5-2000

Rotational Structure of Extremely Floppy van der Waals Complexes: Adiabatic Separation of Angular and Radial Motion

P. Daniel Ward
Utah State University

ROTATIONAL STRUCTURE OF EXTREMELY FLOPPY VAN DER WAALS COMPLEXES: ADIABATIC SEPARATION OF ANGULAR AND RADIAL MOTION

by

P. Daniel Ward

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Chemistry

UTAH STATE UNIVERSITY
Logan, Utah

2000

ABSTRACT

Rotational Structure of Extremely Floppy van der Waals Complexes: Adiabatic Separation of Angular and Radial Motion

by

P. Daniel Ward, Master of Science

Utah State University, 2000

Major Professor: Dr. David Farrelly
Department: Chemistry and Biochemistry

The adiabatic or Born-Oppenheimer approximation is often used in molecular calculations to simplify the solution to the Schrödinger equation. The basis of the approximation is the large difference in the relative motions of the nuclei and electrons in the molecule—the electrons are able to respond almost instantly to the movements of the nuclei. Thus, the nuclei may be regarded as being fixed in a certain position and the Schrödinger equation can then be solved using the potential obtained by solving the electronic problem at fixed nuclear configuration.

A similar argument can be used to decouple the angular and radial motions of many van der Waals complexes because, like nuclei in molecules, the radial motions in many van der Waals complexes are strongly localized. Fixing the radial separation between the atoms and molecules in the complex to a particular value results in a Schrödinger equation that is much simpler to solve because it is only dependent on angles. van der Waals complexes containing helium atoms, however, present a dilemma because the extremely weak interactions present also lead to large amplitude radial as well as angular motions. Because the basis of the adiabatic approximation

is a large difference in time scale between the angular and radial motions, the validity of the adiabatic approximation for helium complexes is uncertain.

In this thesis, the adiabatic separation of angular and radial motion is shown to be accurate for extremely floppy complexes of helium by demonstrating its use on the van der Waals molecule He-HCN. A major application of this method is expected to be the quick calculation of approximate wavefunctions for Diffusion Monte Carlo studies of the rotation of impurity molecules inside ultra-cold droplets of helium. The method presented here is significantly faster than other methods (e.g., Variational Monte Carlo) that have been used to calculate approximate wavefunctions for Diffusion Monte Carlo.

(136 pages)

ACKNOWLEDGMENTS

I wish to thank all those who have assisted me during my time as a graduate student at Utah State University. First, I thank my advisor, Professor David Farrelly, for "the calm excellence of his wisdom" as well as the assistance he provided in all aspects of my graduate education. During the past two years, he taught me quite a bit about chemistry and other things "vegetable, animal, and mineral"; however, I will most appreciate the things he taught me that were not necessarily related to my field of study.

I also wish to thank my other thesis committee members, Dr. Alexander Boldyrev and Dr. Robert Brown, for their flexibility and support. In addition, thanks are due to other former members of the Farrelly group—James and Terri for their support and friendship and Ernestine and Andrea for their patience and advice.

Finally, I give special thanks to my wife, Janie, and our three children, Andy, Abby, and Jacob. I especially appreciate the patience, support, and encouragement they gave me during my career as a graduate student.

Dan Ward

CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION AND BACKGROUND

1.1. Introduction

For millennia, philosophers and scientists have been interested in how matter is made up on the small scale. This quest for knowledge about the microscopic has been aided by the recent technological explosion of the past century. One example of this is a particularly powerful technique, spectroscopy, which has provided numerous new insights into the structure and behavior of atoms and molecules. While the use of light to study matter is not new, the way it is used in modern spectroscopy was only made possible after the invention and refinement of quantum mechanics. This improvement in the model of the physical world provides a way to interpret the complex interactions between light and matter. These interactions are examined using spectroscopy by perturbing the sample of interest with light and recording the sample's response to the perturbation as its spectrum. Analysis of the sample's spectrum provides valuable information about molecules and their environment, e.g., intra- and intermolecular bonding.

Other influences besides light, of course, exist, which can perturb the sample being studied. These intrusive perturbations lead to broadening of spectral lines and a subsequent decrease in spectral clarity, which makes extraction of information from spectra more difficult. Many of the undesirable perturbations seen in spectroscopy are a result of interactions of the sample molecule with its environment. Minimization or, ideally, removal of such perturbations generally leads to much sharper spectral lines and an increase in the amount of information that can be extracted from molecular spectra.

An important way to minimize unwanted perturbations and increase spectral res-

olution is to cool the sample down to extremely low temperatures, thereby reducing molecular rotations and vibrations. These molecular motions are especially a problem for large molecules because the many modes present make their spectra difficult to interpret. This difficulty, in practice, excludes many large molecules from spectroscopic study—a significant problem because many of the molecules of medical and biological interest fall into this category.

Ideally, for spectroscopy, the sample molecule would be trapped inside an ultracold matrix that interacts only minimally with the sample. A matrix that has these properties is a nanometer-sized droplet of liquid helium [1] in a molecular beam. Condensed phase (bulk) helium might be expected to work well in this role because of its low boiling and melting points and its weak interactions with other molecules. However, in practice, using liquid or solid helium as the host matrix for spectroscopy is difficult; the extremely cold temperature of the helium leads to condensation of the sample onto the container walls immediately when the sample is inserted into the matrix [1].

Recently, experiments have shown that liquid helium in the form of otherwise gas phase droplets (see Fig. 1) may overcome this problem. Many studies and experiments have been done to explore the possibility of using helium droplets as spectroscopic matrices. These studies, which have mostly concentrated on the behavior of simple chromophore molecules inside the clusters, have revealed a number of surprising and exciting properties of helium droplets. The pioneering experiments, which were carried out primarily by two different research groups (the Toennies group at the Max Planck Institute in Göttingen and the Scoles group at Princeton), use an apparatus like the one shown in Fig. 2 to obtain spectra of molecules in helium droplets.

The spectra collected for these molecules are similar to the respective gas phase

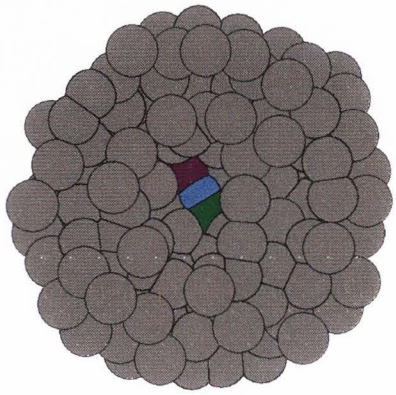

Fig. 1. Schematic of a molecule trapped inside a helium cluster in a molecular beam. Adapted from [2].

molecular spectra although they differ in one key aspect—the rotation of the molecule inside a helium droplet is much slower than it is in the gas phase. How the helium cluster is able to reduce the rate of the molecule's rotation is an important question that needs to be answered in order to better understand the properties and behaviors of these systems. Thus, a major goal of theoretical studies of these systems is the calculation of their rotational states, which will provide valuable information about how the helium environment affects the behavior of impurity molecules trapped inside.

Just as the calculation of wavefunctions and energies for molecules is complicated by the presence of many electrons, calculations of the states of molecules inside helium clusters are complicated by the many helium atoms in the system. In fact, solutions to the Schrödinger equation of these systems are not just difficult to calculate, they are analytically impossible to obtain, making the use of approximate methods necessary.

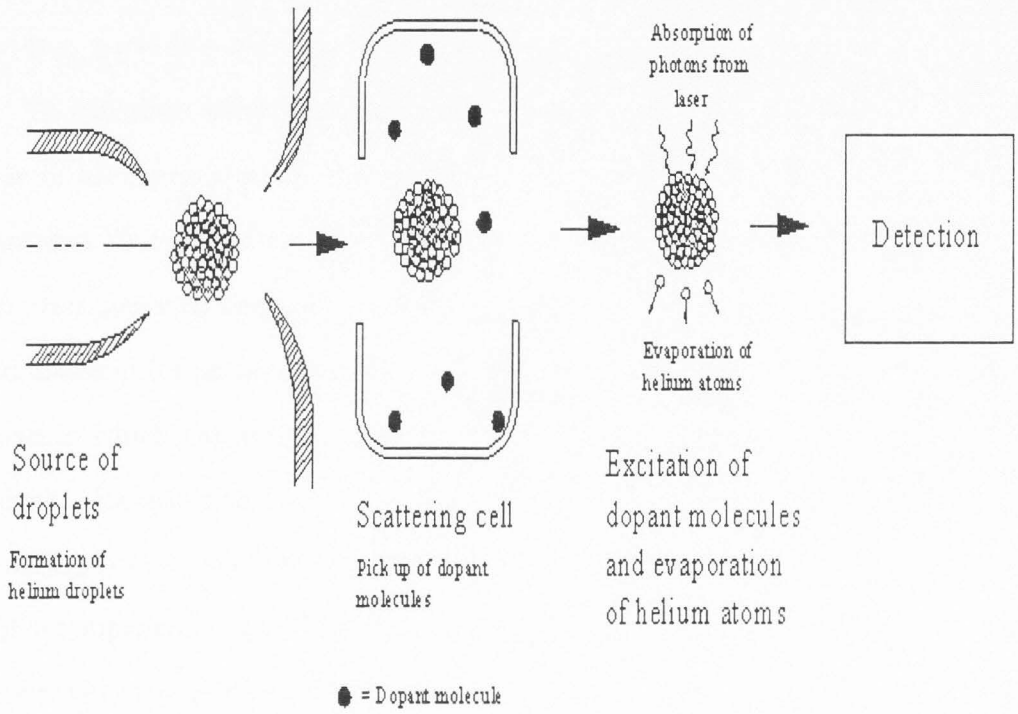

Fig. 2. Schematic diagram of a typical experimental apparatus. The clusters are initially formed by expanding gaseous helium through a cold, small diameter nozzle. They are then passed through a cell containing gas phase dopant molecules. Impact between dopant and droplet results in pick-up of the dopant by the droplet. After exciting the dopant molecule inside the cluster using a laser, helium atoms evaporate. This change in the size of the droplet is then detected.

A common approach used to calculate approximate solutions for quantum mechanical systems is to expand the wavefunction of the system as a sum of basis functions, which are approximate eigenfunctions of the Hamiltonian of the system. For many-body systems, the wavefunction is dependent on the wavefunctions of the individual particles, each of which is expressed as a series expansion in some basis set. To calculate wavefunctions and energies for these systems to any reasonable degree of accuracy requires the use of extremely large basis set expansions leading to matrices that are impractical to diagonalize because of their large size.

An alternative to basis set expansion methods that is commonly used are the so-called classical (or semi-classical) methods. There are many problems from classical mechanics which can only be solved approximately because they have analytically unsolvable Hamiltonians. For example, any realistic system that has more than two interacting bodies, such as the Sun, Earth, and Moon, has a solution that can only be solved approximately. The Sun-Earth-Moon system can be approximately solved by restricting the motion of the three bodies to the same plane and assuming that the mass of the moon is negligible compared to the other two bodies.

Often, the methods used to solve for more complicated classical systems can be applied to quantum systems. To do this, the quantum system must be within the classical limit, i.e., the masses of the particles involved must be large and their velocity must not be too fast. The small mass of helium causes helium complexes to move with very large-amplitude motions. The resulting "floppiness" of helium complexes eliminates the possibility of using classical methods to solve for the properties of these systems because the centers of mass of the helium atoms cannot be used to obtain an approximation to the wavefunction.

Although direct application of classical methods to helium cluster systems is not an acceptable option, a method that is closely related to a classical problem, Diffu-

sion Monte Carlo, can be used to accurately calculate the wavefunctions and energies for these systems. As will be described in detail in chapter 2, the similarity between the equation which describes the diffusion of particles subject to a "source" or "sink" term and the nonrelativistic, imaginary time-dependent Schrödinger equation allows a numerically exact ground state solution to the time-independent Schrödinger equation to be calculated.

The basis of the method can easily be seen by comparing the probability distribution of charged particles in solution and in an electric field with the wavefunction of a quantum system. If the diffusion-electric field system is examined a long time after the electric field is turned on, the particles will be located in the areas where their potential energy is minimized. In other words, the probability of finding a particle is highest in regions of low potential. For quantum systems, this process can be simulated by using theoretical particles called "walkers" subject to the quantum system's potential, which allows the ground state wavefunction of the system to be calculated. The process works by randomly moving the walkers to new positions and then evaluating the move to see if it resulted in the particle being at higher or lower potential. The walkers that move to regions of lower potential are duplicated while those that move to regions of higher potential are removed. After enough of these moves, the probability distribution of the walkers is concentrated in regions of low potential. This final probability distribution of the walkers represents the wavefunction of the system.

It is important to note that the probability distribution of the hypothetical walkers represents ψ, not $|\psi|^2$. This limits the basic Diffusion Monte Carlo method to systems that have wavefunctions that do not change sign, i.e., they do not have nodes. This is perhaps the major drawback to the method because it excludes excited states and fermions from consideration. This hurdle can be overcome through the use of

the fixed-node approximation, which works by restricting the random movement of the walkers to regions where the wavefunction is either always positive or always negative. In other words, it does not allow the walkers to cross the nodes of the wavefunction. To fence the walkers into these regions, infinite potentials are placed at the nodes, which requires that the location of the nodes be known. As neither the wavefunction nor its nodes are known beforehand, an approximate way to accurately predict nodal structure is critical.

In general, the complexity of the nodal surface of the wavefunction increases as the number of bodies in the system does. There are several methods which can be used to calculate approximate wavefunctions with complicated nodal structure and while each has different virtues, all methods are ultimately judged on two factors—accuracy and speed. A method that is often used to calculate approximate wavefunctions is another Monte Carlo method called Variational Monte Carlo, which uses Monte Carlo techniques in conjunction with the variational method.

The basis of the Variational Monte Carlo method is to first select a "trial" wavefunction that contains several parameters. An approximate wavefunction is calculated by adjusting the values of these parameters such that the expectation value of the Hamiltonian,

$$\langle \psi_{trial} | H | \psi_{trial} \rangle \,, \tag{1}$$

is minimized. The complex integrals that arise in these calculations are solved using Monte Carlo methods.

In addition to providing an approximate wavefunction, the variational method is also useful because the energy calculated using the method is always an upper bound to the true energy of the system. In other words, the energy calculated using the variational method is always greater than or equal to the true energy of the system.

The approximate wavefunctions calculated using Variational Monte Carlo allow accurate excited state energies and wavefunctions to be calculated using Diffusion Monte Carlo and the fixed-node approximation; however, Variational Monte Carlo calculations require a significant computational investment, which makes the method impractical to use in many circumstances. The nature of the bonding in van der Waals complexes suggests that there might be a way to calculate wavefunctions for these systems using a method that can achieve accuracy comparable to Variational Monte Carlo at a much lower computational expense.

As van der Waals forces are the weakest of all intermolecular forces, complexes that are held together by them are very weakly bound. This weak bonding allows the atoms and molecules in the complex to move with much larger amplitude motions than is seen in complexes held together by other forces. More specifically, van der Waals complexes exhibit very broad angular motions with much more localized radial motions. The strong directional dependence of these complexes combined with their relatively weak dependence on radial separation suggests that a decoupling of the angular motion from the radial might lead to a highly accurate approximate wavefunction that is dependent only on angle. This adiabatic separation of angular and radial motion is similar to the adiabatic separation of nuclear and radial motion in the well-known Born-Oppenheimer[1] approximation. In this comparison, the angular and radial motion of helium complexes are respectively analogous to the nuclear and electronic motion in molecules.

While the adiabatic approximation was previously shown to be valid for the van der Waals complex of Ar-HCl by Holmgren [3], whether the approximation will work for

[1]To prevent confusion between the adiabatic approximation for angular and radial motion and the adiabatic approximation for electronic and nuclear motion, the latter is referred to throughout this thesis as the "electronic" or "molecular" Born-Oppenheimer approximation.

helium complexes is not known. The approximation works well for argon complexes for basically three reasons:

1. Their potentials are strongly anisotropic,
2. They have relatively deep potential energy wells, and
3. They have small zero point energies.

While helium complexes have strongly anisotropic potentials, they have much shallower potential energy wells and higher zero point energies than the complexes of argon. These differences lead to much less radial localization for helium complexes than is seen in those of argon. The radial delocalization present in helium complexes is easily seen by comparing the approximate isotropic wavefunctions for a helium complex and an argon complex (Fig. 3). The wavefunction for the argon complex shown in Fig. 3 is tall and narrow while the wavefunction for the helium complex is short and broad indicating that the helium complex is able to move with much larger radial motions than the argon complex can.

As the basis of the adiabatic approximation for van der Waals complexes is the large difference between the radial and angular motions, whether the approximation is still valid for helium complexes is not known. This thesis, which applies the adiabatic approximation to the van der Waals complex He-HCN, shows that the approximation is valid for complexes with extremely floppy radial character. In addition, the thesis demonstrates that the method can be used to calculate approximate wavefunctions for Diffusion Monte Carlo studies of rotations of impurity molecules inside helium clusters.

The remainder of this thesis is organized as outlined below. Chapter 2 describes the Diffusion Monte Carlo method, its application to excited states, and the importance of calculating an accurate approximate wavefunction. In addition, a brief discussion of the Variational Monte Carlo method is given. Chapter 3 presents a

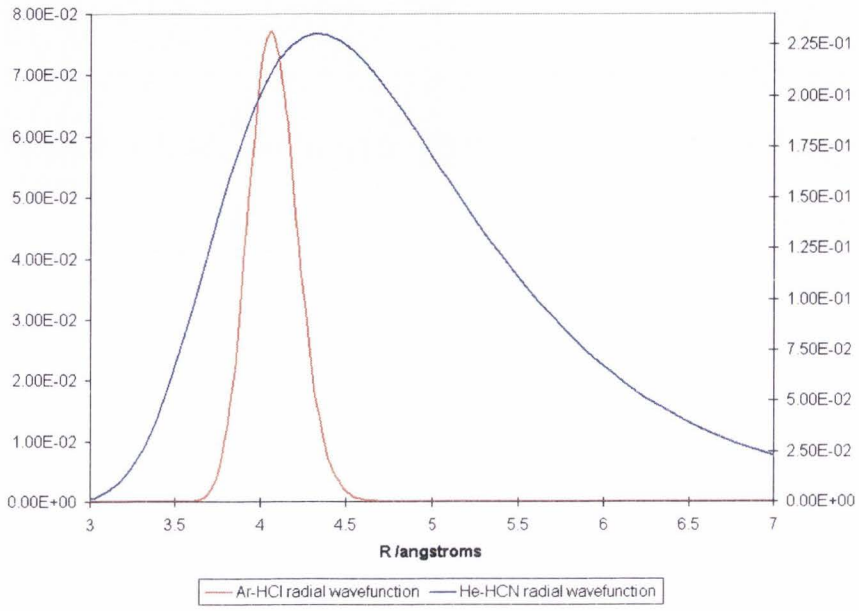

Fig. 3. Approximate radial wavefunctions for Ar-HCl and He-HCN.

description of the adiabatic method and how it was used to calculate approximate, excited state wavefunctions for the van der Waals complex He-HCN. The remainder of this chapter provides some background on the properties of helium germane to its use as a spectroscopic matrix as well as a description of how doped helium droplets are produced and spectroscopically studied.

1.2. Properties of helium

Helium is the second most abundant element in the universe and has the unique property of being the only element identified extraterrestrially before it was detected on earth. Its discovery came about through the observation of a new line in the yellow region of the spectrum of the sun's chromosphere during a solar eclipse in 1868 [4]. This observation led J. N. Lockyer (founder of the journal *Nature*) and E. Frankland to propose the existence of a new element, which they named helium (after the Greek

name for the Sun, Helios). In 1881, L. Palmieri observed the same spectral line in the spectrum of volcanic gas emitted from Mount Vesuvius and a few years later, William Ramsay finally confirmed the existence of helium on earth during his studies of atmospheric gases.

Helium consists of two isotopes, ^{4}He, which comprises 99.999863% of terrestrial helium, and ^{3}He, which is produced as a by-product of nuclear reactions. Thus, this minor isotope has only been available since the 1950s when nuclear weapon production began in earnest. The only interatomic interactions of helium are very weak van der Waals forces; the absence of any stronger forces results in several interesting properties, such as a low boiling point (4.215 K) and enthalpy of vaporization (ΔH_{vap}=0.08 $KJ\ mol^{-1}$, the lowest value for any substance).

