
12
11
10
9
8
7
6
5
4
0
0,002
0,004
0,006
0,008
0,01
Injection rate (packets per cycle per node)

# Hotspots

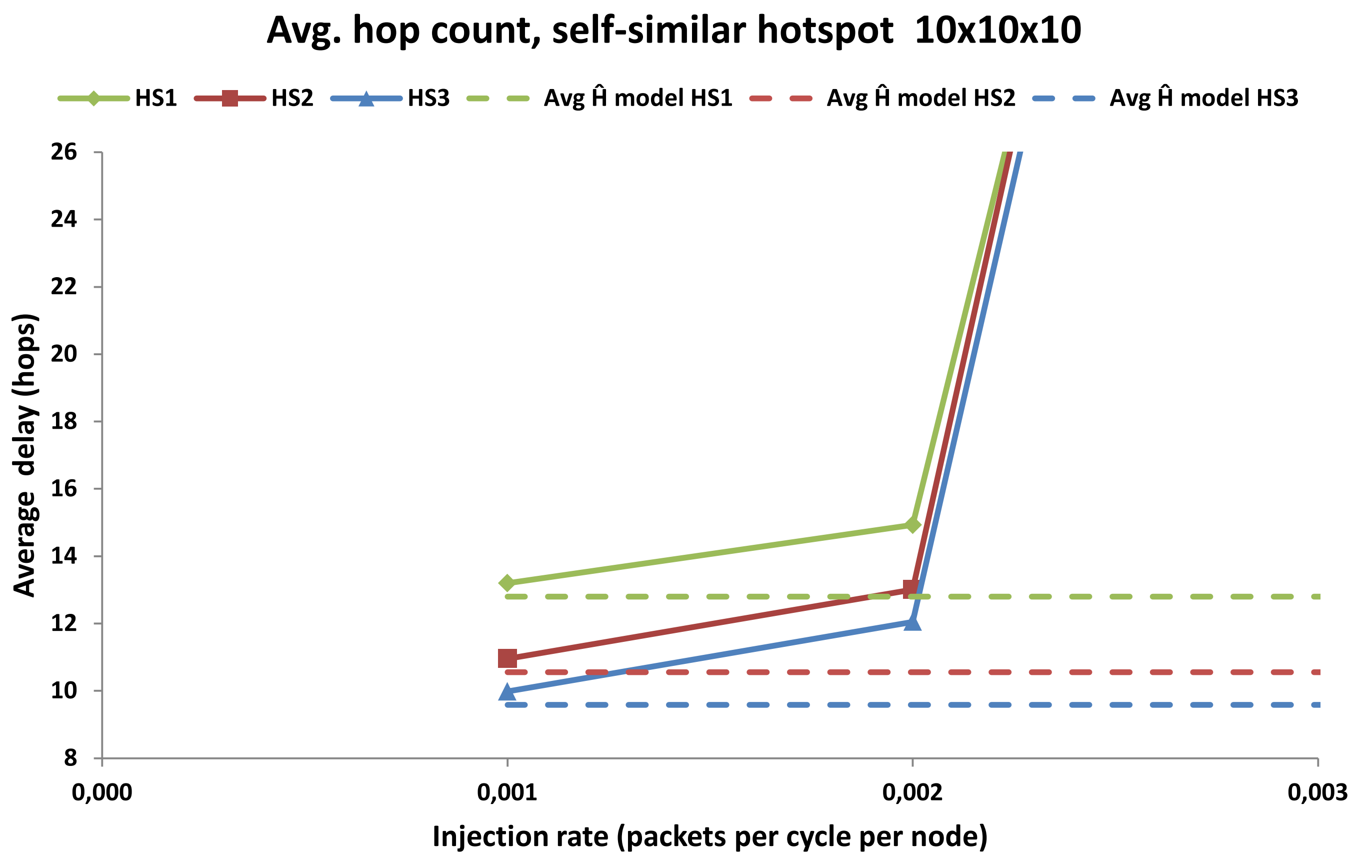

Avg. hop count, self-similar hotspot 7x7x7
HS1
HS2
HS3
Avg Ĥ model HS1
Avg Ĥ model HS2
Avg Ĥ model HS3
Average delay (hops)
20
18
16
14
12
10
8
6
0
0,001
0,002
0,003
0,004
0,005
0,006
0,007
0,008
Injection rate (packets per cycle per node)

