PATO
- User Guide for Version 1.3.x -

Jean Lachaud
University of California Santa Cruz
jlachaud@ucsc.edu

May 18, 2015

1 Introduction

PATO stands for a **P**orous-material **A**nalysis **T**oolbox based on **O**penFOAM. PATO is a NASA class-E EAR-99 non-safety-critical software. Class-E non-safety-critical means that PATO is not classified to be used for design. It is willfully designed and optimized to be used for research and analysis.

PATO is a fully portable OpenFOAM library (www.openfoam.org). The estimated time to install and become fully familiar with OpenFOAM is 1 week full time (40 hours). The estimated time to install PATO within OpenFOAM is 15 minutes. With previous knowledge of the material-response models implemented, the estimated time to understand and run the tutorials included in PATO 's distribution is 1 day. Then, users should be able to modify the tutorials and the solvers to run other cases and models as needed.

This user guide is specific to PATO. It is in no way meant to replace the official OpenFOAM user manual. However, it should provide sufficient information/references to guide a new OpenFOAM user through the following steps: choosing an operating system, installing OpenFOAM, running a few official OpenFOAM tutorials, installing PATO, running PATO tutorials, modifying PATO tutorials to run other cases.

Contents

2 Notices

3 Directory Structure

The PATO library is meant to be copied and compiled within the OpenFOAM directory. The directory structure of PATO is presented in figure 1.

- `OpenFoam` .... OpenFoam directory
  - `PATO_version#` .... directory of current PATO library version
    - `applications` .... sources and executables of PATO applications
      - `solvers` .... holds solvers
        - `COACO` .... Carbon Oxidation Analysis Code based on OpenFOAM
        - `PAM` .... Pyrolysis Ablation Model
      - `utilities` .... holds pre- and post-processors
        - `MakeAxialMesh` pre-processor to transform a 2D mesh in a wedge mesh [by B. Gschaider]
        - `processSets` ...... post-processor to extract interpolated probed data -eg. thermocouple values
    - `data` .... chemistry data for Mutation++ and OpenFOAM
    - `libPATO` .... sources and executable of PATO classes
      - `BLinput` .... reads and interpolates boundary layer data
      - `BprimeSI` .... reads and interpolates B' tables
      - `fluxFactor` .... reads and interpolates axi-symmetrical fluxes
      - `gasProperties` .... reads and interpolates tabulated gas properties
      - `gradTempAccess` .... OpenFOAM -derived boundary-condition
      - `materialProperties` .... reads and interpolates tabulated material properties
      - `PATOchemistryModel` ...... PATO ODE chemistry model based on the native OpenFOAM chemistry library
    - `documentation` .... root directory for PATO documentation
      - `manuals` .... PATO and OpenFOAM manuals
      - `references` .... holds useful reference documents
    - `thirdParty` .... third-party software directory
      - `Mutation++` .... sources and executable of Mutation++
    - `tutorials` .... root directory for all PATO tutorials
      - `COACO` .... COACO tutorial
      - `PAM` .... PAM tutorials

Figure 1: PATO 's directory structure.

4 Installation

This section describes how to properly install and setup the PATO library on your machine: 1- install OpenFoam, 2- learn how to use OpenFOAM, 3- install PATO within OpenFOAM.

4.1 Install OpenFOAM

Note: It is recommended to start using OpenFOAM on an Ubuntu operating system (OS) because OpenFOAM is heavily tested and well supported for this OS. Ubuntu is easily installed on any machine, either as a single boot, double boot, or in a virtual machine. It should not take more than one hour to install it, and it would save you time and would keep you out of trouble for a first installation of OpenFOAM. This being said, OpenFoam should work equally well on any Unix/Linux machine.

Download OpenFOAM 2.3.0 from the OpenCFD website (http://www.openfoam.org/download/) and install it by following carefully their installation instructions. It is recommended to compile OpenFOAM from sources (rather than just getting the executables) because it will insure full compatibility on your system and it is an excellent way to verify that all the components (e.g. compilers) are installed.

4.2 Learn some basics of OpenFOAM

Please learn how to use OpenFOAM before installing PATO, as the PATO manual is not meant to teach you the OpenFOAM basics. The OpenFOAM user and programmer manuals are available from www.openfoam.com - they are included in the folder documentation/manuals of PATO 's distribution for convenience. Both OpenFOAM manuals ('User guide' and 'Programmer's guide') contain throughly documented tutorials. In general people like to explore OpenFOAM by running several of the OpenFoam tutorials. The 'cavity' tutorial - which is the first tutorial of the user guide - provides a very nice overview of OpenFOAM architecture, capabilities, and pre/post-processing tools.

