
pyadjoint Documentation

Release 2019.1.2

Sebastian Mitusch

Sep 20, 2023

CONTENTS

1	Contributors	3
2	Licence	5

pyadjoint is an operator-overloading algorithmic differentiation framework for Python. It is employed as the basis for the automatic adjoint and tangent linear model capabilities of the [FEniCS](#) and [Firedrake](#) projects.

These adjoint and tangent linear models are key ingredients in many important algorithms, such as data assimilation, optimal control, sensitivity analysis, design optimisation, and error estimation. Such models have made an enormous impact in fields such as meteorology and oceanography, but their use in other scientific fields has been hampered by the great practical difficulty of their derivation and implementation. In [his book](#), Naumann (2011) states that

[T]he automatic generation of optimal (in terms of robustness and efficiency) adjoint versions of large-scale simulation code is **one of the great open challenges in the field of High-Performance Scientific Computing**.

pyadjoint aims to solve this problem for the case where the model is implemented in the Python interface to FEniCS/Firedrake.

For more technical details on pyadjoint and dolfin-adjoint, see the papers.

CONTRIBUTORS

The pyadjoint project was originally developed by:

- [Sebastian Mitusch](#) (Simula Research Laboratory)
- [Simon W. Funke](#) (Deeptech Consulting)
- [Jørgen S. Dokken](#) (Simula Research Laboratory)

Based on previous work in [Dolfin-adjoint](#) by:

- [Patrick E. Farrell](#) (Mathematical Institute, University of Oxford)
- [Simon W. Funke](#) (Deeptech Consulting)
- [David A. Ham](#) (Department of Mathematics, Imperial College London)
- [Marie E. Rognes](#) (Simula Research Laboratory)

It is currently maintained by Jørgen Dokken and David Ham. For further contributors see [GitHub](#).

LICENCE

pyadjoint is freely available under the [GNU LGPL](#), version 3.