Chemical mechanisms and KPP

Ravan Ahmadov

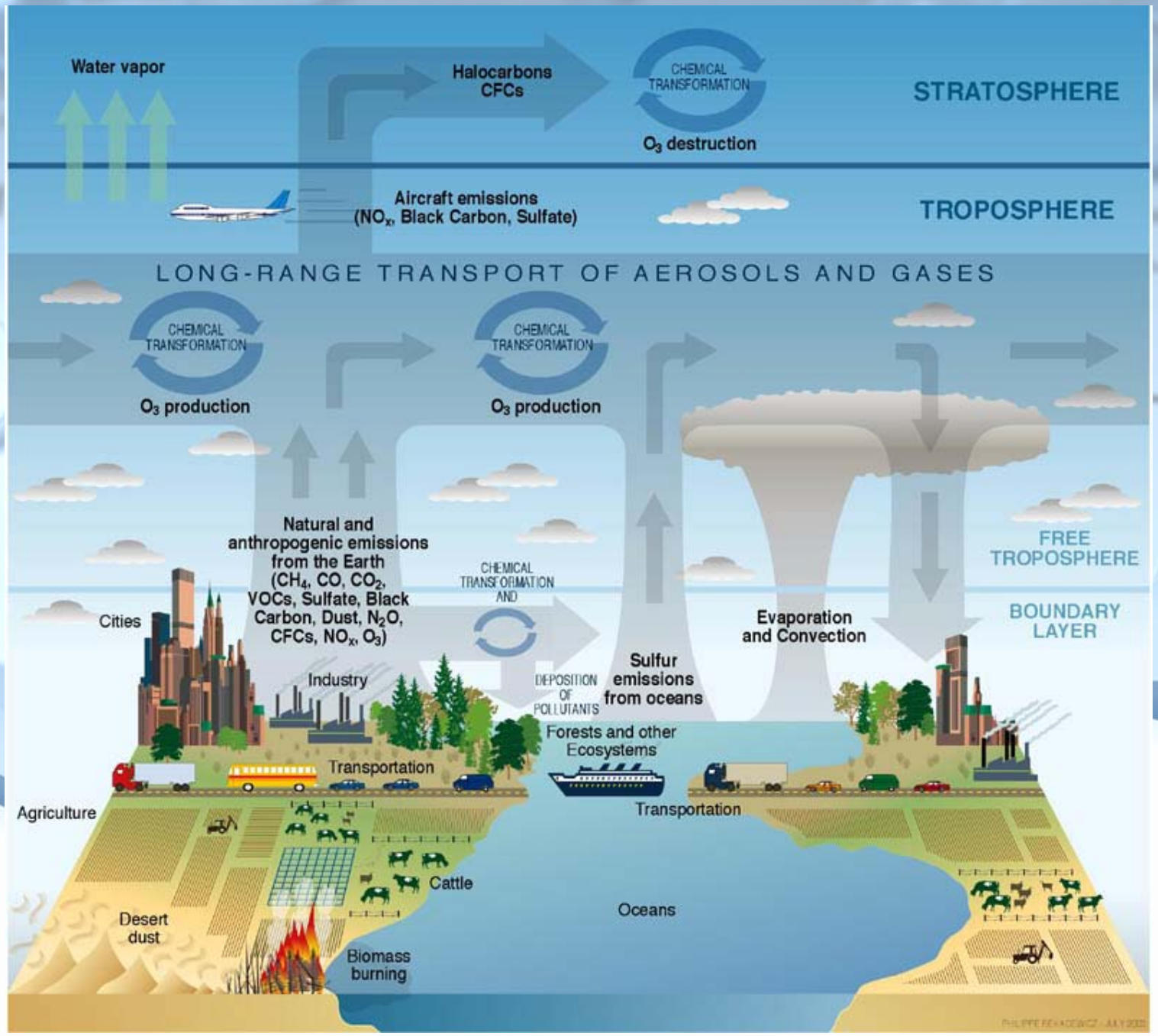
Cooperative Institute for Research in Environmental Sciences, CU Boulder

Earth System Research Laboratory, NOAA, Boulder

Acknowledgements: S.W.Kim, M.Salzmann, S.McKeen

Outline

- Chemical mechanisms in the WRF-Chem model
- Kinetic PreProcessor (KPP) and WRF-Chem KPP Coupler (WKC)
- Adding chemical mechanisms to the WRF-Chem model using KPP
- Suggestions