4.3 Install PATO in OpenFOAM

4.3.1 Extract PATO

PATO is distributed as a GnuPG encrypted file. The file and the password are provided to authorized users only. GnuPG should be installed by default on any system. If not, it is open source and can be downloaded for freefor Linux/Windows/Mac.

```
gpg PATO_v1.x.x.tar.gpg
enter password: ....... (provided to authorized users in a blank email)
```

GnuPG will decrypt the file and write a tar file: PATO_v1.x.x.tar Decompress the tar file with the following command:

```
tar -xz PATO_v1.x.x.tar
```

Copy the directory 'PATO_v1.x.x' in your OpenFOAM directory.

```
cp path-to-where-you-extracted-PATO/PATO_v1.x.x $WM_PROJECT_INST_DIR
```

4.3.2 Compile PATO

As a convenience, the thirdParty library Mutation++ is provided together with PATO. Mutation++ stands for a MUlticomponent Thermodynamic And Transport property library for IONized plasmas written in C++. It is developed at the von Karman Institute for fluid dynamics (VKI), Belgium, by J.B. Scoggins and T. Magin. J.B. Scoggins and T. Magin have kindly accepted to let us include Mutation++ as a third party software in the PATO releases. However, you must also obtain the permission of using Mutation++ from the VKI (scoggins@vki.ac.be or magian@vki.ac.be).

The following Mutation++ paths need to be copied in you .bashrc file (adapt the path to your own system installation if needed)

```
export MPP_DATA_DIRECTORY=$WM_PROJECT_INST_DIR/PATO_v1.x.x/data/mutation++
export PATH=${PATH}:$WM_PROJECT_INST_DIR/PATO_v1.x.x/thirdParty/mutation++/
    mutation++/install/bin
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$WM_PROJECT_INST_DIR/PATO_v1.x.x/
    thirdParty/mutation++/mutation++/install/lib
```

The compilation script 'Allwmake' in the 'PATO_v1.x.x' directory will compile Mutation++, and PATO's libraries, solvers, and utilities. Open a new terminal and run the following commands:

```
cd $WM_PROJECT_INST_DIR
cd PATO_v1.x.x
./Allwmake
```

Compilation warnings may appear for Mutation++ (e.g., comparison of signed and unsigned integers) but this is not a problem.

If you run into any compilation error, please contact me.

5 Description of the solvers

Supporting documentation for the models implemented in the solvers is found in the directory: 'documentation/references'. Important note: To increase robustness and ease of use for first-time users, several warnings are ignored. For example, data may be extrapolated outside of the table furnished by the users. In the source code, these are labeled by the flag "CheckPoint". Please make sure to look-up and remove these flags in all applications and libraries if you are using the code for applied analysis.

5.1 COACO: Carbon Oxidation Analysis Code based on OpenFOAM

COACO is being developed for the fundamental analysis of the oxidation of porous carbon-fiber materials. COACO is used as an inverse analysis tool to extract the intrinsic oxidation properties of materials in ground test conditions - rather than experimental set-up dependent properties. It is also used to assist with the development of physics-based macroscopic oxidation model for macroscopic/multi-physics tools like PAM. COACO model, solver, and applications are described in the papers "Lachaud2011_AIAA".

5.2 PAM: Pyrolysis Ablation Model

PAM is a type 2 and 3 ablative-material response code as described in the papers "Lachaud2013_AIAA", "Lachaud2014_JTHT". It can run 1D, 2D, 2D-axi-symmetrical, and 3D problems. Seven (7) tutorials show the capabilities and options available in PAM. They start with a simple 1D case with fixed surface temperature and no recession to finish with a full 3D case with variable surface heat flux. The source code of the solver is readable (OpenFOAM operators are used) and commented (application/solver/PAM) if detailed information on the numerical method implemented is sought.

6 Tutorials

The available tutorials are shown in the directory tree of Fig. 2. All units used in PATO are SI units.

```
Tutorials ................................................ Tutorial directory
├── COACO ................................................ COACO tutorial
│   └── fiberFormOxidation  Simulates the oxidation of the fibers in a fiber preform - see
│                           ref. doc.
└── PAM .................................................. PAM tutorials
    ├── AblationTestCase_1.0 ......... 1D, imposed surface temperature, no recession
    ├── AblationTestCase_2.x ..... 1D, boundary-layer approximation, generic case for
    │                              series 2
    ├── AblationTestCase_2.x_multiMat ..................... includes material stack
    ├── AblationTestCase_3.x .................. 2D-axi IsoQ, generic case for series 3.
    ├── AblationTestCase_3.x_multiMat ..................... includes material stack
    ├── Cylinder3D ..................... 3D, boundary-layer approximation, recession
    └── Stardust ................ 1D, fight case - Stardust stagnation point, multiMat
```

Figure 2: List of PATO tutorials.

