OpenMKM

Opensource **M**icro**K**inetic **M**odeling Toolkit

Bharat Medasani, Sashank Kasiraju, Dionisios Vlachos*
University of Delaware

https://github.com/VlachosGroup/openmkm
Medasani, B. et al. *J. Chem. Inf. Model.* **2023**, *63 (11)*, 4319–4355.

VlachosLab
AT THE UNIVERSITY OF DELAWARE

Motivation: Why use OpenMKM?

- OpenMKM is an open-source micro-kinetic modeling toolkit built in C++
- It automates the generation and simulation of the differential algebraic equations (DAE) that make up the MKM and performs sensitivity analysis (SA)
- No hand-calculations or programming is required to run OpenMKM
- Uses standardized input files (YAML) and (generates) output files (OUT, CSV) which are human-readable
- Inputs can be generated systematically using pMuTT[1] (also part of VLab)
- Uses state-of-the-art numerical libraries (Cantera[2], Sundials[3], Boost C++[4])
- Interfaces to Python (sensitivity analysis), RenView[5] for Path Visualization

[1] https://vlachosgroup.github.io/pMuTT/

[2] https://cantera.org

[3] https://computing.llnl.gov/projects/sundials

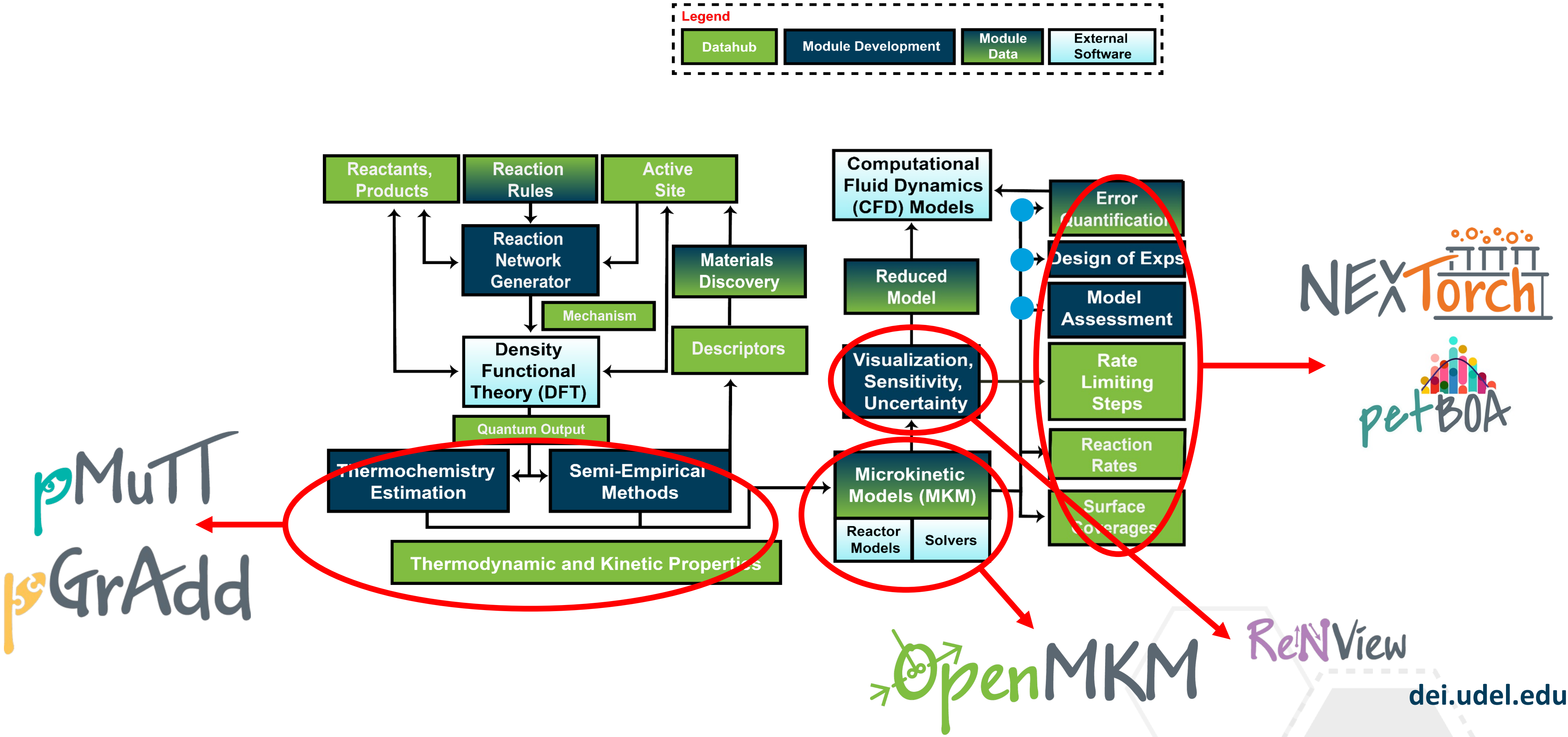
[4] https://www.boost.org/

[5] https://github.com/VlachosGroup/renview

RAPID
Transforming Process Industries

Where does OpenMKM fit in V-Lab?

Bridge between atomic/meso scale data and macro-scale observables.



Legend
Datahub | Module Development | Module Data | External Software

Reactants, Products

Reaction Rules

Active Site

Reaction Network Generator

Mechanism

Materials Discovery

Descriptors

Density Functional Theory (DFT)

Quantum Output

Thermochemistry Estimation

Semi-Empirical Methods

Thermodynamic and Kinetic Properties

Computational Fluid Dynamics (CFD) Models

Reduced Model

Visualization, Sensitivity, Uncertainty

Microkinetic Models (MKM)

Reactor Models

Solvers

Error Quantification

Design of Exps

Model Assessment

Rate Limiting Steps

Reaction Rates

Surface Coverages

pMuTT

pGrAdd

NEXTorch

petBOA

OpenMKM

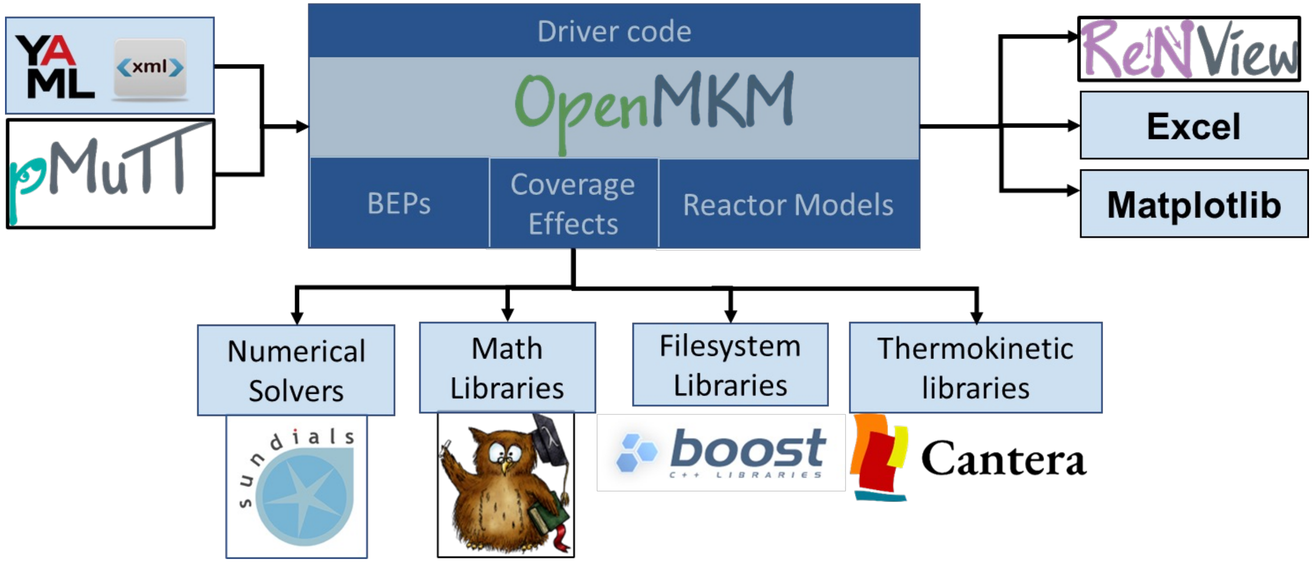
ReNView

dei.udel.edu/rapid

OpenMKM: Core Modules

Uses state-of-the art libraries for stability and robustness.

