Galaxies of Supercomputers and their underlying interconnect topologies hierarchies

Lukasz P. Orlowski[1], Yuefan Deng[1,2,3] and Marek T. Michalewicz[1]

[1] A*STAR Computational Resource Centre, Singapore 138632, Singapore; [2] Stony Brook University, New York 11794-3600, USA; [3] National Supercomputer Centre in Jinan, Shandong Province, P. R. China

Abstract

Galaxies of Supercomputers are defined as tightly interconnected supercomputing resources with latencies of interconnects limited by time-of-flight of the physical optical fibre infrastructure. With the continual increase of the number of peta-scale Supercomputers, and with very fast progress of interconnect technologies, such as Longbow^{TM} from Obsidan Strategics [1] and MetroX from Mellanox [2], allowing RDMA and message passing across global scale distances, as well as research networks growth to attain Terabits per second bandwidth [3], we may imagine certain classess of applications, which would run well on globally federated resources spanning Supercomputer centres across the globe. In order to build a Galaxy of Supercomputers, an optimal hierarchy of interconnect topologies, spanning from an inter-node, intra-rack, inter-rack through to a single Supercomputer centre and up to long-distance, inter-Centres, is needed.

We propose a "graph of graphs hierarchy" methodology, and a graph embedding algorithm which may be used to explore colossal-scale space of all possible interconnect topologies. These embedded graphs could be considered simple model for building Galaxy of Supercomputers connecting tens of millions of processing cores residing at the multiple locations of the globe.

Here we present just one of our optimal graphs $32k5 \otimes 32k5$ and compare it with existing topologies: K-Computer's TOFU and 5-D torus of BlueGene/Q.

Topic Areas: Architectures - Network technology; and Future Trends - Exascale HPC **Keywords**: Federated Supercomputers, Galaxy of Supercomputers, interconnects, topology, graph theory, graph of graphs hierarchy, graph embedding

Introduction

Interconnects in supercomputers are designed to optimize communication by minimizing node-to-node hop distance for message passing. Additional design criteria are the complexity of the topology and the total length of all links. Traditionally, these topologies corresponded to most frequently occuring computational stencil communications, hence prevalence of torus and hypercube topologies.

We have discovered regular graphs for up to $N = 32$ nodes with corresponding node degrees $k = 2, 3, 4, 5$. These graphs minimize the graph average node-to-node hop distances and graph diameters. We also present general formalism of graph embedding. The hierarchy of graph topologies resulting from such embeddings can form topologies of the future Galaxy of Supercomputers.The authors believe that it's only a matter of time before federated multiple resources of Supercomputer class, combined with efficient network technologies [1, 2, 3] and appropriate scientific work-flow and I/O frameworks [4] will make Galaxy of Supercomputers possible.

Graph Embedding Formalism

We represent the graph through an adjacency matrix. An adjacency matrix is a matrix that describes which vertices are connected i.e. adjacent. If *i*-th vertex in a graph is connected with *j*-th vertex then *j*-th entry in *i*-th row of the corresponding adjacency matrix is non-zero, otherwise it's zero. Here we focus exclusively on topologies, rather then the physical fiber, therefore we explore unweighted, undirected graphs with no self-loops.

Definition: Embedding a graph into another graph is a substitution of nodes of a graph by entire graphs. We introduce notation $N_1kK_1 \otimes N_2kK_2$ to mean embed a graph of N_2 nodes, each of degree K_2 into a graph on N_1 nodes, each of degree K_1. The embedding operation can be repeated (i.e. $N_1kK_1 \otimes N_2kK_2 \otimes ... \otimes N_nkK_n$) to lead to more complex graphs. Multiple embeddings are referred to as a *chain of embedding*.