Water vapor
Halocarbons
CFCs
CHEMICAL TRANSFORMATION
O_3 destruction
STRATOSPHERE
Aircraft emissions
(NO_x, Black Carbon, Sulfate)
TROPOSPHERE
LONG-RANGE TRANSPORT OF AEROSOLS AND GASES
CHEMICAL TRANSFORMATION
O_3 production
CHEMICAL TRANSFORMATION
O_3 production
FREE TROPOSPHERE
Natural and anthropogenic emissions from the Earth (CH_4, CO, CO_2, VOCs, Sulfate, Black Carbon, Dust, N_2O, CFCs, NO_x, O_3)
CHEMICAL TRANSFORMATION AND
BOUNDARY LAYER
Evaporation and Convection
Cities
Industry
DEPOSITION OF POLLUTANTS
Sulfur emissions from oceans
Forests and other Ecosystems
Transportation
Transportation
Agriculture
Cattle
Oceans
Desert dust
Biomass burning

Main gas phase chemistry mechanisms in WRF-Chem3.3

- Regional Acid Deposition Model, 2nd generation (RADM2)
- Regional Atmospheric Chemistry Mechanism (RACM)
- RACM - Mainz Isoprene Mechanism (RACM-MIM)
- RACM - Earth System Research Laboratory (RACM-ESRL), *updated RACM-MIM*
- Carbon Bond Mechanism (CB4)
- Carbon-Bond Mechanism version Z (CBMZ)
- Model of Ozone and Related Chemical Tracers (MOZART)
- Nonmethane Hydrocarbons Chemistry (NMHC9)
- Statewide Air Pollution Research Center (SAPRC99)
- *RACM2 - ongoing work*

different implementations, coupled to different aerosol schemes and aqueous chemistry, suitable for different applications ranging from regional air quality to global atmospheric chemistry simulations (no halogen chemistry yet!)

Some references

- Stockwell, W. R., et al. (1997), A new mechanism for regional atmospheric chemistry modeling, *J. Geophys. Res.-Atmos., 102(D22), 25847-25879.*
- Stockwell, W. R., et al. (1990), The 2nd generation regional acid deposition model chemical mechanism for regional air-quality modeling, *J. Geophys. Res.-Atmos., 95(D10), 16343-16367.*
- Horowitz, L. W., et al. (2003), A global simulation of tropospheric ozone and related tracers: Description and evaluation of MOZART, version 2, *J. Geophys. Res., 108(D24), 4784, doi:4710.1029/2002JD002853.*
- Gross, A., and W. R. Stockwell (2003), Comparison of the EMEP, RADM2 and RACM mechanisms, *Journal of Atmospheric Chemistry, 44(2), 151-170.*
- Geiger, H., et al. (2003), The tropospheric degradation of isoprene: an updated module for the regional atmospheric chemistry mechanism, *Atmos. Environ., 37(11), 1503-1519.*
- Luecken, D. J., et al. (2008), Effects of using the CB05 vs. SAPRC99 vs. CB4 chemical mechanism on model predictions: Ozone and gas-phase photochemical precursor concentrations, *Atmos. Environ., 42(23), 5805-5820.*
- *Cai et al. (2011),* Photochemical Modeling in California with Two Chemical Mechanisms: Model Intercomparison and Response to Emission Reductions, *J. Air & Waste Manage. Assoc.*
- Peckham S. et al. (2011), WRF-CHEM 3.3 User's Guide.

Chemistry mechanisms in WRF-Chem3.3

Chemical mechanisms	*Fixed versions*	*KPP*	*Coupled to the aerosol schemes*
RADM2	**Yes**	**Yes**	**MADE/SORGAM, GOCART**
RACM	**None**	**Yes**	**MADE/SORGAM, GOCART**
RACM-MIM	**None**	**Yes**	**None**
RACM-ESRL	**None**	**Yes**	**MADE/SORGAM**
CB4	**None**	**Yes**	**None**
CBMZ	**Yes**	**Yes**	**MOSAIC**
MOZART	**None**	**Yes**	**GOCART**
SAPRC99	**None**	**Yes**	**MOSAIC**
NMHC9	**None**	**Yes**	**None**