- OpenMKM's core modules are derived from:
  - The Cantera[1] library
  - Sundials[2] ODE/DAE solvers
  - Boost[3] C++ libraries
- YAML Input files are generated easily using
  - The pMuTT[4] library
  - Optionally also using YAML[2] API



[1] https://cantera.org
[2] https://computing.llnl.gov/projects/sundials
[3] https://www.boost.org/
[4] https://vlachosgroup.github.io/pMuTT/
[5] https://yaml.org/

Microkinetic Model Inputs: Reactor Conditions

Reactor

(e.g., Batch, CSTR, PFR)

$Q_0 = 1\ \mathrm{mL\ s^{-1}}$

Inlet composition, x_0:

- N_2
- H_2
- Inert

$T = 673\ \mathrm{K}$

$P = 100\ \mathrm{bar}$

$V = 1\ \mathrm{cm^3}$

$m_{cat} = 0.1$ g Ru/Al_2O_3 catalyst

Legend

Set by user

Calculated by MKM

Derived from DFT

i – Species i

j – Reaction j

RAPID Transforming Process Industries

Microkinetic Model Inputs: Elementary Steps

Reactor
(e.g. Batch, CSTR, PFR)

Q_0 = 1 mL s^{-1}

Inlet composition, x_0:

- N_2
- H_2
- Inert

T = 673 K

P = 100 bar

V = 1 cm^3

m_{cat} = 0.1 g Ru/Al_2O_3 catalyst

Legend

Set by user

Calculated by MKM

Derived from DFT

i – Species i

j – Reaction j

Elementary reversible reactions:

- Adsorption/Desorption
  1. $N_2 + * \leftrightarrow N_2*$
  2. $H_2 + * \leftrightarrow 2H*$
  3. $NH_3 + * \leftrightarrow NH_3*$
- Surface Reactions
  4. $N_2* + * \leftrightarrow 2N*$
  5. $N* + H* \leftrightarrow NH*$
  6. $NH* + H* \leftrightarrow NH_2*$
  7. $NH_2* + H* \leftrightarrow NH_3*$

Microkinetic Model Inputs: Mass Balance for N_2*

$$\frac{d}{dt}[N_2^*] = \underbrace{k_1^{fwd}P_{N_2}[*] - k_1^{rev}[N_2^*]}_{\text{Adsorption}} - \underbrace{k_4^{fwd}[N_2^*][*] + k_4^{rev}[N^*]^2}_{\text{Surface Reactions}}$$

Legend

Set by user

Calculated by MKM

Derived from DFT

i – Species i

j – Reaction j

Elementary reversible reactions:

- Adsorption/Desorption
  1. $N_2 + * \leftrightarrow N_2^*$
  2. $H_2 + * \leftrightarrow 2H^*$
  3. $NH_3 + * \leftrightarrow NH_3^*$
- Surface Reactions
  4. $N_2^* + * \leftrightarrow 2N^*$
  5. $N^* + H^* \leftrightarrow NH^*$
  6. $NH^* + H^* \leftrightarrow NH_2^*$
  7. $NH_2^* + H^* \leftrightarrow NH_3^*$

Microkinetic Model Inputs: Modified Arrhenius Rate Constant

$$\frac{d}{dt}[N_2^*] = k_1^{fwd} P_{N_2}[*] - k_1^{rev}[N_2^*] - k_4^{fwd}[N_2^*][*] + k_4^{rev}[N^*]^2$$

$$k_j^{fwd} = \underbrace{A'_j \left(\cancelto{T}{\frac{T}{T_{ref,j}}}\right)^{\beta_j}}_{A_{Arrhenius}} \exp\left(-\frac{E_{A,j}}{RT}\right)$$