Let's call the base topology graph (*i.e.* the one into which we're embedding) $\mathbf{G}_1$, with 1 meaning "the lowest level" of embedding. Let $\mathbf{G}_1$ be represented by an adjacency matrix G_1 of the following form.:

$$G_1 = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 1 & 0 & 1 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix} \tag{1}$$

We embed a graph $\mathbf{G}_2$ represented by adjacency matrix G_2 into one with adjacency matrix G_1. The resulting adjacency matrix $G_1 \otimes G_2$ will be of the following form:

$$G_1 \otimes G_2 = \begin{pmatrix} G_2 & \mathbf{1}_m & \mathbf{0}_m & \dots & \mathbf{0}_m \\ \mathbf{1}_m & G_2 & \mathbf{1}_m & \dots & \mathbf{0}_m \\ \mathbf{0}_m & \mathbf{1}_m & G_2 & \dots & \mathbf{0}_m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_m & \mathbf{0}_m & \mathbf{0}_m & \dots & G_2 \end{pmatrix} \tag{2}$$

The diagonal is populated with blocks of embedded graph G_2, $\mathbf{0}_m$ are $m \times m$ blocks of all zeros and $\mathbf{1}_m$ is defined as follows:

$$\mathbf{1}_m[i][j] = \begin{cases} 1 & \text{if i = j = 1} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Note that 1's in G_1, are substituted by $\mathbf{1}_m$'s in the block representation of $G_1 \otimes G_2$. Similarily with 0's and $\mathbf{0}_m$'s. Here m is the *order* of graph $\mathbf{G}_2$, i.e. $m = |\mathbf{G}_2|$.

We can generalise the embedding process to more than two graphs. For example, given $G_1 \otimes G_2$ we can further embed G_3:

$$G_1 \otimes G_2 \otimes G_3 = \begin{pmatrix} G_2 \otimes G_3 & \mathbf{1}_{m'} & \mathbf{0}_{m'} & \dots & \mathbf{0}_{m'} \\ \mathbf{1}_{m'} & G_2 \otimes G_3 & \mathbf{1}_{m'} & \dots & \mathbf{0}_{m'} \\ \mathbf{0}_{m'} & \mathbf{1}_{m'} & G_2 \otimes G_3 & \dots & \mathbf{0}_{m'} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{m'} & \mathbf{0}_{m'} & \mathbf{0}_{m'} & \dots & G_2 \otimes G_3 \end{pmatrix} \tag{4}$$

The size of blocks will grow from $m \times m$ to $m' \times m'$, where $m' = |\mathbf{G}_2| + |\mathbf{G}_3|$ other than that similarity to the form of G_1 is preserved. Embedding is an associative operation.

Results

We first built regular ($k = const$ for all nodes) graphs of up-to 32 nodes. Next, we studied all possible embeddings of two graphs with up-to 1024 nodes. Of course, this is a very small number of nodes, compared to the real supercomputers' scales, but it illustrates the principle and gives clear guidance to further extensions of this work. Below we compare the topological properties of our embedded graph with those of the 5D torus in IBM Blue Gene/Q [6] and the TOFU in the Fujitsu K-Computer [7].

Name	Number of nodes	Number of links	Diameter	Mean path distance
5-D Torus ($4 \times 4 \times 4 \times 4 \times 4$)	1024	5120	10	5.00
TOFU ($3 \times 4 \times 8 \times 2 \times 3 \times 2$)	1152	5760	10	5.38
32k5 $\otimes$ 32k5	1024	2640	9	6.31

Table 1: Topologies of interconnect networks and their properties

Our *Embedded Graph 32k5 $\otimes$ 32k5* has the smallest diameter and might be a good candidate for a cost effective interconnect topology, since it has only 50% links of comparable 5-D torus and TOFU topologies. The number of links of the graph is directly translated to the length of copper/optical fibre "wires" necessary to interconnect the supercomputer. Hence, minimizing this graph property simplifies the graph complexity and lowers the real cost of the interconnect. The graph diameter and mean path length are very important for irregular communication problems and especially for the *Big Data* problems. Clearly, our *Embedded Graph 32k5 $\otimes$ 32k5* scores reasonably well on all three criteria, and is better in two of them.