Perhaps the most interesting property of helium is its behavior at low temperatures. When helium is under its own vapor pressure (in a vacuum), it never freezes—additional pressure must be applied to produce solid helium. The reason for this is due to the high zero point energy of helium (caused by its small mass), which allows the atoms to vibrate with large amplitude vibrations even at extremely low temperatures. At these low temperatures, quantum mechanical effects become important and cause helium to behave in a bizarre fashion.

Below around 2.2 K (the λ-point temperature, named after the λ-like shape of helium's phase diagram, see Fig. 4) for ^{4}He and 0.003 K for ^{3}He, helium undergoes a transition to superfluidity. As the temperature of the helium is cooled down to T_λ, its tumultuous boiling suddenly stops, its specific heat increases by a factor of 10, its thermal conductivity increases by 10^6, and its viscosity (as measured by its flow through a small capillary) approaches zero [5]. In addition, helium at temperatures below T_λ is able to cover all solid surfaces that are also below T_λ with a film that is a few hundred helium atoms thick. These strange properties, first observed in

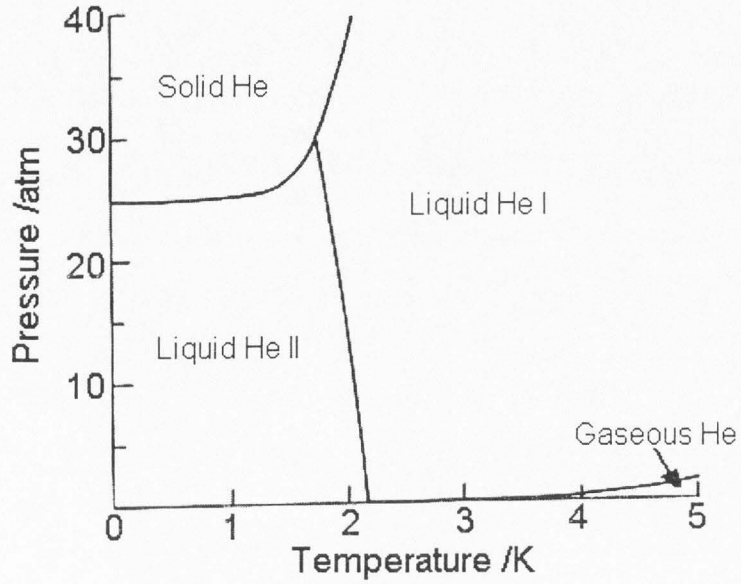

Fig. 4. Phase diagram for ^{4}He.

the 1930s, led to the recognition of a new state of matter—superfluidity, so named because the minimal resistance it experiences when flowing through a capillary is analogous to the lack of resistance electrons experience as they flow through a superconductor.

The reason why superfluidity occurs at a much lower temperature for ^{3}He than for ^{4}He arises from a difference between the two isotopes that becomes very important at low temperatures—^{4}He is a boson (it has integer nuclear spin) while ^{3}He is a fermion (it has half-integer nuclear spin). Bosons can all fall into the lowest energy state to form a Bose-Einstein condensate; fermions, on the other hand, must obey the Pauli Exclusion Principle, which prohibits two identical particles from occupying the same state. This allows bosons to form superfluids at much higher temperatures than fermions.

Just as the behavior of bulk ^{3}He and ^{4}He differ, the behavior of droplets of the two isotopes is also different. Clusters consisting of only three atoms of ^{4}He are predicted to exist [6] while the smallest number of atoms in a ^{3}He clusters is predicted to be 29 [7–11] . In addition, the size and temperature of the clusters vary with isotope. The sizes of the two clusters are $2.22N^{1/2}$ and $2.44N^{1/3}$ (where N is the number of atoms in the cluster) for ^{4}He and ^{3}He, respectively [12]. The temperature inside a ^{4}He cluster is around 0.4 K while the temperature in a ^{3}He cluster is around 0.15 K [12–14].

The placement of a strongly interacting impurity, such as SF_6, in a helium droplet creates a series of compressed shells of helium [15–20]. The first of these shells has a density that is roughly four times greater than that of bulk helium and contains 22 or 23 frozen helium atoms on average [17]. The second solvation shell contains 50 helium atoms and has a density that is around $\frac{4}{3}$ times greater than that of bulk helium [18]. In clusters containing a mixture of ^{3}He and ^{4}He atoms, the greater mass and resultant lower zero point energy of ^{4}He causes the fraction of ^{4}He in the cluster to coagulate in the center of the cluster forming a ^{4}He core that is surrounded by ^{3}He atoms [11]. Any additional impurity or dopant molecule, such as SF_6, is found inside this ^{4}He core [11].

The next section describes some of the key experimental results of studies on these systems.

1.3. Experimental studies of helium clusters

1.3.1. Production of helium clusters

Helium droplets, i.e., clusters of helium atoms consisting of more than 1000 helium atoms, were first produced by Kamerlingh Onnes in 1908 during his initial attempts to liquefy helium [21]. In 1961, Becker was able to produce a molecular beam of ^{4}He

droplets [22,23] and in 1977, Gspann duplicated the feat using droplets of ^{3}He [24].

Fig. 2 shows the typical experimental setup used to produce helium beams. The droplets are formed in a molecular beam by expanding gaseous helium through a 5-20 μm diameter nozzle at source temperatures between 5 and 30 K and source pressures of 5-80 bar [25–29]. As the gaseous helium passes through the nozzle, it immediately expands. This adiabatic expansion results in a precipitous drop in temperature and leads to condensation of the gaseous helium into droplets consisting of 10^3 to 10^8 helium atoms [5]. A few millimeters away from the nozzle, collisions between helium droplets cease and the droplets are further cooled through the evaporation of more helium atoms. At this point, the velocity of the droplets is somewhere between 200 and 400 m/s [26,28].

1.3.2. Doping of helium clusters

The sample molecule is inserted into the helium droplet using the "pickup" technique first demonstrated by Gough *et al.* in 1985 [30]. In this method, the clusters are passed through a gas cell that contains the molecules of interest. As the droplets pass through the gas cell, collisions between the droplets and the sample molecules result in the insertion of the dopant into the helium droplet.

The probability of a cluster picking up a molecule is quite good (the pickup cross section is on the order of 5000 Å) and thus the vapor pressure of the sample in the cell need only be approximately 10^{-6} to 10^{-5} mbar [5]. In fact, Lewerenz *et al.* predict that for a cluster containing 2650 helium atoms with a scattering cell dopant (the dopant they used was SF_6) pressure of 3×10^{-6} mbar, 7.7% of the clusters are expected to have captured one dopant molecule [31]. This allows for the easy insertion of species, such as amino acids, which are not very volatile. Samples such as metals and large organic molecules may be inserted after sublimation in a heated

cell [5].

As the vapor pressure of the sample in the cell is increased, the likelihood of picking up more than one sample molecule increases. The probability of capturing k molecules in a given helium cluster is given by the Poisson distribution:

$$P_k = \frac{(\sigma n l)^k}{k!} e^{-\sigma n l}, \tag{2}$$

where n is the density of impurity molecules in the scattering cell, σ is the capture cross section, and l is the length of the scattering cell. The maximum probability is achieved when $\sigma \cdot n \cdot l = k$. Assuming that the scattering cell length and the capture cross section remain constant, the maximum directly indicates the number of captured molecules. The low viscosity of the liquid helium droplet allows the captured molecules to move about virtually unimpeded in the cluster. This allows molecules to coagulate and form interesting van der Waals complexes such as $(SF_6)_4$ [32] and $(H_2O)_{16}$ [31] inside the helium droplet.

The size of these complexes is limited by the fact that the sample's internal energy, the kinetic energy of the collision between the droplet and molecule, and the binding energy of the molecule to the droplet lead to evaporation of some of the helium atoms from the droplet [5]. Measurements on a 10,000 atom helium droplet before and after insertion of a sample molecule reveal that approximately 600 helium atoms are lost for each molecule that is inserted [31]. This sets a limit on the minimum size of a helium droplet that can be "doped" using this method.

1.3.3. Spectroscopic study of doped helium clusters

After pickup of the dopant, the droplet of helium is probed using a laser. The laser, which is positioned perpendicular or antiparallel to the molecular beam, is used to excite the sample in the cluster. The absorption of a photon by the molecule causes it to become vibrationally excited and, as it relaxes back into its ground state, it releases

the excess energy into the cluster causing additional helium atoms to evaporate from the surface. Absorption of a photon can thus be detected by either measuring the decrease in the droplet ionization cross section in the mass spectrometer or by using a sensitive on-axis semiconductor bolometer[2] located downstream from the lasers [5].

1.4. Studies of molecules trapped inside helium clusters

The group of G. Scoles at Princeton performed the first spectroscopic study of a molecule trapped inside a helium droplet in 1992 [33]. In this study, a line-tunable CO_2 laser was used to excite the ν_3, or cage, vibrational mode of SF_6 (see Fig. 5). The spectrum they obtained is shown in Fig. 6. The two vibrational bands located at 945.8 and 946.4 cm^{-1} were assigned to a single molecule of SF_6 located on the surface of the helium cluster. The outer two peaks were later determined to be from contamination with nitrogen gas [34].

The estimated linewidth they found was approximately 0.3 cm^{-1}, a value much lower than the 1-10 cm^{-1} normally seen in other, heavier rare gas clusters, indicating that perturbations of the molecule by the matrix of helium atoms are significantly reduced compared to other types of clusters.

Later, Fröchtenicht *et al.* [36] obtained the first high-resolution spectrum of the ν_3 vibrational mode of SF_6 (shown in Fig. 7) using diode lasers. The spectrum obtained exhibits the P, Q, and R bands characteristic of a spherical top. In addition, the pronounced sharpness of the central line at 946.3 cm^{-1} indicates that most of the

[2]A bolometer is an instrument used to measure small amounts of radiation. It was invented in 1860 by the American scientist Samuel Pierpont Langley and is now used primarily to detect heat energy from distant sources. In astronomy, for example, bolometers measure the heat of stars. In its most basic form, the instrument consists of two platinum strips. When one strip receives radiation, the small change in resistance it experiences is compared to the other strip and measured. This difference indicates the amount of radiation received.

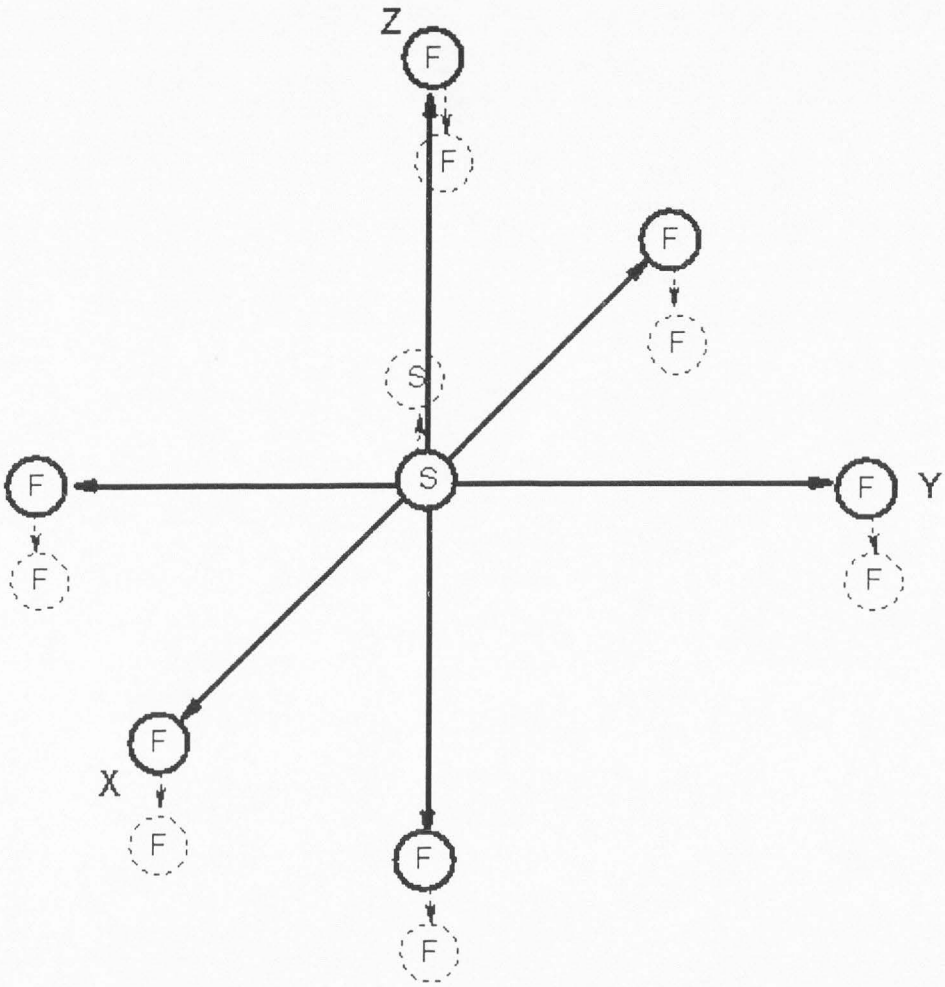

Fig. 5. The ν_3 or cage vibrational mode of SF_6. Adapted from [35].

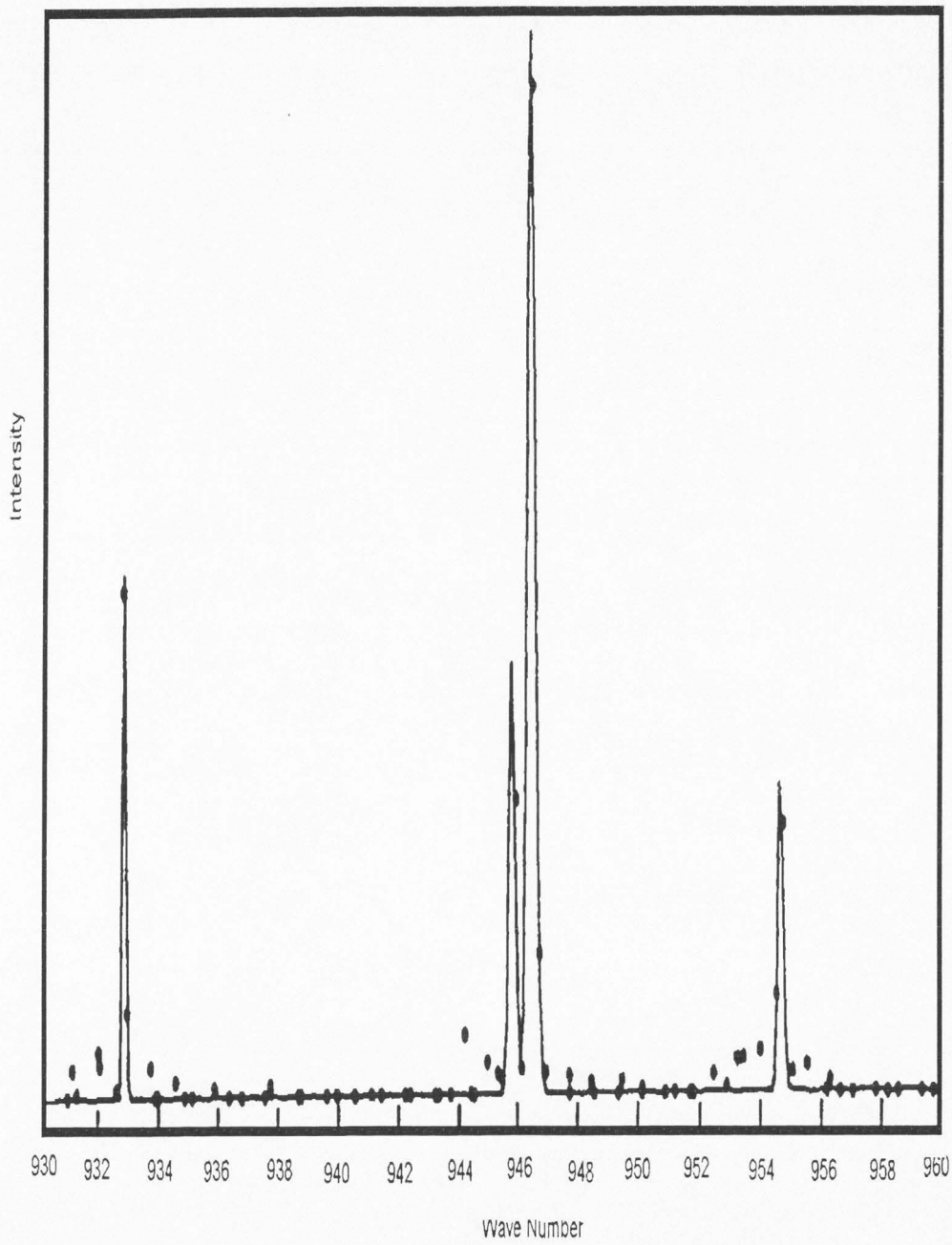

Fig. 6. IR spectrum obtained by Goyal in 1992. Adapted from [33].

perturbations from the host matrix are absent—a surprising find because of the significant inhomogeneous broadening that is usually seen in liquid matrices. These findings, together with several other experimental and theoretical studies [15,37–39], led them to the conclusion that the SF_6 was trapped at the center of the cluster and not on its surface as the previous Scoles study suggested.

Attempts to fit the spectrum to a free rotor Hamiltonian revealed that the rotational constant of the SF_6 in the cluster was reduced from its gas phase value. Using a rotational constant reduced by a factor of five from its gas phase value, Frochtenicht was able to obtain an accurate fit to the spectrum (shown in Fig. 8). As the rotational constant of a molecule is inversely proportional to its moment of inertia[3], this reduction in the rotational constant must be the result of an increase in the molecule's moment of inertia. The reasonably good fit to experimental data obtained with the reduced rotational constant led to the hypothesis that a portion of the helium density is rigidly attached to the SF_6.

This assumption was strengthened in 1995 when the first rotationally resolved spectrum of a molecule (SF_6) in a helium cluster was obtained directly [34]. The spectrum (shown in Fig. 9) shows no observable splittings of the ν_3 spectral lines, implying that the molecule resides in a symmetric environment. This rather conclusively shows that the molecule must be in the center of the cluster (a symmetric environment) and not on the surface (an antisymmetric environment). The spectrum also allowed the determination of the actual rotational and centrifugal distortionconstants of the molecule in the helium cluster.

By examining the relative intensities of the peaks, the first experimental determination of the temperature (0.37±0.01 K, a temperature in close agreement with previous theoretical studies [12–14]) inside a helium cluster was determined. Most

[3]The rotational constant, B, of a molecule is defined as $B = \hbar/4\pi cI$.