- Arrhenius equation when $\beta_i = 1$ and $T_{ref,i} = 1\ K$

Legend

Set by user

Calculated by MKM

Derived from DFT

i – Species i

j – Reaction j

RAPID
Transforming Process Industries

Microkinetic Model Inputs: Reverse Rate Constant

$$\frac{d}{dt}[N_2^*] = k_1^{fwd} P_{N_2}[*] - k_1^{rev}[N_2^*] - k_4^{fwd}[N_2^*][*] + k_4^{rev}[N^*]^2$$

$$\begin{tikzcd}
k_j^{fwd} = A_j' \left(\frac{T}{T_{ref,j}}\right)^{\beta_j} \exp\left(-\frac{E_{A,j}}{RT}\right) \arrow[r, "\text{By microscopic reversibility}"'] & k_j^{rev} = \frac{k_j^{fwd}}{K_j^{eq}} \arrow[d] \\
 & K_j^{eq} = \exp\left(-\frac{\Delta G_j}{RT}\right)
\end{tikzcd}$$

Legend

Set by user

Calculated by MKM

Derived from DFT

i – Species i

j – Reaction j

Microkinetic Model Inputs: Adsorption Preexponential

$$\frac{d}{dt}[N_2^*] = k_1^{fwd} P_{N_2}[*] - k_1^{rev}[N_2^*] - k_4^{fwd}[N_2^*][*] + k_4^{rev}[N^*]^2$$

$$k_{ads,j}^{fwd} = A'_{ads,j}\left(\frac{T}{T_{ref,j}}\right)^{\beta_j}\exp\left(-\frac{E_{A,j}}{RT}\right)$$

By collision theory

$$A'_{ads,j} = \frac{s_j}{(\sigma)^{n_{surf}}}\sqrt{\frac{RT}{2\pi M_i}}$$

Legend

Set by user

Calculated by MKM

Derived from DFT

i – Species i

j – Reaction j

Microkinetic Model Inputs: Surface Reaction Preexponential

$$\frac{d}{dt}[N_2^*] = k_1^{fwd} P_{N_2}[*] - k_1^{rev}[N_2^*] - k_4^{fwd}[N_2^*][*] + k_4^{rev}[N^*]^2$$

$$k_{surf,j}^{fwd} = A'_{surf,j}\left(\frac{T}{T_{ref,j}}\right)^{\beta_j} \exp\left(-\frac{E_{A,j}}{RT}\right)$$

By transition-state theory

$$A'_{surf,j} = \frac{k_B}{h}\frac{1}{(\sigma)^{n_{surf}-1}}\exp\left(\frac{\Delta S_j^{\neq}}{R}\right)$$

$$E_{A,j} = \Delta H_j^{\neq}$$

Legend

Set by user

Calculated by MKM

Derived from DFT

i – Species i

j – Reaction j

Microkinetic Modeling Input Summary

Set by user

- Reactor conditions
  - T, P, Q, V, x_{i0}, m_{cat}
- Elementary steps
  - Adsorption
  - Surface reactions
- Reaction parameters
  - s_j, σ, $T_{ref,j}$, β_j

