Molecular graphics and the classical adiabatic and nonadiabatic dynamics of small polyatomic reactions

L. J. Dunne and N. L. Bourner

South Bank Polytechnic, London, SE1 OAA, UK

A molecular graphics system for displaying the classical trajectories of small polyatomic molecule dynamics has been developed. The system can be used in adiabatic and nonadiabatic dynamic studies. Applications to the photodissociation of water at 10eV illustrate the usefulness of computer graphics in molecular reaction dynamics.

Most applications of molecular graphics to date have concerned the display of the stereochemistry of complex molecules, and this has contributed much to an understanding of a variety of chemical problems that are well documented in this journal and elsewhere. However, most of these applications have concerned static molecules in which dynamic considerations have been of little interest. One class of problems of considerable interest to theoretical chemists is the dynamic study of reactive scattering processes by classical trajectory methods.[1] Although an accurate treatment of molecular rate processes requires fully quantal methods,[1] it has long been established that classical trajectory methods produce good results provided the atoms are sufficiently heavy.[1-3] Nonadiabatic classical trajectory methods have also been developed.[4,5]

The classical trajectory approach has an advantage over quantal methods in that the computed trajectories of atoms can be followed in time, and this has high interpretive value. The regions of the dynamic phase space mapped out by the moving atoms in a chemical reaction can be related to features in the molecular potential energy surface, and identifying these may allow chemical inferences to be drawn. The reader is referred to the extensive literature on classical trajectory methods for fuller details.[6]

When dealing with nonadiabatic processes, it is essential to study the time evolution of a wavefunction or its corresponding density matrix.[4]

In seeking a solution to a problem of chemical interest, the study of a single classical trajectory is insufficient. Usually a batch of trajectories — typically a few hundred — is computed, and initial conditions are chosen randomly with phase space distributions appropriate to the problem under consideration. Classical trajectory computer programs calculate phase space variables, which are the time, t, and sets of generalized atomic coordinates (Q_i) and momenta (P_i) and any functions of these.

Nonadiabatic classical trajectory programs also output the wavefunction Ψ or density matrix ρ as a function of time. Traditionally, the interpretation of such a set of classical trajectories has been carried out by plotting various sets of intramolecular distances and Ψ or ρ against time so that a chemical reaction can be followed.

This task can be automated, but so far little has been done to explore the usefulness of computer-generated displays in this context. In this paper we show that the display in scaled real time of three-dimensional trajectories is useful for the study of problems in molecular reaction dynamics. Many computer graphics packages currently available appear to consider dynamic displays, but these are not truly dynamic because the atomic positions are calculated on the basis of minimum energy pathways instead of using classical or quantal equations of motion. Furthermore, currently available graphics packages have not considered nonadiabatic effects.

GRAPHIC SYSTEM AND PROGRAMS

The computer system used to develop these programs was a VAX 11-785 with a Tektronix color graphics terminal. The programs were written in FORTRAN 77, together with the PICASO drawing system.[7] The PICASO system consists of a set of subroutines for

*Also Visiting Research Fellow, School of Molecular Sciences, University of Sussex, Falmer, Brighton, BN1 9QJ, UK

creating and displaying two- and three-dimensional images on a variety of output devices. Calls to the subroutines can be made in a program linked with the appropriate library. This system can display only line drawings rather than filled solids, but we found these sufficient to allow interpretation of the displays produced. Many packages equivalent to PICASO are available.

Displaying the paths taken by an atom during a reaction is different from displaying a static molecule, and different problems must be overcome. The main problem is to produce a display with enough information to convey the relative positions of the atoms in space at any given time during the reaction, while preventing the display from becoming so cluttered as to defy all attempts at interpretation. The best compromise between useful information and clutter was achieved by drawing spheres at some of the atomic positions calculated and by drawing dots to represent the calculated positions of the atomic centers between the spheres.

The sphere sizes were scaled to convey the depth information needed to show the three-dimensional nature of the atomic paths. This solution raises the problem of deciding at which atomic positions to draw spheres and at which to draw dots. The method used regards the atomic path as a function of time; for example:

$$f(t) = \sqrt{(x_i^2 + y_i^2 + z_i^2)} \tag{1}$$

where x_i, y_i and z_i are Cartesian coordinates at time t and spheres are then drawn at the turning points of this function, found by calculating the first derivative of the function for each atom at each time t and looking for zeros or sign changes between time intervals. The derivatives were calculated using the central difference method. The complexity of the display can be varied by examining the derivatives of particular atom paths or by changing the function used, depending on the nature of the reaction paths being examined.