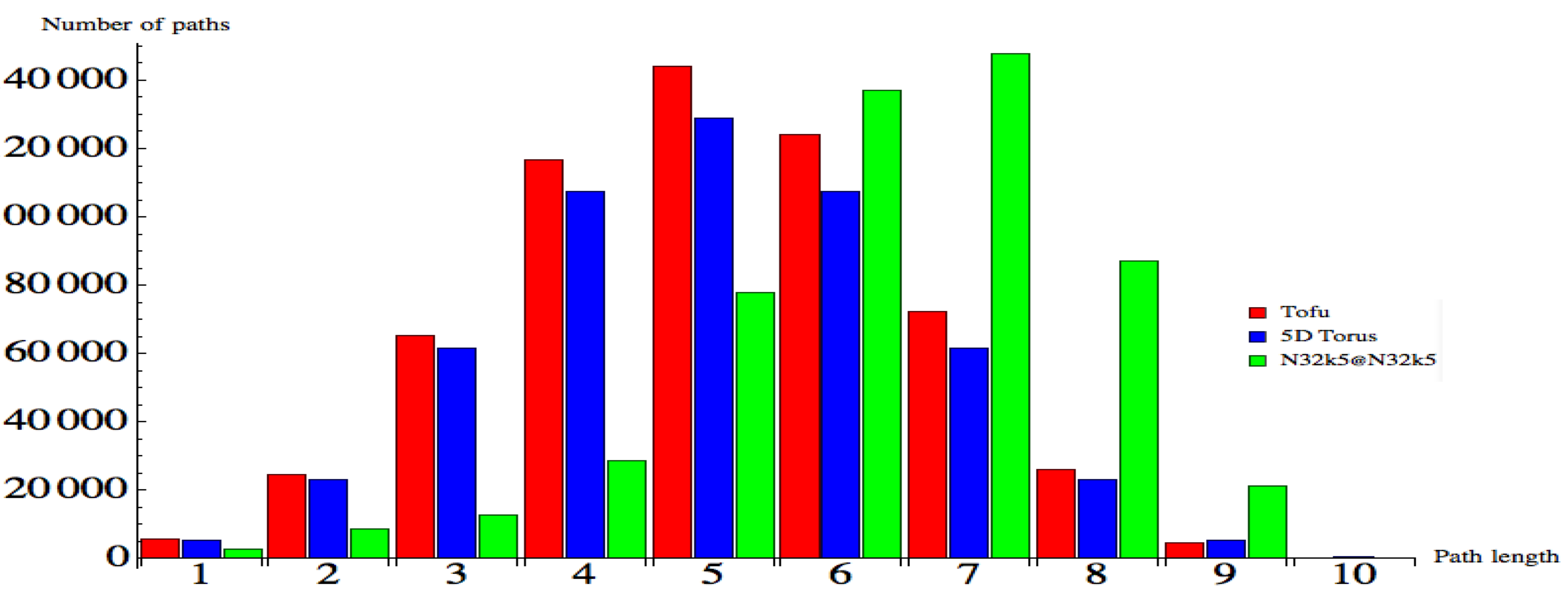

Figure 1: Distribution of path lengths for three top graphs from Table 1.

Figure 1 above shows that both TOFU of K-computer and 5-D torus of BlueGene/Q are very well designed topologies. It also explains why BlueGene/Q computers score well in the Graph500 contest [5]. 5-D torus has greater *over-provisioning* of links and greater fail-proof resiliency, but it results in greater complexity and cost. The TOFU network is similarly *over-provisioned.*

