A Tutorial on Chemkin

Niket Kaisare
(11-12-2007)
Indian Institute of Technology – Madras

Original Presentation: Dr. Ashish Mhadeshwar (June 2004)

Why use Chemkin?

1. Efficient handling of large reaction mechanisms
2. Generalized framework, simple and standard inputs
3. Minimization of common mistakes
4. Common platform for distributing new products
Requirements
We intend to solve a combined flow-diffusion-reaction problem

1. Reaction stoichiometry and rate constants
2. Reaction thermo-chemistry
3. Thermodynamic properties
4. Transport properties (diffusion, viscosity and thermal conductivity)

Example: Ammonia Decomposition on Ruthenium

Matrix of stoichiometric coefficients

<table>
<thead>
<tr>
<th>Reaction</th>
<th>R₁</th>
<th>R₂</th>
<th>R₃</th>
<th>R₄</th>
<th>R₅</th>
<th>R₆</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. N₂ + 2* ↔ 2N*</td>
<td>N*</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. H₂ + 2* ↔ 2H*</td>
<td>H*</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3. NH₄⁺ + * ↔ N⁺ + H⁺</td>
<td>NH⁺</td>
<td>-1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. NH₂⁺ + * ↔ NH⁺ + H⁺</td>
<td>NH₂⁺</td>
<td>-1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. NH₃⁺ + * ↔ NH₂⁺ + H⁺</td>
<td>NH₃⁺</td>
<td>-1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. NH₃ + * ↔ NH₃⁺</td>
<td>NH₃</td>
<td></td>
<td></td>
<td>-1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Kinetic parameters for each reaction

\[ r_j = A_{ij} \left( \frac{T}{T_{ref}} \right)^{n_j} e^{-\frac{E_j}{RT}} \prod a_{ij}^{m_j} \]

Reaction thermo-chemistry \( \Delta H, \Delta G, \Delta S \)

Thermodynamic and Transport properties \( c_p, \hat{H}, \rho, k_g, D_{ik}, \mu \)
Example: Ammonia Decomposition on Ruthenium

Matrix of stoichiometric coefficients

<table>
<thead>
<tr>
<th>Reaction</th>
<th>( R_1 )</th>
<th>( R_2 )</th>
<th>( R_3 )</th>
<th>( R_4 )</th>
<th>( R_5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ( \text{N}_2 + 2^* \leftrightarrow 2N^* )</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. ( \text{H}_2 + 2^* \leftrightarrow 2H^* )</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

1. Prone to errors
2. Detection of errors and debugging is not straightforward
3. Adding / removing reactions from mechanism is cumbersome

Reaction thermo-chemistry \( \Delta H, \Delta G, \Delta S \)

Thermodynamic and Transport properties
- \( c_p, \dot{H}, \rho \)
- \( k_g, D_{lk}, \mu \)

Utilities

AURORA/SENKIN
AURORA predicts the time-evolution or steady state of a well mixed reactor.

CRESLAF simulates laminar, chemically reacting, boundary-layer flow in cylindrical or planar channels.
**Utilities**

- **EQUIL** calculates the equilibrium state of systems containing ideal gas mixtures or ideal solutions.

- **PLUG** simulates a plug-flow reactor with gas-phase and surface chemistry.

- **OPPDIF** predicts the structure of opposed-flow diffusion flames in cylindrical or planar coordinate systems.

