**Visualization Tool – CHEMKIN**

The following is the outline for the documentation:

1. Accessing the files
2. Modifications to input files
3. Output files
4. Bugs
5. Post-processing files
6. Verbose Mode

# **Accessing the files**

The necessary files have been uploaded to the [GitHub page](https://github.com/VlachosGroup/VisualizationTool-Chemkin). The files include two shell scripts (chemkin.sh and refineVisualization.sh) and one folder (graphviz).

1. chemkin.sh – shell script for running the chemkin code (**FILE**)
2. refineVisualization.sh – performing post-processing on the visualization output file (**FILE**)
3. graphviz – Open source code used for generating the visualizations (**FOLDER**)

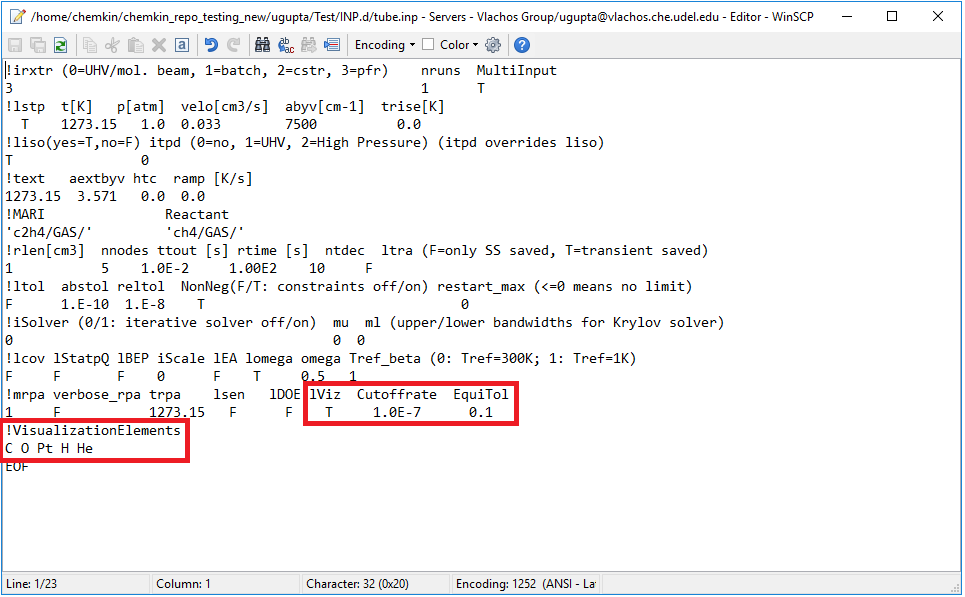
**All the above files/folders need to be copied into your working directory.**

A quick description about the above files.

1. chemkin.sh – The file has been modified to generate the visualization files
2. refineVisualization.sh – The file generates the visualization files without rerunning the chemkin code. The file uses Cutoffrate, EquilibriumTolerance, to refine the reaction network to be visualized. rpa\_visualizationMaster.out file, generated by the chemkin code in the OUT.d folder, is used as an input for this script.
3. graphviz – Graphviz, an open-source package specified in DOT language is used for the visualization. DOT is a graph description language that defines the graph involving nodes and edges. Graphviz manipulates and renders the graph. The folder contains executables necessary to generate the visualization files.

# **Modifications to input files**

An example input (tube.inp) has been put in the uploaded files.



**lViz** – boolean variable takes in input values T or F. This is a switch to turn on or off the visualization procedure when running Chemkin.

**Cutoffrate** – Uses a float variable. This is used to refine the reaction network to be visualized. Any reaction with a rate below the cutoffrate is not considered in the visualization. This is done to remove the insignificant reactions.

**EquilTol** – Uses a float variable between 0 – 0.5. This is used to identify reactions that are close to equilibrium. The value denotes the deviation in pei value from 0.5. In above case, any reaction falling within the range of 0.4 – 0.6 (0.5 ± EquilTol) pei values is considered equilibrated.

**VisualizationElements**\*– Requires elements as mentioned in gas.inp file. Any species having atleast one of the specified elements is considered in the visualization. The text case needs to be consistent with the gas.inp file.

# **Output files**

The code generates additional output files in OUT.d folder. These are:

1. **rpa\_visualization.out** – No splines and normalized rates
2. **rpa\_visualizationNormalized.out** – Splines and rates normalized by the net rate of initial reactant

The reaction rates of each edge are normalized by the net generation rate of the initial reactant. The thickness of each edge is also set based on the normalized reaction rate.

