Aix*Marseille université
Socialement engagée

cnrs

PIIM
Physique des Interactions
Ioniques et Moléculaires

Parrallelizing calls to CRM

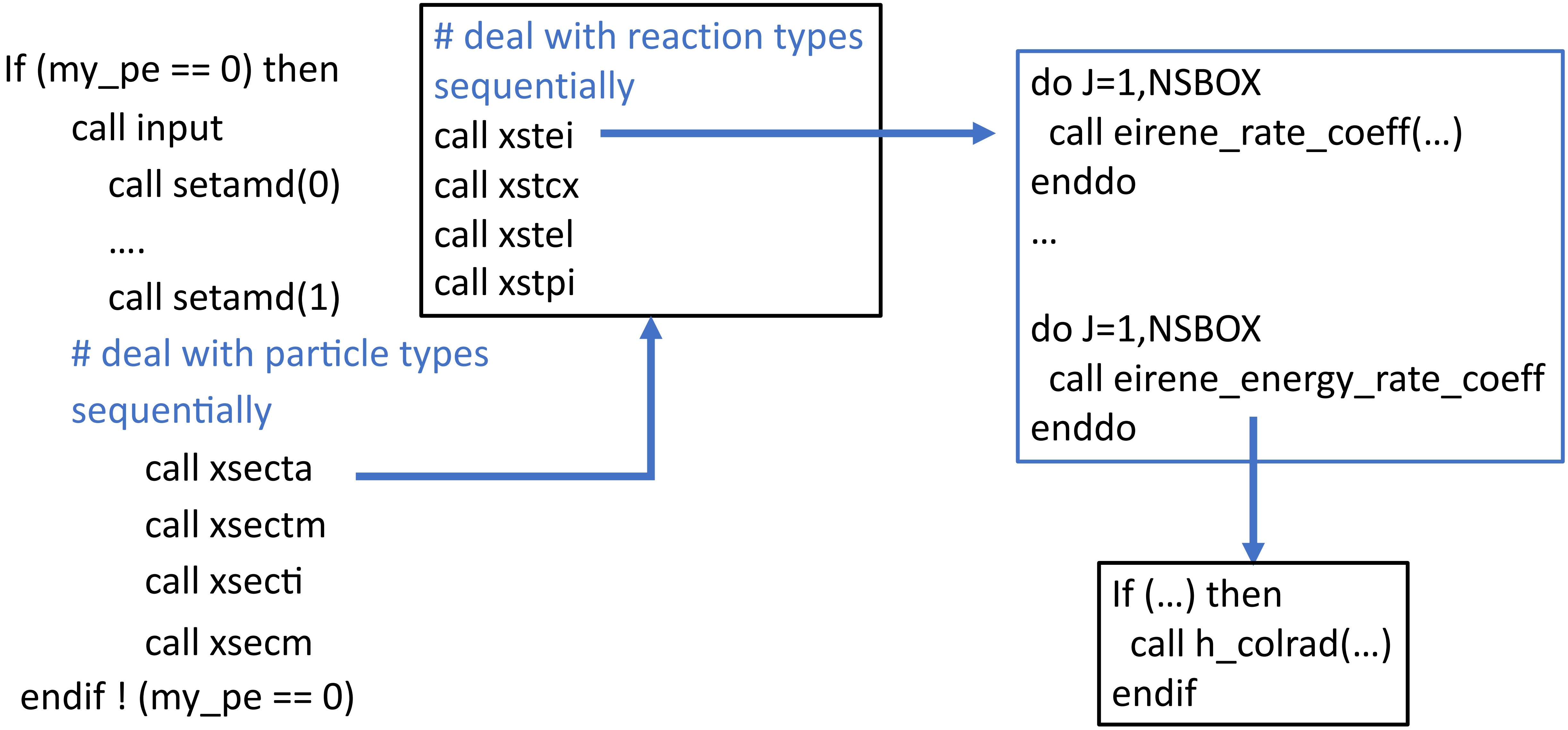
Y. Marandet & P. Genesio

(with some advice from Petra …)

Why & How ?

