pySpawn User Manual

Version 0.0

by Benjamin G. Levine

Michigan State University

2017

Contents

[2 Introduction 3](#_Toc495602962)

[2.1 What Is pySpawn? 3](#_Toc495602963)

[2.2 What Are The Primary Goals of pySpawn? 3](#_Toc495602964)

[2.3 Features 3](#_Toc495602965)

[2.4 Citing pySpawn 3](#_Toc495602966)

[2.5 License 3](#_Toc495602967)

[2.6 Acknowledgements 4](#_Toc495602968)

[3 Obtaining and Installing PySpawn 5](#_Toc495602969)

[3.1 Obtaining PySpawn 5](#_Toc495602970)

[3.2 Dependencies 5](#_Toc495602971)

[3.3 Installing PySpawn 5](#_Toc495602972)

[4 Guided Tour of Your First pySpawn Simulation 6](#_Toc495602973)

# Introduction

## What Is pySpawn?

PySpawn is an implementation of ab initio multiple spawning (AIMS) in python 2.7.

## What Are The Primary Goals of pySpawn?

PySpawn is designed with the following goals in mind:

* The core of pySpawn is designed to be small and thus easy to maintain.
* PySpawn is designed to be extensible. At present, the following functionality can, in principle, be implemented without modification of existing code:
  + Interfaces to additional electronic structure codes/methods
  + Quantum mechanical Hamiltonians (e.g. adiabatic representation vs. diabatics vs. diabatized Gaussians on adiabatic surfaces, interpolated vs. analytical derivative couplings, spin orbit Hamiltonian, explicit light field)
  + Quantum mechanical integrators
  + Classical mechanical integrators
* PySpawn is designed with large, shared computer resources in mind. Specifically, pySpawn is designed to facilitate
  + Subdivision of large jobs into small chunks
  + Restart
  + Parallelization

## Features

* Runs FMS/AIMS jobs in the adiabatic representation
* Derivative couplings computed by norm preserving interpolation (NPI)
* PySpawn interfaces to a development version of TeraChem via a protobuf interface

## Citing pySpawn

If you use pyspawn please cite the following:

“pySpawn is a nonadiabatic molecular dynamics software package written by Benjamin G. Levine”

This citation will eventually be replaced by a true publication.

## License

PySpawn is distributed under the MIT License:

Copyright (c) 2017 Benjamin G. Levine

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Acknowledgements

* The protobuf interface was developed by Stefan Seritan in the group of Todd J. Martínez at Stanford University.
* Many thanks to B. Scott Fales, Todd J. Martínez, and Stefan Seritan for their contributions, input, and guidance.

# Obtaining and Installing PySpawn

## Obtaining PySpawn

PySpawn can be obtained from github: <https://github.com/blevine37/pySpawn17>.

## Dependencies

PySpawn requires the following dependencies:

* numpy
* h5py
* google (for TeraChem interface)
* protobuf (for TeraChem interface)

These can be easily be installed using pip:

pip install numpy h5py google protobuf

## Installing PySpawn

Having obtained pySpawn and installed the necessary dependencies, cd to the top directory and run

python setup.py install

# Guided Tour of Your First pySpawn Simulation