Down to Earth – How to Visualize Traffic on High-dimensional Torus Networks

Lucas Theisen*, Aamer Shah*†, and Felix Wolf*†

*RWTH Aachen University, Department of Computer Science, Aachen, Germany
lucas.theisen@rwth-aachen.de
† German Research School for Simulation Sciences, Laboratory for Parallel Programming, Aachen, Germany
{a.shah, f.wolf}@grs-sim.de

***Abstract*—High-dimensional torus networks are becoming common in flagship HPC systems, with five of the top ten systems in June 2014 having networks with more than three dimensions. Although such networks combine performance with scalability at reasonable cost, the challenge of how to achieve optimal performance remains. Tools are needed to help understand how well the traffic is distributed among the many dimensions. This involves not only capturing network traffic but also its comprehensible visualization. However, visualizing such networks requires projecting multiple dimensions onto a two-dimensional screen, which is naturally challenging. To tackle this problem, in this position paper, we propose a visualization technique which can display traffic on torus networks with up to six dimensions. Our fundamental approach is to simultaneously present multiple views of the same network section, with each view visualizing different dimensions. Furthermore, we leverage the multiple-coordinate system concept and combine it with a customized polygon view to provide both a global and a zoomed-in perspective of the network. By interactively linking all the views, our technique makes it possible to analyze how the communication pattern of an application is mapped onto a network.**

I. Introduction

High performance requirements of flagship HPC system are pushing them to compliment commodity hardware with specialized components and architectures. Centralized parallel file systems, high-bandwidth message passing networks and even integrated cooling systems have become essential parts of new systems. One such development is the usage of distributed switched communication networks in the form of a loop-around torus. Compute nodes of such systems are directly connected to their neighbors, without the centralized switches of fat-tree networks. The result is a low-cost, highly-scalable communication network with high nearest-neighbor bandwidth [1]. This is the reason why such networks have become common in flagship systems from Cray, IBM, and Fujitsu.

A torus network is characterized by its dimensionality, i.e; the number of dimensions along which a node is connected to its neighboring nodes. Early versions of the torus network typically had three dimensions. The demands for better performance has increased the dimensionality even further in flagship systems, with IBM's BlueGene/Q systems employing a five-dimensional network and Fujitsu's K computer having a six-dimensional network. The performance of the network has increased as a result, but utilizing the network to its full potential still remains challenging. Application processes have to be properly mapped to compute nodes to efficiently use the available network resources. The high dimensionality not only makes it difficult to find a suitable mapping but also to analyze how the network was utilized by the application.

Analysis of network utilization requires capturing communication traffic as well as visualizing it to provide a feedback to the user. A challenge that arises then is projecting the multi-dimensional network onto a two-dimensional display in an intelligible way for a useful feedback. This naturally becomes harder as the number of dimensions increase. For example, a previous attempt to visualize network traffic on torus networks is applicable only to three dimensions [2]. To tackle this problem, in this position paper, we propose a technique to visualize torus networks with up to six dimensions. We also present an example visualization of an artificially generated application data using a prototype of our VisTorus tool.

The basic principle of our technique is to provide multiple views of the network concurrently to the user, with each view projecting and displaying traffic only along a few dimensions. We use multi-coordinate axes, such as used in multi-pixel bar graphs [3], to visualize network planes across axes. Dimensions are also collapsed to effectively summarize network traffic on the global scale. While the polygon view provides the ability to zoom in on traffic to-and-from individual nodes. By linking all the views interactively, the technique allows a user to look at the global trend in network traffic and to zoom-in to areas of interest.

The rest of the paper is organized as follows. Section II covers the state of the art in visualizing multi-dimensional data sets. In Section III, we outline how we extended the performance monitoring tool LWM^2 [4] to capture network traffic. We then describe our approach in Section IV, introducing two different visualization techniques. In Section V, we present an example visualization of an artificially generated application data using VisTorus, a prototypical implementation of our visualization technique. We conclude the paper with a summary in Section VI and provide directions for future work in Section VII.

