

Pharmaceutical
Multiphase Reactors
CHE.782

Design of Multiphase
Flow Processes
669.266

Hands on the „ParScale“ Library

A – General Information
B – Guide - Verification Cases

Ass.Prof. Dr. Stefan Radl,
M. Sc. Thomas Forgber
Email: radl@tugraz.at
Institute of Process and
Particle Engineering
Inffeldgasse 13/III
TU Graz

A part of this teaching material has been
prepared for NanoSim (<http://sintef.no/NanoSim/>)



NanoSim - A Multi-scale Simulation-Based Design Platform

Why verification/test cases:

- Ensure correct implementation (**verification cases**: compare simulation result with exact solution)
- Showcases to demonstrate software capabilities (**test cases** produce „eyecandy“)
- Ensure software backward capability

Where to find:

- Verification cases: „*examples/verificationCases/*“
- Test cases: „*examples/testCases*“
- Used octave functions and plotting/formatting routines: „*examples/octaveFunctions*“

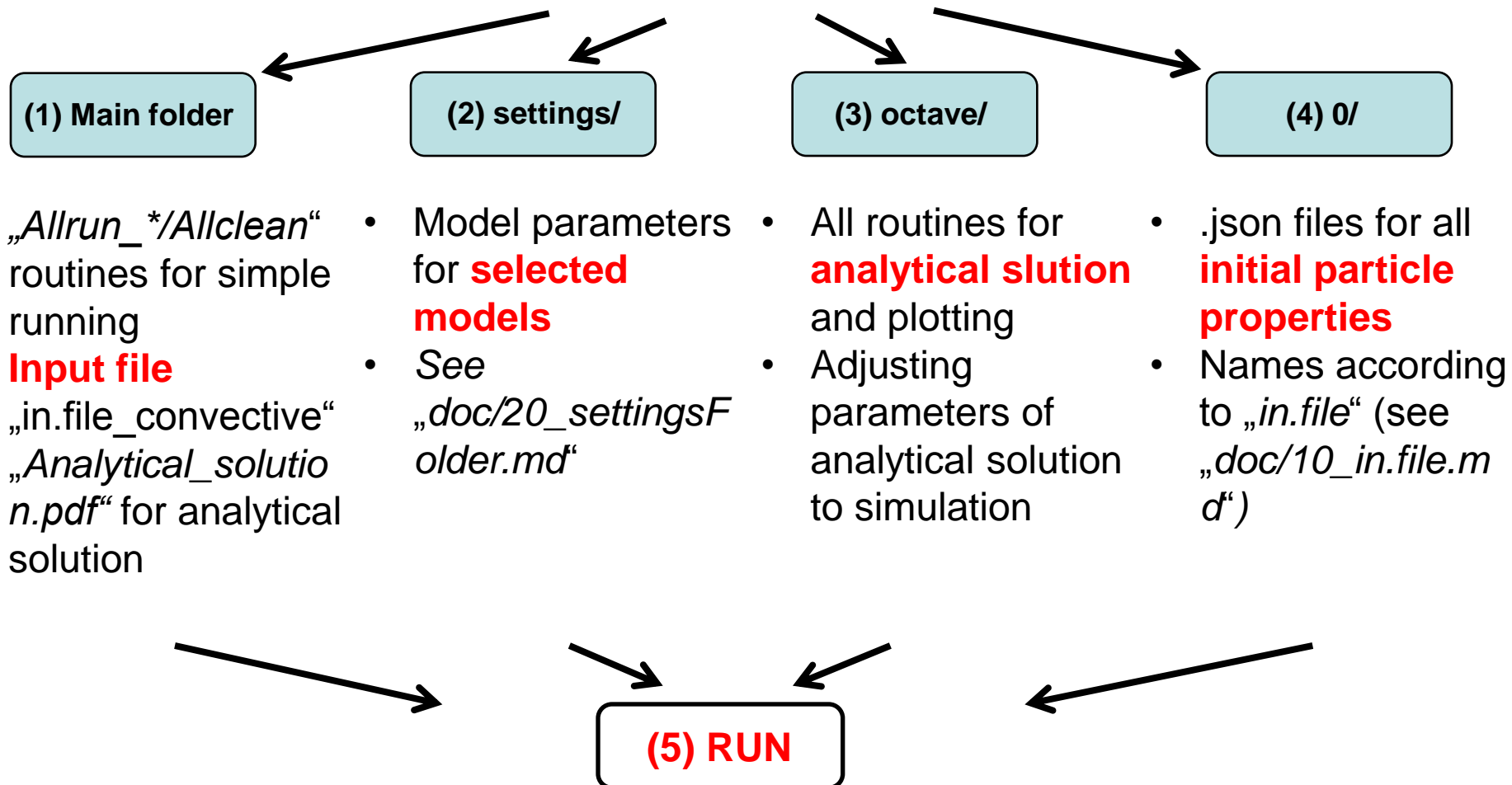
When to run:

- After compiling *ParScale* as a **stand-alone application**
- **AFTER** reading all *.md files in „*doc/*“ for general informations regarding structure and input of *ParScale*

Verificastion Case – Particle Drying:

- Go to „*examples/verificationCases/Drying_particle*“

(Main) Structure of verificastion cases



Particle Drying – A Walk-through (1):

- Have a look at the **main input script** „*verificationCases/Drying_particle/in.file_convective*“ for basic information on:
 - Grid points
 - Coupling
 - Models
 - Model equations
 - Boundary conditions
 - Physical state (liquid, solid, gas)
 - Phase Change models
 - Simulation time, output time, time step
 - Also see „*doc/10_in.file.md*“)
- Readme.md for **short description**
- „*Allrun/Allclean*“ for complete automatic running/cleaning of the case
- „*Analytical-solution.pdf*“ for documentation of solved equations

```
particle_mesh nGridPoints 40
```

Particle mesh information

```
particle_data number_particles 1 verbose 0
```

Number of particles

```
coupling none myCoupling
```

Coupling information

```
model propertiesThermo liquidSpeciesDiffusivity
```

```
model propertiesThermo liquidSpeciesTortuosity
```

```
model propertiesThermo liquidSpeciesPermeability
```

```
model propertiesThermo liquidSpeciesViscosity
```

```
model propertiesThermo liquidSpeciesSurface_tension
```

```
model propertiesThermo liquidSpeciesFilm_flow
```

```
model propertiesThermo liquidSpeciesTransferCoeff
```

```
model propertiesThermo speciesDiffusivity
```

```
#model propertiesThermo speciesTransferCoeff
```

```
model propertiesThermo speciesTortuosity
```

```
model propertiesThermo speciesDiffusivity
```

```
model propertiesThermo speciesTransferCoeff
```

```
model propertiesThermo speciesTortuosity
```

Model section

- naming MUST match modelEqn section!

Solved model equation (1D spherical) +
name of model equation (heat,species,...)

Boundary conditions

Additional options (see documentation)

```
#Equations
```

```
modelEqn 1DSpherical heat
```

```
modelEqn 1DSpherical species
```

```
modelEqn 1DSpherical liquidSpecies liquid BC0 1 BC1 2 updatePhaseFraction averagePhaseFraction writeDebugContainer
```

```
gas
```

```
BC0 1 BC1 1 inactive #solveConvectiveFlux #writeDebugContainers
```

```
BC0 1 BC1 1 writeDebugContainers #solveConvectiveFlux phaseFractionMinim
```

```
BC0 1 BC1 2 updatePhaseFraction averagePhaseFraction writeDebugContainer
```

Physical state of model equation (only available for species)

```
#Phase Change
```

```
modelPhaseChange Evaporation liquidToGasEvaporation
```

Phase change models

```
control outputTimeStep 2e-2
```

```
control timeStep 1e-5
```

```
control run 400e-2
```

Control section (output time-step, simulation time step,
simulation run time)

Particle Drying – A Walk-through (2):

- Go to **„setting/“** to model parameters for your selected models
- „*coupling_none.json*“ contains information regarding coupling
 - See „*doc/13_CouplingModel.md*“
- **„integrator.json“** has settings for the integrator (CVODE)
 - See „*doc/20_settingsFolder_1_integrator.md*“
- **„liquidToGasEvaporation.json“** refers to the phase change model
 - See „*doc/18_PhaseChange_1_Evaporation.md*“
- **„parscale.json“** handles output data format
 - See „*doc/20_settingsFolder_2_parscale.md*“
- **„verbose.json“** enables extended display output
- **„model_*.json“** assigns model parameter to models specified in „*in.file*“
 - „*doc/20_settingsFolder.md*“
 - „*doc/16_EquationProperties.md*“
 - „*doc/14_ModelEquation.md*“

 coupling_none.json

rename, bounding of tempPha

 integrator.json

Major upgrade: liquid phase u

 liquidToGasEvaporation.json

Included derivative of phase fi

 model_liquidSpeciesDiffusivity.json

Removed hard-coded values i

 model_liquidSpeciesFilm_flow.json

rename, bounding of tempPha

 model_liquidSpeciesPermeability.j...

rename, bounding of tempPha

 model_liquidSpeciesSurface_tensi...

