

Second MODE Workshop on Differentiable Programming for Experiment Design



Contribution ID: 75

Type: Poster

Automatic Optimization of Derivatives with Polyhedral Compiler

Wednesday 14 September 2022 19:10 (10 minutes)

Derivatives, mainly in the form of gradients and Hessians, are ubiquitous in machine learning and Bayesian inference. Automatic differentiation (AD) techniques transform a program into a derivative (adjoint) program, which is run to compute the gradient.

Traditionally, most AD systems have been high level, and unable to extract good performance on scalar code or loops modifying memory. These have instead been reliant on extensive libraries of optimized kernels –e.g. for linear algebra, convolutions, or probability distributions –combined with associated adjoint rules, forcing practitioners to express their models in terms of these kernels to attain satisfactory performance.

New low level AD tools such as Enzyme allow for differentiating kernels implemented as naive loops. However, since the derivative code is generated programmatically during the application of reverse mode AD, the reverse passes are likely to access memory in patterns suboptimal with respect to both cache and SIMD performance. This opens a new opportunity for polyhedral loop-and-kernel compilers to provide the aggressive transforms needed for high performance.

We report on work in progress on the implementation of LoopModels –an automatic loop optimization library based on LLVM IR using polyhedral dependency analysis methods to attain performance competitive with vendor libraries on many challenging loop nests, with applications from linear algebra to the derivative code produced by AD.

Author: DEMIN, Alexander (Max Planck Institute for Informatics)

Co-authors: Dr ELROD, Chris (Julia Computing); Mr MA, Yingbo (Julia Computing)

Presenter: DEMIN, Alexander (Max Planck Institute for Informatics)

Session Classification: Wine Tasting and Poster Session

Track Classification: Computer Science