II. Related Work

Multi-dimensional data can have different structural specifications and can be visualized accordingly. For example, an independent parallel coordinate system [5] can be used when the individual elements of a tuple are independent of each other. Similarly, a multi-pixel bar graph [3] can be used to visualize a five tuple data set, when there is no structural relationship between the tuples. There are a number of general graph visualization techniques surveyed in [6]. However, these techniques focus on revealing the structure of the graphs, which in our case is already known. Radial graphs with edge bundling [7] can also be used to visualize networks, but with a loss of structural relationship between individual nodes.

In the HPC context, Schulz et al. [8] propose a holistic approach to visualization, combining metrics from three different domains. However, they do not address the problem of displaying multi-dimensional networks in the hardware domain. CUBE4 [9] visualizes a multi-dimensional torus as a set of cubes, but is limited to node-level metrics and cannot show link-level metrics. Another tool, called Boxfish [2], can visualize traffic on individual links of a torus networks but is restricted to three dimensions. We build upon these concepts to visualize torus networks with up to six dimensions. Similar to the dual view technique [10], we utilize multiple simultaneous views, but each view projects the same section of the structured network in a different way. Furthermore, we utilize zooming and panning [11], but not under a single view. We form a hierarchy of views with each step being a zoomed-in version of the previous one, projecting the network using different techniques simultaneously in each step.

III. Network monitoring

Visualizing network traffic on a torus network requires monitoring each network link used by the application. For this purpose, we extended LWM^2 [4], a lightweight monitoring module, and added the capability of capturing the number of packets sent on each link used by an application on a BlueGene/Q systems. To capture the network traffic, we used the BlueGene/Q Performance Monitoring API (BGPM) [12].

BGPM allows all the eleven links attached to a compute node to be monitored. Ten of these links represent network communication in both the positive and negative direction of the five dimensional torus network. The eleventh link represent the I/O traffic between the compute node and the I/O node. For each of these links, it is possible to separately capture the number of 32 bytes packets sent based on several criteria, such as statically routed or dynamically routed, user generated or system generated, point-to-point or collective, etc.. Similarly, for receiving information, the average queue length and the packet arrival rate can be captured to indicate the level of congestion on each network link. Combining all these, we separately capture the following metrics for each of the eleven links attached to a compute node.

- Number of 32 bytes user point-to-point packets statically routed.
- Number of 32 bytes user point-to-point packets dynamically routed.
- Number of 32 bytes user global communicator packets.
- Number of 32 bytes user subcommunicator packets.
- The average packet arrival rate.
- The average length of the packet arrival queue.

We capture the network traffic in detail this way. However, to demonstrate the visualization, we currently use only an aggregated derived metric, representing the total number of packets along the ten torus network links. Later, we plan to make the full spectrum of metrics available.

IV. Approach

Visualizing multi-dimensional torus network on a two dimensional screen is naturally challenging, especially when the number of dimensions increase beyond three. The difficulty arises when trying to transform the network structure for projection onto a two dimensional display in an intelligible way. Boxfish achieves this by projecting a view of the three dimensional torus network from the top, displaying a plane of concentric rectangles. This technique is, however, limited to three dimensions.

We propose to visualize torus networks beyond three dimensions by providing simultaneous views to the user. Each view displays traffic along a few dimensions of the same network section, and at the same time projects the network in a different way. The views range from detailed (showing individual links) to summarized (averaging traffic between whole planes into a single link) projections. All the views are interactively linked among each other, providing the ability to look at the complete network occupied by the application or to magnify a particular section. We provide a global perspective by using a multi-axes plane, an intermediate perspective by collapsing some dimensions, and a zoomed-in perspective by using our polygon view. We combine these views in different configurations to provide two complete approaches to network traffic visualization.