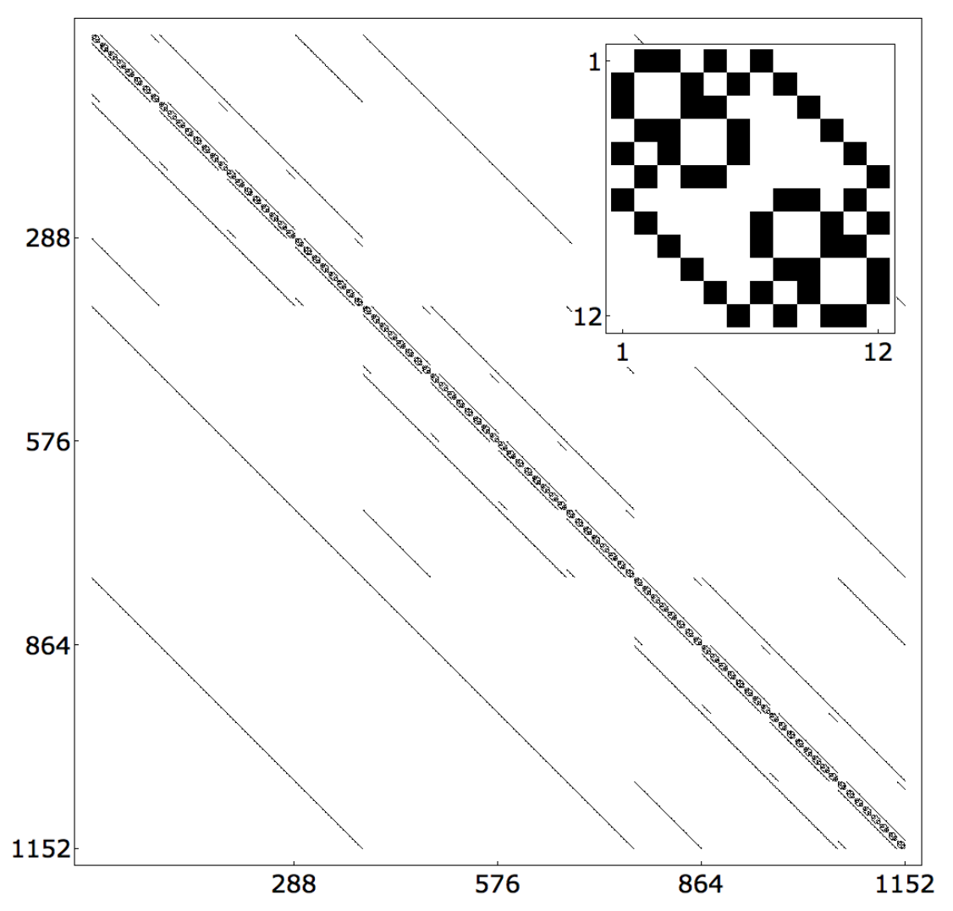

Figure 2: Adjacency matrices of TOFU ...