6.1 Overview

Each tutorial may be run (from mesh generation to data plotting) by executing its script in the terminal

```
./Allrun
```

or by executing each command contained in the script in sequence. For first time users, it is recommended to edit the script and look at the sequence.

We provide below a description of the base sequence. It follows the usual OpenFOAM sequence, and it is similar to other traditional CFD codes: mesh generation, update boundary-conditions and physical properties (if needed), code execution, post-processing.

For simple geometries, blockMesh is used to generate the meshes.

```
blockMesh
```

The solver is executed:

```
solverName (e.g. PAM)
```

Output data are placed in the 'output' directory (specific to PATO) and in the time directories automatically generated by OpenFOAM.

In most of the tutorial, there are gnuplot scripts in the 'plots' directory. They may be executed as follows to plot the results:

```
gnuplot plot<relevantName>.gnu
```

For example, in the PAM tutorials, the temperature profiles may be plotted:

```
cd plots
gnuplot plotT.gnu
```

Gnuplot will export a "PATO_Energy_TestCase_2.x.ps" file.

Alternatively, 'paraview' may also be used, as explained in the 'cavity' tutorial.

The 'sample' post-processor may be used to extract interpolated probed values (like 'thermocouple' data):

- set the right positions in the sampleDict (in systems/)
- run sample:

  ```
  sample
  ```

  This will create a directory 'sets' and place inside runTime directories containing the required values for each time step separately.
- run PATO post-processor to grab the data produced by sample and reorganize them in a single file:

  ```
  processSets
  ```

  This last step will produce a list_T file that can be plotted with gnuplot.

6.2 COACO tutorial

COACO tutorial is extremely simple. The fiberFormOxidation case is described in the paper "Lachaud2011_AIAA".

6.3 PAM tutorials

As shown in figure 3, the directory structure of PAM tutorials follows OpenFOAM tutorial structure with a few additions aiming at improving user friendliness.

To set a new case, start from an existing tutorial, first copy it and rename it, then open the file 'setCase'. This file is heavily commented to help the user set a new case in an efficient way. Follow carefully the guidelines provided in the 'setCase' file, it will guide you through the steps and tell you what options are available. Then, please observe carefully figure 3, and update the input files as needed.

All units used in PATO are SI units. SI units are recalled in the headers of the input files to avoid confusions. If needed, British unit to SI unit conversion factors are available in the TACOT material property table (excel file). PAM's Ablation-Test-Case_2.i cases are described in the document "AblationTestCase#2_v2.8.pdf". When available, Amaryllis results are provided as a reference in the directory: plots/refs.

```
tutorials ............................................ tutorial directory
└── PAM ................................................... PAM tutorials
    └── 'Tutorial_x' ................................. Tutorial directory
        ├── 0 ............ time '0' directory - see OpenFOAM user manual
        ├── constant ......................... mesh and constant directory
        │   ├── ablaMat ....................... ablative material properties
        │   │   ├── char ...................... charred material properties
        │   │   ├── gasProperties ............... tabulated gas properties
        │   │   ├── virgin ...................... virgin material properties
        │   │   ├── chemistryProperties ..... finite-rate chemistry solver options
        │   │   ├── constantProperties ...... constant material and pyrolysis properties
        │   │   ├── dynamicMeshDict ..................... mesh motion options
        │   │   └── thermophysicalProperties .... chemistry mixtures, mechanisms, and
        │   │       thermo database selection
        │   ├── environmentProperties ........ external properties: heat flux, etc
        │   │   ├── boundaryConditions ................ boundary conditions
        │   │   ├── Bprime ............................................ B' table
        │   │   └── fluxFactorMap .............. map for axi-symmetrical loads
        │   ├── polyMesh ......... mesh directory - see OpenFOAM user manual
        │   ├── regionProperties ........................ for multiMat option
        │   └── setCase ............. studied problem options − >update first
        ├── output ................................................ output directory
        ├── plots ................... plot directory (for post-processing)
        │   └── refs ................... reference results for comparison
        │       ├── Amaryllis ............................. Amaryllis results
        │       └── PAM ........................................... PAM results
        └── system ....... 'system' directory - see OpenFOAM user manual
```

Figure 3: Typical tutorial directory structure in PAM.