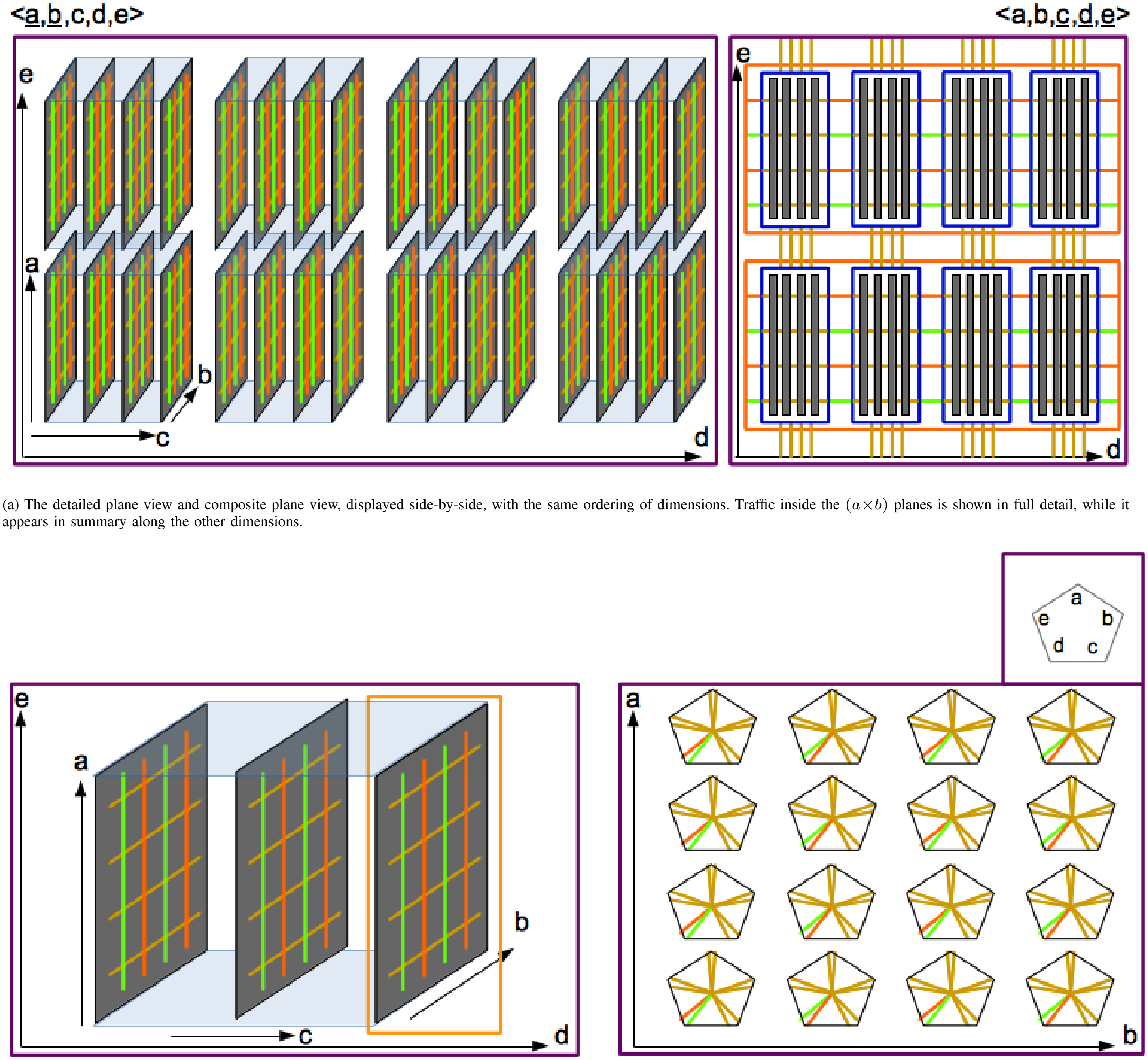
A. Global plane approach

The global plane approach uses two multi-axes views to display all the planes along selected dimensions in a torus network. A multi-axes view has multiple dimensions hierarchically represented along a single horizontal or vertical axis. The multiple axes are ordered such that fully traversing the "inner" dimension changes the outer dimension by one. The concept is similar to the memory representation of multidimensional arrays. Regardless of the number of dimensions, they are represented linearly in memory with a hierarchical ordering of the dimensions.