Chemistry options in WRF-Chem3.3

chem_opt =0 **no chemistry**

= 1 RADM2 chemical mechanism - no aerosols

= 2 RADM2 chemical mechanism and MADE/SORGAM aerosols

No indirect effect To have radiative feed back with the chemistry/aerosols use ra_sw_physics = 2 (Goddard shortwave scheme). For dust and sea salt use dust_opt=2, seas_opt=2

= 5 CBMZ chemical mechanism with Dimethylsulfide

= 6 CBMZ chemical mechanism without DMS

= 7 CBMZ chemical mechanism (chem_opt=6) and MOSAIC using 4 sectional aerosol bins

No indirect effect To have radiative feed back with the chemistry/aerosols use , ra_sw_physics = 2, for dust and seasalt use dust_opt=2, seas_opt=2

= 8 CBMZ chemical mechanism (chem_opt=6) and MOSAIC using 8 sectional aerosol bins.

No indirect effect To have radiative feed back with the chemistry/aerosols use, ra_sw_physics = 2, for dust and seasalt use dust_opt=2, seas_opt=2.

= 9 CBMZ chemical mechanism (chem_opt=6) and MOSAIC using 4 sectional aerosol bins) including some aqueous reactions

For direct and indirect effect use: phot_opt=2; ra_sw_physics=2; progn=1; mp_physics=2; aer_ra_feedback=1;wetscav_onoff=1; cldchem_onoff=1,

For dust and sea salt use dust_opt=2, seas_opt=2

= 10 CBMZ chemical mechanism (chem_opt=6) and MOSAIC using 8 sectional aerosol bins) including some aqueous reactions

For direct and indirect effect use: phot_opt=2; ra_sw_physics=2; progn=1; mp_physics=2; aer_ra_feedback=1; wetscav_onoff=1; cldchem_onoff=1

For dust and seasalt use dust_opt=2, seas_opt=2.

= 11 RADM2 chemical mechanism and MADE/SORGAM aerosols including some aqueous reactions

For direct and indirect effect use: phot_opt=2; ra_sw_physics=2; progn=1; mp_physics=2; aer_ra_feedback=1; wetscav_onoff=1; cldchem_onoff=1

For dust and seasalt use dust_opt=2, seas_opt=2.

....

KPP in WRF-Chem

Kinetic PreProcessor (KPP) reads chemical reactions and rate constants from ASCII input files and automatically generates code for chemistry integration using the Rosenbrok solver

No KPP for aerosols!

Advantages:

- **less time consuming than manual coding**
- **less error prone**
- **numerically efficient**
- **flexibility in updating mechanism with additional species and equations**
- **suitable for adjoint code development**

References:

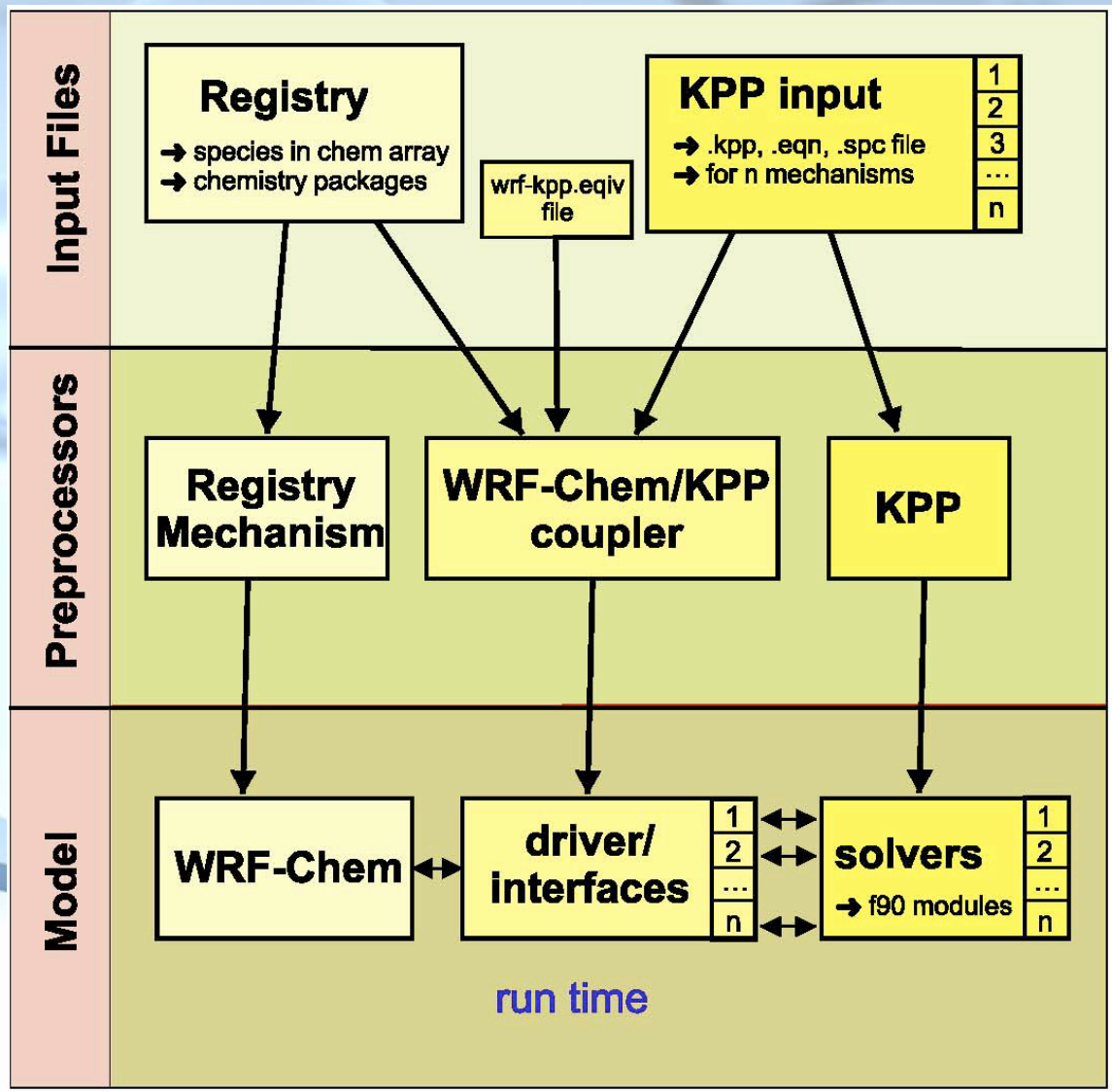
• Damian, V., et al. (2002), The kinetic preprocessor KPP - a software environment for solving chemical kinetics, *Comput. Chem. Eng., 26(11), 1567-1579*.

• Sandu, A., and R. Sander (2006), Technical note: Simulating chemical systems in Fortran90 and Matlab with the Kinetic PreProcessor KPP-2.1, *Atmos. Chem. Phys., 6, 187-195*.

• Verwer, J., Spee, E., Blom, J. G., and Hunsdorfer, W. (1999), A second order Rosenbrock method applied to photochemical dispersion problems, *SIAM Journal on Scientific Computing, 20, 1456–1480*.

• www.mpch-mainz.mpg.de/~salzmann/my_home/sub/wkc.html

WKC (WRF-Chem/KPP Coupler) ($WRF-CHEM/chem/KPP/util/wkc/)



WRF-CHEM3.2 User's guide, 2010

racm_esrlsorg.eqiv file:

```
! use this file for species that have different
! names in WRF and KPP
!
! Currently case sensitive
!
! left column        right column
! name in WRF       name in KPP
rpho                 pho
```

Input files of KPP

***.spc file**

Definition of chemical species as variable or fixed value.

***.eqn file**

Writing chemical reactions in kpp format

***.kpp file**

Model description, computer language, precision, integrator (e.g. Rosenbrock solver) etc.

***.def file**

User defined functions

(also check $WRF-CHEM/chem/KPP/kpp/kpp-2.1/util/ WRF_conform/ UserRateLaws.f90)

Pre-defined variables in WKC

	KPP equation file	Equation file units	Registry
Photolysis rate	J(Pj_no2)	s^{-1}	ph_no2
Temperature	TEMP	K	t_phy
Third body concentration	C_M	(molecular moist air) cm^{-3}	Calculated from density
Water vapor concentration	C_H20	Molecules cm^{-3}	Calculated from qvapor

WRF-CHEM3.2 User's guide, 2010

How to add a new KPP chemistry mechanism to WRF-Chem?