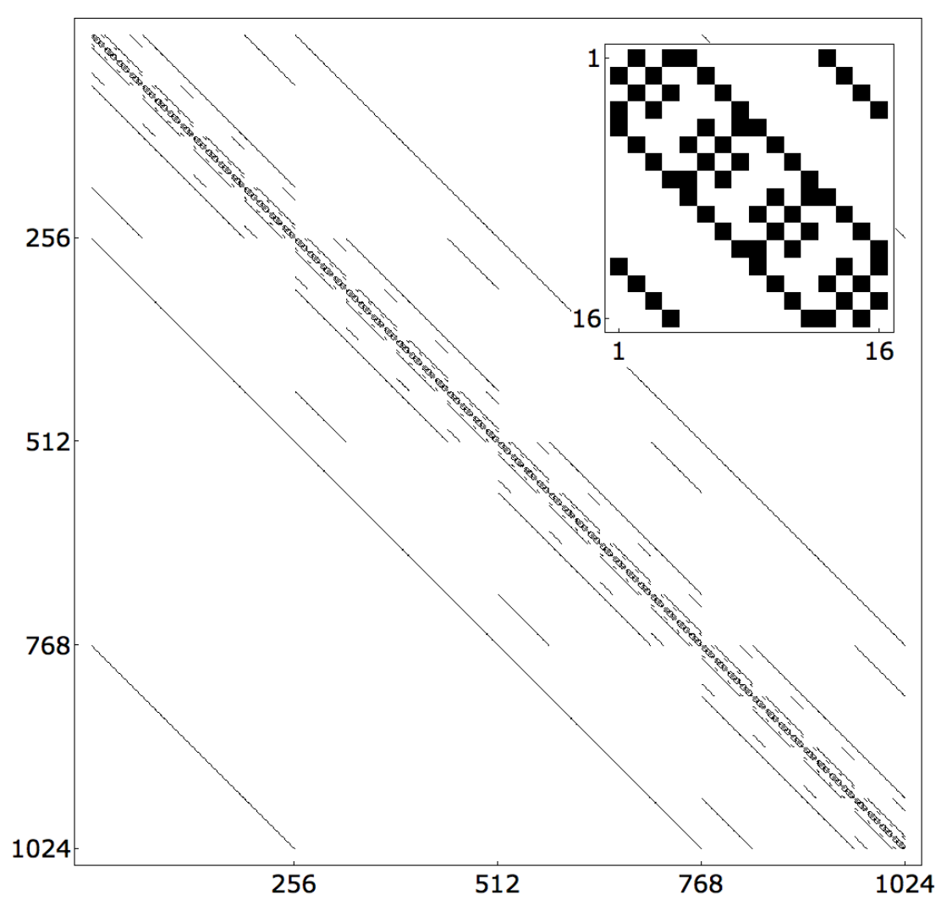

Figure 3: 5D torus

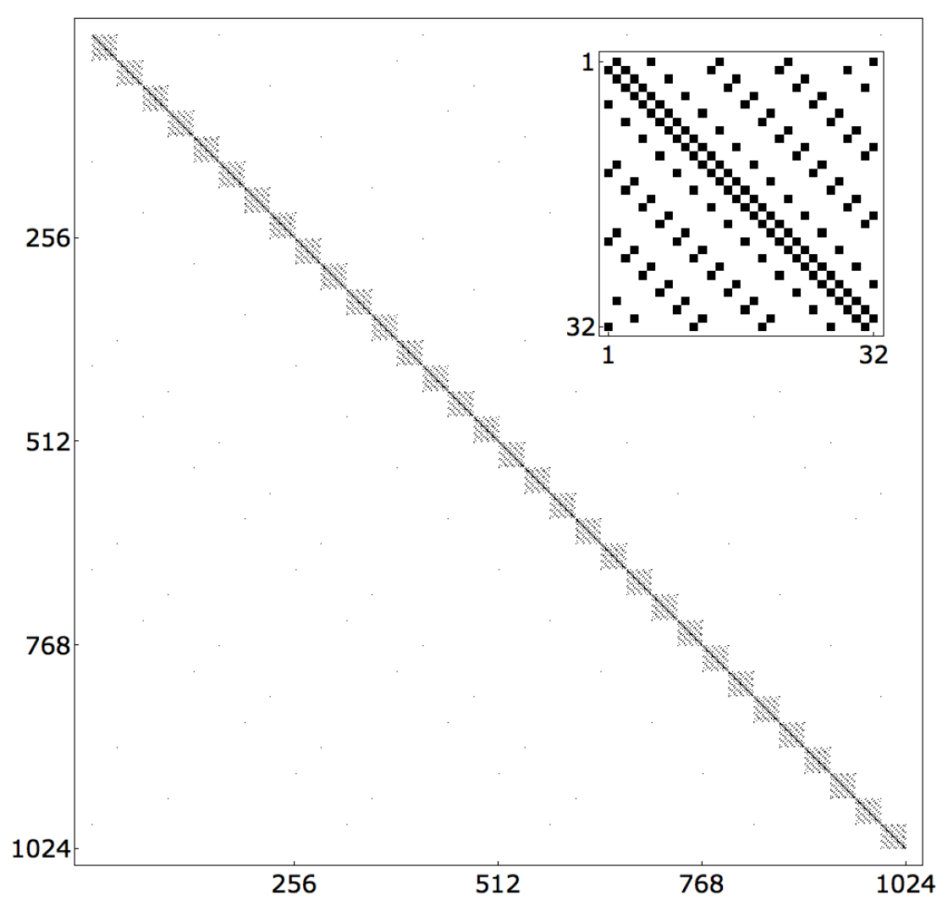

Figure 4: and 32k5 $\otimes$ 32k5

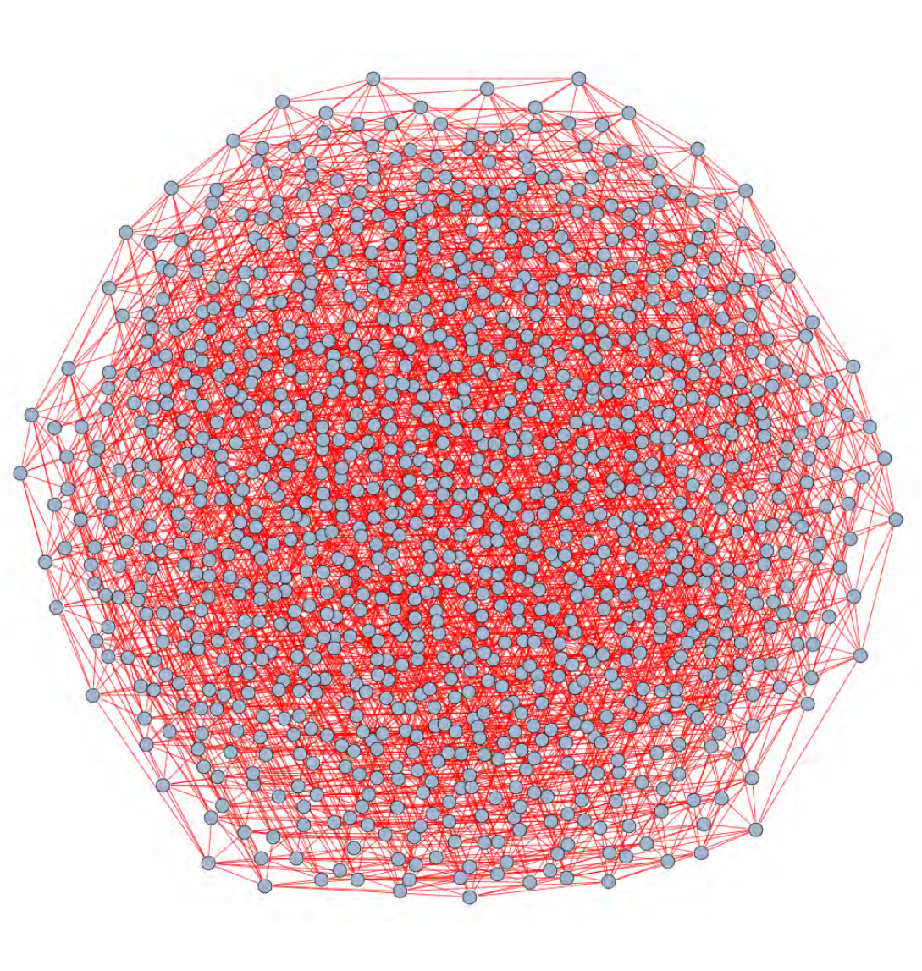

Figure 5: Graph of TOFU

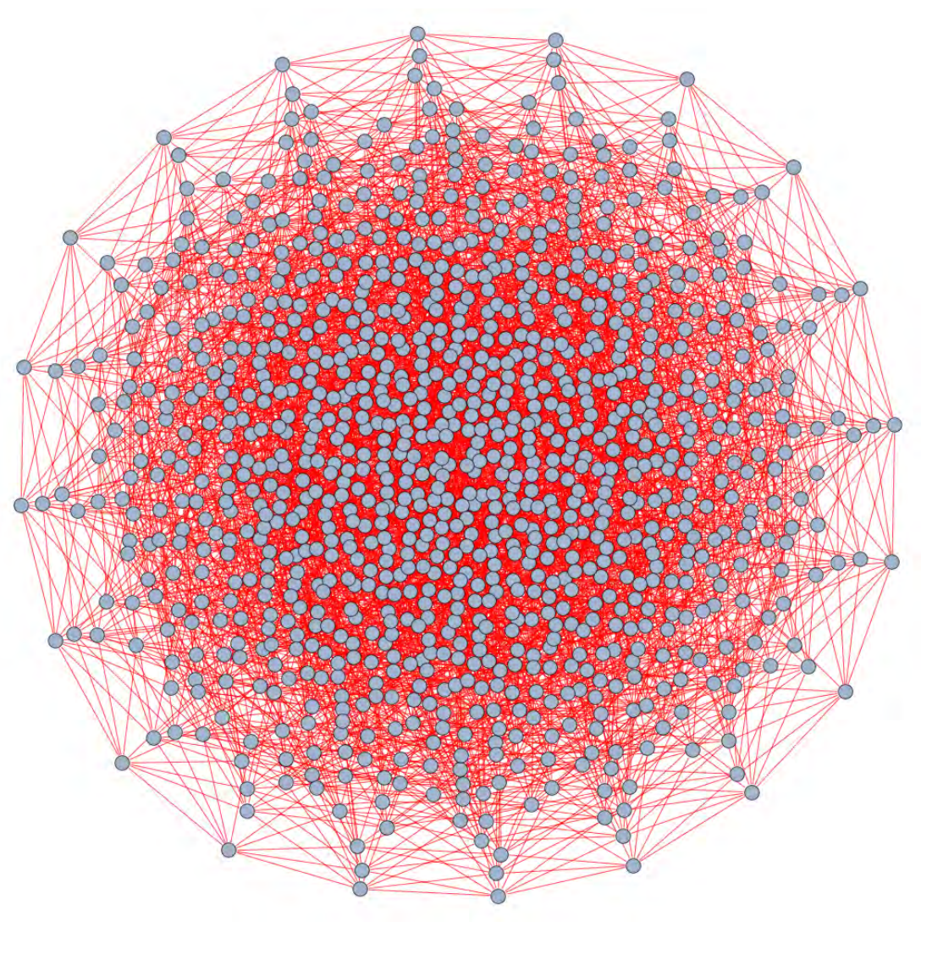

Figure 6: Graph of 5D torus

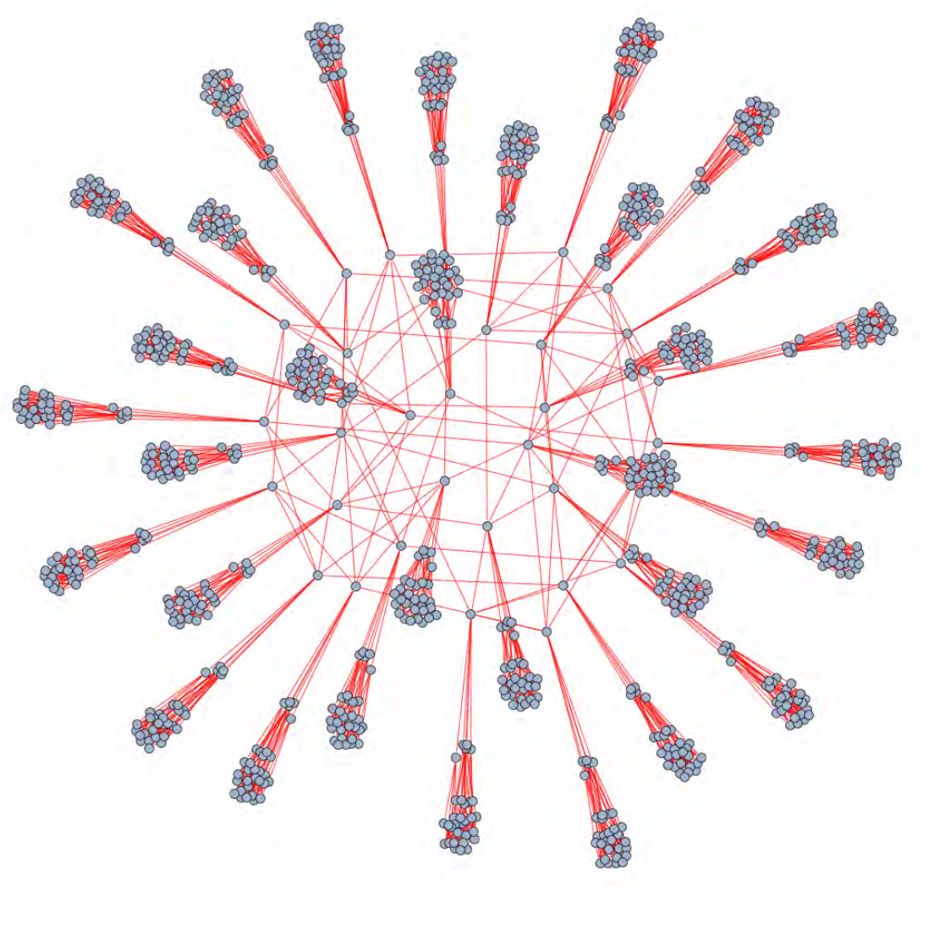

Figure 7: Graph of 32k5 $\otimes$ 32k5

The Figures 2,3,4 above depict non-zero 1 link elements ($zeros$ are empty spaces) in adjacency matrices of the three interconnect topologies analysed above. Clearly seen in case of TOFU and 5D torus are side-bands of 1's resulting from **I** identity matrices blocks running parallel to the main diagonal. Our *Embedded Topology 32k5 $\otimes$ 32k5* graph's Adjacency Matrix (Figure 4) has only the sparsely dotted side-band entries along the main diagonal, corresponding to $\mathbf{1}_m[i][j]$ block matrices.