To describe the global plane approach, we use an example of a five-dimensional torus network, as shown in Figure 1.

1) Detailed plane view: The detailed plane view of the network is shown on the left side of Figure 1a. The view shows all the $(a \times b)$ planes in form of a grid and uses a colormap to show the traffic intensities of each link. The planes are ordered horizontally, first along the c dimension and then along the d dimension. Vertical ordering of the planes is along the e dimension.

2) Composite plane view: The ordering of the planes becomes important when looking at the global plane view in conjunction with the composite plane view, as shown in the right side of Figure 1a. This view shows the same $(a \times b)$ planes in the same order, but without their internal network traffic. The blue rectangles group the planes along the c dimension. The lines inside the blue rectangles show the traffic between the planes, summarized along the horizontal lines inside the



(a) The detailed plane view and composite plane view, displayed side-by-side, with the same ordering of dimensions. Traffic inside the $(a \times b)$ planes is shown in full detail, while it appears in summary along the other dimensions.

(b) Left: A zoomed-in view of the $(a \times b)$ planes. Right: The polygon view shows individual nodes in the selected plane, displaying the traffic to and from the nodes along each torus network link.

Fig. 1: Global plane approach allows viewing the traffic from a global scope down to individual nodes scope.

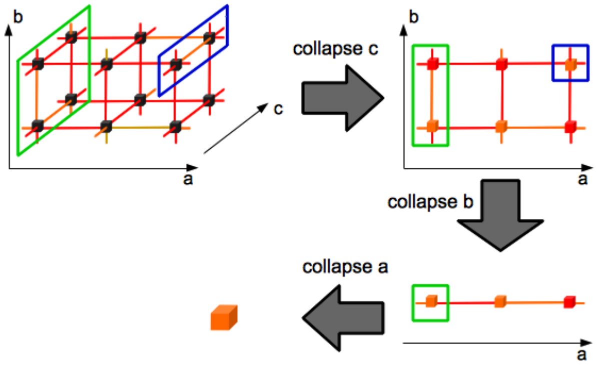


Fig. 2: Collapsing of dimensions of a cuboid.

plane. This means that a link at position n represents the average traffic between nodes of adjacent $(a \times b)$ planes with $a = n$. The loop around links of the torus network are shown at the edges of the boundary planes.

The horizontal lines between the blue rectangles show the traffic between $(a \times b)$ planes along the d dimension. Here, one line represents the summarized traffic between two complete planes. Similarly, the vertical lines between the blue rectangles show the traffic between $(a \times b)$ planes along the e dimension.

For six dimensions, multiple axes can also be used for the vertical axis, similar to the horizontal axis. Following this scheme, a torus network with up to six dimensions can be visualized, at various levels of detail. Giving the user the ability to select which dimensions are represented in the detailed plane view and in the composite plane view, it is possible to quickly explore the general trend of the network traffic.

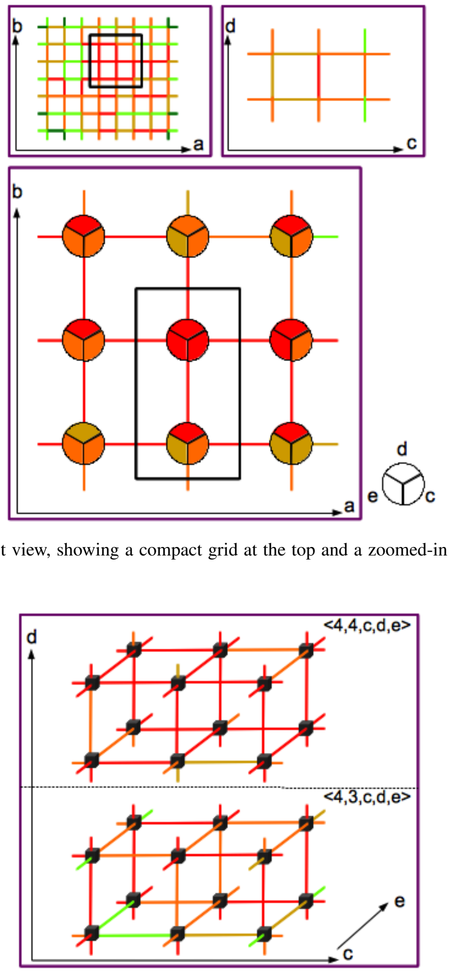
Following this general scheme, a network with more than 6 dimensions can also be visualized. However, we currently restrict ourselves to 6 dimensions.

