
Victor Eijkhout
**Toward mechanical derivation of Krylov solver methods
and libraries**

Texas Advanced Computing Center
Research Office Complex 1 101
J J Pickle Research Campus
Building 196
10100 Burnet Road (R8700)
Austin
Texas 78758-4497
`eijkhout@tacc.utexas.edu`
Paolo Bientinesi
Robert van de Geijn

In a series of papers, it has been shown that from the mere mathematical specification of a target operation it is possible to systematically and even mechanically derive families of loop-based algorithms for dense linear algebra operations. A framework named FLAME (Formal Linear Algebra Methods Environment) has been developed to realize this aim. FLAME starts with a non-algorithmic specification of the inputs and outputs of an operation; in the case of matrix inversion this would state that the output is the inverse of the input. Once the algebraic description is written as a Partitioned Matrix Expression (PME), it becomes possible to choose an invariant that is to hold in each iteration of the loop-based algorithm, for instance declaring that parts of the result have been computed.

This talk will first of all show how the FLAME methodology, so far mostly applied to direct dense matrix algorithms, can be applied to vector-oriented iterative processes such as Krylov space methods. For this we use the block formulation of iterative methods, going back to Householder, and reason in terms of matrices that comprise the vectors computed. The PME for an iterative method then describes the basic generating relations between the vectors, as well as the condition of orthogonality. We show that from this high level description we can derive specific methods. We will also show that it is possible to derive multiple algorithms from the same PME.

Secondly, we will establish that the reasoning outlined so far can be made sufficiently systematic that mechanical derivation, complete with correctness proof, is within reach. We show that the derivation of the essential steps in a method is in fact driven by the generation of Hoare triples, a central tool proving correctness of algorithms.

Thus, this research makes a case for the feasibility of mechanical generation

of proved correct algorithms and corresponding library software for iterative methods for solving linear systems.