Automated Code Generation for High-Performance, Future-Compatible Graph Libraries

THE GRAPHBLAS APPLICATION PROGRAMMING INTERFACE (API) SPECIFICATION released this year allows graph analytics experts and hardware experts to more easily come together to develop high performance graph algorithms. In the coming year we will be developing automated code generation tools to help hardware experts tune the graph primitives defined in the API for each new hardware architecture that is developed.

Goal: Write Once Run Everywhere
The GraphBLAS API Specification was released in May 2017. It defines a set of primitive operations that can be used to implement a wide variety of graph algorithms. It allows the separation of concerns where:

- Graph experts can more easily develop algorithms at a high level using the primitives defined by the API.
- Hardware experts can finely tune the primitives for their present and future hardware architectures.

Goal: Automated Code Generation for High-Performance
It still takes a large team of developers with knowledge of the complex hardware architectures to finely tune the GraphBLAS primitives. With the rapid development of new hardware architectures, the burden on these teams to develop high-performance implementations for each architecture is high. Using formal specifications of hardware capabilities, CMU’s Spiral code generation technology can already automatically generate high performance signal processing codes. We are currently augmenting Spiral to support the GraphBLAS primitives so it can generate high performance code for graph algorithms.

By combining the mathematical descriptions of the graph primitives specified by GraphBLAS and the formal specifications of hardware used by Spiral, our technology will be able to automatically generate high-performance graph library code for today’s hardware platforms as well as future architectures still being designed.

Achieving high-performance from today’s complex hardware architectures traditionally requires large teams of developers with hardware expertise. This year, we are building on automated code generation technology called Spiral (spiral.net) to use the mathematical formalization of the GraphBLAS algorithms to automatically generate the high-performance code for targeted hardware platforms.

GraphBLAS Math
C\(L_{ij}\) = \(L \circ \circ L'\)
count = \(\sum_{i,j} C(i,j)\)

Algorithm Development

Hardware Architectures

Abstraction

Rewriting

Kernel:
problem size, algorithm choice

Architectural parameters:
Vector length, #processors, ...

Optimization

By separating the concerns between algorithm development and hardware architectures, the GraphBLAS API (graphblas.org) frees graph experts to develop algorithms while allowing hardware experts to fine tune primitives for existing and new hardware architectures.