3) Polygon view: The global plane approach also allows the user to select a group of planes for closer inspection, as shown in Figure 1b. After zooming in, the internal traffic inside a plane is accentuated. A further selection in the magnified planes leads to the polygon view, as shown on the right side of Figure 1b. Each polygon here represents a single node. A pair of lines originating from the center of the polygon represents the traffic, to and from the node, along both the positive and negative directions of a dimension. A legend at the top of the view shows the dimension represented by each corner of the polygon. The polygons themselves are ordered according to the ordering of nodes in the magnified plane, which in this example is along the a and b dimensions. As a result of the same ordering, a user is able to maintain the context of the data while zooming in to node-level details.

B. Compact dimension approach

The compact dimension approach uses the concept of collapsing a dimension into a node to visualize network traffic.

1) Collapsing dimensions: Figure 2 shows an example of a step-wise collapsing a $(a \times b \times c)$ cuboid. In the first step, the c dimension is collapsed, resulting in a single, condensed $(a \times b)$ plane. The collapsing not only removes the c dimension but also condenses all the $(a \times b)$ planes into a single plane. In the condensed plane, the traffic along the c dimension of the original cuboid is summarized in the resulting nodes, while the traffic along the a and b dimensions of the cuboid is now summarized in the respective edges of the plane. The condensed plane can further be collapsed along the b dimension, resulting in a single line along the a dimension. Now a node summarizes the traffic along both the b and c dimensions, while the respective edges represent the summarized traffic along the a dimension. In the final step, the a dimension is collapsed, resulting in a single node representing the traffic of the whole cuboid.