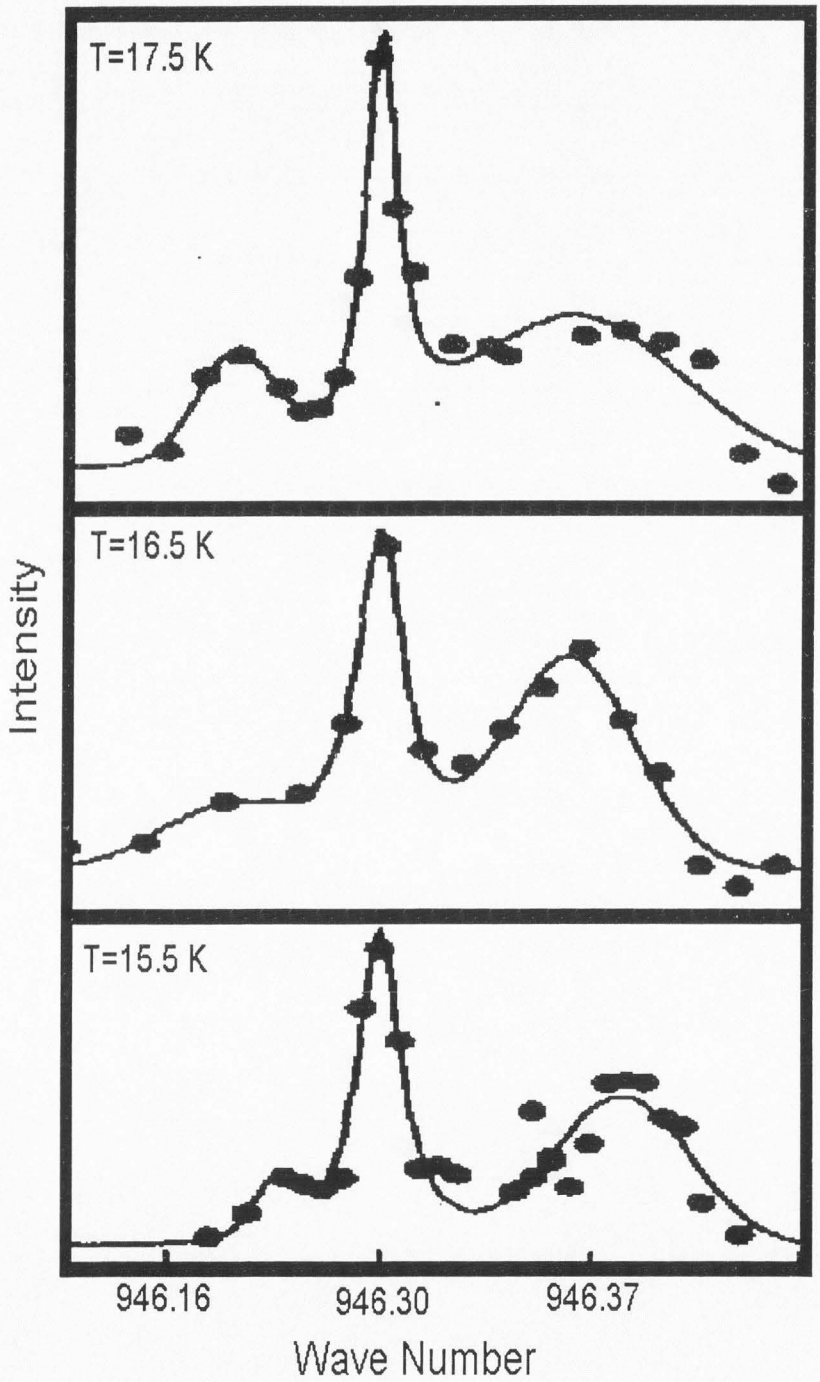

Fig. 7. High resolution spectrum of SF_6. Adapted from [36].

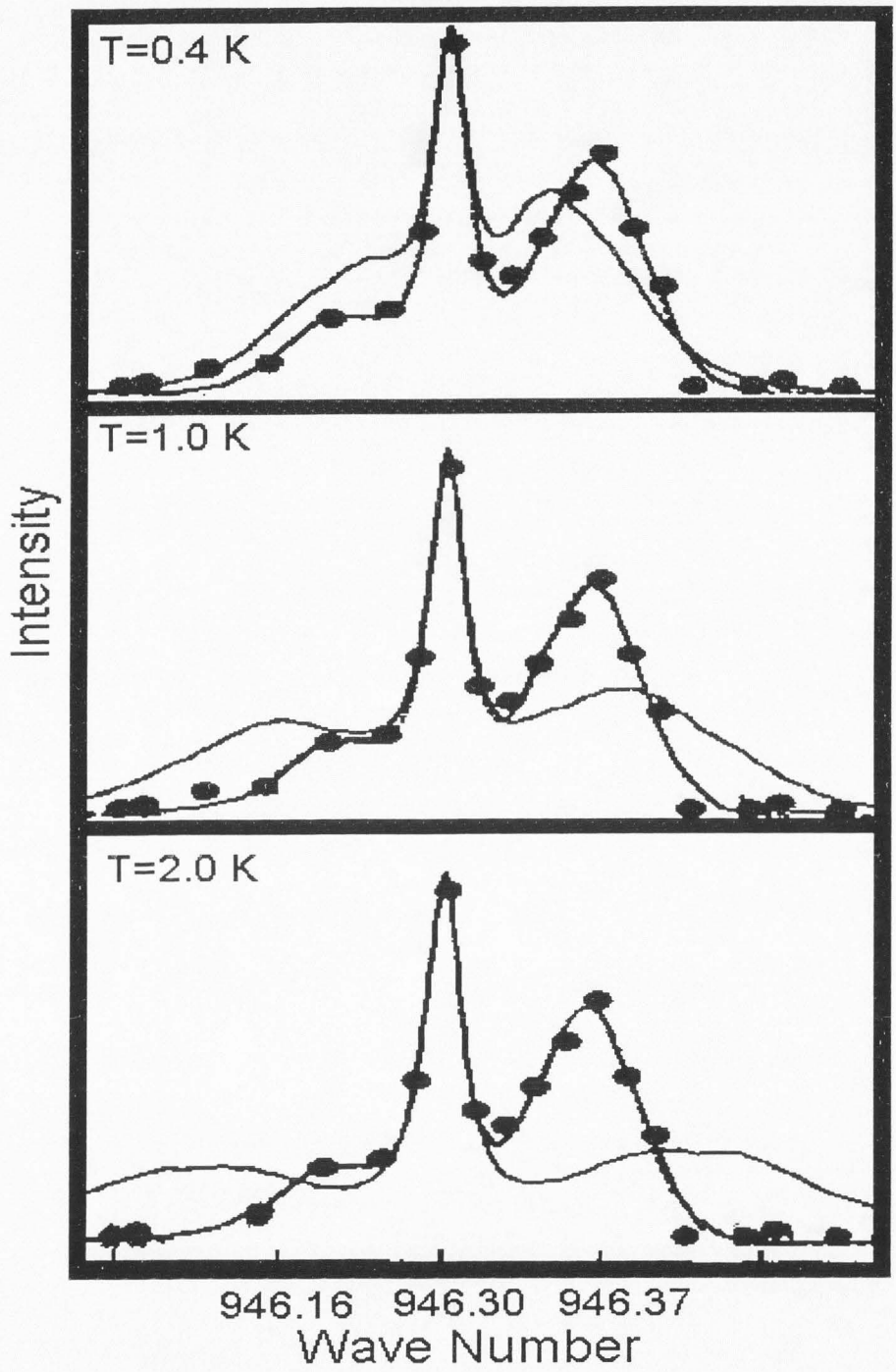

Fig. 8. Fit of high resolution spectrum of SF_6. The spectrum was calculated using a rotational constant reduced by a factor of five. Adapted from [36].

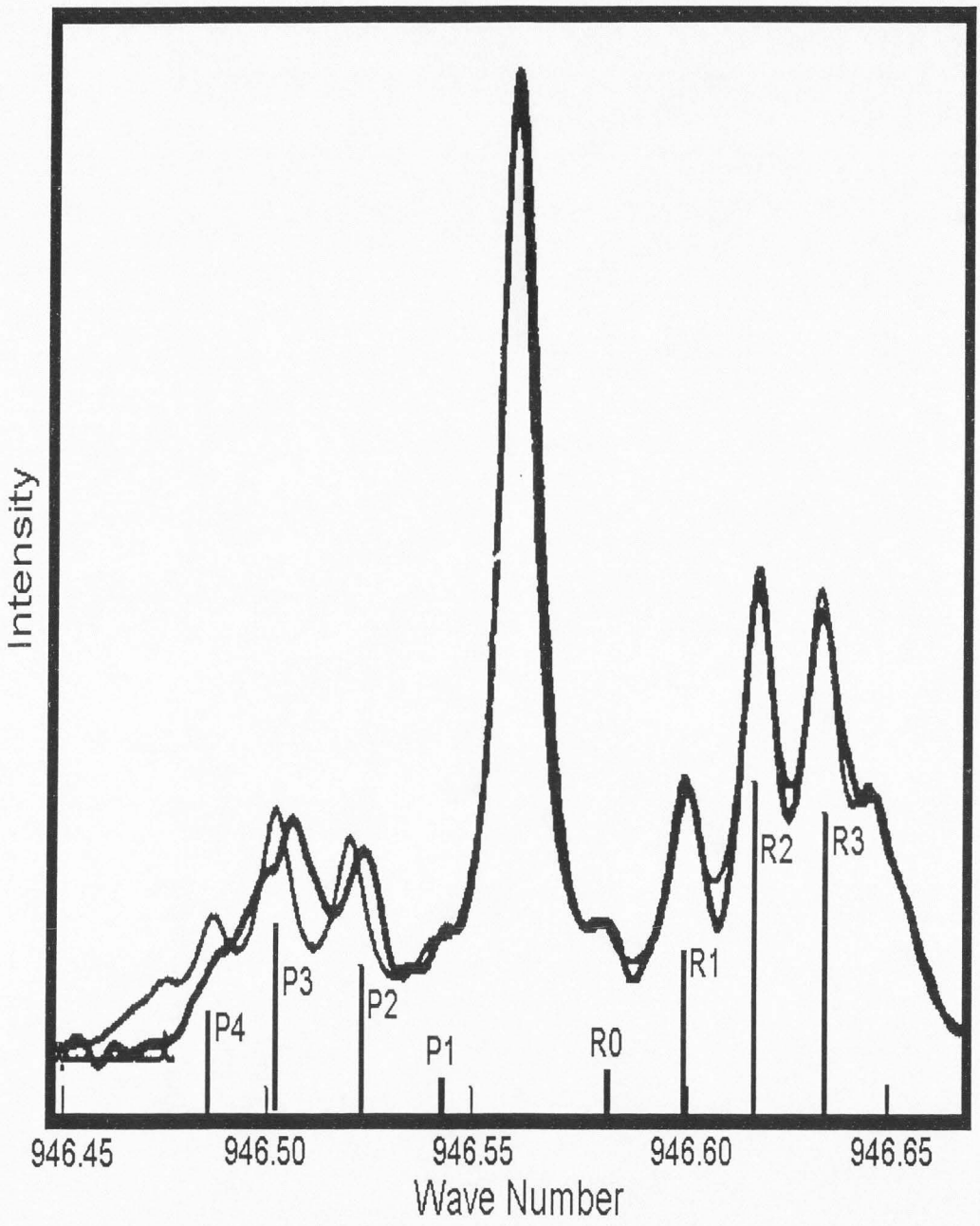

Fig. 9. Rotationally resolved spectrum of SF_6 in a helium cluster. Adapted from [34].

importantly, the observation of sharp rotational lines verified that SF_6 apparently rotates uninhibited in helium clusters.

The spacings between transitions in the P and R branches of the spectrum of SF_6 in a helium cluster are much smaller than the respective spacings in the gas phase spectrum, implying that the rotational constant of the SF_6 in the helium cluster is reduced to a fraction (37%) of its gas phase value. The authors presented a possible explanation for this—the rotation of the SF_6 is slowed down by the almost-rigid attachment of a helium atom at each of the eight global minima of the He-SF_6 potential (see Fig. 10).

This idea of eight "frozen" helium atoms surrounding the central molecule is appealing in part because theoretical calculations done by Barnett [18] show that the first solvation shell of helium atoms around an impurity molecule is frozen. However, Barnett's calculations suggest that the number of helium atoms in the first frozen shell is actually as many as 22 or 23, a number that would give a much smaller rotational constant than that actually observed in this spectroscopic study. In addition, the model is also weakened by the fact that when the dopant molecule used is OCS, the difference between the number of attached atoms and the number in the first shell is even more pronounced—only two rigidly attached helium atoms are needed to explain the reduction in the rotational constant [5].

Since the first rotationally resolved spectrum of SF_6 in a helium cluster was found, the spectra of other molecules in helium clusters have been obtained. These experiments have verified the temperature that was determined in the SF_6 experiment, indicating that the temperature of the droplet is independent of the molecule inside the cluster [14]. In addition, the apparent free rotation of several molecules has been verified though the degree by which the rotational constants of these molecules are reduced varies from molecule to molecule (see Table 1).

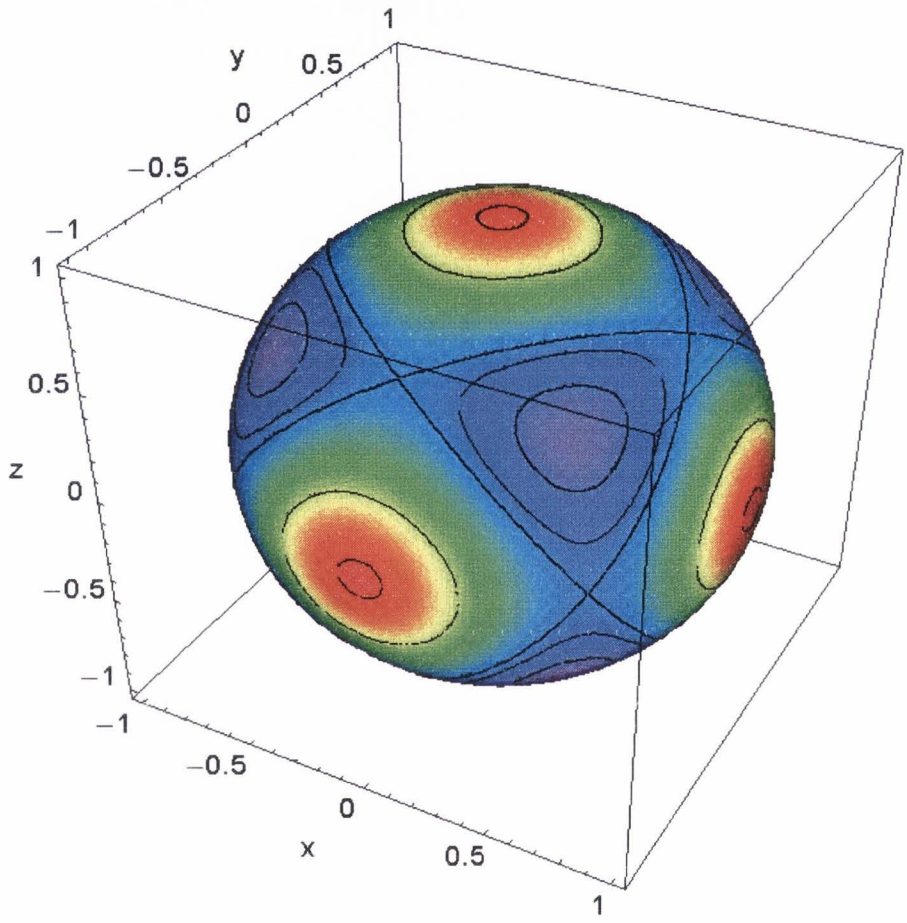

Fig. 10. He-SF_6 potential projected onto a sphere of radius r=8 a.u. The sulfur is located at the center of the sphere while the six flourine atoms are located along the coordinate axes. The potential minima are shown in purple and all distances are given in a.u.

Table 1
The rotational constants of some molecules inside helium clusters.

Molecule	B_0 (cm^{-1})	B in ^4He_N (% of B_0)	Reference
H_2O	27.8, 14.5, 9.3	~ 100	[40]
HF	20.56	~ 100	[41]
NH_3	9.94	76	[42]
$(CHO)_2$	1.8	39	[43]
HCN	1.47	81	[44]
HCCH	1.17	89	[45]
DCCH	0.99	88	[45]
CH_3CCH	0.28	25	[45]
OCS	0.20	33	[46]
HCCCN	0.15	33	[47]
HCCCCH	0.146	32	[45]
CF_3CCH	0.096	36	[45]
SF_6	0.091	37	[34]
$(CH_3)_3CCCH$	0.089	33	[48]
$(CH_3)_3SiCCH$	0.065	22	[48]

Lee recently presented a compelling argument as to why the degree of reduction of the rotational constant of molecules in helium clusters varies from molecule to molecule [49]. Her numerically exact diffusion Monte Carlo calculations of the low-lying rotational energy levels of SF_6 in a helium cluster revealed that a fraction of the helium density (eight helium atoms) is able to instantaneously adjust to the rotation of the SF_6 if the rotation of the molecule is slow enough. Conversely, for molecules with large enough rotational constants, the helium atoms are unable to keep up with the fast rotation of the molecule resulting in the molecule's rotational constant being unaffected. Thus, as can be seen in Table 1, the degree by which the rotational constant in a helium cluster is reduced is dependent on the magnitude of the molecule's gas phase rotational constant.

Another important question which faced researchers was whether the behavior of impurity molecules in liquid helium clusters is due simply to the extremely cold temperatures of the cluster or to their superfluid nature. This question began to be answered through a series of experiments done by Grebenev *et al.* [2], which they called the "Microscopic Andronikashvili Experiment."

In the original Andronikashvili experiment [50], performed in 1946, a small disk torsional oscillator was immersed in superfluid helium. As the temperature of the liquid helium was decreased, the rotational constant of the disk did not decrease as would be expected in a classical fluid; rather, it sharply increased as the temperature went below T_λ.

The Grebenev experiment replaced the disk used in the original experiment with the rod-like molecule OCS and the infrared spectrum of OCS in both ^{3}He and ^{4}He was collected. ^{3}He clusters are colder than ^{4}He clusters ($T_{^3He}$=0.15 K, $T_{^4He}$=0.40 K); however, as previously mentioned, the superfluid transition temperature for ^{3}He is much lower than that of ^{4}He ($T_{\lambda,^3He}$=3 mK, $T_{\lambda,^4He}$=2.2 K). Thus, at the temper-

atures in the experiment, the ^{4}He clusters are superfluid while the ^{3}He clusters are not even though the ^{3}He clusters are considerably colder than the ^{4}He clusters.

As can be seen from the ν_3 infrared spectrum of OCS in ^{4}He and ^{3}He (Fig. 11), the ^{3}He spectrum does not display sharp rotational peaks as the ^{4}He spectrum does. Rather, the spectrum is characteristic of a molecule in a classical fluid, i.e., it only has a broad band indicative of rotational diffusion [51–53]. As the number of ^{4}He atoms is increased in the cluster, the rotationally resolved spectrum gradually begins to appear (see Fig. 12). When the number of ^{4}He atoms in the cluster approaches 60, corresponding to two solvation shells surrounding the OCS, the spectrum of the freely rotating OCS appears. This experiment conclusively showed that the surprising behavior of molecules in liquid helium clusters is due to the superfluid character of the clusters although it did not fully explain the mechanism by which the helium cluster is able to slow down the rotation of some molecules.

In addition to this result, the experiment also revealed the minimum number helium atoms that are needed to form a superfluid. Their experimental result of 60 helium atoms necessary for superfluidity closely agrees with the 64 atoms Sindzingre predicted in 1989 [54].

1.5. Conclusions

Helium clusters offer a tremendous opportunity in both chemistry and physics. From the perspective of chemical physics, helium clusters may become a valuable spectroscopic tool that can be used to study molecules which currently elude spectroscopic study. From the physics standpoint, doped helium clusters provide a way to understand superfluidity in finite-sized droplets. More specifically, they offer the possibility to explore the role of size on superfluidity. This is an important objective because while there are many different ways to describe superfluidity on the

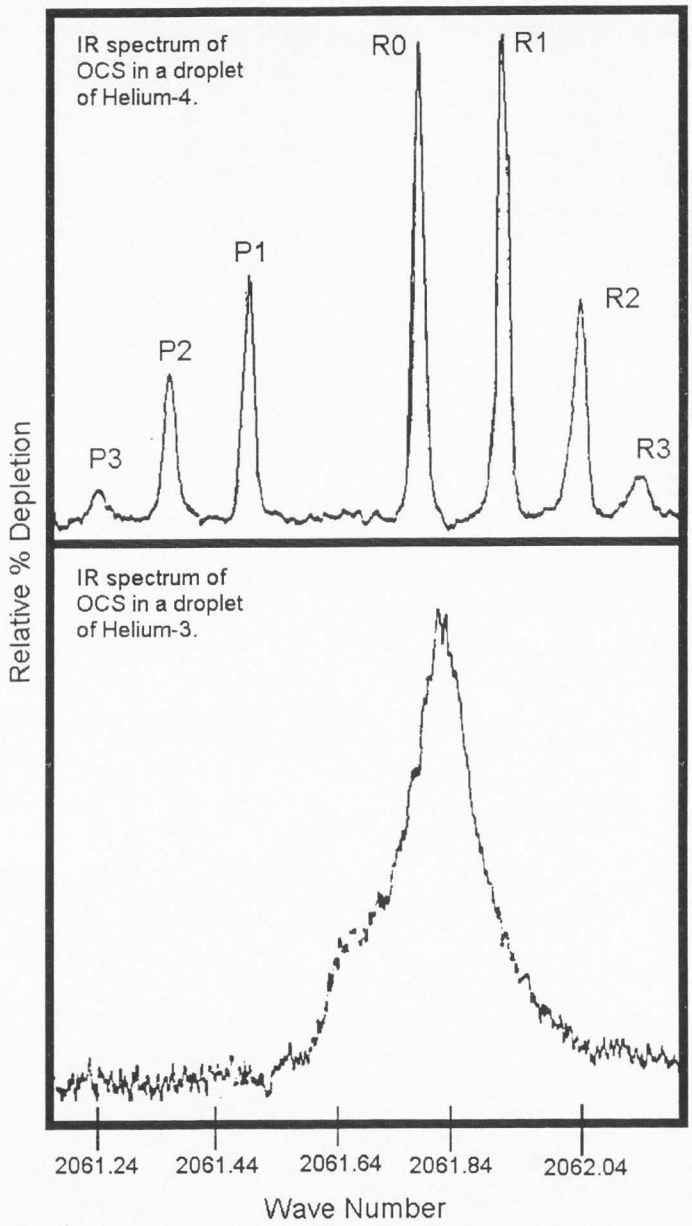

Fig. 11. Spectrum of OCS in (a) ^{4}He and (b) ^{3}He. Adapted from [2].