- **PREMIX** predicts the structure of steady, laminar, 1D, pre-mixed flames.
**Utilities**

**SHOCK**: A program for predicting chemical behavior behind incident and reflected shock waves.

**SPIN**: A program for modeling one-dimensional, rotating-disk or stagnation-flow chemical-vapor-deposition reactors.

---

**Components of Chemkin Suite**

1. Chemkin
2. Surface Chemkin
3. Transport
Components of Chemkin Suite

- Chemistry Input
- Properties Input
- ChemKin Utilities
- Interpreter
- Library
- Link File
- Application

<table>
<thead>
<tr>
<th></th>
<th>CHEMKIN</th>
<th>SURFACE</th>
<th>TRANSPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interpreter</td>
<td>ckinterp.x</td>
<td>skinterp.x</td>
<td>tranfit.x</td>
</tr>
<tr>
<td>Library</td>
<td>cklb.f</td>
<td>sklb.f</td>
<td>tranlb.f</td>
</tr>
<tr>
<td>Chemistry Input</td>
<td>fort.15</td>
<td>surf.inp</td>
<td>---</td>
</tr>
<tr>
<td>Physical Properties Input</td>
<td>thermdat</td>
<td>trandat</td>
<td></td>
</tr>
<tr>
<td>Link File</td>
<td>cklk</td>
<td>sklk</td>
<td>tklk</td>
</tr>
</tbody>
</table>

Note: Surface Chemkin and Transport Libraries require cklk for species information

Example: Ammonia Decomposition on Ruthenium

1. CHEMKIN: To define gas phase species and thermodynamics
2. SURFACE CHEMKIN: To define surface species and reactions
3. TRANSPORT: To define gas phase transport properties (diffusion, viscosity and thermal conductivity) if applicable
Steps:

- Create *fort.15* (gas phase reaction mechanism)
  - Input of elements, species, and reactions in *fort.15*
  - Input of thermodynamic database: *thermdat*
  - Compile *ckinterp.f* to create *ckinterp.x*
  - Run *ckinterp.x* to create *fort.16* and *cklink* (*MUST READ fort.16*)
  - Compile *cklib.f* to create *cklib.o* and link with your code

1. Chemkin inputs: Sample *fort.15*

```
ELEMENTS
N  H  AR  RH
END

SPECIES
N2   H2
NH3  AR
END

REATIONS
END
```

5 elements

6 species

0 reactions

Chemkin peculiarity:

- Catalyst should be declared with gas phase species
2. Chemkin inputs: Thermodynamic database \textit{thermdat}

- **Low**
  - $300,000$
  - $1,000,000$
  - $5,000,000$

- **Mid**
  - $1,000,000$

- **High**
  - $5,000,000$

<table>
<thead>
<tr>
<th>Species</th>
<th>L 7/88N</th>
<th>1H</th>
<th>3</th>
<th>G 200,000</th>
<th>3500,000</th>
<th>1000,000</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>NH$_3$</td>
<td>3.85746029E+00</td>
<td>4.41437026E-03</td>
<td>-2.21481404E-06</td>
<td>5.23490188E-10</td>
<td>-4.72084164E-14</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-4.87591660E+04</td>
<td>2.27163806E+00</td>
<td>2.35677352E+00</td>
<td>8.98459677E-03</td>
<td>-7.12356269E-06</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.45919022E-09</td>
<td>1.43699548E-13</td>
<td>-4.83719697E+04</td>
<td>9.90105222E+00</td>
<td></td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

Chemkin peculiarity:

Fixed format that must be followed for every new entry

3. Check output file \textit{fort.16} for errors

Run \textit{ckinterp.x}. Two output files are created: \textit{cklink} and \textit{fort.16}

---

Defining Surface Conditions in Chemkin

**Steps:**

- Create \textit{surf.inp} (surface reaction mechanism)
  - Input surface species and reactions
- Input of thermodynamic database: \textit{thermdat}
- Compile \textit{skinterp.f} to create \textit{skinterp.x}
- Run \textit{skinterp.x} to create \textit{surf.out} and \textit{sklink} (MUST READ \textit{surf.out})
- Compile \textit{cklib.f} to create \textit{cklib.o} and link with your code
1. Surface Chemkin inputs: Sample surf.inp

[Chemkin input code]

Site density (mol/cm²S) → surface species → units → reaction → sticking keyword → Coverage dependent parameters → k = A₀ \left( \frac{T}{T_{ref}} \right)^n \exp \left( -\frac{E}{RT} \right)

1. Surface Chemkin inputs: Peculiarities in surf.inp

Site density is a 3 element vector (gas, surface, bulk). Surface site density, i.e., sden(2) is typically used in the calculations, other two are zero.