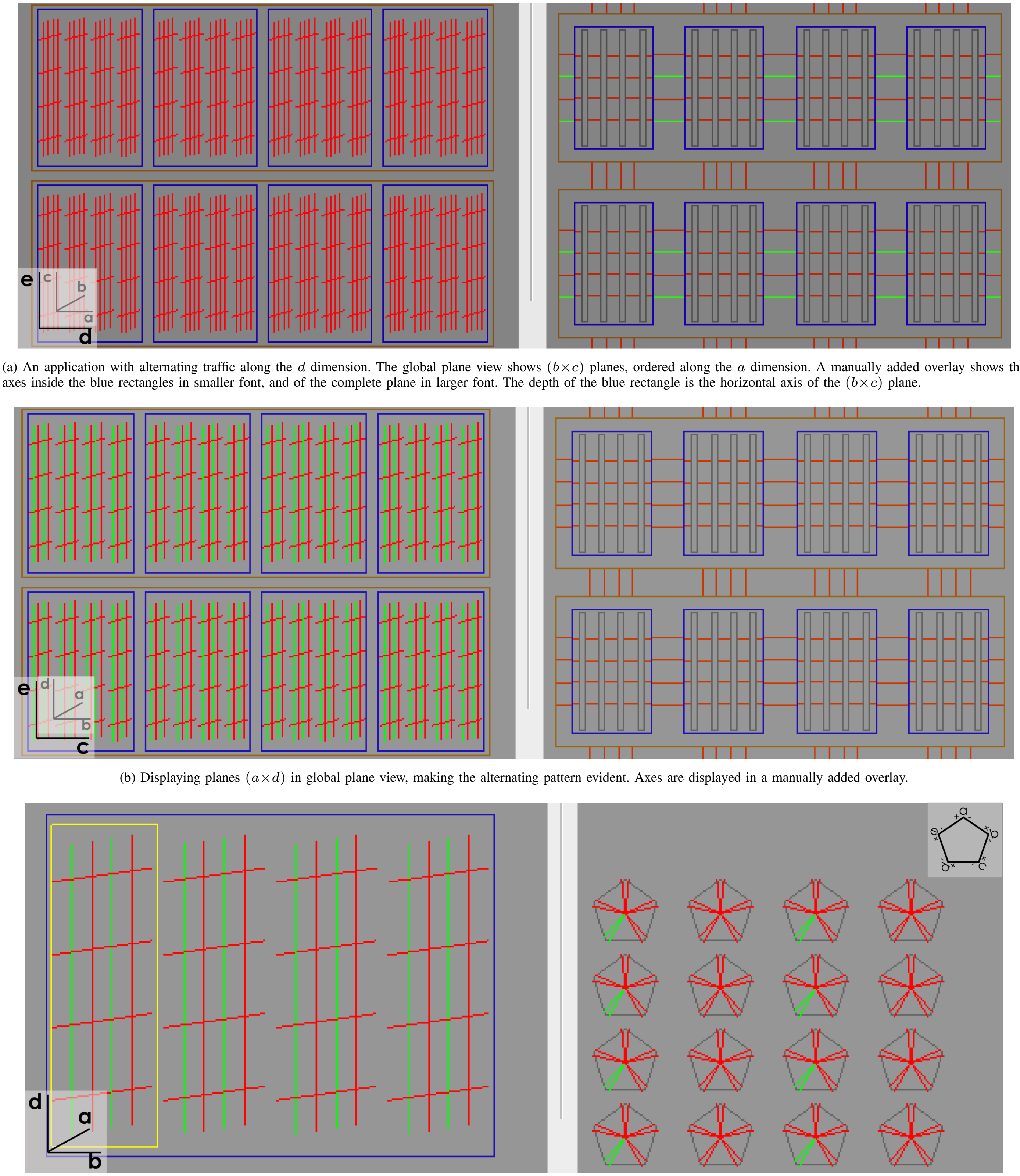


(a) The compact view, showing a compact grid at the top and a zoomed-in section at the bottom.

(b) A cuboid formed from the collapsed dimensions summarized inside a node.

Fig. 3: The compact dimension approach, with a compact grid and a zoomed-in view for global navigation, and a three-dimensional cube for closer inspection.

2) Collapsed view: The collapsed view for a five-dimensional torus is shown in Figure 3a. The top most navigational widget of the view consist of three grids, each showing only two dimensions. The remaining dimensions in the grid are collapsed. For brevity, Figure 3a shows only two grids. A selection in any of the grids results in a zoomed-in version, where the individual nodes are also visible. As the traffic of the collapsed dimensions is summarized in the node, a segmented circle is used to represent the average traffic in each collapsed dimension. In Figure 3a, three dimensions are collapsed and are represented by the three segments of a node.



(a) An application with alternating traffic along the d dimension. The global plane view shows $(b{\times}c)$ planes, ordered along the a dimension. A manually added overlay shows the axes inside the blue rectangles in smaller font, and of the complete plane in larger font. The depth of the blue rectangle is the horizontal axis of the $(b{\times}c)$ plane.

(b) Displaying planes $(a{\times}d)$ in global plane view, making the alternating pattern evident. Axes are displayed in a manually added overlay.

(c) Zoomed-in $(a{\times}d)$ planes and the individual nodes of a selected plane shown in polygon view. The nodes are ordered along the a and d dimensions. The lines show traffic along positive and negative axes respectively, starting with dimension a with a vertical line up north and then changing when moving clock-wise. Manually added overlays show the axes of both the views.