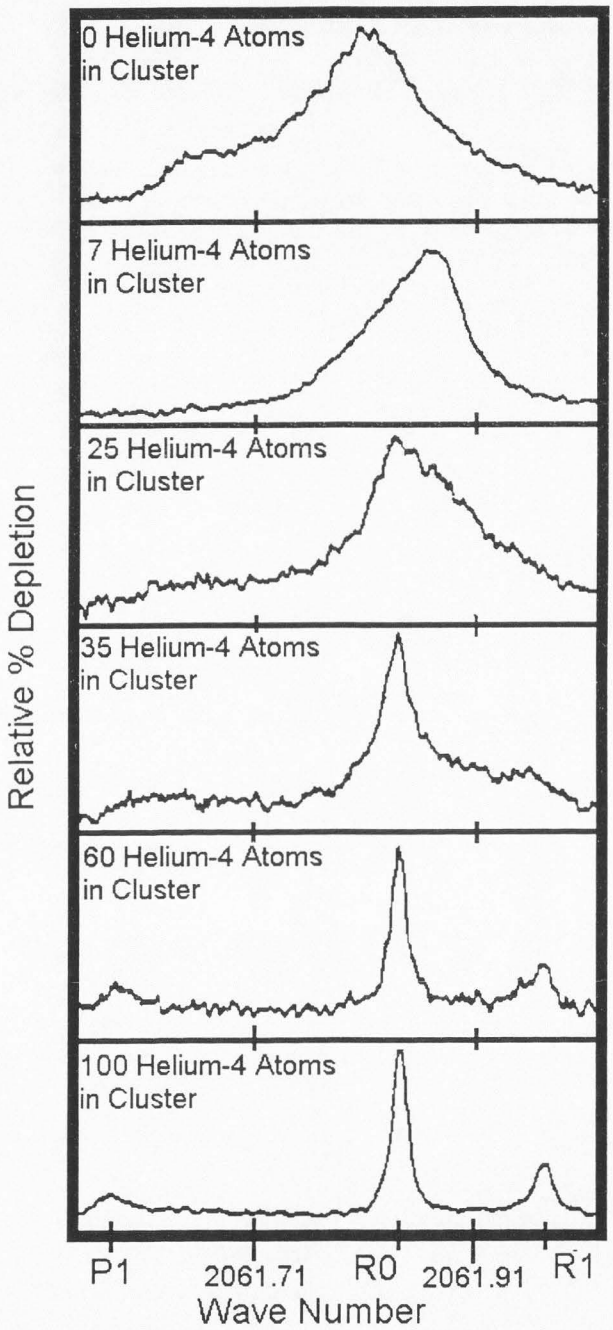

Fig. 12. Spectrum of OCS as the number of ^{4}He atoms increases. Adapted from [2].

bulk level, not much is known about the behavior of superfluids on a finite scale. To explore these avenues, a better understanding of the behavior and properties of impurity molecules trapped inside helium clusters must be reached.

The next chapter deals with a method that can be used to perform calculations on the rotational states of these systems; more specifically, it concentrates on the use of the method to calculate excited state wavefunctions and energies for helium clusters. To use this method to calculate excited states, an approximate wavefunction that can accurately predict the location of the nodes of the wavefunction must be calculated. In addition to allowing the calculation of excited states, an approximate wavefunction is also useful because it can be used to increase the efficiency of the method.

References

[1] G. Scoles, K. K. Lehmann, Science 279 (1998) 2065.

[2] S. Grebenev, J. Toennies, A. Vilesov, Science 279 (1998) 2083.

[3] S. Holmgren, M. Waldman, W. Klemperer, J. Chem. Phys. 67 (1977), 4414.

[4] N. N. Greenwood, A. Earnshaw, Chemistry of the Elements 2nd edition, Butterworth-Heinemann, Oxford, 1998.

[5] J. Toennies, A. F. Vilesov, Ann. Rev. Phys. Chem 49 (1998) 1.

[6] M. Lewerenz, J. Chem. Phys. 106 (1997) 4596.

[7] V. Pandharipande, S. Pieper, R. Wiringa, Phys. Rev. B 32 (1985) 3341.

[8] S. Stringari and J. Treiner, J. Chem. Phys 87 (1987) 5021.

[9] P. Joyes, R. Tarento, J. Van de Walle Z. Phys. D 20 (1993) 233.

[10] S. Stringari, Phys. Lett. A 107 (1985) 36.

[11] M. Barranco, M. Pi, S. M. Gatica, E. S. Hernandez, J. Navarro, Phys. Rev. B 56 (1997) 8997.

[12] D. Brink, S. Stringari, Z. Phys. D 15 (1990) 257.

[13] A. Guirao, M. Pi, M. Barranco, Z. Phys. D 21 (1991) 185.

[14] J. Gspann, in: S. Datz (Ed.), Physics of Electronic and Atomic Collisions, North Holland, Amsterdam, 1982, p. 79.

[15] F. Dalfovo, Z. Phys. D 29 (1994) 61.

[16] S. M. Gatica, E. S. Hernandez, M. Barranco, J. Chem. Phys. 107 (1997) 927.

[17] R. N. Barnett, K. B. Whaley, J. Chem. Phys. 99 (1993) 9730.

[18] R. N. Barnett, K. B. Whaley, Z. Phys. D 31 (1994) 75.

[19] S. A. Chin, E. Krotscheck, Phys. Rev. B 52 (1995) 10405.

[20] M. Barranco, E. S. Hernandez, Phys. Rev. B 49 (1994) 12078.

[21] H. Kamerlingh Onnes, Commun. Phys. Lab. Leiden 105 (1908) 744.

[22] E. W. Becker, R. Klingelhöfer, P. Lohse, Z. Naturforsch. Teil A 16 (1961) 1259.

[23] E. W. Becker, Z. Phys. D 3 (1986) 101.

[24] J. Gspann, G. Krieg, H. Vollmar, J. Phys 38 (1977) C2171.

[25] H. Buchenau, R. Götting, R. Minuth, A. Scheidemann, J. P. Toennies, in: G. E. A. Meier and F. Obermeier (Eds.), Flow of Real Fluids, Springer-Verlag, Berlin, 1985, p. 157.

[26] H. Buchenau, R. Götting, R. Minuth, A. Scheidemann, J. P. Toennies, Proc. Int. Sym. Rerefied Gas Dyn., 15th, Grado, Italy, July, 1986, p. 197.

[27] J. P. Toennies, Proc. Int. School Phys. "Enrico Fermi," Course CVII, Varenna, June, 1990, p. 597.

[28] H. Buchenau, E. L. Knuth, J. Northby, J. P. Toennies, J. Chem. Phys. 92 (1990) 6875.

[29] H. Buchenau, J. P. Toennies, J. A. Northby, J. Chem. Phys. 95 (1991) 8134.

[30] T. E. Gough, M. Mengel, P. A. Rowntree, G. Scoles, J. Chem. Phys., 83 (1985) 4958.

[31] M. Lewerenz, B. Schilling, J. P. Toennies, J. Chem. Phys. 102 (1995) 8191.

[32] M. Hartmann, R. Miller, J. Toennies, A. Vilesov, Science 272 (1996) 1631.

[33] S. Goyal, D. L. Schutt, G. Scoles, Phys. Rev. Lett. 69 (1992) 933.

[34] M. Hartmann, R. Miller, J. Toennies, A. Vilesov, Phys. Rev. Lett. 75 (1995) 1566.

[35] D. Eichenauer, R. LeRoy, J. Chem. Phys. 88 (1987) 2898.

[36] R. Fröchtenicht, J. Toennies, A. Vilesov, Chem. Phys. Lett. 229 (1994) 1.

[37] M. A. McMahon, R. N. Barnett, K. B. Whaley, J. Chem. Phys. 104 (1996) 5080.

[38] F. Ancilotto, G. DeToffol, F. Toigo, Phys. Rev. B 52 (1990) 16125.

[39] F. Ancilotto, M. W. Cole, G. DeToffol, P. B. Lerner, F. Toigo, J. Low Temp. Phys. 101 (1995) 325.

[40] F. Huisken, private communication.

[41] D. Blume, M. Lewerenz, F. Huisken, M. Kaloudis, J. Chem. Phys. 105 (1996) 8666.

[42] M. Behrens, U. Buck, R. Fröchtenicht, M. Hartmann, F. Huisken, J. Chem. Phys. 105 (1996) 6128.

[43] J. Toennies, private communication.

[44] K. Nauta and R. Miller, Science 283 (1999) 1895.

[45] G. Scoles, private communication.

[46] S. Grebenev, M. Hartmann, M. Havenith, B. Sartakov, J. Toennies, A. Vilesov, J. Chem. Phys. 112 (2000) 4485.

[47] K. Nauta, R. Miller, Faraday Discussions 113 (1999) 261.

[48] R. Miller, private communication.

[49] E. Lee, D. Farrelly, K. B. Whaley, Phys. Rev. Lett. 83 (1999) 3812.

[50] E. L. Andronikashvili, J. Phys. U.S.S.R. 10 (1946) 201.

[51] A. Burshstein and S. Temkin, Spectroscopy of Molecular Rotation in Gases and Liquids, Cambridge University Press, Cambridge, 1994.

[52] R. Clark, R. Hester (Eds.), Advances in Spectroscopy, vol. 23, Wiley, Singapore, 1995.

[53] R. Gordon, J. Chem. Phys. 44 (1966) 1830.

[54] P. Sindzingre, M. L. Klein, D. M. Ceperley, Phys. Rev. Lett. 63 (1989) 1601.

CHAPTER 2

MONTE CARLO METHODS

In the previous chapter, the properties and behavior of impurity molecules inside ultra-cold clusters of helium atoms were examined. The intriguing results found from the numerous experimental and theoretical studies of these systems suggest that a closer theoretical examination could offer valuable insight into the molecular properties of superfluidity. In addition, an improved understanding of doped helium clusters could lead to the eventual use of helium clusters as a spectroscopic matrix that can be used to study large molecules.

The most surprising result of these studies was the appearance of rotational structure in the spectra of molecules inside liquid helium clusters. To better understand the cause of this behavior, a detailed study of the rotational dynamics of impurity molecules in helium clusters is necessary.

The calculation of rotational states for these systems is complicated by the fact that an analytical solution to the Schrödinger equation, the equation which governs the behavior of quantum systems at the nonrelativistic level, is impossible to obtain because of the many interacting bodies that must be considered. In fact, any realistic system, whether it be quantum mechanical or classical, with three or more interacting bodies leads to a Hamiltonian that is too complicated to solve analytically. This fact means that approximations and numerical methods must be relied on to calculate the eigenvalues and eigenfunctions of these systems.

One of the most useful sets of numerical methods that can be used to solve for many-body systems are the Monte Carlo methods. Monte Carlo methods, named after the reliance on probabilities they share with the games played in the casinos of Monte Carlo, are very powerful because of their accuracy and efficiency. This

chapter provides a description of two Monte Carlo methods, Diffusion Monte Carlo, which can be used to calculate rotational states for impurity molecules in helium clusters, and Variational Monte Carlo, which is often used to calculate trial wavefunctions for Diffusion Monte Carlo studies. In addition, results of the calculation of a trial wavefunction for the ground state of the van der Waals complex He-SF_6 using Variational Monte Carlo are given.

2.1. Monte Carlo methods

Monte Carlo methods use statistical sampling to approximate solutions of mathematical problems. The use of statistics to solve problems dates back at least to the eighteenth century to the work of the French mathematician Georges Buffon who used a statistical method to estimate the value of π. With a few isolated exceptions, Monte Carlo methods were not used as a research tool until they were formally developed by von Neumann and Ulam during their work on the Manhattan project [1–4].

The ability to quickly and easily generate random numbers that came as a result of the improvement in computing technologies in the past sixty years has allowed Monte Carlo methods to become one of the most widely used approximation methods in quantum physics. The motivation for using Monte Carlo methods comes from the accuracy the methods afford as well as the relatively minimal computer time they require to obtain that high level of accuracy [5].

The basic idea of Monte Carlo is easily illustrated with the use of an example. Consider any function $f(x)$; the area under f from point a to point b, is given simply by

$$\int_a^b f(x)dx. \tag{1}$$

Suppose that this integral is impossible to solve analytically. Monte Carlo methods

can be used to estimate the area under the curve by picking N random, uniformly distributed points in an area A that contains $f(x)$. The ratio of the number of points that lie under the curve, N_{under}, to the total number of points is equal to the area under the curve, A_{under}, divided by the total area, i.e.,

$$\frac{N_{under}}{N} = \frac{A_{under}}{A}. \tag{2}$$

Solving for A_{under} gives

$$A_{under} = A\frac{N_{under}}{N}. \tag{3}$$

Thus, by simply counting the number of random points under the curve, the area under it can be determined.

There are several different Monte Carlo methods that are often used in quantum calculations. One of the most powerful Monte Carlo methods is the Diffusion Monte Carlo method. This method is able to calculate numerically exact ground state energies and wavefunctions without any prior knowledge of the wavefunction of the system. In addition, it is the only viable option that can be used for calculations on large helium clusters because it is the only method that can deal with the many degrees of freedom at a uniform level of accuracy [6,7]. The next section outlines the basis of the DMC method.

2.2. Diffusion Monte Carlo

The name Diffusion Monte Carlo comes from the similarity between the time-dependent Schrödinger equation and a modified general diffusion equation. This can be seen by examining the equations, which are shown respectively in equations 4 and 5:

$$i\hbar\frac{\partial\psi}{\partial t} = H\psi = -\frac{\hbar^2}{2\mu}\nabla^2\psi + V\psi \tag{4}$$

$$\frac{\partial C}{\partial t} = D\nabla^2 C, \tag{5}$$

where C is the concentration of the particles undergoing diffusion and D is the diffusional constant. Modification of the diffusion equation by the addition of a first-order rate term,

$$-kC, \tag{6}$$

results in an equation,

$$\frac{\partial C}{\partial t} = D\nabla^2 C - kC, \tag{7}$$

that is close to having the same form as the time-dependent Schrödinger equation.

In fact, the only structural difference between the two equations is the presence of imaginary terms in the Schrödinger equation. These terms may be removed by replacing the real time, t, with imaginary time, $\tau \equiv \frac{it}{\hbar}$. This transformation, also known as a Wick rotation of time [8], leads to the imaginary-time-dependent Schrödinger equation,

$$\frac{\partial \psi}{\partial \tau} = \frac{\hbar^2}{2\mu}\nabla^2\psi - V\psi, \tag{8}$$

which is an ordinary differential equation of the exact same form as a diffusion equation modified by the presence of a first-order rate term (a "source" or "sink").

As was first suggested by Fermi as cited in [8] and Wigner [9], a diffusion equation modified by the addition of a rate term can be solved using a random walk procedure. As equations 7 and 8 are completely analogous, any method that can be used to solve one can be used to solve the other.

To use a random walk to solve equation 8, a shift in energy scale is necessary [8]. This shift, by an arbitrary energy E_{ref}, results in the following Schrödinger equation,

$$\frac{\partial \psi}{\partial \tau} = \frac{\hbar^2}{2\mu}\nabla^2\psi - [V - E_{ref}]\psi. \tag{9}$$

After this shift in energy, the probability that a particle "reacts" is now proportional to -$[V - E_{ref}]$ rather than V.

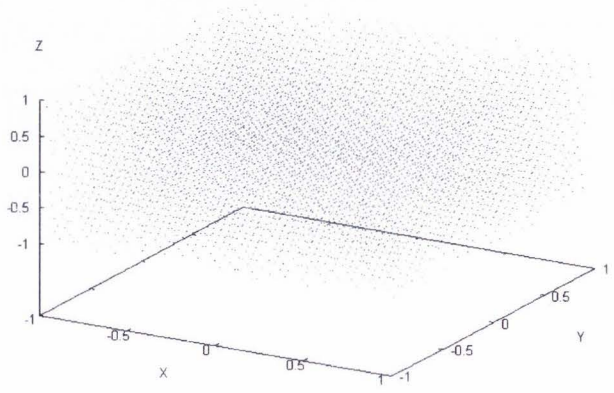

Fig. 13. An initial distribution of walkers. A random distribution of walkers for an arbitrary system with a potential minimum located at the origin.

The random walk procedure begins with an initial ensemble of theoretical particles termed "walkers" that are distributed throughout position space (see Fig. 13). Each walker is randomly moved to a new position at each time step, $\delta\tau$; after each move, the potential at the new position is evaluated to see if the move resulted in the walker being in a region of higher or lower potential. If the potential, $V(\mathbf{r})$, at the new position is greater than E_{ref}, the walker is destroyed; alternatively, if the move resulted in the walker being in a position that has a lower potential value than E_{ref} the walker gives birth to additional[1] walkers. This "birth/death" process the walkers undergo eventually results in a "steady-state" distribution that fluctuates about an

[1]The number of walkers that are produced is dependent on the magnitude of the difference between E_{ref} and V. Typically, the maximum number of walkers that are "born" after a "good" move is not more than three. If the move results in a potential value equal to E_{ref}, then neither birth nor death occurs.

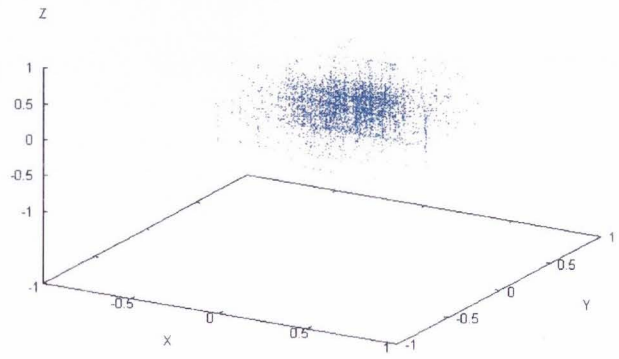

Fig. 14. A converged distribution of walkers. A distribution of walkers for an arbitrary system with a potential minimum located at the origin after many time steps.

average steady-state distribution centered around regions where the potential or sink term is lowest (see Fig. 14 and Fig. 15).