For every consumed vacant surface site, a bulk site is produced. However, for reactions in the category of A(S) + B(S) → C(S) + D(S), no vacant surface sites or bulk sites are involved.

Reactions are written in irreversible manner, due to unavailability of thermodynamic database (specific heats) for surface species.

Units of pre-exponential factors are dependent on the type of reaction in the mechanism. The pre-exponential factor should be defined such that the reaction rate is calculated in mol/cm²sec.
2. Surface Chemkin inputs: Thermodynamic database `thermdat`

| Surface | Temperature | Specific Volume | Specific Enthalpy | Specific Entropy | Specific Excess Entropy | Specific Excess Entropy
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RU(S)</td>
<td></td>
<td>S 300.00</td>
<td>200.00</td>
<td>1000.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>N(S)</td>
<td></td>
<td>S 300.00</td>
<td>3000.00</td>
<td>1000.00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Follows same format as earlier

Can be provided in surf.inp or thermdat

Thermodynamic inputs for surface species are usually not available. Put all specific heats as zero, since all heats of reactions are calculated using BOC

Thermodynamic consistency is therefore important

---

### Defining Surface Conditions in Chemkin

**Steps:**

- Input of transport database: `thermdat`
- Compile `tranfit.f` and `dmath.f` to create `tranfit.x`
- Run `tranfit.x` to create `tlink`
- Compile `tranlib.f` to create `tranlib.o` and link with your code
1. Transport inputs: Transport database \textit{trandat} (optional)

<table>
<thead>
<tr>
<th>species</th>
<th>molecular index</th>
<th>L-J potential well depth</th>
<th>rotational relaxation collision number</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR</td>
<td>0</td>
<td>136.500</td>
<td>0.000</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>71.400</td>
<td>0.000</td>
</tr>
<tr>
<td>C2</td>
<td>1</td>
<td>97.530</td>
<td>1.760</td>
</tr>
<tr>
<td>C2O</td>
<td>1</td>
<td>232.400</td>
<td>1.000 ! *</td>
</tr>
<tr>
<td>CN2</td>
<td>1</td>
<td>232.400</td>
<td>1.000 ! OIS</td>
</tr>
</tbody>
</table>

\textbf{Interfacing with Application codes}

Typical structure of Chemkin subroutines:

\texttt{subroutine ckcpbs(t, yk, ickwrk, rckwrk, cpmean)}

Name Input(s) Work arrays Output(s)

\textbf{Interfacing with Application codes: Chemkin Mnemonics}

- \texttt{ck} = Chemkin
- \texttt{sk} = Surface Chemkin
- \texttt{mc} = Transport
- \texttt{iwk} = integer work array
- \texttt{rwk} = real work array
- \texttt{cwk} = character work array
Interfacing with Application codes: Dimensionalize Work Arrays

! *** File identifiers
integer, parameter :: linc = 25, linksk = 26, linkmc = 35, lout = 6

! *** Dimensionalizing work arrays
integer, parameter :: leniwk = 6000, lenrwk = 6000, lencwk = 6000
integer, parameter :: lsiwk = 6000, lsrwk = 6000, lscwk = 6000
integer, parameter :: lenimc = 6000, lenrmc = 6000

integer            :: ickwrk(leniwk), iskwrk(lsiwk), imcwrk(lenimc)
double precision   :: rckwrk(lenrwk), rskwrk(lsrwk), rmcwrk(lenrmc)
character (len=16) :: cckwrk(lencwk), cskwrk(lscwk)

Interfacing with Application codes: Initialization

! *** Open chemkin link files
open (linc, file='cklink', form='unformatted')
open (linkmc, file='tplink', form='unformatted')
open (linksk, file='sklink', form='unformatted')