Fig. 4: An artificially generated example application, occupying a $(4{\times}4{\times}4{\times}4{\times}2)$ section of a $(a{\times}b{\times}c{\times}d{\times}e)$ torus network. The traffic along the d dimension alternates for even- and odd-ranked nodes.

A user can inspect the traffic along the collapsed dimensions by selecting any of the nodes. In the example, selecting two nodes reveals the corresponding three-dimensional cuboids, as shown in Figure 3b. The cuboids visualize the network traffic along the collapsed dimensions.

Following this approach, a three step visualization workflow is established, starting from a summary of the traffic in the navigational grids to detailed three dimensional visualization in the cuboids. The simultaneously displayed zoomed-in collapsed view and the three-dimensional cuboids compliment each other by visualizing traffic in up to five dimensions at the same time. In case of six dimensions, the zoomed-in view can also be represented by a cube with a configurable depth along the sixth dimension. This way all the six dimensions can be properly visualized.

V. Example

We have implemented our visualization technique at a prototypical level in a tool called VisTorus. The following presents an example of visualizing artificially generated application data, occupying a $(4{\times}4{\times}4{\times}4{\times}2)$ section of a $(a{\times}b{\times}c{\times}d{\times}e)$ torus network. The behavior of the application is such that the network traffic is equally distributed on all the links, except for the d dimension. Along the d dimension, the link in the positive direction of even-ranked nodes have less traffic compared to odd-ranked nodes. The node ranks are assigned using a permutation, starting from the a dimension.

Figure 4a shows the application visualized using the global plane approach. The planes on the left side have dimension b as their horizontal axis and dimension c as their vertical axis. The planes themselves are first ordered along the a dimension inside the blue rectangles, and then along the d dimension inside the brown rectangles. This means that all the nodes in the left most $(b{\times}c)$ plane in a blue rectangle have $a = 0$, in the plane after that have $a = 1$, and so on. The e dimension is represented by the two brown rectangles. The axes are shown in a manually added overlay; smaller fonts show axes inside the blue rectangles where as larger fonts show axes of the complete plane. The depth of the blue rectangle represents the horizontal axis of the $(b{\times}c)$ planes.

The right side of the figure shows the composite plane view where each plane is shown as an empty rectangle. The ordering of the dimensions is the same as in the detailed plane view. The links of interest are the ones representing traffic along the d dimension, visualized by lines between the blue rectangles. Each link represent traffic between two $(b{\times}c)$ planes. The links are ordered along the a dimension, same as the ordering of the planes inside the blue rectangle. Since the node ranks are generated using a permutation starting from a, moving along the a dimension changes the node ranks from even to odd and vice versa. This means that a plane has either all even-ranked nodes or all odd-ranked nodes. As the traffic alternates along the d dimension between even- and odd-ranked nodes, the trend is clearly visible in the links between the blue rectangles.

The communication pattern can also be closely investigated by displaying $(a{\times}d)$ planes in the global plane view, as shown in Figure 4b. It is now clear that moving along the a dimension changes the traffic intensity along the d dimension.

We can also zoom in on a few $(a{\times}d)$ planes and even select a few nodes for closer inspection, as shown in Figure 4c. The nodes in the polygon view on the left side are ordered along a and d dimensions, the ordering of the selected plane. The nodes show traffic in both the positive direction and negative direction, starting from dimension a with a vertical line towards *north* and then changing when moving clockwise. The view clearly shows alternating traffic along the d dimension between even- and odd-ranked nodes.

VI. Conclusion

Torus networks with more than three dimensions are being used in flagship HPC systems. Such networks offer high scalability at low cost, but at the same time make it harder to utilize the network to its potential. A suboptimal mapping of an application communication pattern to compute nodes can result in uneven distribution of traffic on the network links. Identifying such disparity not only requires capturing network traffic, but also providing visual feedback to the user. However, visualizing a multi-dimensional network on two dimensional display is, by its very nature, challenging.