# Hotspots



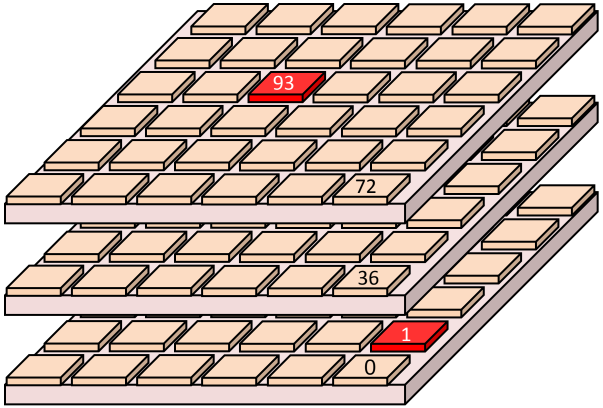
Avg. hop count, self-similar hotspot 8x8x8
HS1
HS2
HS3
Avg Ĥ model HS1
Avg Ĥ model HS2
Avg Ĥ model HS3
Average delay (hops)
20
18
16
14
12
10
8
6
0
0,001
0,002
0,003
0,004
0,005
Injection rate (packets per cycle per node)

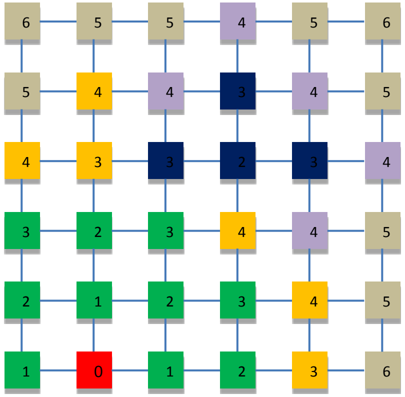
# Hotspots


Avg. hop count, self-similar hotspot 10x10x10
HS1
HS2
HS3
Avg Ĥ model HS1
Avg Ĥ model HS2
Avg Ĥ model HS3
26
24
22
20
18
16
14
12
10
8
Average delay (hops)
0,000
0,001
0,002
0,003
Injection rate (packets per cycle per node)

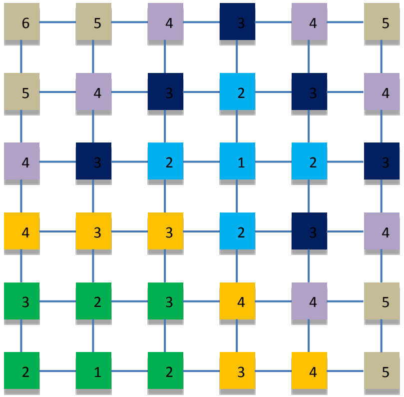
# Configuration

| Number of cores | Memory access | Off-chip access | With other cores |
|---|---|---|---|
| A=18 | 7.14% | 7.14% | 85.71% |
| B=18 | 14.29% | 14.29% | 71.43% |
| C=18 | 21.43% | 21.43% | 57.14% |
| D=18 | 28.57% | 28.57% | 42.86% |
| E=18 | 35.71% | 35.71% | 28.57% |
| F=18 | 42.86% | 42.86% | 14.29% |