! *** Initialize chemkin
 call ckinit (leniwk, lenrwk, lencwk, linc, lout, ickwrk, rckwrk, cckwrk)
call mcinit (linkmc, lout, lenimc, lenrmc, imcwrk, rmcwrk)
call skinit (lenisk, lenrsk, lencsk, linksk, lout, iskwrk, rskwrk, cskwrk)

!.......tutorial
open (unit=394, file='tutorial.out', status='unknown')
write(394,*)'Inlet mole fractions'
write(394,39)xin
call ckxty(xin, ickwrk, rckwrk, yin)
write(394,*)'Inlet mass fractions'
write(394,39)yin

Inlet mole fractions
0.300000 0.600000 0.100000 0.000000
Inlet mass fractions
0.743363 0.106195 0.150442 0.000000
**Interfacing with Application codes: Using subroutines**

```fortran
write(394,'*') 'Initial coverages'
write(394,39) (act(ii),ii=kmax+1,kmax-1)
call skrat (p,temper,act,sden,iskwrk,rskwrk,sdot,sitdot)
write(394,'*') 'Rates of species production'
write(394,39)sdot
```

**Initial coverages**

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1000E-05</td>
<td>0.99990E+00</td>
<td>0.10000E-05</td>
<td>0.98000E-04</td>
<td></td>
</tr>
</tbody>
</table>

**Rates of species production**

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.53416E-05</td>
<td>-0.21331E-05</td>
<td>0.45925E-05</td>
<td>0.00000E+00</td>
<td>0.10678E-10</td>
</tr>
<tr>
<td>0.21331E-05</td>
<td>-0.45925E-05</td>
<td>0.24594E-05</td>
<td>-0.24594E-05</td>
<td></td>
</tr>
</tbody>
</table>

**Recent modifications in Surface Chemkin**

1. UBI-QEP implementation using a new subroutine (subroutine ubi) internally called from subroutine skrrop
2. Modification of sklib.f to incorporate
   - Molecular beam effects (subroutines skatcz, skden)
   - Large reaction mechanisms (subroutine skinu)
   - Replacement of temperature exponents using a new subroutine (subroutine skrbex)
   - Modification of Arrhenius temperature exponent formula (subroutine skrrop)
### Useful links

**Reaction Design:**  
http://www.reactiondesign.com

**History of chemkin:**  
http://egweb.mines.edu/rjkee/

**Overview of chemkin:**  

**Equilibrium calculations:**  
http://www.gaseq.co.uk/

**GRI-Mech:**  
http://www.me.berkeley.edu/gri_mech/

**Sensitivity, reaction path, and principal component analysis modules:**  
http://www.chem.leeds.ac.uk/Combustion/links.htm  
http://www.cstl.nist.gov/div836/836.03/xsenkplot/index.html  
http://www.chem.leeds.ac.uk/Combustion/kinalc.htm

---

### Useful links

**Cantera - MATLAB based Chemkin:**  
http://rayleigh.cds.caltech.edu/~goodwin/cantera/index.html

**Burcat’s extended thermodynamic database:**  
http://garfield.chem.elte.hu/Burcat/burcat.html

**Chemical reaction network toolbox:**  
ftp://ftp.che.rochester.edu/pub/feinberg/

**EGLIB – A multicomponent transport software:**  
http://www.cmap.polytechnique.fr/www.eglib/

**Caltech gas phase mechanism library:**  
http://www.galcit.caltech.edu/EDL/mechanisms/library/library.html

**Software aid for developing mechanisms (unit conversion, deleting species, etc.):**  
http://www.chem.leeds.ac.uk/Combustion/mechmod.htm
Useful links

Information about Equil, Aurora, Senkin, Premix, Spin, Creslaf, Surftherm:
http://www.me.umn.edu/courses/me8646/chemkin.html

A general link for chemical engineers:
http://www.che.ufl.edu/www-che/topics/software.html