We propose a network traffic visualization technique for this purpose, capable of displaying up to six dimensions. Our basic approach is to use multiple concurrent views that display the same network section. Each view visualizes a different set of dimensions and uses a different projection technique. Using this concept, we proposed plane views, with multi-coordinate axes; compact views, with collapsed dimensions; and polygon views, that can zoom-in on individual nodes, to visualize network traffic at different levels and in different forms. These views are grouped together in two different approaches to network traffic visualization. We also present a prototype of our visualization tool, VisTorus, and show the utility of our technique with an artificially generated example application on a five-dimensional torus network.

VII. Future work

One of the proposed visualization techniques is implemented as a prototype in the VisTorus tool. The next step is to implement all the techniques in a stable version of VisTorus and visualize the traffic generated by production applications. At the same time, the visualization technique has to be extended to utilize statistics to provide users with a quick feedback. Examples of such statistics are box plots and histograms to show the distribution of traffic along different axes and communication links. The network monitoring of LWM^2 currently supports only BlueGene/Q systems. The next step is to extend it to support Cray and Fujitsu systems that also use multi-dimensional torus networks.

Visualization of performance data on multi-dimensional networks is a broad research topic. Providing an all encompassing solution, if at all possible, will require extensive efforts. We realize this and also the fact that gaining experience with actual applications will reveal further techniques to better visualize network performance data.

References

[1] M. Blumrich, D. Chen, P. Coteus, A. Gara, M. Giampapa, P. Heidelberger, S. Singh, B. Steinmacher-burow, T. Takken, P. Vranas, M. Blumrich, D. Chen, P. Coteus, A. Gara, M. Giampapa, P. Heidelberger, S. Singh, B. Steinmacher-burrow, T. Takken, and P. Vranas, "Design and analysis of the bluegene/l torus interconnection network," *IBM Research Report*, 2003.

[2] A. Landge, J. Levine, A. Bhatele, K. Isaacs, T. Gamblin, M. Schulz, S. Langer, P.-T. Bremer, and V. Pascucci, "Visualizing network traffic to understand the performance of massively parallel simulations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2467–2476, 2012.

[3] D. Keim, M. Hao, U. Dayal, M. Hsu, and J. Ladisch, "Pixel bar charts: A new technique for visualizing large multi-attribute data sets without aggregation," in *Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01)*, ser. INFOVIS '01. Washington, DC, USA: IEEE Computer Society, 2001, pp. 113–.

[4] A. Shah, F. Wolf, S. Zhumatiy, and V. Voevodin, "Capturing inter-application interference on clusters," in *IEEE International Conference on Cluster Computing (CLUSTER), 2013*, 2013, pp. 1–5.

[5] K. T. McDonnell and K. Mueller, "Illustrative parallel coordinates," in *Proceedings of the 10th Joint Eurographics / IEEE - VGTC Conference on Visualization*, ser. EuroVis'08. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2008, pp. 1031–1038.

[6] I. Herman, G. Melancon, and M. Marshall, "Graph visualization and navigation in information visualization: A survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, no. 1, pp. 24–43, Jan 2000.

[7] D. Holten, "Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 741–748, Sept 2006.

[8] M. Schulz, J. Levine, P.-T. Bremer, T. Gamblin, and V. Pascucci, "Interpreting performance data across intuitive domains," in *International Conference on Parallel Processing (ICPP), 2011*, Sept 2011, pp. 206–215.

[9] "CUBE 4 Visualization tool. See: http://www.vi-hps.org/projects/score-p/#cube4."

[10] G. M. Namata, B. Staats, L. Getoor, and B. Shneiderman, "A dual-view approach to interactive network visualization," in *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management*, ser. CIKM '07. New York, NY, USA: ACM, 2007, pp. 939–942.

[11] T. Moscovich, F. Chevalier, N. Henry, E. Pietriga, and J.-D. Fekete, "Topology-aware navigation in large networks," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '09. New York, NY, USA: ACM, 2009, pp. 2319–2328.

[12] "BGPM - BG/Q Hardware Perf Monitoring API. See: https://bgq1.epfl.ch/navigator/resources/doc/bgpm/index.html."