- Issue : running a CRM into each cells (10^3-10^7 cells …) prior to each EIRENE run on a single processor takes wall clock time
- Need to parallelize the process
- Split the grid into chunks and give chunks to processors
- Splitting can be done in arbitrary way (independant calculations in each cells)
- First thought : easy, just parallelize the loop calling the CRM model
- well …

How are collisional radiative model called ?



```
If (my_pe == 0) then
      call input
         call setamd(0)
         ….
         call setamd(1)
      # deal with particle types
      sequentially
            call xsecta
            call xsectm
            call xsecti
            call xsecm
 endif ! (my_pe == 0)
```

```
# deal with reaction types
sequentially
call xstei
call xstcx
call xstel
call xstpi
```

```
do J=1,NSBOX
  call eirene_rate_coeff(…)
enddo
…

do J=1,NSBOX
  call eirene_energy_rate_coeff
enddo
```

```
If (…) then
  call h_colrad(…)
endif
```

Implementation of MPI parallelisation (1)

- divide the grid in chunks (= ncell/n_processors, not necessarily divisible but make the chunk length as identical as possible) ☑
- call input with all processors, execute most of it with only one processor) ☑
- rate coefficients already broadcasted to all processes, need to adjust initialization/broadcast and so on) ☑
- Modify all loops (explicit or implicit), e.g. replace 1:NSBOX by grid_chunk(1):grid_chunk(2)) ☑
- Each processor has to broadcast his chunk and receive chunks calculated by others

Implementation of MPI parallelisation (2)

call Input

call setamd(0)

….

call setamd(1)