1) Add a new mechanism to $WRF-CHEM/Registry/registry.chem
2) Add new variables to the registry if necessary (new species, namelist options etc.)
3) Create a subdirectory in $WRF-CHEM/chem/KPP/mechanisms/
4) Create new KPP files with new reactions, rates and species
5) Modify some $WRF-CHEM/chem/*.F (e.g. chem_driver.F) programs in order to include a new chemical mechanism
6) Modify $WRF-CHEM/chem/convert_emiss.F for new species
7) Compile a new WRF-CHEM code and run using updated emission files

KPP requires the UNUX tool programs flex, yacc, and sed to be installed on your system before compiling the code !

setenv FLEX_LIB_DIR = /usr/local/lib

setenv WRF_KPP =1

Example: Adding a new chemistry mechanism to WRF-Chem for the state of the art SOA parameterization

In order to implement the new SOA mechanism we need to modify the RACM_ESRLSORG gas chemistry:

- **Separate MBO from internal alkenes - OLI species**
- **Add SESQ (biogenic VOC) reactions**

Adding the new chemistry package and species to $WRF-CHEM/Registry/registry.chem

state real sesq ikjftb chem 1 - irhusdf=(bdy_interp:dt) "sesq" "SESQ concentration" "ppmv"

state real mbo ikjftb chem 1 - irhusdf=(bdy_interp:dt) "mbo" "MBO concentration" "ppmv"

The new package racm_esrlsoa01_kpp chem_opt==108

chem:

so2,sulf,no2,no,o3,hno3,h2o2,ald,hcho,op1,op2,paa,ora1,ora2,n2o5,no3,pan,hc3,hc5,hc8,eth,co,ete,olt, oli,tol,xyl,aco3,tpan,hono,hno4,ket,gly,mgly,dcb,onit,csl,iso,co2,ch4,udd,hket,api,lim,dien,macr,hace, ishp,ison,mahp,mpan,nald,sesq,mbo,ho,ho2,

so4aj,so4ai,nh4aj,nh4ai,no3aj,no3ai, asoa1j,asoa1i,asoa2j,asoa2i,asoa3j,asoa3i,asoa4j,asoa4i, bsoa1j,bsoa1i,bsoa2j,bsoa2i,bsoa3j,bsoa3i,bsoa4j,bsoa4i, orgpaj,orgpai,ecj,eci,p25j,p25i,antha,seas,soila,nu0,ac0,corn

All species within "chem" array are advected and mixed by WRF!

Some species are part of the “misc” array

$WRF-CHEM/Registry/registry.chem:

 # non-transported radical species for RACM

```
state    real  addt       ikj     misc      1       -      r       "addt"            "Radicals" "ppm"
state    real  addx       ikj     misc      1       -      r       "addx"            "Radicals" "ppm"
state    real  addc       ikj     misc      1       -      r       "addc"            "Radicals" "ppm"
state    real  etep       ikj     misc      1       -      r       "etep"            "Radicals" "ppm"
state    real  oltp       ikj     misc      1       -      r       "oltp"            "Radicals" "ppm"
state    real  olip       ikj     misc      1       -      r       "olip"            "Radicals" "ppm"
state    real  cslp       ikj     misc      1       -      r       "cslp"            "Radicals" "ppm"
state    real  limp       ikj     misc      1       -      r       "limp"            "Radicals" "ppm"
state    real  hc5p       ikj     misc      1       -      r       "hc5p"            "Radicals" "ppm"
state    real  hc8p       ikj     misc      1       -      r       "hc8p"            "Radicals" "ppm"
state    real  tolp       ikj     misc      1       -      r       "tolp"            "Radicals" "ppm"
state    real  xylp       ikj     misc      1       -      r       "xylp"            "Radicals" "ppm"
state    real  apip       ikj     misc      1       -      r       "apip"            "Radicals" "ppm"
state    real  isop       ikj     misc      1       -      r       "isop"            "Radicals" "ppm"
state    real  hc3p       ikj     misc      1       -      r       "hc3p"            "Radicals" "ppm“
```

……

Make a new subdirectory in $WRFV-CHEM/chem/KPP/mechanisms/

The name of this directory should be the same as the package name in the Registry without the "_kpp" suffix.

