Electron-Molecule Collision Calculations on Vector and MPP Systems

Carl Winstead
Vincent McKoy

Cray site: JPL

Work supported by

JPL

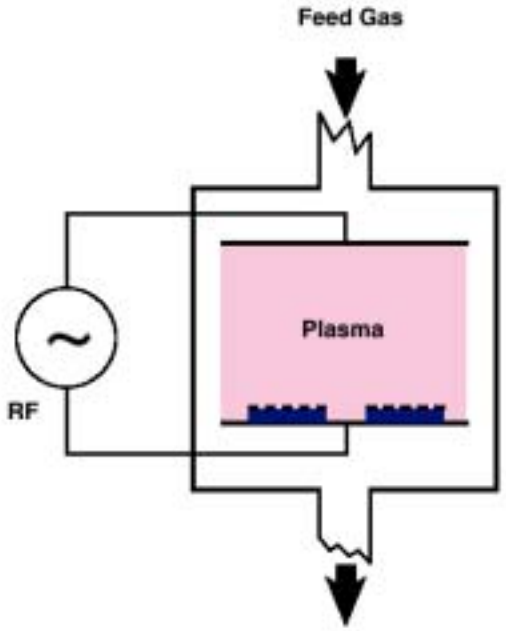
intel

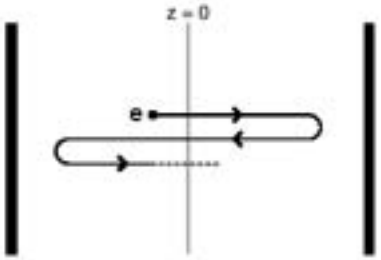
INTERNATIONAL SEMATECH

Electron-molecule collisions in plasmas

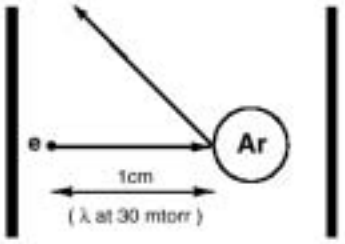
- Elastic collisions affect electron transport and energy deposition
- Inelastic collisions deposit large amounts of energy and create reactive fragments
  - ionization
  - dissociation

Electron-impact dissociation in plasmas

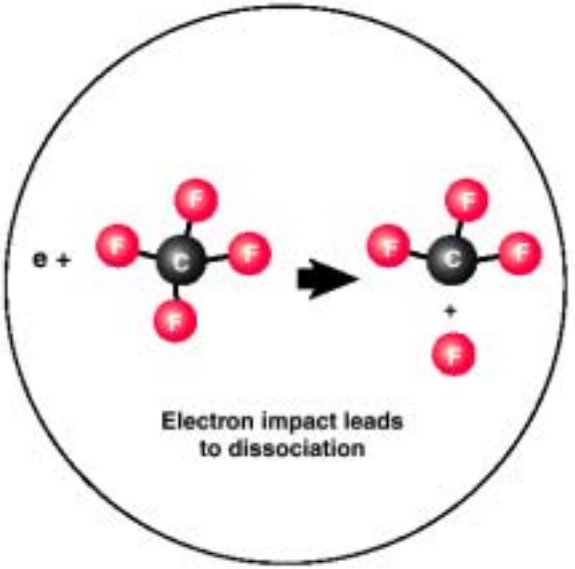


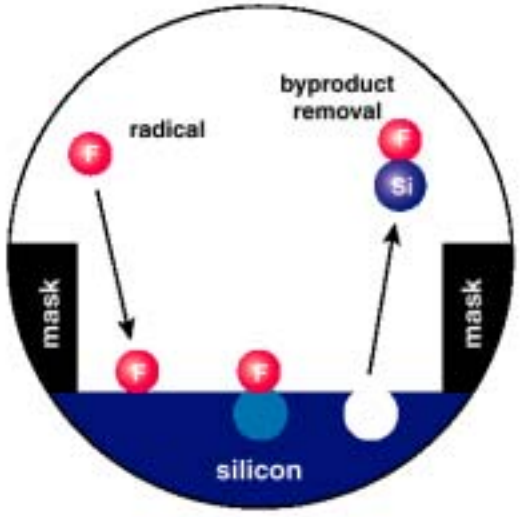


Electron trajectory in an AC field



Heating of electrons





Electron-molecule collision data

- Measurements are often unavailable
  - few groups engaged in the work
  - some gases hazardous or difficult to work with
  - measurements of inelastic cross sections especially challenging
- Calculations are an alternative

