

Randomized Heuristics for Exploiting Jacobian Scarcity

Andrew Lyons^{1,2}, Ilya Safro²

¹ Computation Institute, University of Chicago
5640 S. Ellis Avenue, RI 405, Chicago, Illinois 60637, USA
lyonsam@gmail.com

² Mathematics and Computer Science Division, Argonne National Laboratory
9700 South Cass Avenue, Argonne, Illinois 60439-4844, USA
safro@mcs.anl.gov

Keywords. Jacobian, scarcity, accumulation, directed acyclic graph

1 The Problem

Griewank and Vogel [1] introduced the notion of Jacobian scarcity, which generalizes the properties of sparsity and rank to capture a kind of deficiency in the degrees of freedom of the Jacobian matrix $F'(\mathbf{x})$. We describe new randomized heuristics that exploit scarcity for the optimized evaluation of collections of Jacobian-vector or Jacobian-transpose-vector products.

Let $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$ be a vector function whose evaluation procedure P is given in the form of a straight-line program. We are interested in algorithmically applying the chain rule in order to obtain a new program P^+ that evaluates F along with some derivative information for F . In particular, we are interested in computing either a collection of p Jacobian-vector products $F'(\mathbf{x})\dot{\mathbf{x}}^1, F'(\mathbf{x})\dot{\mathbf{x}}^2, \dots, F'(\mathbf{x})\dot{\mathbf{x}}^p$, or a collection of p Jacobian-transpose-vector products $F'(\mathbf{x})^T\bar{\mathbf{y}}^1, F'(\mathbf{x})^T\bar{\mathbf{y}}^2, \dots, F'(\mathbf{x})^T\bar{\mathbf{y}}^p$. Without loss of generality, we will consider the case where we want the most efficient program $\hat{F} \equiv \hat{F}(\mathbf{x}, \dot{\mathbf{x}})$ for computing Jacobian-vector products $F'(\mathbf{x})\dot{\mathbf{x}}$.

P can be represented as a directed acyclic graph (DAG) $G = (V, E)$, where each edge $e \in E$ is assigned a label c_e that corresponds to the partial derivative of its target variable with respect to its source variable. In particular, we recognize that some edges may be given labels in $\{1, -1\}$, and multiplication with such *unit labeled* edges can be performed without using any floating point operations. Let $E^+ = \{e \in E \mid c_e \notin \{1, -1\}\}$ define the set of *nonunit edges* in G . Traditional automatic differentiation [2] prescribes the *forward mode* for evaluating Jacobian-vector products, whereby derivatives are *propagated* through G by traversing the vertices in topological order. This process can be performed by either propagating p directions $\dot{\mathbf{x}}$ separately (scalar mode), or by propagating $\dot{\mathbf{X}} \in \mathbb{R}^{p \times n}$ in a single pass (vector mode). In both scalar and vector modes, the cost of evaluating p Jacobian-vector products is $p(|E^+|)$ multiplications.

Various graph transformations (eliminations, reroutings, and normalizations) allow us to transform G into the *remainder graph* $\tilde{G} = (\tilde{V}, \tilde{E})$ while preserving the property that Jacobian-vector products can be obtained by propagating

derivative values through \tilde{G} . For functions with scarce Jacobians, judicious application of these transformations can yield a \tilde{G} that has significantly fewer nonunit edges than G . Consequently, propagation can be performed through \tilde{G} at a cost of $p|\tilde{E}^+| < p|E^+|$ multiplications. We distinguish between the *preaccumulation phase*, which consists of transformation operations and has a cost independent of p , and the *propagation phase*, which has cost $p(|E^+|)$. When p is sufficiently large, the cost of the propagation phase dominates the computation. With this as our motivation, we focus on finding a suitable sequence of transformations to minimize $|\tilde{E}^+|$.

2 New Heuristics

Lyons and Utke [3] describe a greedy heuristic for exploiting scarcity, which was tested experimentally on a handful of sample numerical codes. Our new heuristics, which are briefly described here, compare quite favorably with the greedy heuristic when applied to the same set of sample codes.

Consider the *meta-graph* \mathcal{G} of all possible states of the graph G as it undergoes a sequence of edge eliminations. We began with a series of very aggressive random walks over \mathcal{G} in which all transitions were equiprobable. This resulted in an improvement over the greedy heuristic of more than 10%. We also applied various simulated annealing schemes, which improved the results by up to 20%. Our best results were obtained, however, using a classical Metropolis scheme in which a transition probability was proportional to the change in the number of nonunit edges. The Metropolis algorithm with a limited number of state changes and reinforced by several restarts exhibited an improvement of up to 35% over the greedy algorithm. The running time of the scheme was bounded by $100|V|$, where V is a number of vertices in the original graph.

Furthermore, our edge elimination heuristics achieve better results than a greedy combination of edge eliminations and reroutings; our results indicate that inclusion of reroutings into the randomized heuristics is unlikely to yield any significant improvement.

References

1. Griewank, A., Vogel, O.: Analysis and exploitation of Jacobian scarcity. In Bock, H., Kostina, E., Phu, H., Rannacher, R., eds.: *Modelling, Simulation and Optimization of Complex Processes*, New York, NY, Springer (2004) 149–164
2. Griewank, A., Walther, A.: *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*. 2nd edn. Number 105 in *Other Titles in Applied Mathematics*. SIAM, Philadelphia, PA (2008)
3. Lyons, A., Utke, J.: On the practical exploitation of scarcity. In Bischof, C.H., Bücker, H.M., Hovland, P.D., Naumann, U., Utke, J., eds.: *Advances in Automatic Differentiation*. Volume 64 of *Lecture Notes in Computational Science and Engineering*. Springer, Berlin (2008) 103–114