This final, converged distribution of the walkers is identical to the solution of the time-independent Schrödinger equation. Why this is so can be seen by considering the solution to the imaginary-time-dependent wavefunction, $\psi(\mathbf{r}, \tau)$, which is obtained by integrating Eq. 8. This solution may be expressed in terms of an expansion of the eigenfunctions, ϕ_i, and eigenvalues, E_i, of the Hamiltonian (i.e., $H\phi = E\phi$),

$$\psi(\mathbf{r}, \tau) = \sum_{i=0}^{\infty} c_i \phi_i(\mathbf{r}) e^{-\frac{1}{\hbar}[E_i - E_{ref}]\tau}. \tag{10}$$

Examination of this sum reveals that as $\tau \rightarrow \infty$, all time dependence in the solution is removed. In addition, the only state that will appreciably contribute at large values of τ is the one with the lowest eigenvalue—the ground state. Thus as $\tau \rightarrow \infty$, the

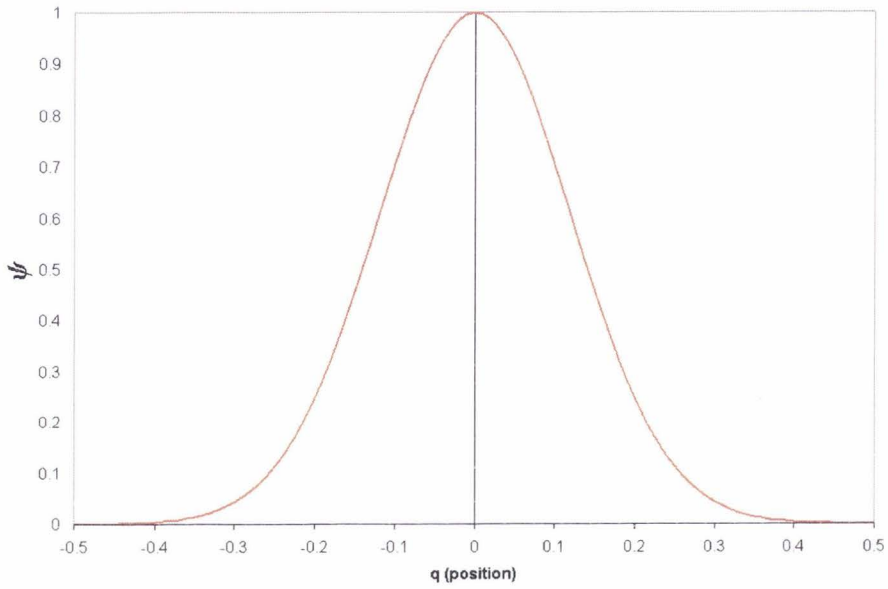

Fig. 15. The probability distribution of the walkers. The probability of finding a walker for an arbitrary system with a potential minimum located at the origin after convergence as a function of position.

converged walker distribution corresponds to the ground state wavefunction of the time-independent Schrödinger equation.

2.3. Calculation of excited states using Diffusion Monte Carlo

2.3.1. The fixed-node approximation

One aim of the study of dopant molecules inside helium clusters is to be able to better understand the behavior and properties of excited rotational states of the molecule in the cluster. To do this, the energies and wavefunctions of the excited states must be calculated. DMC is limited in this regard because it can only be applied to systems that have wavefunctions that are positive-definite.

The reason for this arises from the interpretation of the probability distribution of the walkers as the wavefunction, ψ, of the system. Physically, the probability distribution is represented by $|\psi|^2$, not ψ. This fact eliminates any wavefunction with negative regions (i.e., excited states and fermions) from consideration because a probability distribution with negative regions is nonsensical.

The problem may be approximately overcome through the use of the so-called fixed-node approximation, first introduced by Anderson in 1975 [10]. The basic idea of the approximation is to place an infinite potential at each node of the wavefunction and then study each nodal region separately. To illustrate the use of the method, consider the well-known and oft-used example of a particle in a one-dimensional box.

The solution to such a system with box of length L is given by

$$\psi = \left(\frac{2}{L}\right)^{\frac{1}{2}} \sin\left[\frac{n\pi x}{L}\right] \tag{11}$$

where

$$n = 1, 2, 3, ... \tag{12}$$

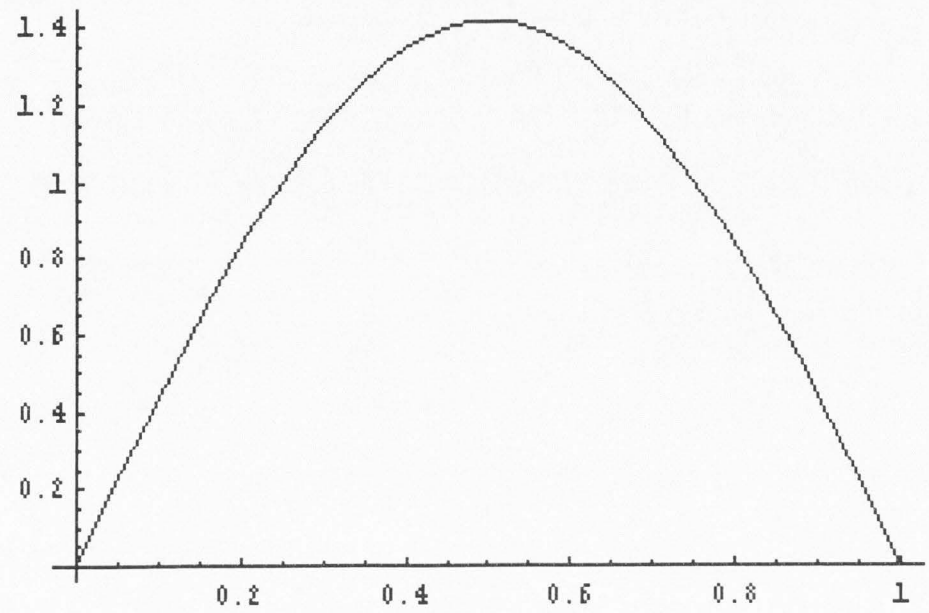

Fig. 16. Ground state wavefunction for a particle in a box of length L.

The ground state of this system,

$$\psi_{n=1} = \left(\frac{2}{L}\right)^{\frac{1}{2}} \sin\left[\frac{\pi x}{L}\right], \tag{13}$$

can easily be solved for using DMC because it has no nodes (see figure 16). However, the first excited state,

$$\psi_{n=2} = \left(\frac{2}{L}\right)^{\frac{1}{2}} \sin\left[\frac{2\pi x}{L}\right], \tag{14}$$

has a node at $\frac{L}{2}$ and is negative for all $x > \frac{L}{2}$ (see figure 17).

With the use of the fixed-node approximation, DMC can be used to find the solution of the first excited state. How to do this can be seen by noticing that for $0 \leq x \leq \frac{L}{2}$, the wavefunction for the first excited state is just the ground state solution for a particle in a box of length $\frac{L}{2}$. Noting the symmetry between the two halves of the first excited state reveals that the solution for the region $\frac{L}{2} \leq x \leq L$

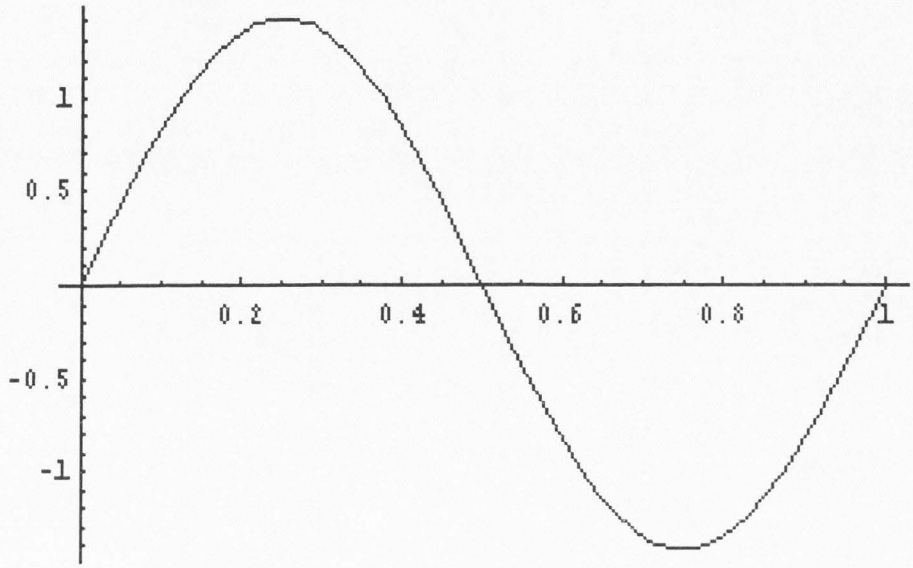

Fig. 17. First excited state wavefunction for a particle in a box of length L.

is just the negative of the solution to the first region. Piecing these two solutions together gives the solution for the first excited state.

Stated shortly, the fixed-node approximation works by imposing appropriate boundary conditions on the excited-state to obtain continuous and contiguous regions containing no nodes themselves, but whose boundaries are the nodal surfaces. Of course, the positions of the nodes are not known beforehand. In these cases, an approximate or trial wavefunction, Ψ_T, must be used to predict the location of the nodes of the wavefunction. A "good" trial wavefunction that can accurately predict nodal location is important because the error in the approximation is directly dependent on the difference between the predicted node and the true node. In fact, the error associated with the fixed-node approximation is zero if the exact location of the nodes is known; otherwise, the approximation will always increase the energy [11].

2.3.2. Importance sampling

To decrease the amount of time required for a DMC calculation, another modification called importance sampling is often used. Variations in the potential at different values of $\delta\tau$ make the DMC method inefficient. Using importance sampling, which was first introduced by Kalos in 1974 [12], the efficiency of DMC can be improved by a factor on the order of two or three orders of magnitude.

The essence of the method is to sample the function

$$f = \psi(\mathbf{r}, \tau)\Psi_T, \tag{15}$$

where Ψ_T is a trial wavefunction, instead of the function $\psi(\mathbf{r}, \tau)$. Multiplication of equation 8 by Ψ_T and making use of the definition of f results in

$$\frac{\partial f}{\partial \tau} = \frac{\hbar^2}{2\mu}\nabla^2 f - \frac{\hbar^2}{2\mu}\nabla \cdot (f \nabla \ln \Psi_T) - [\frac{H\Psi_T}{\Psi_T} - E_{ref}]f(\mathbf{r}, \tau) \tag{16}$$

The term

$$\nabla \ln \Psi_T$$

is a vector field, often called the quantum force [13] that shepherds the walkers to regions where $|\Psi_T|^2$ is greatest, thus modifying the diffusion process in a manner analogous to the way particles undergoing Brownian motion are affected by an external field [13]. In addition to reducing variations in the potential, this modification also increases the efficiency of DMC by replacing the potential, V, with the "local energy,"

$$E_L = \frac{H\Psi_T}{\Psi_T}. \tag{17}$$

This causes walkers to be reproduced with a probability that is now dependent on $[\frac{H\Psi_T}{\Psi_T} - E_{ref}]$ rather than $[V(\mathbf{r}) - E_R]$. For a reasonably good trial wavefunction, this leads to significantly smaller fluctuations.

As with the fixed-node approximation, importance sampling requires the use of a trial wavefunction. Importance sampling is what often makes DMC the method of choice for systems containing hundreds or even thousands of particles (such as helium clusters) because it can determine accurate energies and wavefunctions in a much shorter time than can other methods that provide similar accuracy. This is shown in Table 2, which was adapted from reference [14]. The table compares the accuracy and time needed to calculate energies for a cluster of ten carbon atoms by comparing the percentage of correlation energy[2] recovered and the relative time needed for the calculation.

2.3.3. The trial wavefunction

For both importance sampling and the fixed-node approximation in DMC, a good trial wavefunction is critical. Thus one of the challenges that must be overcome for DMC calculations that utilize either of these techniques is the calculation of a trial wavefunction that provides a reasonably good estimate of the behavior of the true wavefunction of the system. In particular, for calculations of excited states of molecules in helium clusters, the location of the nodes of the wavefunction must be predicted to a reasonable level of accuracy.

In importance sampling, the form of the approximate wavefunction in all regions is important. With this method, as the trial wavefunction approaches the true wavefunction, the local energy approaches the exact eigenvalue quadratically fast [14]. Thus, a good trial wavefunction for importance sampling is one that can accurately mimic the behavior of the true wavefunction everywhere.

The requirements for a wavefunction that will be used in the fixed-node approxi

[2]The correlation energy, E_{corr}, is defined as the difference between the exact energy, E_{exact} and the Hartree-Fock energy, E_{HF}, i.e., $E_{corr} = E_{exact} - E_{HF}$.

Table 2
A comparison of different methods used for many body calculations

Method	Percentage of correlation energy recovered, $E_{corr} = E_{exact} - E_{Hartree-Fock}$	Relative time needed to calculate the energy of a C_{10} cluster
Hartree-Fock	0	14
Local Density Approximation	N/A	1
Variational Monte Carlo	$\approx 85\%$	16
Diffusion Monte Carlo	$\approx 95\%$	300
Coupled Cluster[3]	$\approx 75\%$	1500

[3]In the case of infinite basis set, the coupled cluster method is exact and can provide exact results for small, few-atom systems [14].

mation differ slightly from those needed for approximate wavefunctions that will be used in importance sampling. While an approximate wavefunction that is able to accurately predict the behavior of the true wavefunction in all regions is desirable for the fixed-node approximation, it is not necessary. In the fixed-node approximation, the accurate prediction of the nodes of the wavefunction is the only real characteristic that distinguishes a good trial wavefunction from a bad one. How accurately the trial wavefunction mimics the behavior of the true wavefunction between the nodes does not matter too much because any deviation of the trial wavefunction from the true wavefunction in the non-nodal regions just results in the walkers taking longer to converge to the steady-state distribution.

2.4. Variational Monte Carlo

There are many different methods which can be used to calculate trial wavefunctions to use in DMC calculations. One of the most often used methods is Variational Monte Carlo (VMC). The essence of VMC is the variational method is to vary the parameters of a guessed, initial trial wavefunction until a best estimate of the true wavefunction is found. This is done by defining $\mathcal{E}$, called the Rayleigh ratio, to be

$$\mathcal{E} = \frac{\int \psi^*_{trial} H \psi_{trial}}{\int \psi^*_{trial} \psi_{trial}}. \tag{18}$$

Next, the minimum values of the parameters of ψ_{trial} are found by calculating the derivatives of $\mathcal{E}$ with respect to those parameters. The optimal values of the parameters are then used to construct a "best guess" trial wavefunction.

In VMC, the calculations are done by moving walkers, which are subject to the system's potential, randomly to new positions and then calculating the expectation value in equation 18. Eventually, the walkers are all distributed around the potential minimum and the value of the expectation value converges. This converged value of the Rayleigh ratio is never less than the system's true energy.

Fig. 18 shows the results of VMC calculations of the ground state wavefunction for He-SF_6. The eight regions of high density are located at the eight-fold minima of the He-SF_6 potential, indicating that the wavefunction calculated using VMC is fairly accurate. The trial wavefunctions found using VMC are certainly accurate enough to use in DMC calculations of excited states; however, VMC requires a significant amount of computer time to achieve that accuracy. In addition, VMC is not a general method. That is, for each new system, an initial trial wavefunction must be calculated. This can be a tremendous investment in human and computer time and thus makes the method cumbersome to use. The computational expense of VMC is significant enough to warrant a search for other methods that can maintain a similar degree of accuracy while minimizing the computer time needed.

2.5. Conclusions

DMC is the preferred method to calculate wavefunctions and energies for systems with a large number of particles because it can provide accurate results in a relatively short period of time. To calculate excited state energies and wavefunctions for these systems using DMC, the fixed-node approximation must be used. The fixed-node approximation requires the use of a trial wavefunction to predict the location of the nodes of the wavefunction. In addition, a trial wavefunction makes the DMC method more efficient. Thus, a trial wavefunction must be calculated in many different circumstances.

One way to calculate accurate trial wavefunctions is to use VMC; however, the high computational cost of VMC often makes it impractical to use. The next chapter describes an attractive alternative to VMC. The method uses an adiabatic separation of angular and radial motion to obtain an approximate wavefunction that depends only on angle. The method has the advantage over VMC and other methods

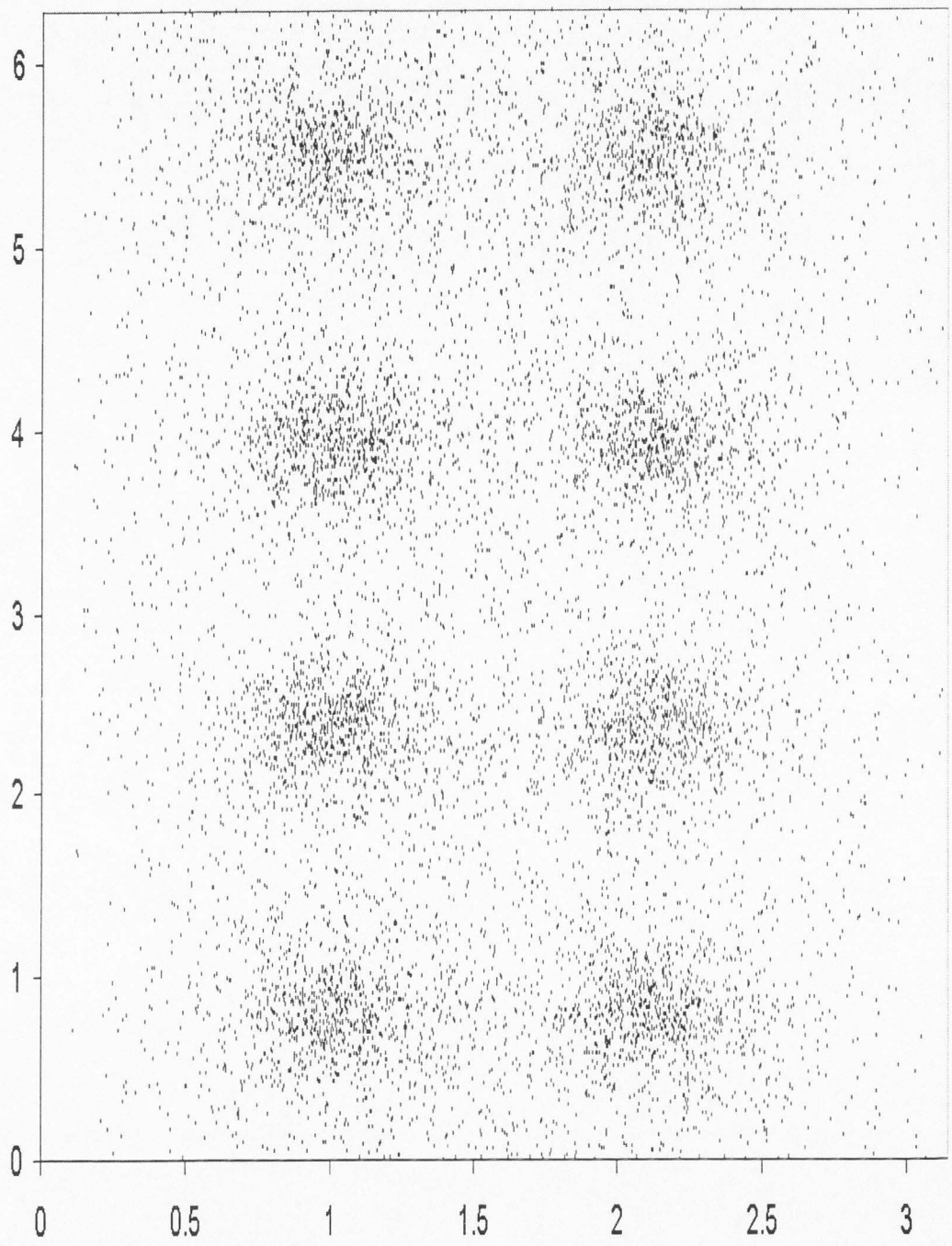

Fig. 18. VMC calculation of the ground state wave function of He-SF_6.

used to calculate trial wavefunctions because it can be calculated at a much cheaper computational cost without sacrificing accuracy.

References

[1] S. Ulam, R. D. Richtmeyer, J. von Neumann, Report LAMS-551, Los Alamos National Laboratory, Los Alamos, NM 1947.

[2] R. Eckhard, Los Alamos Science 15 (1987).

[3] N. Metropolis, Los Alamos Science 15 (1987).

[4] S. Ulam, N. Metropolis, J. of Amer. Stat. Assoc. 44 (1949) 335.

[5] D. Ceperley, L. Mitas, in: I. Prigogine, S. Rice (Eds.) New Methods in Computational Quantum Mechanics Advances in Chemical Physics XCIII, Wiley, New York, 1996, p. 53.

[6] M. Lewerenz, J. Chem. Phys. 106 (1997) 4596.

[7] E. Lee, D. Farrelly, K. B. Whaley, Phys. Rev. Lett. 83 (1999) 3812.

[8] I. Kosztin, B. Faber, K. Schulten, Am. J. Phys. 64 (1996) 633.

[9] E. P. Wigner, Phys. Rev. 40 (1932) 749.

[10] J. Anderson, J. Chem. Phys. 63 (1975) 1499.

[11] J. W. Moskowitz, K. E. Schmidt, M. A. Lee, M. H. Kalos, J. Chem. Phys. 77 (1982) 349.