Requirements

- At the low impact energies of interest, an accurate quantum-mechanical treatment of the collision is necessary
- A method must address
  - Molecular targets of arbitrary symmetry
  - Exchange interactions (indistinguishable particles)
  - Target polarization (distortion of molecular electron density)
  - Electronic excitation (multichannel problem)

Variational approach

- Variational methods are widely used to obtain useful approximate solutions to many-body problems
- Variational methods for collisions generally lead to matrix equations of the form

$$\mathbf{Ax}=\mathbf{b}$$

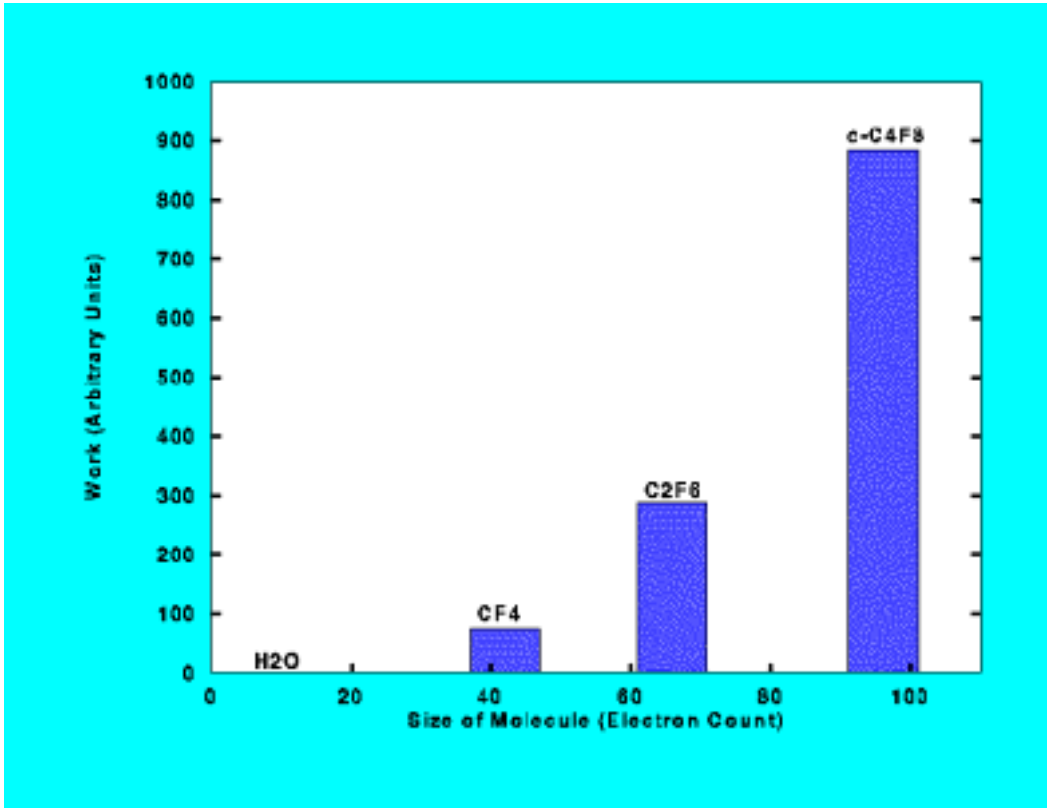
where **A** and **b** are known matrices

The Schwinger multichannel (SMC) method

- We use a multichannel extension of the variational principle introduced by J. Schwinger in 1947
- Applicable to molecules of arbitrary shape
- Treats inelastic as well as elastic collisions

Electron collision calculations

- Accurate calculations scale rapidly with molecular size
- Calculations on larger fluorocarbons such as *c*-C_4F_8, *c*-C_5F_8 require very high operation counts (10^{15}-10^{16})



Integrals, integrals, and more integrals

- Construction of **A** and **b** requires the evaluation and transformation of large numbers of two-electron repulsion integrals of the type

$$d^3r_1\ d^3r_2\ \ (\mathbf{r}_1)\ \ (\mathbf{r}_1)\Omega\mathbf{r}_1\text{-}\mathbf{r}_2\Omega^{-1}\ \ (\mathbf{r}_2)\exp(i\mathbf{k}\cdot\mathbf{r}_2)$$

  where , , and are Cartesian Gaussian functions of the form $f(x, y, z) \exp(- |\mathbf{r}-\mathbf{R}|^2)$.