When used to examine nonadiabatic reaction processes, this technique can be enhanced to show the electronic state of the molecule for each set of calculated atomic positions. This is accomplished by representing a molecule in different states by different colors (e.g., Π by blue, Σ by red).

We felt that the use of hidden-line removal would not add to the usefulness of the display, since these do appear to contain more dynamic information.

Listing 1 (see appendix) is the program we used to produce three-dimensional displays of atom paths in reactions.
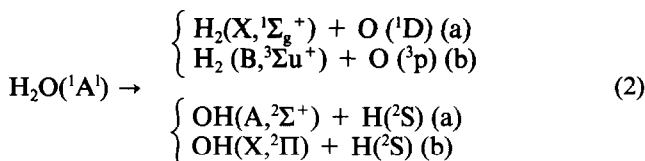
APPLICATION TO THE PHOTODISSOCIATION OF H_2O

There have been several experimental and theoretical studies of reactions occurring on photodissociation of water.[8-16] The outcome depends on the wavelength of the absorbed photons and on the initial rotational distribution of the water molecules.[8-16] The study of such events is contributing greatly to our understanding of molecular photochemical dynamics.[17-19]

The electronic states considered important in the photochemical dynamics of H_2O are summarized in references 15, 20 and 21. It is generally accepted that at excitation energies of approximately 10eV, the photo-

products of H_2O photodissociation are generated by a dissociative pathway that is initially on the $\tilde{B}$ state surface. Hodgson et al.[15] and others[20] have noted that there is an intersection of the $\tilde{B}$ and $\tilde{X}$ surfaces in linear configurations of the water molecule, and crossing between these states may occur. Such crossings are produced by nonadiabatic couplings. Here we investigate graphically the atomic dynamics in such transitions using a two-valued potential energy function for the $\tilde{B}$-$\tilde{X}$ surfaces and a classical trajectory method, which includes nonadiabatic transitions.

It is not our purpose to dwell on the details of the dynamical calculations, which have been discussed elsewhere[22,23] but rather to show how graphic displays can aid work in molecular reaction dynamics. A potential for H_2O has been derived in diabatic form as a 2 × 2 matrix V in which the diagonal elements conform to the dissociative channels (a) and (b) in the following scheme:

$$H_2O(^1A^1) \rightarrow \begin{cases} \begin{cases} H_2(X,^1\Sigma_g^+) + O\ (^1D)\ (a) \\ H_2\ (B,^3\Sigma u^+) + O\ (^3p)\ (b) \end{cases} \\ \begin{cases} OH(A,^2\Sigma^+) + H(^2S)\ (a) \\ OH(X,^2\Pi) + H(^2S)\ (b) \end{cases} \end{cases} \tag{2}$$

The full potential has been published elsewhere.[24] We note that channels a and b allow the OH fragment to emerge in $OH(^2\Sigma^+)$ or $OH(^2\Pi)$ following photodissociation at 10eV.

A recently developed dynamic model for nonadiabatic processes couples Hamilton's equations:

$$\frac{dQk}{dt} = \left(\frac{\partial H}{\partial P_k}\right)_{t,Qk}, \frac{dPk}{dt} = -\left(\frac{\partial H}{\partial Q_k}\right)_{t,P_k} \tag{3}$$

with the quantum mechanical Liouville equation for the electronic density matrix:

$$i\hbar\frac{\partial \rho}{\partial t} = [V, \rho] \tag{4}$$

We work in a diabatic representation in which it is assumed that the electronic transitions arise solely from the off-diagonal elements of V. The Hamiltonian $\mathcal{H}$ for the classical nuclear motion contains the average potential Trace $(V\rho)$.

$$\mathcal{H} = \Sigma\ P_k^2/2\mu_k + \text{Trace } (V\rho) \tag{5}$$

and the trajectories therefore evolve continuously with time.

One assumption of theoretical studies[22,23] in the photofragmentation dynamics of H_2O is that the initial configuration of the molecule, when the $\tilde{B}$ state is initially populated following light absorption, is that the excited state nuclear geometry is close to that in the ground state at equilibrium. Hence, the Franck–Condon principle is invoked.

During the photodissociation process, when nonadiabatic crossings between the two surfaces are involved, the molecule undergoes oscillations of its electronic state in time so that it is predominantly on one of the two electronic surfaces. The factors governing this process are complicated, but the time dependence of the electronic state is contained in the electronic density matrix ρ, and this we can calculate.