In Figures 5,6,7 we show the full graph connectivity of the three interconnect topologies studied here. The clustering of the *sub-graphs* is clearly seen.

Conclusions

This study, together with previous works [8, 9, 10], constitutes the extension of the traditional practice of designing supercomputer interconnects by graph theory analysis and computation [11]. The key methodologies for our study include discovery of the optimal graphs in terms of graph diameter and average path length by exhaustive search and embedding these optimal graphs to form hierarchies of embedded graphs. Our new approach results in optimal embedded graphs. Our proposed graph construction and related graph algebra is a promising method for producing interesting network topologies for interconnecting separate supercomputers into a *Galaxy of Supercomputers* configuration, as well as for the platforms for *Big Data* and graph analytics tasks.

References

[1] `http://www.obsidianresearch.com/products/longbow/index.html`
[2] `http://www.mellanox.com/page/products_dyn?product_family=155&mtag=tx6200`
[3] `http://www.ieee802.org/3/400GSG/`
[4] `https://www.olcf.ornl.gov/center-projects/adios/`
[5] `http://www.graph500.org/results_nov_2013`
[6] D. Chen, et al. The IBM Blue Gene Q Interconnection Network and Message Unit. Proceedings of 2011 SC - International Conference for High Performance Computing, Networking, Storage and Analysis, New York, NY: ACM, 2011.
[7] Y. Ajima, et al. The TOFU Interconnect. Proceedings of the 2011 IEEE 19th Annual Symposium on High Performance Interconnects, 2057600: IEEE Computer Society, 2011, pp. 87-94.
[8] P. Zhang, R. Powell, and Y. Deng. Interlacing Bypass Rings to Torus Networks for More Efficient Networks. IEEE Transactions on Parallel and Distributed Systems. 2011, 22(2) pp. 287-295
[9] R. Feng, P. Zhang, and Y. Deng. Network Design Considerations for Exascale Supercomputers. Proceedings of the 24th IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS 2012), pp. 86-93.
[10] R. Feng, P. Zhang, and Y. Deng. Simulated Performance Evaluation of a 6D Mesh-iBT Interconnect. 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel and Distributed Computing (SNPD 2012), pp. 253-259.
[11] J. M. McQuillian, Graph theory applied to optimal connectivity in computer networks. ACM SIGCOMM Computer Communication Review. Volume 7 Issue 2, 1977, pp. 13–41.

Acknowledgements

The original input graph for 32k5 used to generate our embedded graph was produced by T. Mayer, W. Wissner, and Z. Ye who were interns with YD at the National Supercomputer Centre in Jinan.