Calculated by MKM

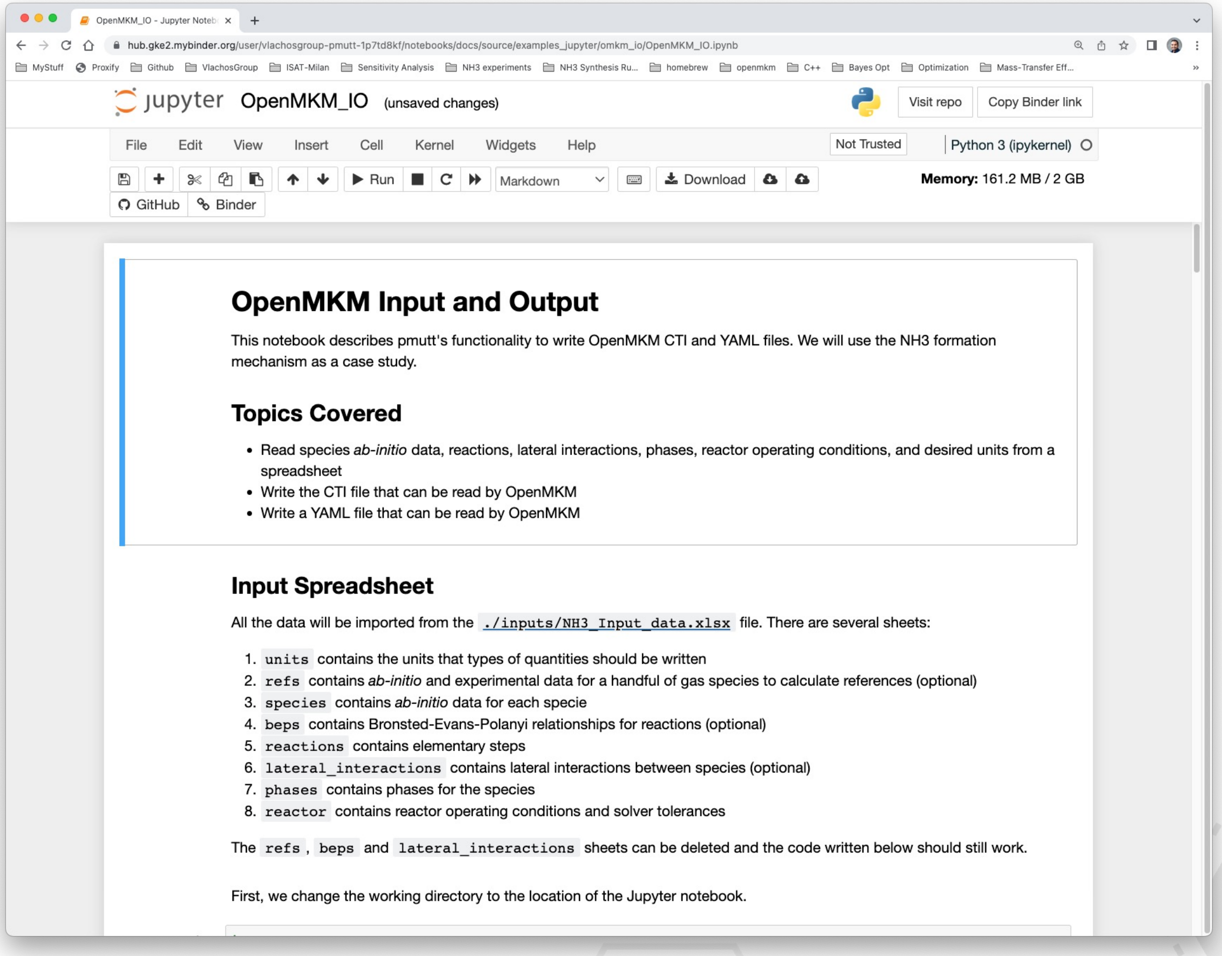
- Surface coverages
- Gas-phase compositions
- Rates

Derived from DFT/GA

- Enthalpies (H) and entropies (S) of:
  - Reactants
  - Products
  - Intermediates
  - Transition states

How to make input files for OpenMKM?

- **YAML Input files are generated easily using pMuTT**
  - Detailed pMuTT[1] documentation to generate the input files.


OpenMKM_IO - Jupyter Noteb
hub.gke2.mybinder.org/user/vlachosgroup-pmutt-1p7td8kf/notebooks/docs/source/examples_jupyter/omkm_io/OpenMKM_IO.ipynb
jupyter OpenMKM_IO (unsaved changes)
Visit repo
Copy Binder link
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
Download
GitHub
Binder
Memory: 161.2 MB / 2 GB

OpenMKM Input and Output

This notebook describes pmutt's functionality to write OpenMKM CTI and YAML files. We will use the NH3 formation mechanism as a case study.