[12] M. H. Kalos, D. Levesque, L. Verlet, Phys. Rev. A 9 (1974) 2178.

[13] B. L. Hammond, W. A. Lester, Jr., P. J. Reynolds, Monte Carlo Methods in Ab Initio Quantum Chemistry, World Scientific, Singapore, New Jersey, London, Hong Kong, 1994.

[14] L. Mitas, in: I. Prigogine, S. Rice (Eds.) New Methods in Computational Quantum Mechanics Advances in Chemical Physics XCIII, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999, p. 178.

CHAPTER 3

THE ADIABATIC SEPARATION

3.1. Introduction

As noted in the previous chapter, calculations of *excited* states of systems containing an impurity molecule trapped inside a helium cluster using Diffusion Monte Carlo (DMC) requires the use of a trial wavefunction. A minimum requirement of a trial wavefunction that is to be used for this purpose is that it predict the location of the nodes of the true wavefunction to an acceptable degree of accuracy. In addition to allowing calculations on excited states to be done using DMC, a trial wavefunction can be used to greatly increase the efficiency of DMC calculations. These facts make calculation of a trial wavefunction a necessary and important part of many DMC studies.

The Variational Monte Carlo method provides trial wavefunctions that can properly predict the location of the nodes of the wavefunction; however, the method is quite expensive computationally. In addition, the trial wavefunction that is required for Variational Monte Carlo varies from problem to problem, meaning that a new form of a trial wavefunction must be found for each new problem. As finding a trial wavefunction requires a significant time investment, a more general method is desirable. This chapter gives a description of an alternative to Variational Monte Carlo—the adiabatic separation of angular and radial motion—as well as the results of the application of the method to the van der Waals "molecule" He-HCN.

3.2. The adiabatic approximation

The adiabatic approximation is useful for quantum calculations because it can greatly simplify the Hamiltonian of the system by decoupling two of its components from each other. For the electronic Born-Oppenheimer approximation, the two

components that are decoupled are the molecule's nuclear and electronic motion. This approximation is widely used and without it, solutions for even the simplest molecules would be impossible to obtain.

Just as the adiabatic approximation is used to simplify molecular electronic calculations, it can also be used to simplify the calculations on van der Waals complexes. In this case, the approximation utilizes the large difference in the amplitudes of the angular and radial motion of the van der Waals complex to decouple their motion from each other leading to a much simpler Hamiltonian.

For example, consider the Hamiltonian for the interaction of a rare gas element with a linear molecule in the laboratory frame (the same Hamiltonian used in this study of He-HCN),

$$\mathcal{H} = -\frac{\hbar^2}{2\mu}\frac{\partial^2}{\partial R^2} + \frac{\hbar^2}{2\mu}\frac{\mathbf{l}^2}{R^2} + V(R,\theta) + b_0\mathbf{j}^2, \tag{1}$$

where μ is the reduced mass of the complex, b_0 is the rotational constant of the linear molecule, and V is the intermolecular potential which is dependent on both R (the distance between the two centers of mass) and θ (the angle the lone element makes with the molecule, see Fig. 19). The operator $\mathbf{j}$ is the angular momentum of the rod-like molecule and $\mathbf{l}$ is the angular momentum of the element and the molecule about each other.

Rewriting the Hamiltonian in a body-fixed frame[1] [1] gives

$$\mathcal{H} = -\frac{\hbar^2}{2\mu}\frac{\partial^2}{\partial R^2} + \frac{\hbar^2}{2\mu}\frac{(\mathbf{J}-\mathbf{j})^2}{R^2} + V(R,\theta) + b_0\mathbf{j}^2, \tag{2}$$

where $\mathbf{j}$ is now the rotational angular momentum operator for the diatomic in the molecule-fixed coordinate system and $\mathbf{J}$ is defined as the total angular momentum,

[1]The body-fixed frame is fixed on the molecule and thus rotates with respect to the laboratory frame. The two reference frames are related to each other by the Euler angles.

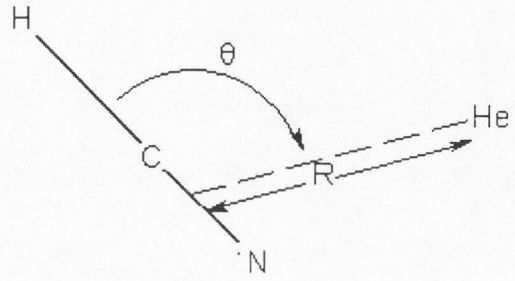

Fig. 19. Definition of terms used in Eq. 1.

i.e.,

$$\mathbf{J} = \mathbf{j} + \mathbf{l}. \tag{3}$$

If the radial motion is assumed to be much smaller than the angular motion, R may be parameterized which allows the first term in the Hamiltonian,

$$-\frac{\hbar^2}{2\mu}\frac{\partial^2}{\partial R^2}, \tag{4}$$

to be ignored leading to a modified Hamiltonian,

$$H(R;\theta) = \frac{\hbar^2}{2\mu}\frac{(\mathbf{J}-\mathbf{j})^2}{R^2} + V(R,\theta) + b_0\mathbf{j}^2. \tag{5}$$

Here, the notation $H(R;\theta)$ denotes a Hamiltonian that is dependent on θ and parametrically dependent on R. Now, instead of solving the full Schrödinger equation,

$$\mathcal{H}\psi(R,\theta) = E\psi(R,\theta), \tag{6}$$

the much simpler approximate Schrödinger equation,

$$H(R;\theta)\Psi_A(R;\theta) = U(R)\Psi_A(R;\theta), \tag{7}$$

can be solved by constructing and diagonalizing $H(R;\theta)$ in an appropriate basis.

As the adiabatic separation for van der Waals complexes is based entirely on the assumption that the radial motion of the complex is localized, whether the

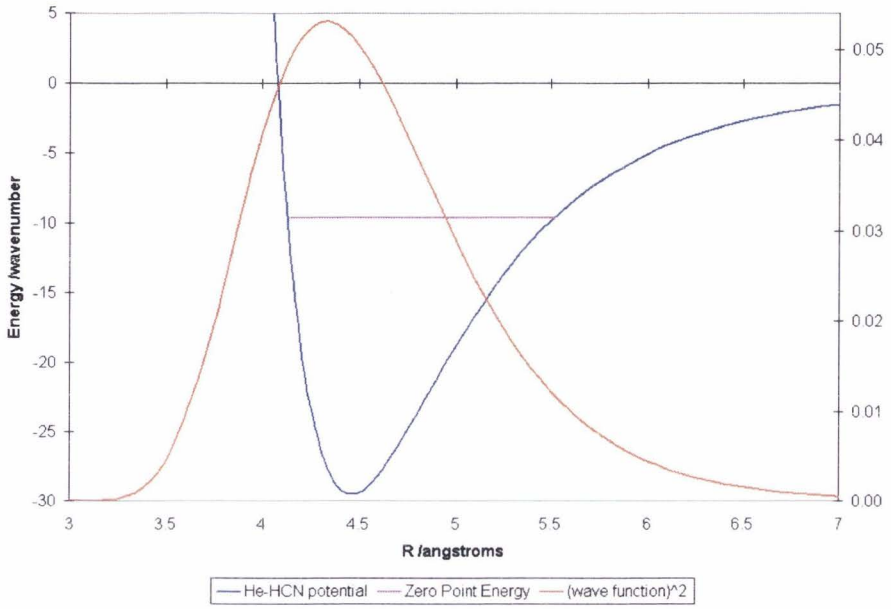

Fig. 20. Radial probability distribution for He-HCN.

approximation is valid for complexes that do not have highly localized radial motion cannot be determined without testing it on a radially floppy complex. The large amplitude radial motions and the extremely weak interactions which cause them can be seen by comparing Fig. 20 and Fig. 21, which show the potential energy curves, ground state energies, and radial probability distributions[2] for Ar-HCl (a classical van der Waals complex) and He-HCN (a radially floppy or quantum van der Waals complex) respectively. The shallow potential energy well and large zero point energy of He-HCN lead to a radial distribution that is much broader than that of the argon complex.

[2]The approximate radial wave function used to determine the probability distributions for He-HCN and Ar-HCl shown in Fig. 20 and Fig. 21 were calculated by solving the respective Hamiltonians for He-HCN and Ar-HCl using the isotropic potential, i.e., $V = \sum_{n=0}^{0} C_n(R)P_n(\cos\theta) = C_0(R)$.

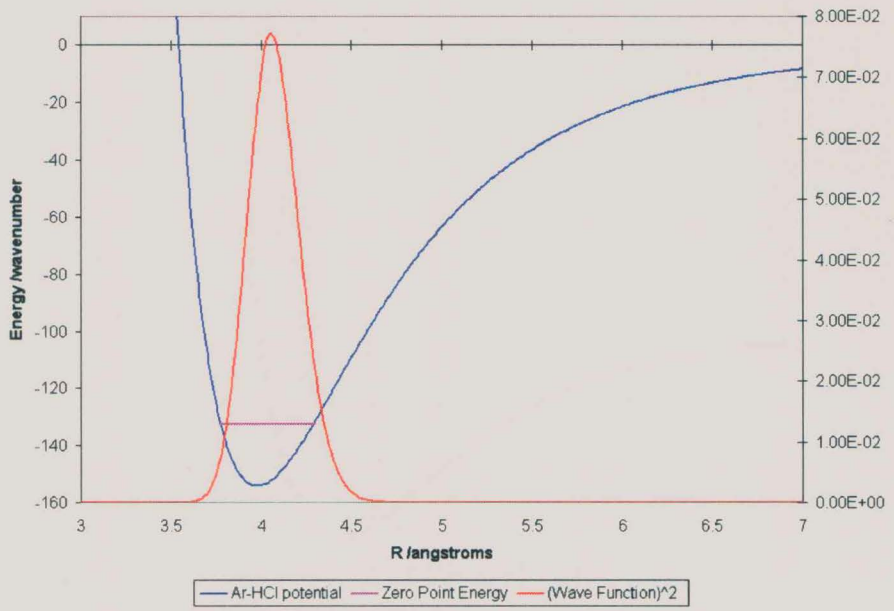

Fig. 21. Radial probability distribution for Ar-HCl.

In spite of the large amplitude radial motions, the wavefunctions for radially floppy van der Waals complexes are still much more sensitive to changes in angle than they are to changes in distance. This is shown in Fig. 22, which shows the interaction potential of He-HCN at fixed angle (Fig. 22a) and fixed radial separation (Fig. 22b). The fact that there is still a significant difference between the angular and radial motions of helium complexes leaves open the possibility of using the adiabatic approximation on them.

By showing that the adiabatic approximation is valid for helium van der Waals complexes, the work presented in this thesis demonstrates that the adiabatic approximation can be used to calculate approximate wavefunctions accurate enough to be used in DMC studies of impurity molecules trapped inside helium clusters. Using the method for this purpose offers significant improvements over other methods (such

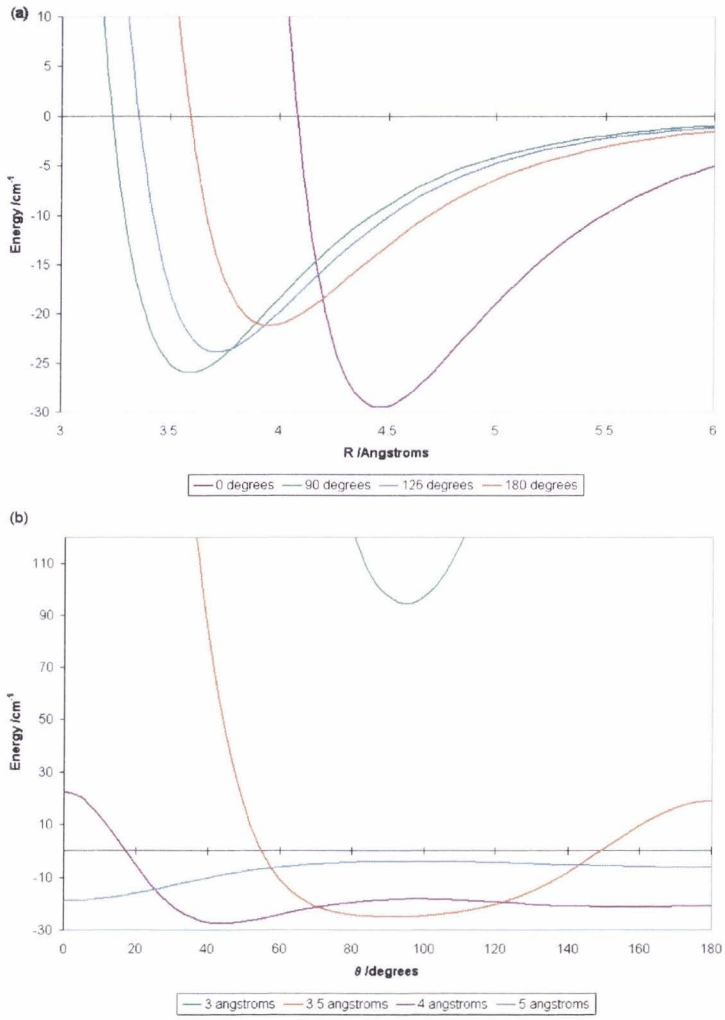

Fig. 22. The potential of He-HCN at several values of (a) θ and (b) R.

as Variational Monte Carlo) commonly used to calculate approximate wavefunctions because of the relatively short amount of time it requires. The time improvement over Variational Monte Carlo this method provides comes not only from a decrease in actual computer time needed, but also from the fact that an initial trial wavefunction does not need to be calculated for each new system.

The next section describes how the adiabatic approximation was applied to the He-HCN van der Waals complex.

3.3. Application of the adiabatic approximation to He-HCN

The Hamiltonian matrix for the He-HCN van der Waals complex in the Y_m^0 basis[3] is given by

$$\langle Y_m^0 \mid H(R;\theta) \mid Y_n^0 \rangle = \left\langle Y_m^0 \left| b_0 \mathbf{j}^2 + \frac{\hbar^2}{2\mu}\frac{(\mathbf{J}-\mathbf{j})^2}{R^2} + V(R,\theta) \right| Y_n^0 \right\rangle . \tag{8}$$

Using

$$\mathbf{j}^2 \left| Y_n^0 \right\rangle = n(n+1) \left| Y_n^0 \right\rangle , \tag{9}$$

leads to the following expression for the matrix elements of the $b_0\mathbf{j}^2$ term:

$$\langle Y_m^0 \mid b_0\mathbf{j}^2 \mid Y_n^0 \rangle = b_0[n(n+1)]\delta_{m,n}. \tag{10}$$

The matrix elements for the $\frac{\hbar^2}{2\mu}\frac{(\mathbf{J}-\mathbf{j})^2}{R^2}$ term can be simplified by expanding $(\mathbf{J}-\mathbf{j})^2$,

$$\left\langle Y_m^0 \left| \frac{\hbar^2}{2\mu}\frac{(\mathbf{J}-\mathbf{j})^2}{R^2} \right| Y_n^0 \right\rangle = \left\langle Y_m^0 \left| \mathbf{J}^2 - \mathbf{j}^2 - 2\mathbf{J}\cdot\mathbf{j} \right| Y_n^0 \right\rangle , \tag{11}$$

and then replacing the operators $\mathbf{J}^2$ and $\mathbf{j}^2$ with their respective eigenvalues, $J(J+1)$ and $n(n+1)$, giving

$$\frac{\hbar^2}{2\mu}\left[\hbar\left\{J(J+1)+n(n+1)\right\}\delta_{m,n} - 2\left\langle Y_m^0 \left| \mathbf{J}\cdot\mathbf{j} \right| Y_n^0 \right\rangle\right] . \tag{12}$$

---

[3] $Y_n^0(\theta)$ are the spherical harmonics, $Y_m^l(\theta,\phi)$, with l=0.

The final term in this expression, $\langle Y_m^0 \,|\mathbf{J}\cdot\mathbf{j}|\, Y_n^0\rangle$, can be shown to be negligible using perturbation theory [2], which gives

$$\begin{aligned}\langle Y_m^0 \;\;|\;\; H(R;\theta) \mid Y_n^0\rangle &= b_0[n(n+1)]\delta_{m,n} + \\ &\frac{\hbar^2}{2\mu}\left[\hbar\left\{J(J+1)+n(n+1)\right\}\delta_{m,n}\right] + \left\langle Y_m^0\,|V(R,\theta)|\,Y_n^0\right\rangle . \end{aligned} \tag{13}$$

Unlike the matrix elements of b_0 and $(\mathbf{J}-\mathbf{j})^2$, the $V(R;\theta)$ matrix elements do not have an analytical form that can be used. To overcome this obstacle, the potential was expressed as an expansion of the associated Legendre functions, i.e.,

$$V(R,\theta) = \sum_{n=0}^{N} C_n(R) P_n(\cos\theta). \tag{14}$$

Using this expansion, the matrix elements of the potential are

$$\left\langle Y_m^0\,|V(R,\theta)|\,Y_n^0\right\rangle \;=\; \left\langle Y_m^0\left|\sum_{k=0}^{K} C_k(R)P_k(\cos\theta)\right|Y_n^0\right\rangle = \tag{15}$$

$$[(2m+1)(2n+1)]^{1/2}\sum_{k=0}^{K} C_k(R)\begin{pmatrix} m & k & n \\ 0 & 0 & 0\end{pmatrix}^2, \tag{16}$$

where $\begin{pmatrix} m & k & n \\ 0 & 0 & 0\end{pmatrix}$ is a Wigner 3-j symbol and $C_k(R)$ is the k^{th} expansion coefficient for the potential expressed in terms of the Legendre polynomials. Thus, the final expression for the approximate Hamiltonian matrix was

$$\begin{aligned}\langle Y_m^0 \;\;|\;\; H(R;\theta) \mid Y_n^0\rangle &= \\ b_0(n(n+1))\delta_{m,n} &+ \frac{\hbar^2}{2\mu}\left[\hbar\left\{J(J+1)+n(n+1)\right\}\delta_{m,n}\right] + \\ \sqrt{[(2m+1)(2n+1)]}&\sum_{k=0}^{K} C_k(R)\begin{pmatrix} m & k & n \\ 0 & 0 & 0\end{pmatrix}^2 . \end{aligned} \tag{17}$$

The matrix was diagonalized using the Jacobi method, which, like most numerical eigensystem routines, nudges the matrix to diagonal form by performing a series of similarity transformations, i.e.,

$$\mathbf{A} \rightarrow T_0^{-1}\mathbf{A}T_0 \rightarrow T_1^{-1}T_0^{-1}\mathbf{A}T_0T_1 \rightarrow T_2^{-1}T_1^{-1}T_0^{-1}\mathbf{A}T_0T_1T_2 \rightarrow etc. \tag{18}$$

In the Jacobi method, each similarity transformation is actually a rotation that eliminates one of the off-diagonal matrix elements. After enough of these transformations, all off-diagonal matrix elements are removed. The method has the advantage of being virtually foolproof—it will diagonalize any $n \times n$ real symmetrical matrix [3].

After diagonalization, the matrix elements remaining on the diagonal are the eigenvalues; the eigenvectors are found by expressing them in an expansion of the basis functions (in this case, Y_l^0). The coefficients for this expansion are found by multiplying the columns of the accumulated transformation together as follows:

$$T_0 \cdot T_1 \cdot T_2 \cdot .... \tag{19}$$

The eigenvectors found from diagonalizing the matrix are the approximate wavefunction of the He-HCN van der Waals complex. The eigenvectors were checked for normalization and convergence by examining the sum of the square of the expansion coefficients, i.e.,

$$\sum_n |C_n|^2 . \tag{20}$$

Convergence was checked by examining this sum as the number of basis vectors increased. Assuming the wavefunction is normalized, the sum will converge to one when the expansion of the basis vectors is equal to the wavefunction. As can be seen in Fig. 23, which shows a plot of $\sum_n |C_n|^2$ versus n, the expansion is converged after approximately $n = 15$; therefore, the number of terms that were used in these calculations was twenty. This examination of the expansion coefficients also allowed a check for normalization because the sum is only equal to one if the wavefunction is normalized since the basis vectors themselves are normalized.