Major upgrade: liquid phase u

 model_liquidSpeciesTortuosity.json

rename, bounding of tempPha

 model_liquidSpeciesTransferCoeff....

rename, bounding of tempPha

 model_liquidSpeciesViscosity.json

rename, bounding of tempPha

 model_speciesDiffusivity.json

Major upgrade: made evapora

 model_speciesTortuosity.json

rename, bounding of tempPha

 model_speciesTransferCoeff.json

rename, bounding of tempPha

 parscale.json

rename, bounding of tempPha

 verbose.json

rename, bounding of tempPha













See documentaiton for:

- coupling_none.json
- integrator.json
- liquidToGasEvaporation.json
- parscale.json
- verbose.json

Model properties

Particle Drying – A Walk-through (3):

- Go to „**octave**“ to see routines and plotting scripts for e.g. analytical solution:
 - „*Main_stefan_diffu.m*“ is the main script which calls sub-routines included in the folder
 - „*parametersDrying.m*“ contains all relevant parameter for the analytical solution
 - „*postProcessing.m*“ is post processing simulation data and plots them against the analytical solution
 - „*preProcessor.m*“ enables the user to easily produce .json files for the particle properties (e.g. temperature of single gridpoints)

 Main_stefan_diffu.m	Re-worked
 StefanDiffusion1DSpherical.m	Drying Cas
 StefanDiffusion1DSphericalFlux.m	Major upgr
 convective_speed.m	verification
 diameter_evo.m	rename, b
 evaporation_time.m	Re-worked
 parametersDrying.m	Included d
 plot_property.m	test case u
 postProcessing.m	rename, b
 preProcessor.m	Included d
 simpleDiffusion1DSpherical.m	Re-worked
 simpleDiffusion1DSphericalFlux.m	Re-worked

Main script for post processing

Main plotting routines

Main plotting routines

Particle Drying – A Walk-through (4):

- Go to „0/“ to see the initial condition of your particle:
 - Phase fractions („gasPhaseFraction.json“/“*liquidPhaseFraction.json*“)
 - Temperature („*heat.json*“)
 - Radius („*radius.json*“)
 - Concentraion („*species.json*“)
 - See („*doc/30_0folder.md*“) for further information

 gasPhaseFraction.json

**Input for initial particle properties
(required for all simulations)**

 heat.json

Changed cal

 liquidPhaseFraction.json

**Name and initial conditons for
solved model equation, name
and existence dependent on
simulation**

 liquidSpecies.json

 radius.json

Major upgrad

 species.json

Changed cal

Particle Drying – A Walk-through (5):

- Run the **pre-processing**
 - „*cd octave*“
 - „*octave < preProcessor.m*“
 - „*cd ..*“
- Run the **main application**
 - „*mpirun -np 1 ParScale < in.file_convective*“ (ParScale is the library name)
- Run the **post-processing**
 - „*cd octave*“
 - „*octave < Main_stefan_diffu.m*“
- **OR:** Fully automatic running by typing „*./Allrun_convective*“
- Have a look at the data and automatically generated plots

Impressum & Disclaimer

©2015 by Stefan Radl, and other members of the „Simulation Science“ Group at the Institute of Process and Particle Engineering, Graz University of Technology.

All rights reserved. No part of the material protected by this copyright notice may be reproduced or utilized in any form or by any means, electronically or mechanically, including photocopying, recording or by any information storage and retrieval system without written permission from the author.

LIGGGHTS® is a registered trade mark of DCS Computing GmbH, the producer of the LIGGGHTS® software. CFDEM® is a registered trade mark of DCS Computing GmbH, the producer of the CFDEM®coupling software. This offering is not approved or endorsed by DCS Computing GmbH, the producer of the LIGGGHTS® and CFDEM® coupling software and owner of the LIGGGHTS and CFDEM® trade marks.