Topics Covered

- Read species *ab-initio* data, reactions, lateral interactions, phases, reactor operating conditions, and desired units from a spreadsheet
- Write the CTI file that can be read by OpenMKM
- Write a YAML file that can be read by OpenMKM

Input Spreadsheet

All the data will be imported from the `./inputs/NH3_Input_data.xlsx` file. There are several sheets:

1. `units` contains the units that types of quantities should be written
2. `refs` contains *ab-initio* and experimental data for a handful of gas species to calculate references (optional)
3. `species` contains *ab-initio* data for each specie
4. `beps` contains Bronsted-Evans-Polanyi relationships for reactions (optional)
5. `reactions` contains elementary steps
6. `lateral_interactions` contains lateral interactions between species (optional)
7. `phases` contains phases for the species
8. `reactor` contains reactor operating conditions and solver tolerances

The `refs`, `beps` and `lateral_interactions` sheets can be deleted and the code written below should still work.

First, we change the working directory to the location of the Jupyter notebook.

[1] https://vlachosgroup.github.io/pMuTT/

OpenMKM: Features Summary

Mechanisms

- Gas phase only
- Surface only
- Gas phase + multiple surfaces

Additional Features

- Support for BEP relations
- Coverage dependent enthalpies
- Parametric Studies
- Sensitivity Analysis
- Reaction Path Analysis
- Parameter Estimation

Supported Reactor Models

- Batch
- CSTR
- PFR
- PFR with T-profile

Temperature Modes

- Isothermal
- Adiabatic
- Heat transfer
- Temperature ramp (For TPD)
- Custom T-Profile (PFR 1-D only)

Demo 1: OpenMKM I/O

See Jupyter Notebook at:

VlachosLab
AT THE UNIVERSITY OF DELAWARE

Demo 2: PFR-1D T-Profile

Ethane dehydrogenation MKM

See Example at:

VlachosLab

AT THE UNIVERSITY OF DELAWARE

Demo 3: Sensitivity Analysis

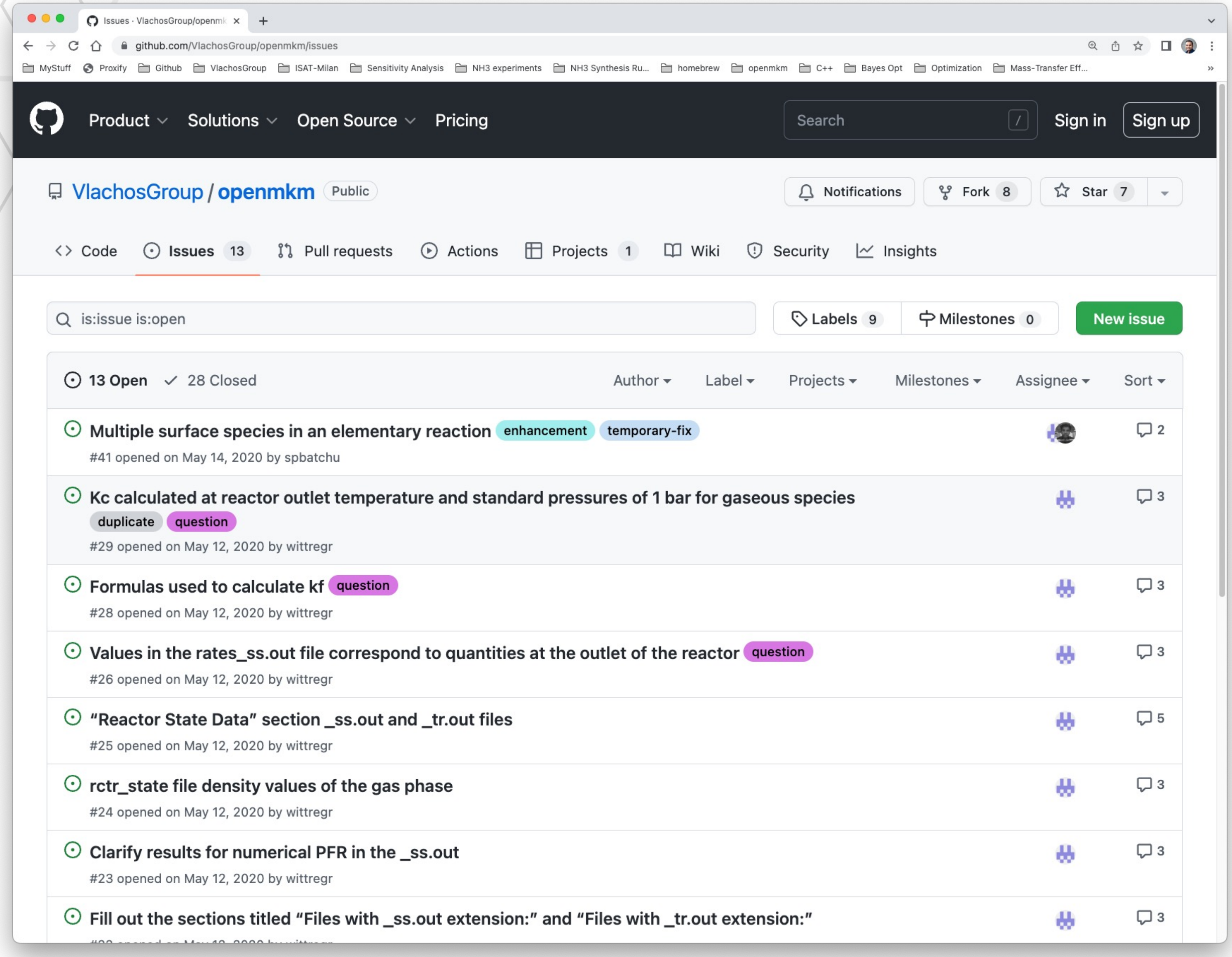
Ethane dehydrogenation MKM

See Example at:

VlachosLab
AT THE UNIVERSITY OF DELAWARE

RAPID
Transforming Process Industries

Active Open-Source Development



Issues · VlachosGroup/openmkm

github.com/VlachosGroup/openmkm/issues

MyStuff | Proxify | Github | VlachosGroup | ISAT-Milan | Sensitivity Analysis | NH3 experiments | NH3 Synthesis Ru... | homebrew | openmkm | C++ | Bayes Opt | Optimization | Mass-Transfer Eff...

Product | Solutions | Open Source | Pricing | Search | Sign in | Sign up

VlachosGroup / openmkm Public

Notifications | Fork 8 | Star 7

Code | Issues 13 | Pull requests | Actions | Projects 1 | Wiki | Security | Insights

is:issue is:open | Labels 9 | Milestones 0 | New issue

13 Open | 28 Closed | Author | Label | Projects | Milestones | Assignee | Sort

- Multiple surface species in an elementary reaction enhancement temporary-fix — 2
  #41 opened on May 14, 2020 by spbatchu
- Kc calculated at reactor outlet temperature and standard pressures of 1 bar for gaseous species duplicate question — 3
  #29 opened on May 12, 2020 by wittregr
- Formulas used to calculate kf question — 3
  #28 opened on May 12, 2020 by wittregr
- Values in the rates_ss.out file correspond to quantities at the outlet of the reactor question — 3
  #26 opened on May 12, 2020 by wittregr
- “Reactor State Data” section _ss.out and _tr.out files — 5
  #25 opened on May 12, 2020 by wittregr
- rctr_state file density values of the gas phase — 3
  #24 opened on May 12, 2020 by wittregr
- Clarify results for numerical PFR in the _ss.out — 3
  #23 opened on May 12, 2020 by wittregr
- Fill out the sections titled “Files with _ss.out extension:” and “Files with _tr.out extension:” — 3

- Vlachos Group active on GitHub for OpenMKM software development.
- Users can report issues and provide feature requests.
- New features can be added.
- Bugs are typically fixed rapidly.

dei.udel.edu/rapid

Acknowledgements

1. Department of Energy
2. RAPID (AIChE)
3. State of Delaware

Dr. Jeffrey Frey, HPC
Kelly Walker – Logos
Jaynell Keely - Logos

DELAWARE ENERGY INSTITUTE

VlachosLab
AT THE UNIVERSITY OF DELAWARE

UNIVERSITY OF DELAWARE