TS2 Custom Rate Constants

Matt Dawson

May 13, 2021

1 Introduction

This document builds on the TS1 Custom Rate Constants document, adding treatments for custom rate constant functions used in the TS2 mechanism that are not also present in TS1.

2 Note on [M] in Troe reactions

Edit Clone	C2H2 + OH + M	0.65*GLYOXAL + 0.65*OH + 0.35*HCOOH + 0.35*HO2 + 0.35*CO + M	C2H2_OH_M	5.5e-30,0,8.3e-13,-2,0.6		40

Figure 1: Example of a Troe reaction in the Chemistry Café.

The species 'M' seems to be included in most (maybe all?) standard Troe reactions as a reactant and a product. However, the CAM preprocessor appears to ignore 'M' as a species, and instead includes it in the calculation of the rate constant:

```
/cam/src/chemistry/pp_trop_strat_mam4_ts2/m_spc_id.F90 (27,27)
      integer, parameter :: id_C2H2 = 25


/cam/src/chemistry/pp_trop_strat_mam4_ts2/m_spc_id.F90 (295,295)
      integer, parameter :: id_OH = 293


/cam/src/chemistry/pp_trop_strat_mam4_ts2/m_rxt_id.F90 (324,324)
      integer, parameter :: rid_C2H2_OH_M = 322


/cam/src/chemistry/pp_trop_strat_mam4_ts2/mo_setrxt.F90 (928,930)
      ko(:) = 5.5e-30_r8
      kinf(:) = 8.3e-13_r8 * itemp(:)**(-2._r8)
      call jpl( rate(:,322), m, 0.6_r8, ko, kinf, n )


/cam/src/chemistry/mozart/mo_jpl.F90 (9,39)
      subroutine jpl( rate, m, factor, ko, kinf, ncol )
!-----------------------------------------------------------------
!       ... Calculate JPL troe rate
!-----------------------------------------------------------------

      use shr_kind_mod, only : r8 => shr_kind_r8

      implicit none

!-----------------------------------------------------------------
!       ... Dummy args
!-----------------------------------------------------------------
      integer, intent(in)    ::   ncol
      real(r8), intent(in)   ::   factor
      real(r8), intent(in)   ::   ko(ncol)
      real(r8), intent(in)   ::   kinf(ncol)
      real(r8), intent(in)   ::   m(ncol)
      real(r8), intent(out)  ::   rate(ncol)

!-----------------------------------------------------------------
!       ... Local variables
!-----------------------------------------------------------------
```

```
      real(r8)  ::  xpo(ncol)

      xpo(:)  = ko(:) * m(:) / kinf(:)
      rate(:) = ko(:) / (1._r8 + xpo(:))
      xpo(:)  = log10( xpo(:) )
      xpo(:)  = 1._r8 / (1._r8 + xpo(:)*xpo(:))
      rate(:) = rate(:) * factor**xpo(:)

      end subroutine jpl
```

```
/cam/src/chemistry/pp_trop_strat_mam4_ts2/mo_prod_loss.F90 (197,198)
        loss(k,84) = (rxt(k,321)* y(k,57) +rxt(k,322)* y(k,293) + het_rates(k,25)) &
               * y(k,25)
```

Is it ok to always ignore 'M' as a species in Troe reactions (with it included in the rate constant calculation)?

3 usr_ISOPNO3_NOa and usr_ISOPNO3_NOn

```
/cam/src/chemistry/mozart/mo_usrrxt.F90 (1213,1231)
!-----------------------------------------------------------------
!       ... ISOPNO3_NOn Temp/Pressure Dependent Nitrate Yield
!-----------------------------------------------------------------
       if( usr_ISOPNO3_NOn_ndx > 0 ) then
          nyield = (1._r8-0.135_r8)/0.135_r8
          natom = 9.0_r8
          exp_natom = exp( natom )
          acorr = (2.0e-22_r8*exp_natom*2.45e19_r8)/(1._r8+((2.0e-22_r8* &
                      exp_natom*2.45e19_r8)/(0.43_r8*(298._r8*(1._r8/293._r8))**8._r8)))* &
                      0.41_r8**(1._r8/(1._r8+(log10((2.0e-22_r8*exp_natom*2.45e19_r8)/ &
                      (0.43_r8*(298._r8*(1._r8/293._r8))**8._r8)))**2._r8))
          aterm(:) = (2.0e-22_r8*exp_natom*m(:,k))/(1._r8+((2.0e-22_r8* &
                      exp_natom*m(:,k))/(0.43_r8*(298._r8*tinv(:))**8._r8)))* &
                      0.41_r8**(1._r8/(1._r8+(log10((2.0e-22_r8*exp_natom*m(:,k))/ &
                      (0.43_r8*(298._r8*tinv(:))**8._r8)))**2._r8))
          call comp_exp( exp_fac, 360._r8*tinv, ncol )
          rxt(:,k,usr_ISOPNO3_NOn_ndx) = 2.7e-12_r8 * exp_fac(:)*aterm(:)/(aterm(:)+acorr*nyield)
          rxt(:,k,usr_ISOPNO3_NOa_ndx) = 2.7e-12_r8 * exp_fac(:)*acorr*nyield/(aterm(:)+acorr*nyield)
       end if
```