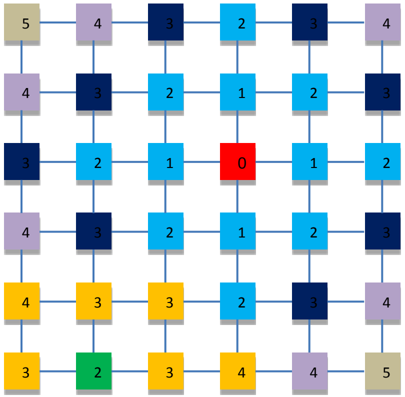



core #0




core #36

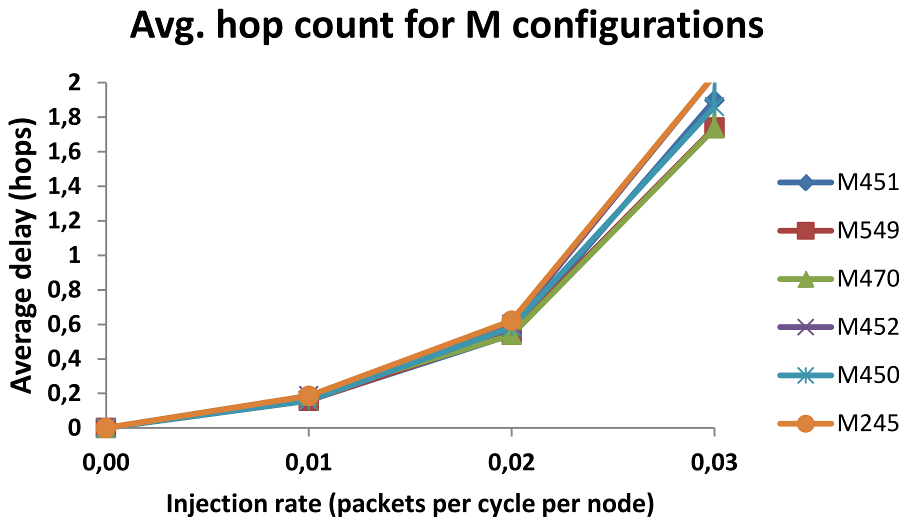



core #72

# Configuration

## ➢ Output

| Injection-r | M451 | M549 | M470 | M452 | M450 | M245 |
|---|---|---|---|---|---|---|
| 0,00 | 4,6041 | 4,6062 | 4,6085 | 4,6122 | 4,6451 | 4,888 |
| 0,01 | 4,82 | 4,85 | 4,83 | 4,84 | 4,90 | 5,13 |
| 0,02 | 5,18 | 5,24 | 5,18 | 5,29 | 5,29 | 5,56 |
| 0,03 | 6,53 | 6,42 | 6,39 | 6,71 | 6,58 | 6,96 |
| 0,04 | 21,09 | 25,25 | 28,83 | 27,51 | 28,24 | 29,01 |
| 0,05 | 118,52 | 132,25 | 131,28 | 132,17 | 135,10 | 162,46 |
| 0,06 | 306,91 | 307,70 | 325,68 | 319,67 | 330,98 | 334,17 |
| 0,07 | 448,99 | 448,91 | 449,40 | 450,26 | 456,25 | 455,13 |
| 0,08 | 537,41 | 541,05 | 543,97 | 543,24 | 548,50 | 535,00 |
| 0,09 | 603,57 | 587,39 | 607,01 | 588,38 | 602,23 | 603,87 |
| 0,10 | 627,37 | 604,70 | 623,91 | 611,32 | 605,12 | 644,24 |



# Cross-Overs

| Injection-r | M451 | M549 | M470 | M452 | M450 | M245 |
|---|---|---|---|---|---|---|
| Model | 4,6041 | 4,6062 | 4,6085 | 4,6122 | 4,6451 | 4,888 |
| c(M451) | 0,00% | -0,05% | -0,10% | -0,18% | -0,88% | -5,81% |
| c(M549) | 0,05% | 0,00% | -0,05% | -0,13% | -0,84% | -5,77% |
| c(M470) | 0,10% | 0,05% | 0,00% | -0,08% | -0,79% | -5,72% |
| c(M452) | 0,18% | 0,13% | 0,08% | 0,00% | -0,71% | -5,64% |
| c(M450) | 0,89% | 0,84% | 0,79% | 0,71% | 0,00% | -4,97% |
| c(M245) | 6,17% | 6,12% | 6,06% | 5,98% | 5,23% | 0,00% |

No Cross-over

Cross-over

Self-reference

$$\text{Difference } \% = \left| \frac{(z(M245) - z(M451))}{\text{z}(451)} \right| * 100$$

# Configuration

- Cross-overs do happen when configurations are very similar.

- In all tested cases, there was never a cross over when the minimum difference in the zero-load prediction of two configuration > ±0.13%.

# Configuration to Minimize Average Delay

- The number of cores and network size is known.
- The preferred network dimensions in X,Y,Z is determined.
- The probabilities of each core sending packets to a hotspot node is estimated.
- The number and position of hotspot nodes is set.
- The cores are equally grouped into clusters based on their probabilities range.
- The Zero-load model finds configuration with the lowest average distance

# Zero Load Predictive Model

- Static performance predictor
- Based on Network geometry and traffic pattern
- High fidelity observed in simulations
- Applications:
  - Fast and early design space exploration
  - Node placement
  - Task mapping
  - …

# Questions