Color Plate 1 presents a display of the classical trajectories produced by a single photodissociating H_2O molecule at 10eV. The oxygen atom is red, while the

two hydrogen atoms are blue. The initial C_{2v} equilibrium geometry of H_2O can be recognized readily at the center of the figure. The inset gives the time (in units of 0.5×10^{-14}s). Segev and Shapiro[16] have classified two limiting types of classical trajectory leading to dissociation on the $\tilde{B}$ adiabatic surface of water. Type (1) trajectories (in Dixon's terminology[14]) linearize outside the $\tilde{B}-\tilde{X}$ conical intersection, while type (2) trajectories linearize inside this intersection. Segev and Shapiro postulate that the former are likely to lead to $OH(^2\Pi)$. We do not find any such simple correlation between the initial behavior of our trajectories and products. A detailed study was made of 12 trajectories to gain some insight into the nuclear dynamics. Typically, the molecule linearizes in approximately 10^{-14}s, and it then undergoes large amplitude vibrations for a few periods with simultaneous extension of one bond and subsequent separation leading to dissociation. The only characteristic feature we noted was that trajectories leading to $OH(^2\Sigma^+)$ took about twice the real time of those leading to $OH(^2\Pi)$. This difference in lifetimes is not surprising, though, since the products in the $OH(^2\Pi)$ channel have generally greater translational energy. Trajectories dissociating to $OH(^2\Pi)$ and $OH(^2\Sigma^+)$ are shown in Color Plates 1 and 2, respectively.

Color Plate 3 shows the dynamics of the small number (5%) of H_2O molecules photodissociating to channel (a) of scheme (1), where the product is $O(^1D) + H_2$.

Color Plate 4, which involves the same trajectories as Color Plate 1, shows the oscillation of the photodissociating water molecule between the $\tilde{B}/\tilde{X}$ electronic surfaces. This is achieved readily by associating a color with each electronic state. (See Color Plate 4 caption.)

OUTLOOK

We are convinced that computer-generated displays are useful in classical trajectory studies of molecular reaction dynamics. With appropriate hardware, the techniques described here can be extended readily to many atoms and to include real-time animation.

ACKNOWLEDGEMENTS

REFERENCES

1 Child, M. S. *Molecular collision theory*. New York: Academic Press, 1974

2 Karplus, M., Porter, R. N. and Sharma, R. D. *J. Chem. Phys.*, 1965, **43**, 3259

3 Dunne, L. J. and Murrell, J. N. *Mol. Phy.*, 1983, **50**, 635

4 Dunne, L. J., Murrell, J. N. and Stamper, J. *Chem. Phys. Letts.*, 1984 **112**, 497

5 Tully, J. C. and Preston, R. K. *J. Chem. Phys.*, 1971, **55**, 562

6 *Dynamics of Molecular Collisions*, ed. W.H. Miller. New York: Plenum Press, 1976.