- Scaling is
  - $N_g^3 N_k$ for evaluating integrals
  - $N_g^4 N_k$ for transforming integrals

How many?

- 10^{10}-10^{13} integrals (10^{12}-10^{15} floating-point operations) are typical for 5-15 atom systems
- Transformation of these integrals requires of the order of 10^{12}-10^{16} floating-point operations
- Single-processor speeds 10^{9} floating-point operations/sec
- 10^{16} operations @ 10^{9} operations/sec 100 processor-days

Parallel computers are necessary

- Complete calculations for polyatomic gases used in plasma processing (C_2F_6, *c*-C_4F_8) are impractical on single-processor computers
- Multiprocessor (parallel) computers provide the aggregate computational power (raw speed, memory, and I/O bandwidth) to make such calculations feasible
- Single-processor computation on PVPs and workstations continues to play a role

Role of PVP Systems

- Not all code worth parallelizing
  - Some steps more disk-intensive than CPU-intensive
  - Others logically intricate but with low operation count
  - If scaling with problem size acceptable, retaining uniprocessor approach preferable
  - Most of our program (by line count) in this category
- Non- or poorly-parallelized third-party applications used in problem setup phase

PVP vs. Workstation/Server

- Find x86/Linux systems increasingly competitive (Moore's Law)
- Our largest uniprocessor problems still use PVP (SV1)
  - Large, fast disk
  - Memory per process
  - CPU performance sufficient

Example: SV1 vs. P4/1.8GHz

- SF_6 electron-impact excitation problem
- Uniprocessor phase:
  - 1.7_10^{12} floating-point operations
  - 88% in 4-index transformation
  - Transformation step involves matrix multiplication and (heavy) disk access

Example: SV1 vs. P4/1.8GHz

- SV1
  - 73 MFLOP overall
  - 175 MFLOP in 4-index transformation
  - Integral generation very slow (11900 s)
- Pentium 4 workstation
  - Not enough disk to complete
  - 100 MFLOP in 4-index transformation
  - Integral generation very fast (~ 780 s)

