CODES-VIS
A Visual Analysis Toolkit for Supporting CODES

Kelvin Li, Takanori Fujiwara, and Kwan-Liu Ma

UCDAVIS  VIDi

CODES Workshop
July 18, 2018
Outline

- CODES-VIS Project Goals
- Past, Present, and Future
- Visualization Techniques and the CODES-VIS System
- Interactive Demo and Tutorial
Goals

- Develop scalable visualization techniques for analyzing large-scale HPC networks and PDES.
  - Support CODES for exploring the design space of HPC networks.
  - Visualize and analyze PDES engine (ROSS) performance.
## Studying Network Performance

<table>
<thead>
<tr>
<th>Exploring design choices</th>
<th>Analyzing network behaviors</th>
</tr>
</thead>
<tbody>
<tr>
<td>● Link arrangements</td>
<td>● Communication pattern</td>
</tr>
<tr>
<td>● Routing strategies</td>
<td>● Network congestion</td>
</tr>
<tr>
<td>● Job placement policies</td>
<td>● Inter-job interference</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
CODES-VIS: Past

Publications and Prototype Systems:

- PMBS 2016
- IEEE Cluster 2017
- Elsevier Journal of Visual Informatics (proc. PacificVAST), 2018

Kelvin Li, Takanori Fujiwara, and Kwan-Liu Ma
Misbah Mubarak and Robert Ross
Caitlin Ross and Christopher Carothers
CODES-VIS: Present

- System Integration and Open Source
- Software Beta
  - https://havex.github.io/codes-netvis/
    - supports Chrome, Firefox, and maybe Sarifi, but no IE/Edge
    - Support Dragonfly, Dragonfly-Custom (1D and 2D), and Dragonfly-Plus (testing)
Data Collection in CODES

- Multiple entities: router, network links, and terminals
- Various performance metrics (e.g., data size, saturation time)
- Time varying
CODES and CODES-VIS

CODES

- Synthetic Traffic Patterns
- Workload Generator
- Network Models
- PDES Engine (ROSS)
- Instrumentation

Simulation Settings
- Node Mappings
- Network Parameters
- Network Link Metrics
- Terminal Metrics

CODES-VIS System

- Data Integration and Preprocessing
- User Interface
  - Data Management
  - Visualization
  - Interaction

Application Traces
- Synthetic Traffic Patterns
- Application Traces
Visual Analytics Techniques

- Hierarchical Data Aggregation
  - Overview of the entire network
  - Different levels of details

- Radial visualization layout
  - Customized visual mappings
  - Correlation of performance
Visual Encoding for Network Links

- **Sender** A
  - Link saturation time
  - Link traffic

- **Receiver** A
  - Link saturation time
  - Link traffic

- Color*: Link saturation time
- Size: Link traffic

Max*(S_{BD}, S_{DB})
Hierarchical Radial Visualization
We want to allow CODES users to easily create scalable and customized visualizations for analyzing large-scale networks exploring various network behaviors.

● Effective User Interface
● A Declarative Language for Creating Visualizations
User Interface
Declarative Language

```json
{
  filter: { group_id : [0, 8] },
  aggregate : "group_id",
  project : "global_links",
  vmap : { size : "traffic"},
  colors : ["white", "purple"]
},
{
  project : "local_links",
  vmap : {
    color : "traffic",
    x : "router_rank",
    y : "router_port"
  },
  colors : ["white", "steelblue"],
},
...
Demo

https://havex.github.io/codes-netvis/
CODES-VIS: Future

- Continue research on scalable visual analytics for network and performance analysis:
  - Conduct co-analysis of model- and simulator-level data
  - In-situ visualization

- Add and integrate existing and new techniques to our online system.
Future CODES-VIS Online System

- Support more network models (e.g., Fat Tree and SlimFly)
- Provide more visualization types and interactions
- Apply and extend current techniques to analyze simulator-level (ROSS) data.
- Allow interactive exploration of time-series data from CODES.
- Adopt machine learning methods (Takanori’s work)
Analyzing Time Series Data
GPU-Accelerated, Interactive Visualization

P4 - Portable Parallel Processing Pipelines
A GPU accelerated JavaScript library for data processing and interactive visualization.

https://jpkli.github.io/p4/
Support for Large Data

- GPU-Accelerated frontend
- Cloud-based backend
Thank You!

Questions?

Email: kelli@ucdavis.edu

Online System (Beta): https://havex.github.io/codes-netvis/

Source code: https://github.com/HAVEX/codes-netvis