This appears to be based on eqs (1)–(6) of [1], which define the rate constant for each branch as a function of four parameters (X, Y, Z, n):

$$
\begin{aligned}
k_{nitrate} &= \left(Xe^{-Y/T}\right)\left(\frac{A(T,[\mathrm{M}],n)}{A(T,[\mathrm{M}],n)+Z}\right) \\
k_{alkoxy} &= \left(Xe^{-Y/T}\right)\left(\frac{Z}{Z+A(T,[\mathrm{M}],n)}\right) \\
A(T,[\mathrm{M}],n) &= \frac{2\times10^{-22}e^{n}[\mathrm{M}]}{1+\frac{2\times10^{-22}e^{n}[\mathrm{M}]}{0.43(T/298)^{-8}}}0.41^{(1+[log(\frac{2\times10^{-22}e^{n}[\mathrm{M}]}{0.43(T/298)^{-8}})]^2)^{-1}}
\end{aligned}
\tag{1}
$$

where T is temperature (K) and [M] is the number density of air (molecules cm^{-3}). To retain the detail of the current implementation of these rate constant functions, Z is defined as a function of two parameters (α_0, n):

Table 1: Wennberg NO + RO_2 rate constant parameters by custom rate constant function name

Function name	X	Y	a_0	n
usr_ISOPB1O2_NO	2.7×10^{-12}	-360	0.14	6
usr_ISOPB4O2_NO	2.7×10^{-12}	-360	0.13	6
usr_ISOPD1O2_NO	2.7×10^{-12}	-360	0.12	6
usr_ISOPD4O2_NO	2.7×10^{-12}	-360	0.12	6
usr_ISOPZD1O2_NO	2.7×10^{-12}	-360	0.12	6
usr_ISOPZD4O2_NO	2.7×10^{-12}	-360	0.12	6
usr_ISOPNO3_NO	2.7×10^{-12}	-360	0.135	9
usr_MVKO2_NO	2.7×10^{-12}	-360	0.04	6
usr_MACRO2_NO	2.7×10^{-12}	-360	0.06	6
usr_IEPOXOO_NO	2.7×10^{-12}	-360	0.025	8
usr_ISOPN1DO2_NO	2.7×10^{-12}	-360	0.084	11
usr_ISOPN2BO2_NO	2.7×10^{-12}	-360	0.065	11
usr_ISOPN3BO2_NO	2.7×10^{-12}	-360	0.053	11
usr_ISOPN4DO2_NO	2.7×10^{-12}	-360	0.165	11
usr_ISOPNBNO3O2_NO	2.7×10^{-12}	-360	0.203	11
usr_ISOPNOOHBO2_NO	2.7×10^{-12}	-360	0.141	12
usr_ISOPNOOHDO2_NO	2.7×10^{-12}	-360	0.045	12
usr_NC4CHOO2_NO	2.7×10^{-12}	-360	0.021	11

$$Z(\alpha_0, n) = A(T = 293\text{K}, [\text{M}] = 2.45 \times 10^{19} \frac{\text{molec}}{\text{cm}^3}, n) \frac{(1 - \alpha_0)}{\alpha_0} \tag{2}$$

This reaction has been added to Music Box as `WENNBERG_NO_RO2`. The usr_ISOPNO3_NOn and usr_ISOPNO3_NOa rate constants can then be calculated according to the above equations with $X = 2.7 \times 10^{-12}$, $Y = -360$, $\alpha_0 = 0.135$, and $n = 9$, with usr_ISOPNO3_NOn corresponding to $k_{nitrate}$ and usr_ISOPNO3_NOa corresponding to k_{alkoxy}.

Similar custom rate constant functions are summarized in Table 1.