The He-HCN potential used in this study was obtained from a recent paper by Atkins and Hutson [4]. Their potential was found by fitting parameterized functional

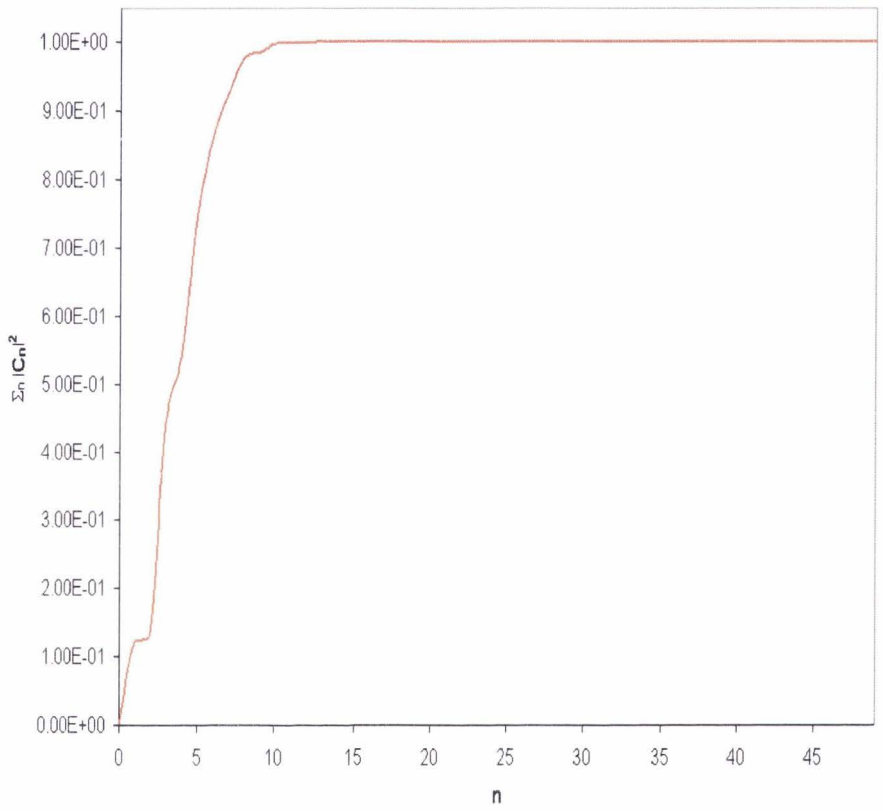

Fig. 23. Graph showing the number of terms, n, that were needed for convergence.

forms to high-resolution microwave and millimeter wave spectroscopic data that was collected by Drucker *et al.* in 1995 [5]. In the Atkins paper, there are several different potential forms presented; the one used in this study was the one termed "1E8" (the E8 signifies the potential was fit to eight experimental data).

The expression for the potential consists of three terms:

$$V(R,\theta) = V_{rep} + V_{ind} + V_{dis}. \tag{21}$$

The first term, V_{rep}, the intermolecular repulsion, is given by

$$V_{rep} = A(\theta)e^{-\beta R}; \tag{22}$$

the intermolecular induction, V_{ind}, is given by

$$V_{ind} = -(\alpha_{He})(\mu_{HCN})^2[1 + P_2(\cos\theta)]R^{-6} - 6(\alpha_{He})(\mu_{HCN})(\Theta_{HCN})(\cos^3\theta)R^{-7}; \tag{23}$$

and the dispersion energy, V_{dis}, is

$$V_{dis} = -\sum_{n=6}^{8} c_n(\theta)D_n(R)R^{-n}, \tag{24}$$

where

$$\begin{aligned} c_6(\theta) &= c_6^{(0)} + c_6^{(2)}P_2(\cos\theta), \\ c_7(\theta) &= c_7^{(1)}\cos\theta + c_7^{(3)}\cos^3\theta, \end{aligned} \tag{25}$$

and $D_n(R)$ are the Tang-Toennies damping functions [6],

$$D_n(R) = 1 - e^{-\beta R}\sum_{m=0}^{n}\frac{(\beta R)^m}{m!}. \tag{26}$$

The values of the parameters used are shown in Table 3. $A(\theta)$ and $c_8(\theta)$ were calculated by expanding the depth, $\epsilon(\theta)$, and the position, $R_m(\theta)$, of the minimum of the potential well as Legendre series,

$$\begin{aligned} \epsilon(\theta) &= \sum_{\lambda=0}^{3}\epsilon_\lambda P_\lambda(\cos\theta) \text{ and} \\ R_m(\theta) &= \sum_{\lambda=0}^{3}R_m^\lambda P_\lambda(\cos\theta). \end{aligned}$$

Table 3
Parameters used in the He-HCN potential

Parameter	Potential 1E8
α_{He} (a_0)	01.383
μ_{HCN} (ea_0)	01.174
Θ_{HCN} (ea_0^2)	01.77
β (Å^{-1})	03.901(15)
ϵ_0 (cm^{-1})	24.825(139)
ϵ_1 (cm^{-1})	03.402
ϵ_2 (cm^{-1})	00.385(3)
ϵ_3 (cm^{-1})	00.854
R_m^0 (Å)	03.715
R_m^1 (Å)	00.200
R_m^2 (Å)	00.071

Using these expansions, the definition of the derivative of a Tang-Toennies damping function,

$$D_n'(x) = \beta[D_{n-1}(x) - D_n(x)], \tag{27}$$

and defining V_{fix} to be

$$V_{fix}(R_m) = V_{ind}(R_m) - c_6 D_6(R_m) R_m^{-6} - c_7 D_7(R_m) R_m^{-7}, \tag{28}$$

allows $c_8(\theta)$ and $A(\theta)$ to be found,

$$c_8(\theta) = \frac{\beta[\epsilon(\theta) + V_{fix}(R_m)] + V_{fix}(R_m)}{D_8'(Rm) - D_8(R_m)\left[\frac{8}{R_m} - \beta\right]} R_m^8 \text{ and}$$

$$A(\theta) = \left(c_8(\theta) D_8(R_m) R_m^{-8} - \epsilon - V_{fix}(R_m)\right) e^{-\beta R_m}.$$

The final form of the potential is shown in Fig. 24.

The associated Legendre polynomials used to obtain an analytical function for the potential were calculated using the following recursion relation to calculate the polynomials:

$$(l-m)P_l^m = x(2l-1)P_{l-1}^m - (l+m-1)P_{l-2}^m. \tag{29}$$

Although there are many different recurrence relations that the associated Legendre functions satisfy, this one was chosen because it is stable and because it has a closed-form expression that can be used to find a starting[4] value.

The expansion coefficients, $C_n(R)$, used to expand the potential were found by integrating the potential with the Legendre polynomials, i.e.,

$$C_n(R) = \langle V_n(R) \,|P_n(\cos\theta)\rangle = \int_0^\pi V(R,\theta) P_n(\cos\theta) d\theta. \tag{30}$$

---

[4]The closed-form expression is $P_m^m = (-1)^m(2m-1)!!(1-x^2)^{m/2}$, where $n!!$ denotes the sum of all odd integers less than or equal to n. If $l = m+1$ and $P_{m-1}^m = 0$, then $P_{m+1}^m = x(2m+1)P_m^m$.

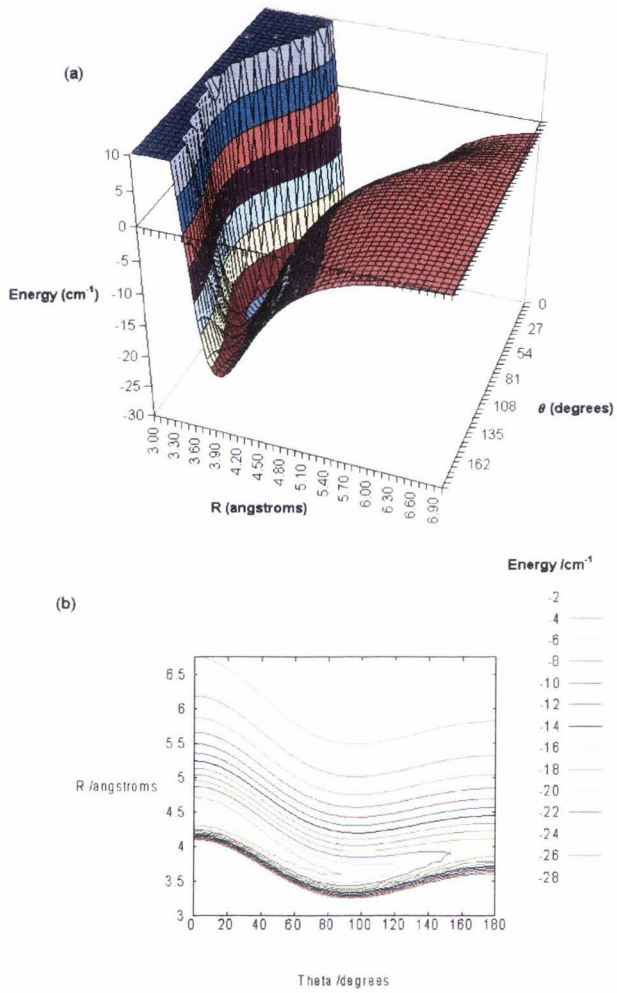

Fig. 24. The He-HCN potential.

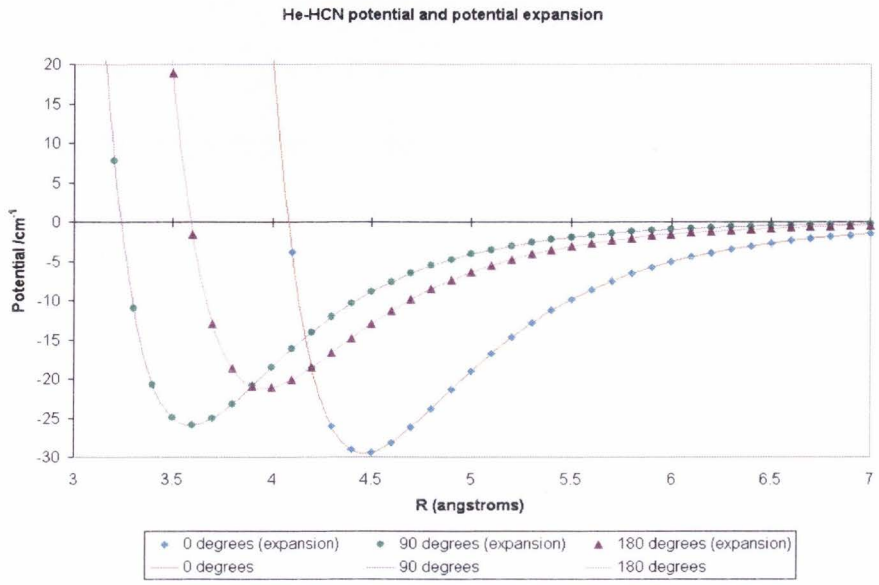

Fig. 25. The expansion of the potential in Legendre polynomials.

Fig. 25 shows the potential plotted with an expansion of the first fifteen terms at several values of θ. As can be seen from Fig. 26, which shows the percent error in the potential expansion, most of the error in the expansion occurs along the minimum of the potential well with the largest error of 2.18% occurring at approximately 3.3 Å and 118°.

3.4. Discussion

The ground state angular wavefunction, $\Psi_{ang}(R;\theta)$ found from diagonalizing the approximate Hamiltonian, $H(R;\theta)$, is shown in Fig. 27. The fact that a reasonable wavefunction was found shows that the adiabatic approximation is indeed valid for extremely floppy van der Waals complexes. As shown in Fig. 28a, which shows $\Psi_{ang}(R;\theta)$ plotted at different fixed values of R, the angular wavefunction does a good job of matching up with the He-HCN potential. However, Fig. 28b, which

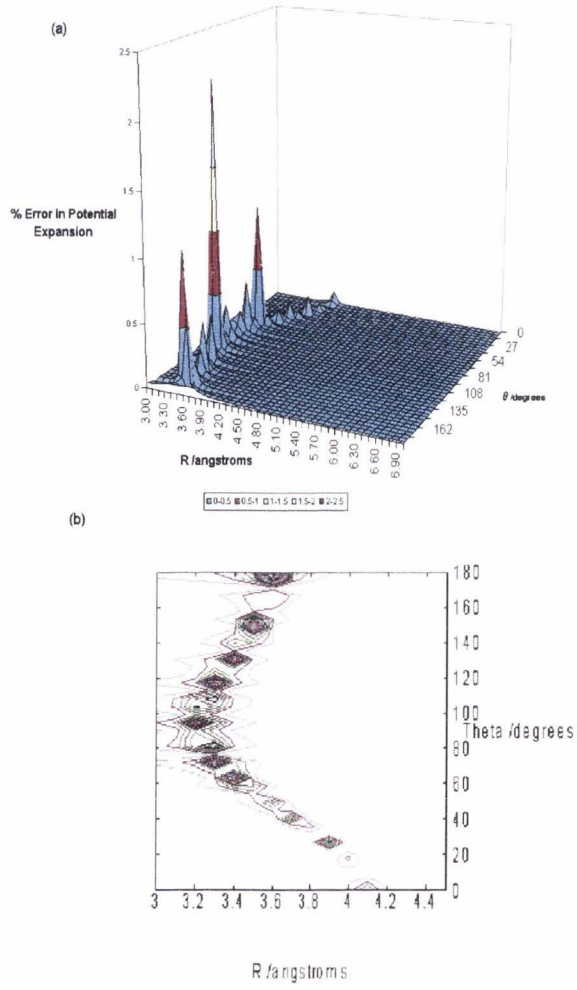

Fig. 26. Error associated with potential expansion.

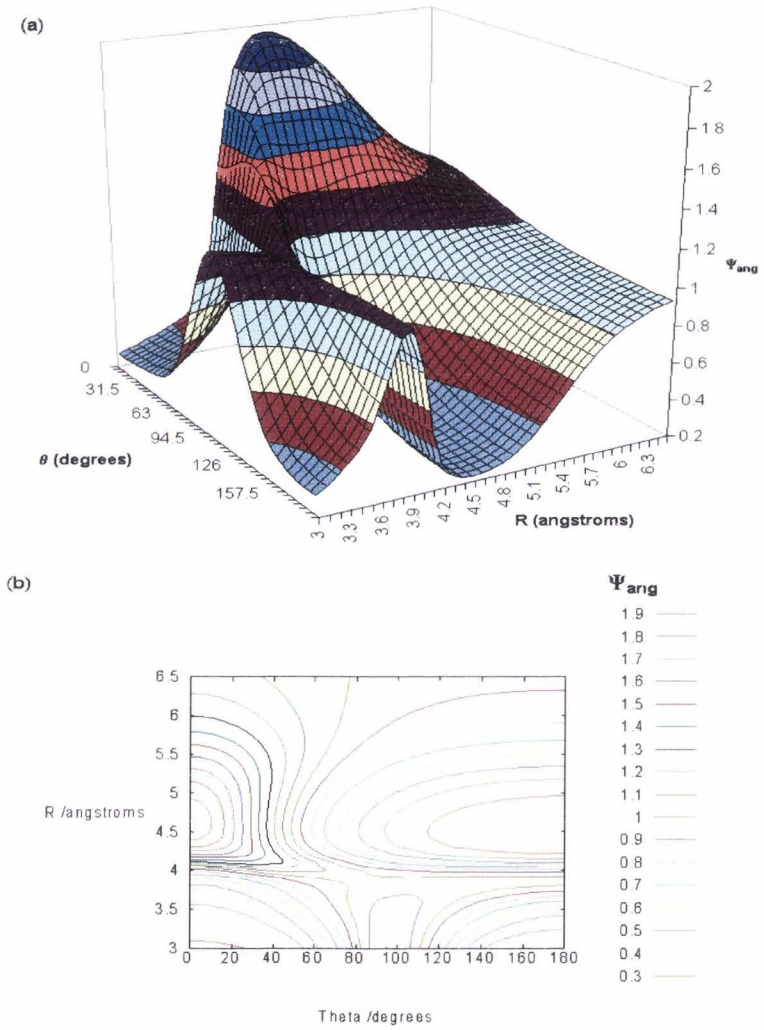

Fig. 27. The angular wavefunction.

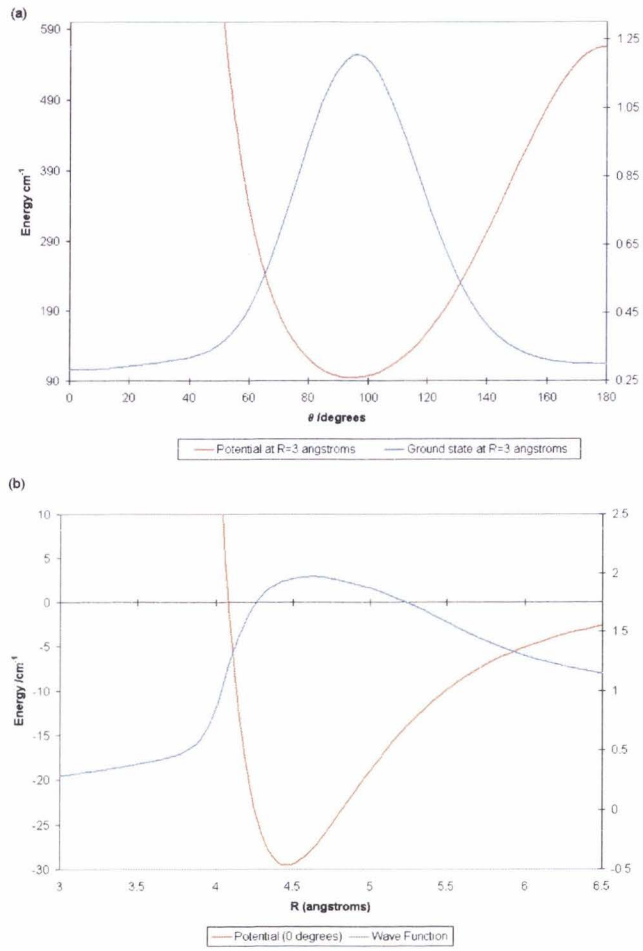

Fig. 28. Comparison of the quality of the angular wavefunction.

shows the ground state angular wavefunction plotted with the potential at $\theta = 0$, reveals that the angular wavefunction does not accurately match up with the He-HCN potential radially.

To improve the form of the angular wavefunction radially, it was multiplied by a radial wavefunction calculated using the radially dependent eigenvalues, $U(R)$ (found from diagonalizing approximate Hamiltonian matrix), as an effective radial potential, i.e.,

$$\left[\frac{\hbar^2}{2\mu}\nabla^2 + U(R)\right]\chi(R) = [E_R]\,\chi(R). \tag{31}$$

This equation was solved using the Fourier Grid Hamiltonian[5] method. The radial wavefunction, $\chi(R)$, and the radial potential, $U(R)$, are both shown in Fig. 29.

The modified ground state wavefunction,

$$\phi(R,\theta) = \Psi_{ang}(R;\theta)\chi(R), \tag{32}$$

is shown in Fig. 30. This modified wavefunction is a much better approximation than the angular wavefunction because in addition to accurately describing the system angularly, it also gives a reasonable radial description of the system. The radial improvement this modification provides can be seen by comparing the modified wavefunction at fixed θ (Fig. 31) with the angular wavefunction at fixed θ (Fig. 28b).

As the major motivation for this study was the development of a method that can be used to calculate excited state trial wavefunctions for DMC studies, the approximation was also tested on the excited states of He-HCN. As shown in Figs. 32-34, which show Ψ_{ang} for the first three excited states of He-HCN, the adiabatic approximation is also valid for the excited states of extremely floppy van der Waals complexes.

---

[5]A description of the Fourier Grid Hamiltonian method is given in Appendix A.