7 PICASO — a graphic software package. D.A. Vince, Middlesex Polytechnic Computer Center, 1985.

8 Vinogradov, I. P. and Vilesov, F. I. *Opt. Spectrosc.*, 1976, **40**, 32

9 Carrington, T. *J. Chem. Phys.*, 1964, **41**, 2012

10 Andresen, P. *J. Chem. Phys.*, 1984, **80**, 2548

11 Fotakis, C., McKendrick, C. B. and Donovan, R. *J. Chem. Phys. Letts.* 1981, **80**, 598

12 Donovan, R. J., *et al. Can. J. Chem.*, 1983, **61**, 1023

13 Hodgson, A., *et al. Chem. Phys. Letts.*, 1984, **107**, 1

14 Dixon, R. N. *Mol. Phys.*, 1985, **54**, 333

15 Hodgson, A. *et al. Molec. Phys.*, 1985, **54**, 351

16 Segev, E. and Shapiro, M *J. Chem. Phys.*, 1982, **77**, 5604

17 Dynamics of molecular photofragmentation. *Faraday Discuss. Chem. Soc.*, 1987, **82**

18 Papagiannokopoulos, P. and Fotakis, C *J. Phys. Chem.*, 1985, 3439

19 Krautwald, H. J., *et al. Faraday Discuss. Chem. Soc.*, 1987, **82** (in press)

20 Docker, M. P. Hodgson, A. and Simons, J. P. *Mol. Phys.*, 1986, **57**

21 Theodorakopoulos, G., *et al. Chem. Phys. Letts.*, 1982, **89**, 164

22 Dunne, L. J. *Faraday Discuss. Chem. Soc.*, 1987, **82**, Dynamics of molecular photofragmentation

23 Dunne, L. J., Murrell, J. N. and Guo, H. *Mol. Phys.*, 1987 (in press)

24 Murrell, J. N., *et al. Mo. Phys.*, 1981, **42**, 605

APPENDIX

```
C          A program to produce 3D displays of atom paths in reactions
C
C          Variables used:-
C                 TM,T,TP  time values
C                 PM       x,y,z values at t-1
C                 P        x,y,z values at t
C                 PP       x,y,z values at t+1
C                 DM       differentials at t-1
C                 D        differentials at t
C                 SIZE     diameters of atoms
C                 SP       array to hold sphere in
C                 DT3      array to hold dot in
C                 ROM      density matrix at t-1
C                 RO       density matrix at t
C                 ROP      density matrix at t+1
C                 ISPHERE  number of spheres drawn so far
C
C          Constants used:-
C                 NATOMS   number of atoms in reaction
C                 XOBS     x coordinate of observer
C                 YOBS     y coordinate of observer
C                 ZOBS     z coordinate of observer
C
C
C
C-------All real variables are double precision
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        PARAMETER(NATOMS=3,XOBS=0.0,YOBS=0.0,ZOBS=30.0)
        DIMENSION PM(NATOMS,3),P(NATOMS,3),PP(NATOMS,3),
      *      INK(2,NATOMS),SIZE(NATOMS),DM(NATOMS),D(NATOMS),
      *      SP1(962),DT3(9)
C
C-------INK and SIZE values in data statements for easy change
C-------adiabatic or non-adiabatic displays acheived by
C-------changing INK values
        DATA SIZE,INK/1.40,1.20,1.20,11,2,11,2,11,2/
C
C
10      FORMAT(1E12.6,1H,1E12.6,1H,1E12.6)
15      FORMAT(1E12.6)
C
C-------These subroutine calls are to PICASO (ref.7) to set up various
C-------parameters such as the viewing position and create the objects
C-------drawn in the program
        CALL START
        CALL SPHERE(SP1,0.2,10.6)
        CALL DOT3D(DT3,0.0,0.0,0.0)
        CALL FRAME(-2.0,2.0,-1.5,1.5)
        CALL DSCALE(4.0,3.0,0.0)
        CALL EYE(XOBS,YOBS,ZOBS,0.0,0.0,0.0)
        CALL HSIZE(0.05,0.045)
C
C-------Read in the first 3 sets of atom coordinates
C-------1st set
        READ(8,15)TM,ROM
        DO I=1,NATOMS
        READ(8,10)PM(I,1),PM(I,2),PM(I,3)
C-------Set the ink to be used according to the value of the density
C-------matrix ROM.  INK values are the same for adiabatic displays
        IF (ROM.LT.0.5) THEN
        CALL NEWPEN(INK(1,I))
        ELSE
        CALL NEWPEN(INK(2,I))
        ENDIF
```

```
C--------Draw a dot at each atom centre
        CALL DRAW3D(DT3,1.0,PM(I,1),PM(I,2),PM(I,3),1)
      ENDDO
C-------Repeat as above for 2nd set
      READ(8,15)T,RO
      DO I=1,NATOMS
        READ(8,10)P(I,1),P(I,2),P(I,3)
        IF (RO.LT.0.5) THEN
          CALL NEWPEN(INK(1,I))
        ELSE
          CALL NEWPEN(INK(2,I))