4 usr_ISOPZD1O2

```
/cam/src/chemistry/mozart/mo_usrrxt.F90 (1081,1089)
!-----------------------------------------------------------------
!       ... ISOPZD1O2 --> HPALD etc. Wennberg 2018 for rate
!-----------------------------------------------------------------
       if( usr_ISOPZD1O2_ndx > 0 ) then
          call comp_exp( exp_fac, -12200._r8*tinv, ncol )
          ko(:) = 5.05e15_r8 * exp_fac(:)
          call comp_exp( exp_fac, 1.e8_r8*tinv**3._r8, ncol )
          rxt(:,k,usr_ISOPZD1O2_ndx) = ko(:)*exp_fac(:)
       end if
```

This appears to be based on eq. (12) of [1]:

$$k_{tunneling} = Ae^{-B/T}e^{C/T^3} \tag{3}$$

with $A = 5.05 \times 10^{15}$, $B = 12200$, and $C = 1 \times 10^8$. This reaction has been added to Music Box as `WENNBERG_TUNNELING`.

5 usr_TERPAPAN_M

```
/cam/src/chemistry/mozart/mo_usrrxt.F90 (971,971)
       call comp_exp( exp_fac, -14000._r8*tinv, ncol )
```

```
/cam/src/chemistry/mozart/mo_usrrxt.F90 (1016,1025)
!-----------------------------------------------------------------
!       ... TERPAPAN + m --> TERPACO3 + no2 + m
!-----------------------------------------------------------------
       if( usr_TERPAPAN_M_ndx > 0 ) then
          if( tag_TERPACO3_NO2_ndx > 0 ) then
             rxt(:,k,usr_TERPAPAN_M_ndx) = rxt(:,k,tag_TERPACO3_NO2_ndx) * 1.111e28_r8 * exp_fac(:)
          else
             rxt(:,k,usr_TERPAPAN_M_ndx) = 0._r8
          end if
       end if
```

Following the same logic as for `usr_PBZNIT_M`, the `tag_TERPACO3_NO2_ndx` reaction is identified as:

```
{
  "type": "TROE",
  "k0_A": 9.7e-29,
  "k0_B": -5.6,
  "kinf_A": 9.3e-12,
  "N": 1.5,
  "reactants": {
    "TERPACO3": { },
    "NO2": { },
    "M": { }
  },
  "products": {
    "TERPAPAN": { },
    "M": { }
  }
},
```

and can be rearranged as a Troe reaction:

$$
\begin{aligned}
k &= \frac{k_0[\mathrm{M}]}{1 + k_0[\mathrm{M}]/k_{\mathrm{inf}}} F_C^{(1+1/N[log_{10}(k_0[\mathrm{M}]/k_{\mathrm{inf}})]^2)^{-1}} \\
k_0 &= A_0 e^{\left(\frac{C_0}{T}\right)} \left(\frac{T}{300}\right)^{B_0} \\
k_{inf} &= A_{inf} e^{\left(\frac{C_{inf}}{T}\right)} \left(\frac{T}{300}\right)^{B_{inf}}
\end{aligned}
\tag{4}
$$

where $F_C = 0.6$, $N = 1.5$, $A_0 = 9.7 \times 10^{-29} \times 1.111 \times 10^{28}$, $B_0 = -5.6$, $C_0 = -14000$, $A_{inf} = 9.3 \times 10^{-12} \times 1.111 \times 10^{28}$, $B_{inf} = 0$, and $C_{inf} = -14000$.

6 usr_TERPA2PAN_M

```
/cam/src/chemistry/mozart/mo_usrrxt.F90 (971,971)
       call comp_exp( exp_fac, -14000._r8*tinv, ncol )
```

```
/cam/src/chemistry/mozart/mo_usrrxt.F90 (1026,1035)
!-----------------------------------------------------------------
!       ... TERPA2PAN + m --> TERPA2CO3 + no2 + m
!-----------------------------------------------------------------
       if( usr_TERPA2PAN_M_ndx > 0 ) then
          if( tag_TERPA2CO3_NO2_ndx > 0 ) then
             rxt(:,k,usr_TERPA2PAN_M_ndx) = rxt(:,k,tag_TERPA2CO3_NO2_ndx) * 1.111e28_r8 * exp_fac(:)
```

```
      else
         rxt(:,k,usr_TERPA2PAN_M_ndx) = 0._r8
      end if
   end if
```

Following the same logic as for `usr_PBZNIT_M`, the `tag_TERPA2CO3_NO2_ndx` reaction is identified as:

```
{
  "type": "TROE",
  "k0_A": 9.7e-29,
  "k0_B": -5.6,
  "kinf_A": 9.3e-12,
  "N": 1.5,
  "reactants": {
    "TERPA2CO3": { },
    "NO2": { },
    "M": { }
  },
  "products": {
    "TERPA2PAN": { },
    "M": { }
  }
},
```

and can be rearranged as a Troe reaction:

$$
\begin{aligned}
k &= \frac{k_0[\mathrm{M}]}{1 + k_0[\mathrm{M}]/k_{\mathrm{inf}}} F_C^{(1+1/N[log_{10}(k_0[\mathrm{M}]/k_{\mathrm{inf}})]^2)^{-1}} \\
k_0 &= A_0 e^{\left(\frac{C_0}{T}\right)} \left(\frac{T}{300}\right)^{B_0} \\
k_{inf} &= A_{inf} e^{\left(\frac{C_{inf}}{T}\right)} \left(\frac{T}{300}\right)^{B_{inf}}
\end{aligned}
\tag{5}
$$

where $F_C = 0.6$, $N = 1.5$, $A_0 = 9.7 \times 10^{-29} \times 1.111 \times 10^{28}$, $B_0 = -5.6$, $C_0 = -14000$, $A_{inf} = 9.3 \times 10^{-12} \times 1.111 \times 10^{28}$, $B_{inf} = 0$, and $C_{inf} = -14000$.

7 usr_TERPA3PAN_M

```
/cam/src/chemistry/mozart/mo_usrrxt.F90 (971,971)
      call comp_exp( exp_fac, -14000._r8*tinv, ncol )


/cam/src/chemistry/mozart/mo_usrrxt.F90 (1036,1045)
!-----------------------------------------------------------------
!       ... TERPA3PAN + m --> TERPA3CO3 + no2 + m
!-----------------------------------------------------------------
      if( usr_TERPA3PAN_M_ndx > 0 ) then
         if( tag_TERPA3CO3_NO2_ndx > 0 ) then
            rxt(:,k,usr_TERPA3PAN_M_ndx) = rxt(:,k,tag_TERPA3CO3_NO2_ndx) * 1.111e28_r8 * exp_fac(:)
         else
            rxt(:,k,usr_TERPA3PAN_M_ndx) = 0._r8
         end if
      end if
```

Following the same logic as for `usr_PBZNIT_M`, the `tag_TERPA3CO3_NO2_ndx` reaction is identified as:

```
{
  "type": "TROE",
  "k0_A": 9.7e-29,
  "k0_B": -5.6,
  "kinf_A": 9.3e-12,
  "N": 1.5,
  "reactants": {
    "TERPA3CO3": { },
    "NO2": { },
    "M": { }
  },
  "products": {
    "TERPA3PAN": { },
    "M": { }
  }
},
```

and can be rearranged as a Troe reaction:

$$
\begin{aligned}
k &= \frac{k_0[\mathrm{M}]}{1 + k_0[\mathrm{M}]/k_{\mathrm{inf}}} F_C^{(1+1/N[log_{10}(k_0[\mathrm{M}]/k_{\mathrm{inf}})]^2)^{-1}} \\
k_0 &= A_0 e^{\left(\frac{C_0}{T}\right)} \left(\frac{T}{300}\right)^{B_0} \\
k_{inf} &= A_{inf} e^{\left(\frac{C_{inf}}{T}\right)} \left(\frac{T}{300}\right)^{B_{inf}}
\end{aligned}
\tag{6}
$$

where $F_C = 0.6$, $N = 1.5$, $A_0 = 9.7 \times 10^{-29} \times 1.111 \times 10^{28}$, $B_0 = -5.6$, $C_0 = -14000$, $A_{inf} = 9.3 \times 10^{-12} \times 1.111 \times 10^{28}$, $B_{inf} = 0$, and $C_{inf} = -14000$.

8 usr_ISOPZD4O2

```
/cam/src/chemistry/mozart/mo_usrrxt.F90 (1090,1098)
!-----------------------------------------------------------------
!       ... ISOPZD4O2 --> HPALD etc. Wennberg 2018 for rate
!-----------------------------------------------------------------
       if( usr_ISOPZD4O2_ndx > 0 ) then
          call comp_exp( exp_fac, -7160._r8*tinv, ncol )
          ko(:) = 2.22e9_r8 * exp_fac(:)
          call comp_exp( exp_fac, 1.e8_r8*tinv**3._r8, ncol )
          rxt(:,k,usr_ISOPZD4O2_ndx) = ko(:)*exp_fac(:)
       end if
```

This appears to be a `WENNBERG_TUNNELING` reaction:

$$
k_{tunneling} = Ae^{-B/T}e^{C/T^3} \tag{7}
$$

with $A = 2.22 \times 10^9$, $B = 7160$, and $C = 1 \times 10^8$.

9 References

References

[1] Paul O. Wennberg et al. “Gas-Phase Reactions of Isoprene and Its Major Oxidation Products”. In: *Chemical Reviews* 118.7 (2018). PMID: 29522327, pp. 3337–3390. DOI: 10.1021/acs.chemrev.7b00439. eprint: https://doi.org/10.1021/acs.chemrev.7b00439. URL: https://doi.org/10.1021/acs.chemrev.7b00439.