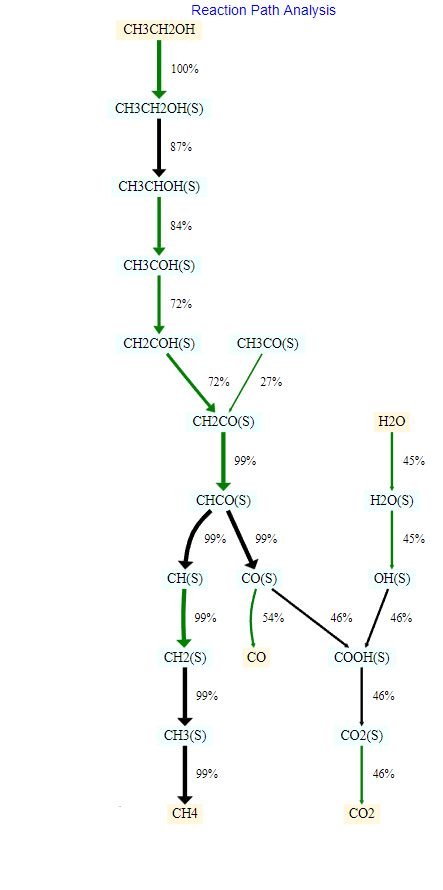
1. **rpa\_visualizationMaxRateNormalized.out** – Splines and rates normalized by the maximum reaction rate

The reaction rates of each edge are normalized by the maximum reaction rate in the reaction network. The thickness of each edge is also set based on the normalized reaction rate.

1. **rpa\_visualizationMaster.out**

The master file stores all the reaction rates and interconnections between species to be used post-simulation for refining the visualization.

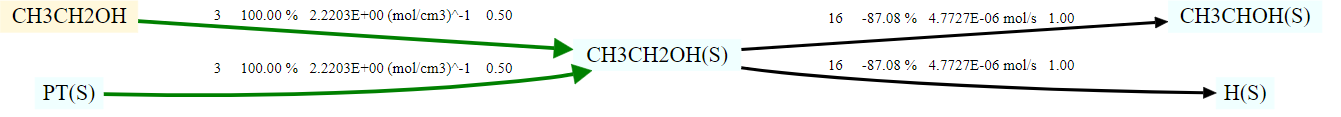
These files are used as input for Graphviz. **The output files from graphviz are stored in folder Species.d/.** To view the visualization, one needs to copy the Species.d folder onto local drive and open rpa\_visualizationNormalized.svg in any internet browser (Google Chrome) or svg editors (Inkscape or Photoshop). The output (with elements C and O) would look like the following:



The species are represented as nodes with their respective strings. The edges represent a reaction between two species and the net rate of the reaction defines the direction of the arrow. The edges are divided into three classes:

1. **Equilibrated reactions** – Represented in Green color
2. **Gas Phase reactions** – Represented in Red color
3. **Surface reactions** – Represented in Black color

Each node in the above overall reaction network can be clicked to see the reactions that the species participates in. For example, if you click on “CH3CH2OH (S)” shown below, you will see species that generate the selected species on the left and species that are generated by the selected species on the right.



For non-equilibrium reactions, we specify the following details (obtained from Chemkin):

1. Reaction Number
2. % Prod/Cons for the specific reaction
3. Net-rate of the reaction
4. Partial Equilibrium Index (pei)

For equilibrated reactions, we specify the following details (obtained from Chemkin):

1. Reaction Number
2. % Prod/Cons for the specific reaction
3. Equilibrium constant for the specific reaction (with units)
4. Partial Equilibrium Index (pei)

# **Bugs**

Running chemkin.sh might result in an error initially with the error being:

-bash: ./chemkin.sh: /bin/bash^M: bad interpreter: No such file or directory

Run the following command to correct the above error:

sed –i –e ‘s/\r$//’ chemkin.sh

# **Post-processing files**

We generate a master file named **rpa\_visualizationMaster**.out. This contains all the information necessary to generate visualization files so that the user does not require running chemkin again. The inputs in this file include cutoffrate and equilibrium tolerance. One can modify these values and run the **refineVisualization**.sh script to generate **RefinedVisualization**.svg that can be visualized similarly to the earlier files.

# **Verbose mode**

We also generate multiple master files for each node when the verbose mode is turned on in Chemkin. The user will need to modify the path to the master file on line 20 of **refineVisualization**.sh file. The individual species files are still generate for the end node.

For using the refineVisualization.sh file, use the following commands:

chmod +x refineVisualization.sh – convert the file into an executable

./refineVisualization.sh – to execute the file