        ENDIF
        CALL DRAW3D(DT3,1.0,P(I,1),P(I,2),P(I,3),1)
      ENDDO
C-------Repeat as above for third set
      READ(8,15)TP,ROP
      DO I=1,NATOMS
        READ(8,10)PP(I,1),PP(I,2),PP(I,3)
        IF (ROP.LT.0.5) THEN
          CALL NEWPEN(INK(1,I))
        ELSE
          CALL NEWPEN(INK(2,I))
        ENDIF
        CALL DRAW3D(DT3,1.0,PP(I,1),PP(I,2),PP(I,3),1)
      ENDDO
C
C-------Write the initial time value at the side of the screen
      CALL NEWPEN(1)
      CALL AMOVE(-2.1,1.45)
      CALL HREAL(TM,2,7)
      ISPHERE=1
C
C-------For each atom
      DO I=1,NATOMS
C
C-------Draw sphere at initial position  (t-1)
C
C--------Scale the sphere according to atom size and z-coordinate
        IF (PM(I,3).GE.0.0) THEN
          S=SIZE(I)*(PM(I,3)+1.0)
        ELSE
          S=-SIZE(I)/(PM(I,3)-1.0)
        ENDIF
C--------Set pen colour according to density matrix
        IF (ROM.LT.0.5) THEN
          CALL NEWPEN(INK(1,I))
        ELSE
          CALL NEWPEN(INK(2,I))
        ENDIF
C--------Draw the sphere
        CALL DRAW3D(SP1,S,PM(I,1),PM(I,2),PM(I,3),-1)
C
C--------Calculate differential at t
        TEMP1=SQRT(PP(I,1)**2+PP(I,2)**2)
        TEMP2=SQRT(PM(I,1)**2+PM(I,2)**2)
        DM(I)=(TEMP1-TEMP2)/2.0
C
C--------Change t so that t->t-1 and t+1->t
        DO J=1,3
          PM(I,J)=P(I,J)
          P(I,J)=PP(I,J)
        ENDDO
      ENDDO
      TM=T
      T=TP
C
C-------Continue to loop unitl end of file reached
      NLOOP=1
      DO WHILE (NLOOP.EQ.1)
C
C--------For each atom
        READ(8,15,END=30,ERR=30)TP,ROP
        DO I=1,NATOMS
C
C----------Read in next set of atom coordinates
          READ(8,10,END=30,ERR=30)PP(I,1),PP(I,2),PP(I,3)
C
C----------Calculate differential
          TEMP1=SQRT(PP(I,1)**2+PP(I,2)**2)
          TEMP2=SQRT(PM(I,1)**2+PM(I,2)**2)
          D(I)=(TEMP1-TEMP2)/2.0
```

```
C
C----------Draw a dot at each atom centre
          IF (RO.LT.0.5) THEN
            CALL NEWPEN(INK(1,I))
          ELSE
            CALL NEWPEN(INK(2,I))
          ENDIF
          CALL DRAW3D(DT3,1.0,P(I,1),P(I,2),P(I,3),1)
        ENDDO
C
C----------If there is a zero or a change of sign then draw a sphere
          IF (((DM(2).GT.0.0).AND.(D(2).LT.0.0)).OR.
     *        ((DM(2).LT.0.0).AND.(D(2).GT.0.0)).OR.
     *        ((DM(3).GT.0.0).AND.(D(3).LT.0.0)).OR.
     *        ((DM(3).LT.0.0).AND.(D(3).GT.0.0))) THEN
            ISPHERE=ISPHERE+1
C------------Write the time value on the screen
            CALL NEWPEN(1)
            CALL AMOVE(-2.1,1.5-(0.06*FLOAT(ISPHERE)))
            CALL HREAL(T,2,7)
C------------Scale the sphere
            DO I=1,NATOMS
              IF (P(I,3).GE.0.0) THEN
                S=SIZE(I)*(P(I,3)+1.0)
              ELSE
                S=-SIZE(I)/(P(I,3)-1.0)
              ENDIF
C------------Set the INK based on density matrix
              IF (RO.LT.0.5) THEN
                CALL NEWPEN(INK(1,I))
              ELSE
                CALL NEWPEN(INK(2,I))
              ENDIF
C------------Draw the sphere
              CALL DRAW3D(SP1,S,P(I,1),P(I,2),P(I,3),-1)
            ENDDO
          ENDIF
C

C----------Change t so that t->t-1 and t+1->t
          DO I=1,NATOMS
            DO J=1,3
              PM(I,J)=P(I,J)
              P(I,J)=PP(I,J)
            ENDDO
            DM(I)=D(I)
          ENDDO
          TM=T
          T=TP
          ROM=RO
          RO=ROP
        ENDDO
C
30      CONTINUE
C
        ISPHERE=ISPHERE+1
        CALL NEWPEN(1)
        CALL AMOVE(-2.1,1.5-(0.06*FLOAT(ISPHERE)))
        CALL HREAL(T,2,7)
C
C-------For each atom
        DO I=1,NATOMS
C
C--------Draw sphere at final position
          IF (PP(I,3).GE.0.0) THEN
            S=SIZE(I)*(PP(I,3)+1.0)
          ELSE
            S=-SIZE(I)/(PP(I,3)-1.0)
          ENDIF
          IF (ROP.LT.0.5) THEN
            CALL NEWPEN(INK(1,I))
          ELSE
            CALL NEWPEN(INK(2,I))
          ENDIF
          CALL DRAW3D(SP1,S,PP(I,1),PP(I,2),PP(I,3),-1)
        ENDDO
        CALL FINISH
        CLOSE(UNIT=8)
        STOP
        END
```