call xsecta

call xsectm

call xsecti

call xsecm

call xstei

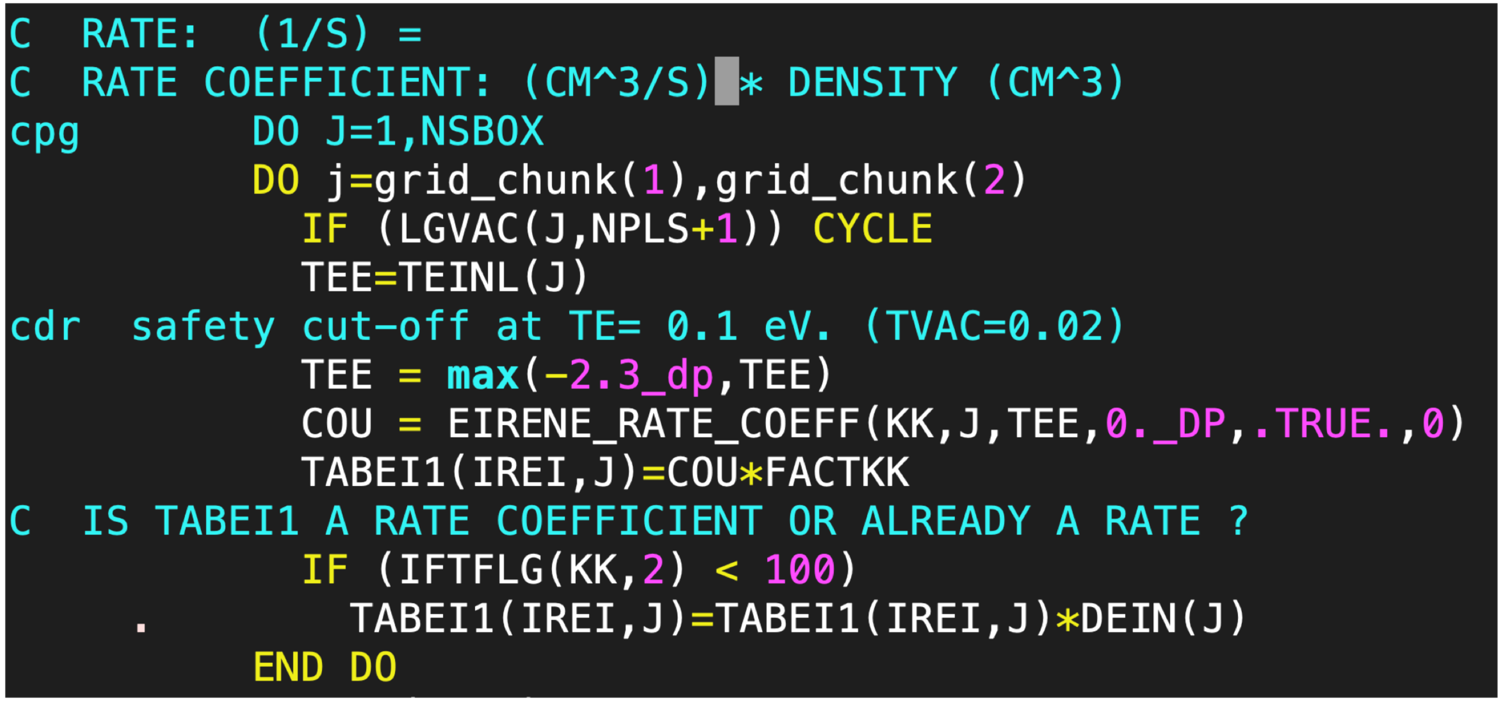
call xstcx

call xstel

call xstpi

25 loops on grid cells

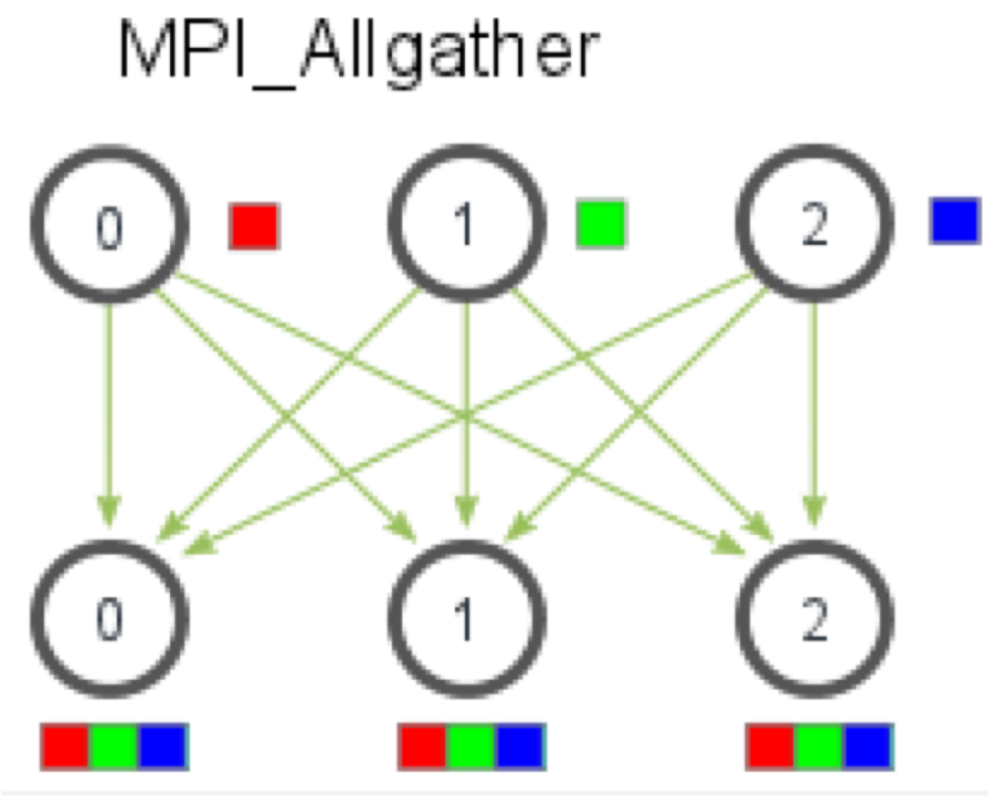
One example



```
C  RATE:  (1/S) =
C  RATE COEFFICIENT: (CM^3/S) * DENSITY (CM^3)
cpg       DO J=1,NSBOX
          DO j=grid_chunk(1),grid_chunk(2)
            IF (LGVAC(J,NPLS+1)) CYCLE
            TEE=TEINL(J)
cdr  safety cut-off at TE= 0.1 eV. (TVAC=0.02)
            TEE = max(-2.3_dp,TEE)
            COU = EIRENE_RATE_COEFF(KK,J,TEE,0._DP,.TRUE.,0)
            TABEI1(IREI,J)=COU*FACTKK
C  IS TABEI1 A RATE COEFFICIENT OR ALREADY A RATE ?
            IF (IFTFLG(KK,2) < 100)
     .        TABEI1(IREI,J)=TABEI1(IREI,J)*DEIN(J)
          END DO
```

Implementation of MPI parallelisation (3)

- Distributing and collecting all information



Use MPI_AllgatherV because all chunks do not have the same size …

Summary & Conclusion

- MPI Implementation ongoing
- Next step : OpenMP layer on the lowest level possible (closest from call to h_colrad) OR for the solver used the CR in each cell (more overhead ?)
- Will be introduced on the EIRENE unified version (currently from develop)