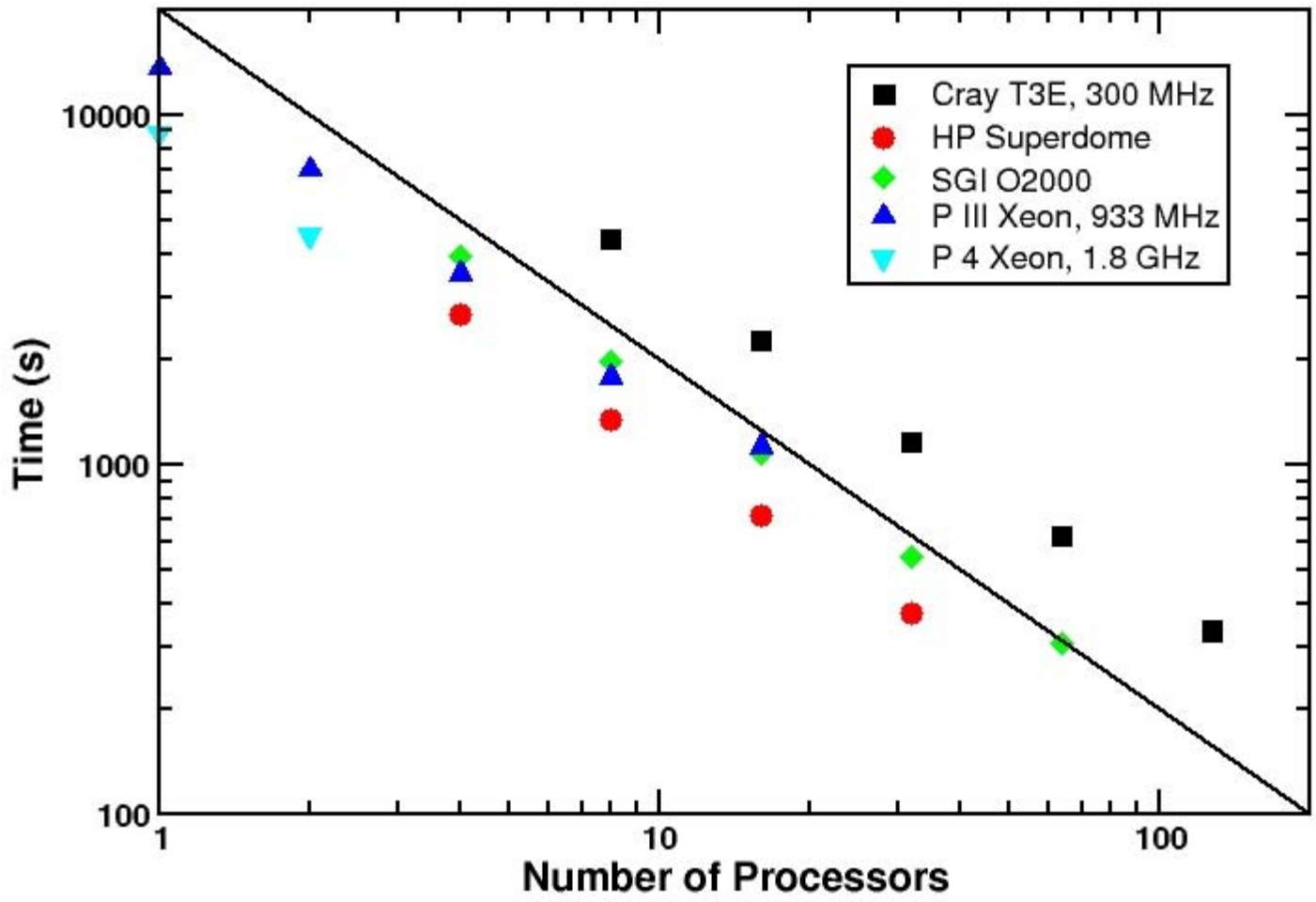
Parallel strategy

- Distribute integral evaluation across processors
  - no interprocessor communication required
- Distributing the transformation is more challenging
  - however, can be mapped to multiplication of large, dense, distributed matrices
- Performance reaches significant fraction of peak for large problems

