

New version announcement for TaylUR, an arbitrary-order diagonal automatic differentiation package for Fortran 95

G.M. von Hippel¹

Department of Physics, University of Regina, Regina, Saskatchewan, S4S 0A2, Canada

Abstract

We present a new version of TaylUR, a Fortran 95 module to automatically compute the numerical values of a complex-valued function's derivatives with respect to several variables up to an arbitrary order in each variable, but excluding mixed derivatives. The new version fixes a potentially serious bug in the code for exponential-related functions that could corrupt the imaginary parts of derivatives, as well as being compatible with a wider range of compilers.

Key words: automatic differentiation, higher derivatives, Fortran 95

PACS: 02.60.Jh, 02.30.Mv

1991 MSC: 41-04, 41A58, 65D25

NEW VERSION PROGRAM SUMMARY

Manuscript Title: New version announcement for TaylUR, an arbitrary-order diagonal automatic differentiation package for Fortran 95

Authors: G.M. von Hippel

Program Title: TaylUR

Journal Reference:

Catalogue identifier:

Licensing provisions: none

Programming language: Fortran 95

Computer: Any computer with a conforming Fortran 95 compiler

Operating system: Any system with a conforming Fortran 95 compiler

Keywords: automatic differentiation, higher derivatives, Fortran 95

PACS: 02.60.Jh, 02.30.Mv

Email address: vonhippg@uregina.ca (G.M. von Hippel).

URL: <http://uregina.ca/~vonhippg/> (G.M. von Hippel).

¹ Corresponding author

Classification: 4.12 Other Numerical Methods, 4.14 Utility

Catalogue identifier of previous version: ADXR_v1.0

Journal reference of previous version: Comput. Phys. Commun. **174** (2006) 569-576

Does the new version supersede the previous version?: yes

Nature of problem:

Problems that require potentially high orders of derivatives with respect to some variables or derivatives of complex-valued functions, such as e.g. expansions of Feynman diagrams in particle masses in perturbative Quantum Field Theory.

Solution method:

Arithmetic operators and Fortran intrinsics are overloaded to act correctly on objects of a defined type `taylor`, which encodes a function along with its first few derivatives with respect to the user-defined independent variables. Derivatives of products and composite functions are computed using Leibniz's rule and Fàa di Bruno's formula.

Reasons for the new version:

The previous version [1] contained a potentially serious bug in the functions overloading the exponential-related intrinsics (`EXP`, `LOG`, `SIN`, `COS`, `TAN`, `SINH`, `COSH`, `TANH`), which could corrupt the imaginary parts of derivatives. It also contained some features which caused it to crash when compiled with certain compilers (notably the NAG and Lahey/Fujitsu compilers).

Summary of revisions:

The bug in the exponential-related intrinsics has been corrected. A number of additional changes have been made to the code to enable better compatibility with a greater range of compilers, including the NAG and Lahey/Fujitsu compilers. Users of some of these compilers may have to define `useintrinsic` as a preprocessor symbol when compiling TaylUR.

Restrictions:

Memory and CPU time constraints may restrict the number of variables and Taylor expansion order that can be achieved. Loss of numerical accuracy due to cancellation may become an issue at very high orders.

Unusual features:

No mixed higher-order derivatives are computed. The complex conjugation operation assumes all independent variables to be real.

Running time:

The running time of TaylUR operations depends linearly on the number of variables. Its dependence on the Taylor expansion order varies from linear (for linear operations) through quadratic (for multiplication) to exponential (for elementary function calls).

References:

- [1] G. M. von Hippel, TaylUR, an arbitrary-order diagonal automatic differentiation package for Fortran 95, *Comput. Phys. Commun.* **174** (2006) 569-576.