

GraphBLAS: C++ Iterators for Sparse Matrices

Benjamin Brock^{*}, Scott McMillan[§], Aydın Buluç^{†*}, Timothy G. Mattson[‡], José E. Moreira[¶]

^{*} EECS Department, University of California, Berkeley, CA

[§] Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA

[†] Computational Research Department, Lawrence Berkeley National Laboratory, Berkeley, CA

[‡] Parallel Computing Lab, Intel, Ocean Park, WA

[¶] IBM Thomas J. Watson Research Center, Yorktown Heights, NY

Abstract—Iteration over opaque, generic data structures is an important feature of many C++ libraries. Aggressive compiler optimization and inlining enables generic C++ iterators to iterate over complex data structures with performance comparable to that of hand-tuned code with C-language (raw) pointers. In this paper, we describe the sparse matrix iterators in the current draft of the C++ GraphBLAS API, their support for a variety of backend data formats, and implementation strategies we have

The C programming language is a difficult language to use when implementing the GraphBLAS. It lacks features of modern programming languages, such as generics, which are essential as we generalize GraphBLAS to other platforms. We need to adopt a modern language such as C++ as we move forward with the GraphBLAS. We describe the problems with C++ and the design of a high-level abstraction of C++ GraphBLAS