$WRF-CHEM/chem/KPP/mechanisms/racm_esrlsoa01

Copy these files from chem/KPP/mechanisms/racm_esrlsorg to the new subdirectory and rename them except "atoms_red":

atoms_red

racm_esrlsoa01.def

racm_esrlsoa01.eqn

racm_esrlsoa01.kpp

racm_esrlsoa01.spc

racm_esrlsoa01_wrfkpp.equiv

racm_esrlsoa01.def file

If necessary update equation sets and rate constants etc.

```
#include atoms_red
#include ./racm_esrlsoa01.spc
#include ./racm_esrlsoa01.eqn

#INLINE F90_RATES
REAL(KIND=dp) FUNCTION k46( TEMP, C_M )
   REAL(KIND=dp), INTENT(IN) :: temp, c_m
   REAL(KIND=dp) :: k0, k2, k3

  k0=2.4E-14_dp * EXP(460._dp/TEMP)
  k2=2.7E-17_dp * EXP(2199._dp/TEMP)
  k3=6.5E-34_dp * EXP(1335._dp/TEMP) * c_m

  k46=k0+k3/(1+k3/k2)

END FUNCTION k46
.....
```

racm_esrlsoa01.spc file:

```
#DEFVAR
 O3 =IGNORE ;
 H2O2 =IGNORE ;
 ……
SESQ =IGNORE ;
MBO  =IGNORE ;
```

racm_esrlsoa01.kpp file:

```
#MODEL racm_esrlsoa01
#LANGUAGE Fortran90
#DOUBLE ON
#INTEGRATOR WRF_conform/rosenbrock
……
```

racm_esrlsoa01.eqn file:

```
#EQUATIONS {} ;                    photolysis:

{001:J01} NO2+hv=O3P+NO                    : j(Pj_no2)  ;

{002:J02} O3+hv=O1D{+O2}                   : j(Pj_o31d) ;

{003:J03} O3+hv=O3P{+O2}                   : j(Pj_o33p)

{004:J04} HONO+hv=HO+NO                    : j(Pj_hno2) ;

{005:J05} HNO3+hv=HO+NO2                   : j(Pj_hno3) ;

.....................................................................................

                              chemical reactions:

{024:001} O3P+M{O2}=O3             : (C_M *6.00D-34*(TEMP/300.0)**(-2.4)) ;

{025:002} O3P+O3=M {2O2}           : ARR2( 8.00D-12 , 2060.0_dp, TEMP) ;

{026:003} O1D + M = O3P            : .78084*ARR2(2.15D-11 , -110.0_dp, TEMP) +

                                       .20946*ARR2( 3.30D-11 , -55.0_dp , TEMP ) ;

{027:004} O1D+H2O=HO+HO            : ARR2( 1.63D-10 , -60.0_dp, TEMP ) ;

{028:005} O3+HO=HO2{+O2}           : ARR2( 1.70D-12 , 940.0_dp, TEMP ) ;
```

Adding new reactions to racm_esrlsoa01.eqn file

{245:222} SESQ+HO=0.36 KET+0.3 HCHO+0.05 ORA1+0.19 OLIP : *2.52D-10 ;*

{246:223} SESQ+O3=0.51 HCHO+0.85 ALD+0.039 ORA1+0.23 KET+0.053 ORA2
+0.63 HO : *5.60D-16 ;*

{247:224} SESQ+NO3=0.9 OLNN+0.10 OLND+0.9 MACR : *2.20D-11 ;*

{248:225} MBO+HO=OLIP : *ARR2(1.33D-11 , -500.0_dp, TEMP) ;*

{249:226} MBO+NO3=0.11 OLNN+0.89 OLND : *ARR2(8.64D-13 , -450.0_dp, TEMP) ;*

{250:227} MBO+O3=0.02 HCHO+0.99 ALD+0.16 KET+0.30 CO+0.011 H2O2
+0.14 ORA2+0.07 CH4+0.22 HO2+0.63 HO+0.23 MO2
+0.12 KETP+0.06 ETH+0.18 ETHP :*ARR2(4.40D-15 , 845.0_dp, TEMP) ;*

Reference:

Papiez, M. R., et al. (2009), The impacts of reactive terpene emissions from plants on air quality in Las Vegas, Nevada, *Atmos. Environ., 43(27), 4109-4123*

Update $WRF-CHEM/chem/ subroutines

In order to call necessary subroutines for the new chemical mechanism, e.g. SO2-> SO4 conversion

chem_driver.F:

……

```
so2so4_select: SELECT CASE(config_flags%chem_opt)
CASE (RADM2SORG_KPP,RACMSORG_KPP,RADM2SORG_NEW_KPP, &
RACM_ESRLSOA01_KPP)
CALL wrf_debug(15,'gocart so2-so4 conversion')
CALL so2so4(chem,p_so2,p_sulf,p_h2o2,p_QV,p_QC,T_PHY,P_PHY,      &
```

……

Also the following subroutines:

chemics_init.F

module_input_chem_data.F

mechanism_driver.F

cloudchem_driver.F

module_wetscav_driver.F

aerosol_driver.F

dry_dep_driver.F

emissions_driver.F

module_bioemi_megan2.F

module_add_emiss_burn.F

module_ftuv_driver.F

optical_driver.F

module_optical_averaging.F

Update chem/convert_emiss.F and generate new emissions (wrfchemi_* and wrfbiochemi* files)

```
.....................................................................
#ifdef DM_PARALLEL
          IF (wrf_dm_on_monitor()) THEN
            READ(26,'(12E9.2)') dumc1(ids:ide-1,jds:jde-1)
          ENDIF
        DM_BCAST_MACRO(dumc1)
#else
            READ(26,'(12E9.2)') dumc1(ids:ide-1,jds:jde-1)
#endif
             grid%sebio_sesq(ips:ipe  ,jps:jpe  ) = dumc1(ips:ipe  ,jps:jpe  )
      ENDIF
.....................................................
```

After compiling WRF-CHEM

$WRF-CHEM/chem/KPP/mechanisms/racm_esrlsoa01:

Makefile -> ../../util/Makefile_kpp

Makefile_racm_esrlsoa01

atoms_red

racm_esrlsoa01.def

racm_esrlsoa01.eqn

racm_esrlsoa01.kpp

racm_esrlsoa01.spc

racm_esrlsoa01_wrfkpp.equiv

racm_esrlsoa01.map

racm_esrlsoa01_Function.f90

racm_esrlsoa01_Global.f90

racm_esrlsoa01_Initialize.f90

racm_esrlsoa01_Integrator.f90

racm_esrlsoa01_Jacobian.f90

racm_esrlsoa01_JacobianSP.f90

racm_esrlsoa01_LinearAlgebra.f90

racm_esrlsoa01_Main.f90

racm_esrlsoa01_Model.f90

racm_esrlsoa01_Monitor.f90

racm_esrlsoa01_Parameters.f90

racm_esrlsoa01_Precision.f90

racm_esrlsoa01_Rates.f90

racm_esrlsoa01_Update_Rconst.f90

racm_esrlsoa01_Util.f90

racm_esrlsoa01_mex_Fun.f90

racm_esrlsoa01_mex_Jac_SP.f90

WKC generated racm_esrlsoa01.map file

Variable species

1 = SULF (n)	29 = ADDX (r)	57 = HC5P (r)
2 = CO2 (n)	30 = ETE (r)	58 = HCHO (r)
3 = ORA1 (n)	31 = ADDC (r)	59 = TOLP (r)
4 = ORA2 (n)	32 = HNO3 (r)	60 = XYLP (r)
5 = SO2 (r)	33 = PAA (r)	61 = OLIP (r)
6 = O1D (r)	34 = ISON (r)	62 = ONIT (r)
7 = ISHP (r)	35 = SESQ (r)	63 = DCB (r)
8 = HC5 (r)	36 = PAN (r)	64 = XO2 (r)
9 = TOL (r)	37 = API (r)	65 = OLT (r)
10 = XYL (r)	38 = CO (r)	66 = ALD (r)
11 = N2O5 (r)	39 = LIM (r)	67 = OLI (r)
12 = HC8 (r)	40 = ISO (r)	68 = OLND (r)
13 = MAHP (r)	41 = MBO (r)	69 = OLNN (r)

.....

Suggestions

1) Always run “clean -a” command after you change any of KPP files
2) When you list chemical species for a new chem_opt in registry.chem, place the gases between “so2 … ho2” and if you add aerosols then place them after “so4aj…”
3) Only species within “chem” array (not “misc”!) are used to initialize from previous simulation data when chem_in_opt=1
4) Check if the added mechanisms work with pre-existing initial and boundary conditions, emissions, photolysis rates, aerosol modules, dry and wet deposition rates: e.g. check module_dep_simple.F
5) You can also simulate some species as passive tracers (tracer_opt)

QUESTIONS ?