Achieving good scaling

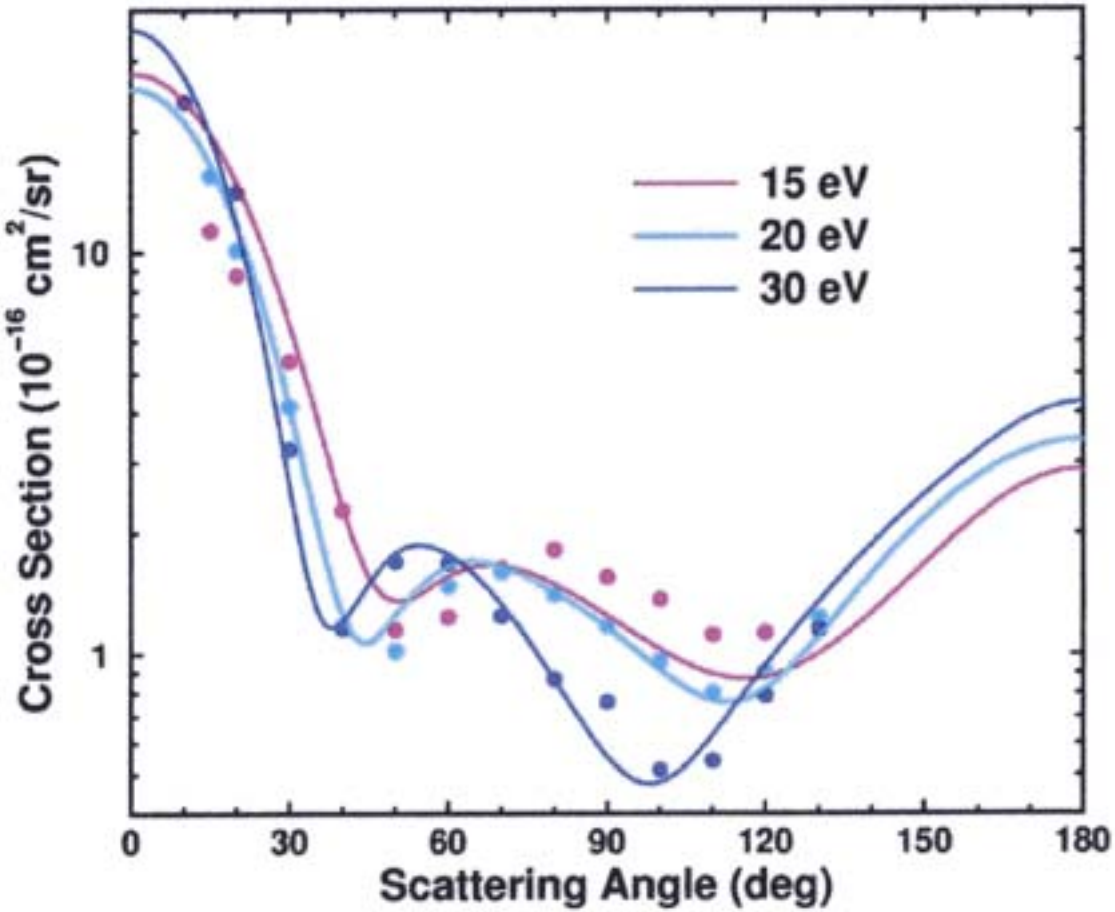
- Critical communication localized in distributed-matrix multiplication
  - Favorable computation-to-communication ratio
  - Easy to optimize
- On T3E, use shared-memory operations in this one step (MPI elsewhere)
- Low latency and flat interconnect helpful
  - Scaling less favorable on some NUMA architectures

Scaling on different platforms



Comparison with experiment: C_2F_6

Calculated elastic differential cross sections at 15, 20, and 30 eV impact energy compared to data of Takagi *et al.*, J. Phys. B **27**, 5389 (1994)

