Scalable Comparative Visualization of Ensembles of Call Graphs using CallFlow

Suraj P. Kesavan1, Harsh Bhatia1, Abhinav Bhatel1, Stephanie Brink2, Olga Pearce2, Todd Gamblin2, Peer-Timo Bremer2, Kwan-Liu Ma1

1University of California, Davis 2Lawrence Livermore National Laboratory 3University of Maryland, College Park

Motivation
- Large-scale parallel applications often support a variety of optimization parameters and programmers typically run the same code multiple times to understand how different application parameters and/or initial conditions may affect the performance.
- It is important to compare multiple executions and analyze the performance variation across executions.
- Visualization of performance metrics along with the dynamic Calling Context Tree (CCT) can play an important role in detecting and understanding the true causes of bottlenecks.
- However, existing tools are constrained to analyzing a single CCT, thereby limiting the usability for analysis on large ensembles of profiles.

CallFlow – What’s new?
CallFlow [1] is an interactive visual analysis tool that provides a high-level overview of CCTs together with semantic refinement operations to progressively explore hierarchical calling contexts. CallFlow introduces a construct called super graph, created by aggregating the nodes of a CCT based on the semantic attributes, e.g., the library name, module name, or file name. In this work, we extend CallFlow’s visual design to scale performance analysis on large ensembles using ensemble super graph. By combining data analysis and visualization, CallFlow’s visual interface along with ensemble super graph representation enables comparison of the performance variability across ensembles.

CallFlow’s Visual Interface
Visual Encoding of a SuperNode
- Color: Metric distribution across runs
- Height: Execution runtime
- Border/Outline: Selected metric: call site, call graph, and MPI rank
- Text-guides (left): Number of runs
- Text-guides (right): Execution runtime
- Target guides: Target run in ensemble

Fig. 2: Screenshot of CallFlow’s interface: Visualization of the ensemble comprising of 218 Caliper performance profiles.

Construction of Ensemble Super Graph
Performance profiles such as those collected from HPCToolkit [2] and Caliper [3] are converted to an ensemble super graph.

Ensemble Super Graph
- Visualizes the aggregated runtimes (histograms) using white-red color gradients.
- Text and target guides provide detailed information of the bins in the aggregation.
- Double-clicking on the text guides recalculates the ensemble super graph of the selected subset.

Supernode Hierarchy
- Visualizes the caller-callee relationship of call sites inside a supernode.

Metric Correlations
- Shows correlations between any two metrics across ensemble runtimes.
- Colored dots help in comparing members of a target run to its ensemble.

Runtime Distribution
- Visualizes the call site runtime distribution.
- Target run’s distribution (in green) is overlaid over the ensemble.

Call Site Correspondence
- Enumerates the call sites inside a selected supernode.
- Dragged call sites can be used to perform super graph splitting operations.

Pairwise Comparison of Profiles using Diff View
(a) 64-cores-27-cores
(b) 216-cores-155 cores

Performance trends in LULESH
We demonstrate 2 comparison studies using LULESH[5] to study effects of weak scaling on performance across an ensemble of 1, 3, 27, 64, 125, 216, 343, and 512 cores.

Conclusion
We have extended CallFlow [1] to create a scalable, interactive visual analytic tool to study ensembles of CCTs. Working closely with domain experts, we formulate the definition of super graph for an ensemble of profiles and map them to concrete visual mediums to support comparative analysis.

References