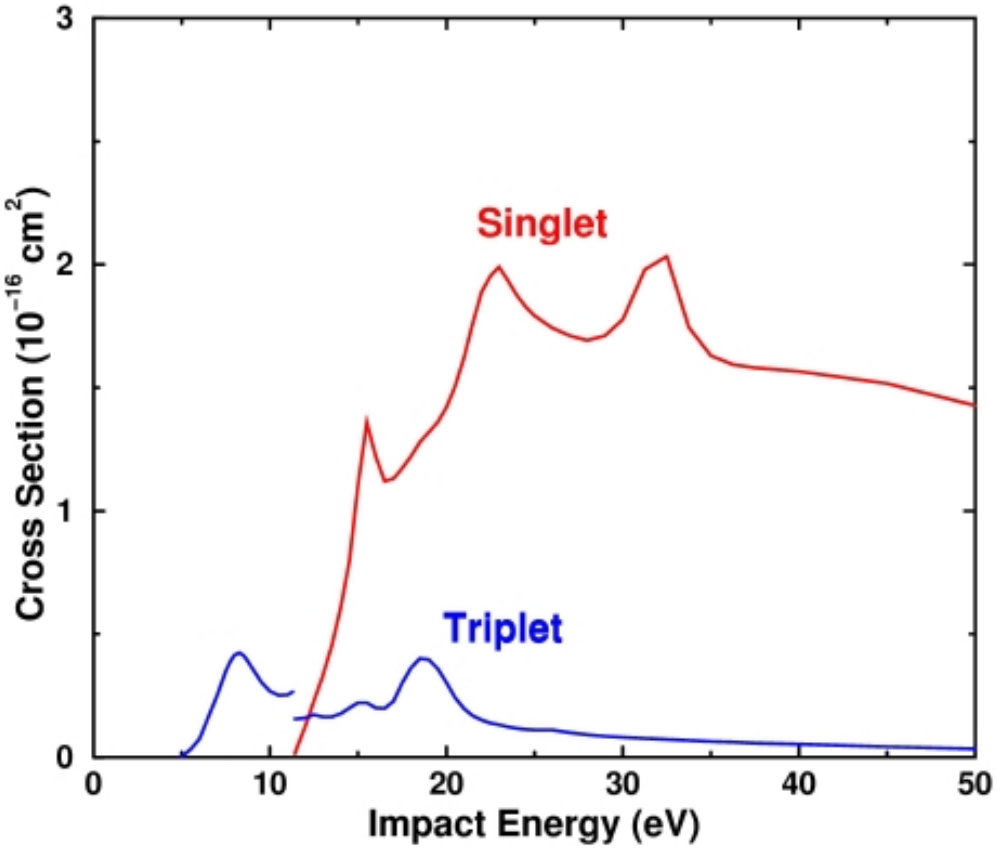


C_2F_4 electron-impact excitation: the 1 $^{1,3}B_{1u}$ (T and V) states

Cross sections for
(*) excitation,

leading to the T (triplet) and V (singlet) states. The V state has a large cross section, as expected.

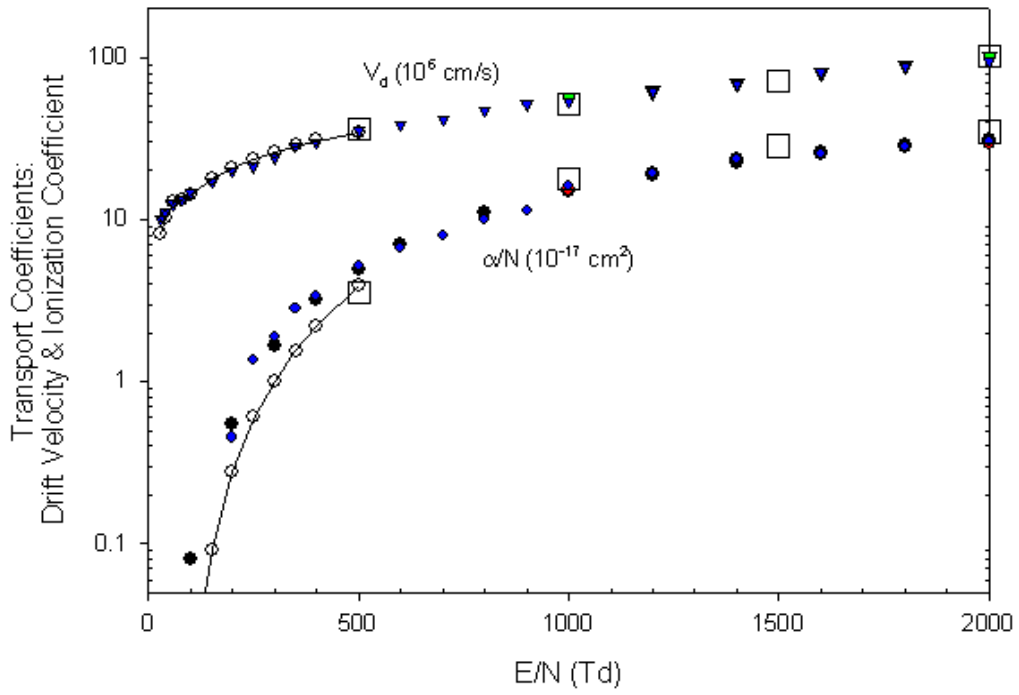
Both processes are expected to contribute to dissociation into neutral fragments, with CF_2 production likely.



Comparison of calculated and measured swarm parameters

The predictions obtained from the final cross section set agree well with the measured swarm data.

At high *E/N,* the two-term approximation fails, and it is necessary to employ Monte Carlo simulation.



Conclusions

- Electron-molecule collision calculations can contribute to plasma modeling
- Need for higher performance continues
- MPP and/or cluster systems vital
- Role for 1- or few-processor systems
  - Vector or IA32/IA64 ?
- Looking forward to X1