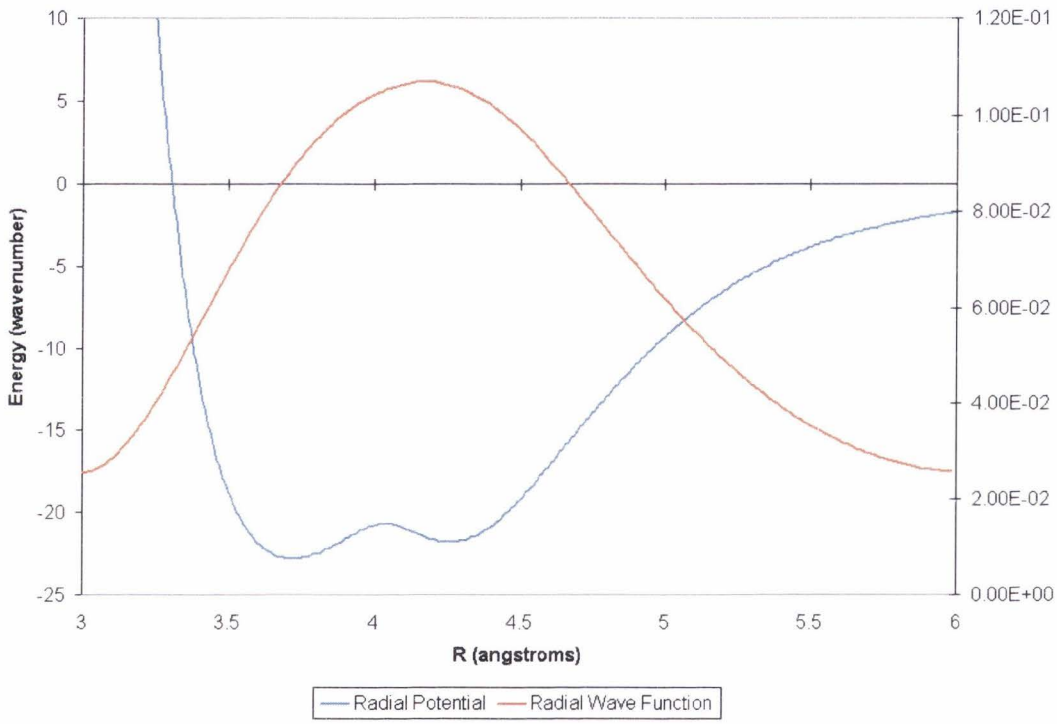

Fig. 29. The radial wave function, $\chi(R)$, with the radial potential, $U(R)$.

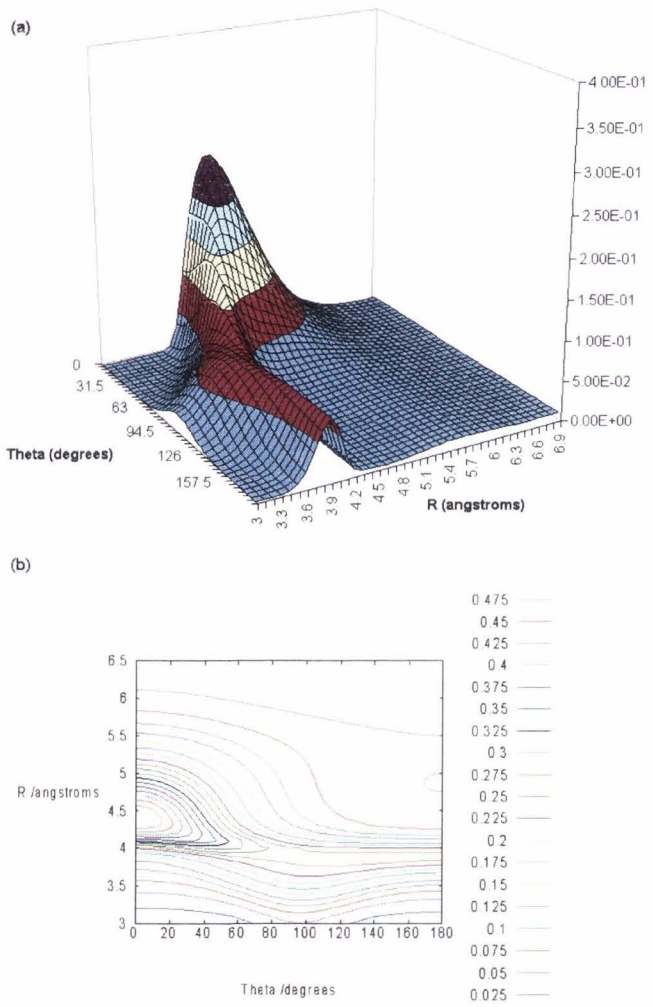

Fig. 30. The modified wavefunction.

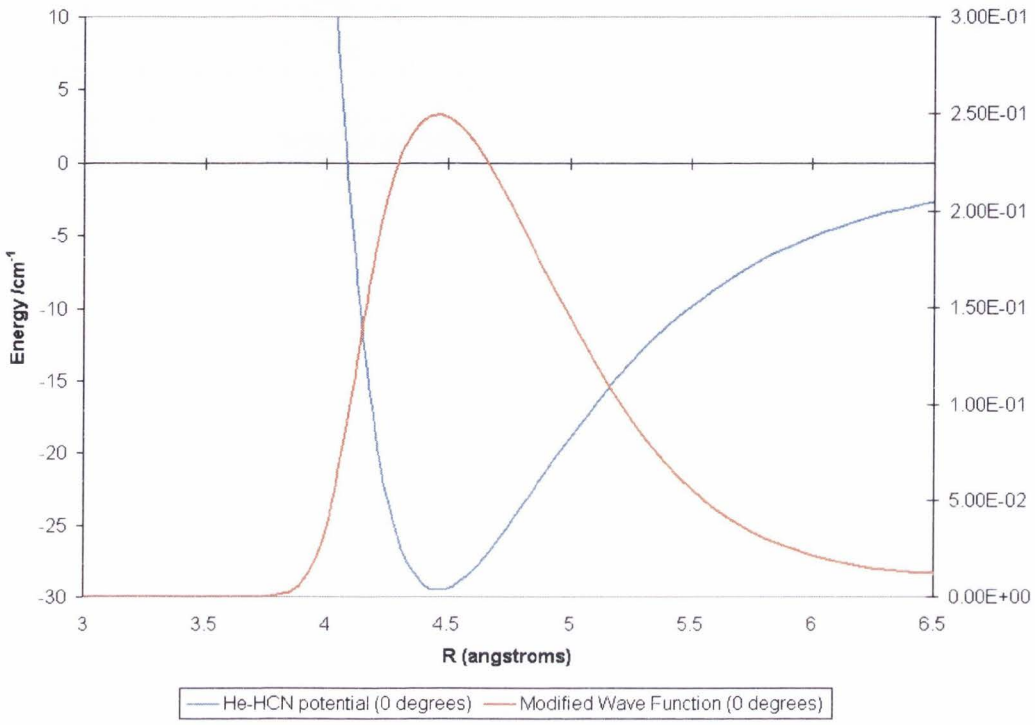

Fig. 31. Plot of the potential with $\phi = \Psi_{ang}(R;\theta)\chi(R)$ at 0°.

As was the case with the ground state wavefunctions, the radial description provided by Ψ_{ang} is lacking. To improve the form of the excited state wavefunctions, the angular wavefunctions were again multiplied by $\chi(R)$. Figs. 35-37 show the result of this modification.

Calculation of excited states using DMC and the fixed-node approximation only require the nodes of the wavefunction to be approximately known. Thus, the nodal structures of the excited state trial wavefunctions are of particular interest. Fig. 38 shows the nodal structure of the angular wavefunction for the first three excited states of He-HCN. These plots reveal the strong angular localization of the nodes of the angular wavefunction. The nodal structure for the first three excited states of the modified wavefunction, ϕ, is shown in Fig. 39. These graphs again reveal the radial improvement provided by multiplying the angular wavefunction by the radial wavefunction.

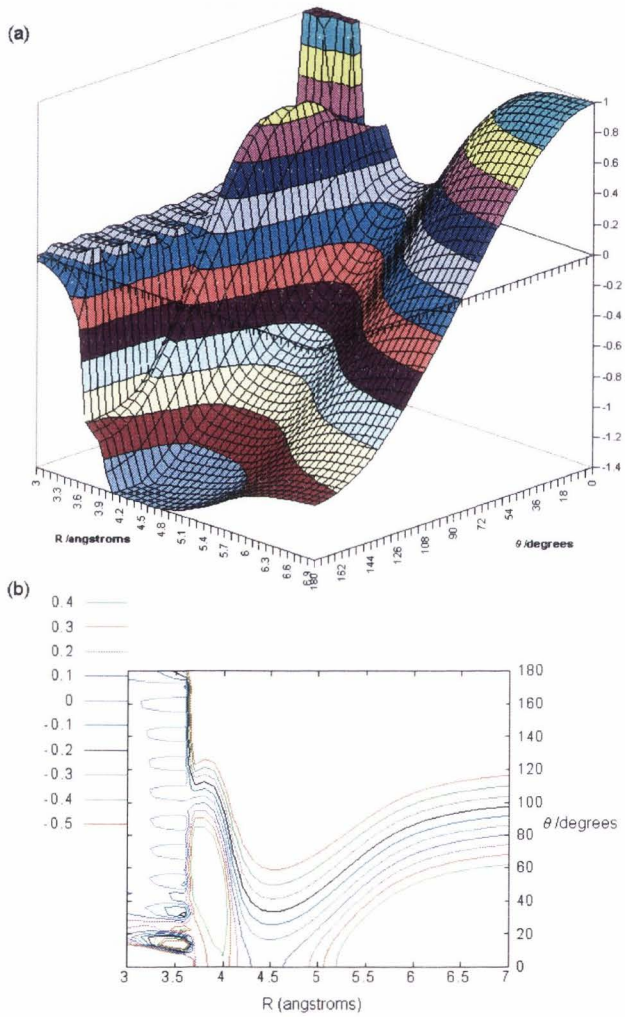

Fig. 32. Angular wavefunction for the first excited state of He-HCN.

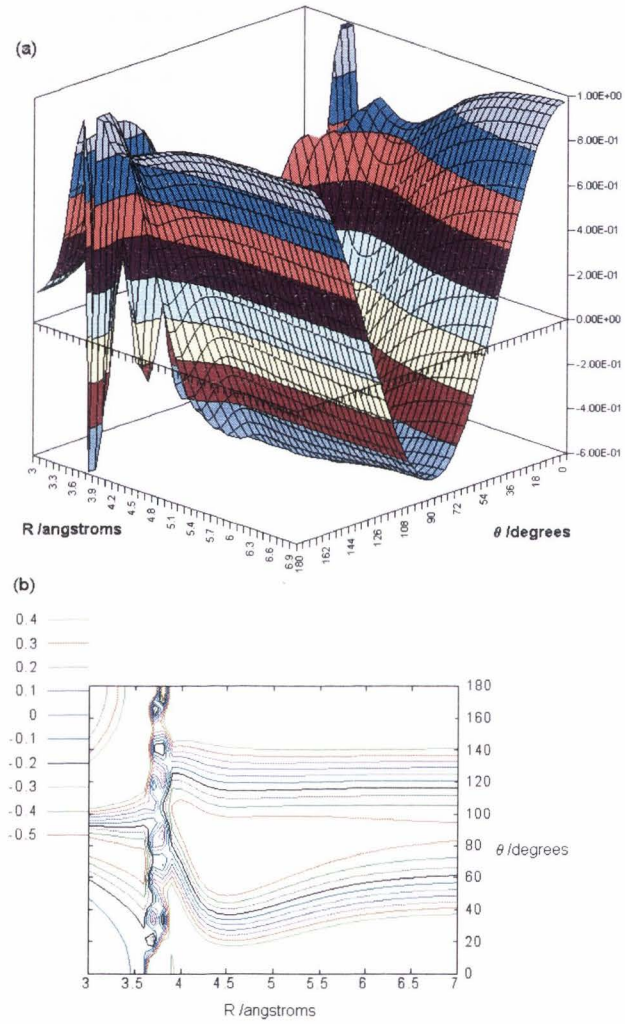

Fig. 33. Angular wavefunction for the second excited state of He-HCN.

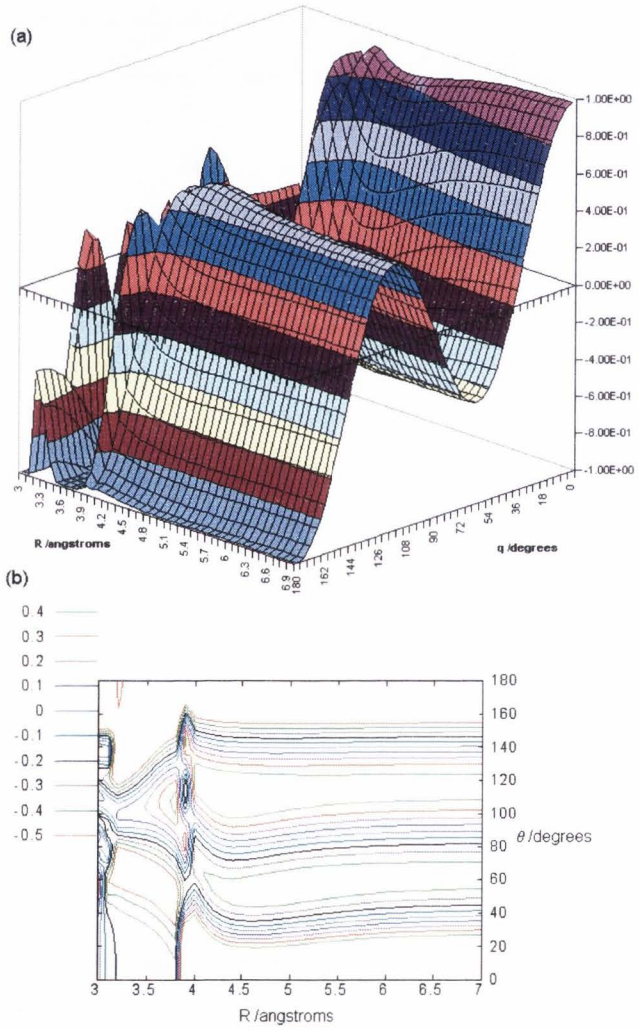

Fig. 34. Angular wavefunction for the third excited state of He-HCN.

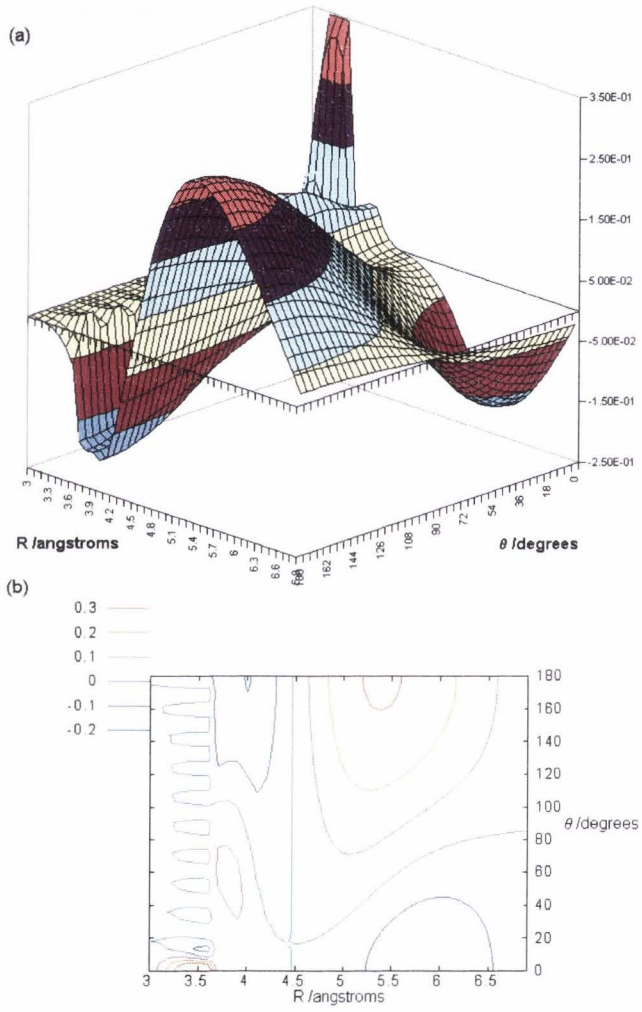

Fig. 35. Plots of the modified wavefunction for the first excited state.

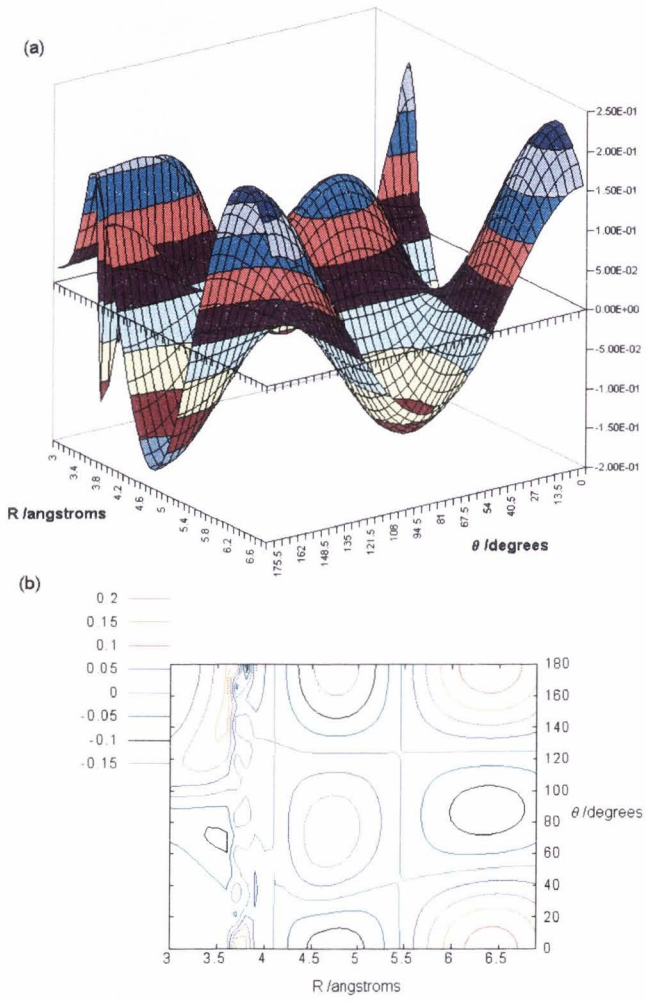

Fig. 36. Plots of the modified wavefunction for the second excited state.

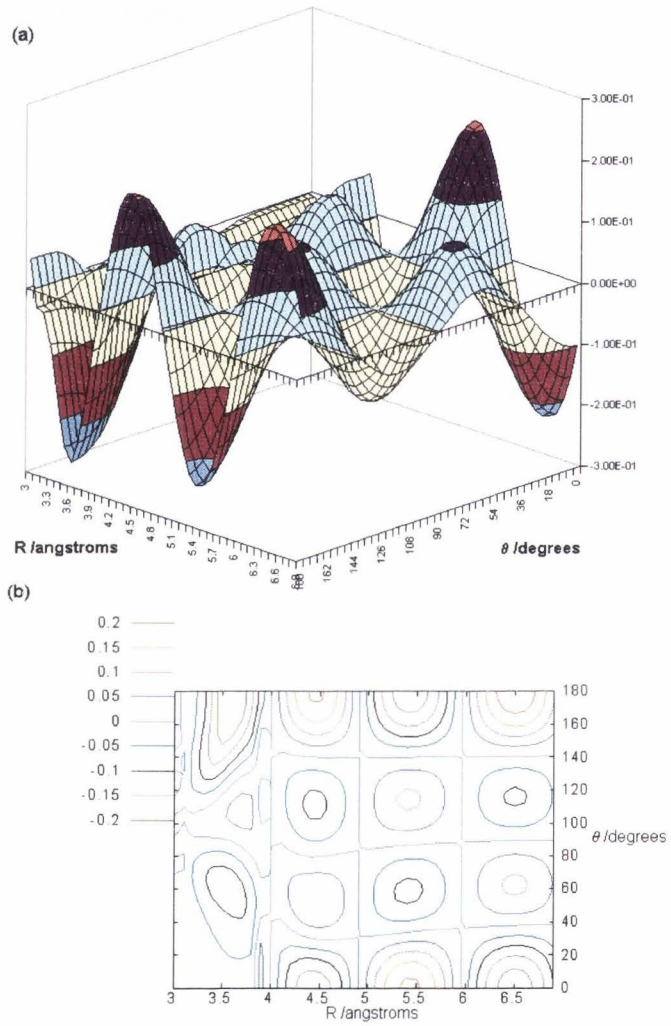

Fig. 37. Plots of the modified wavefunction for the third excited state.

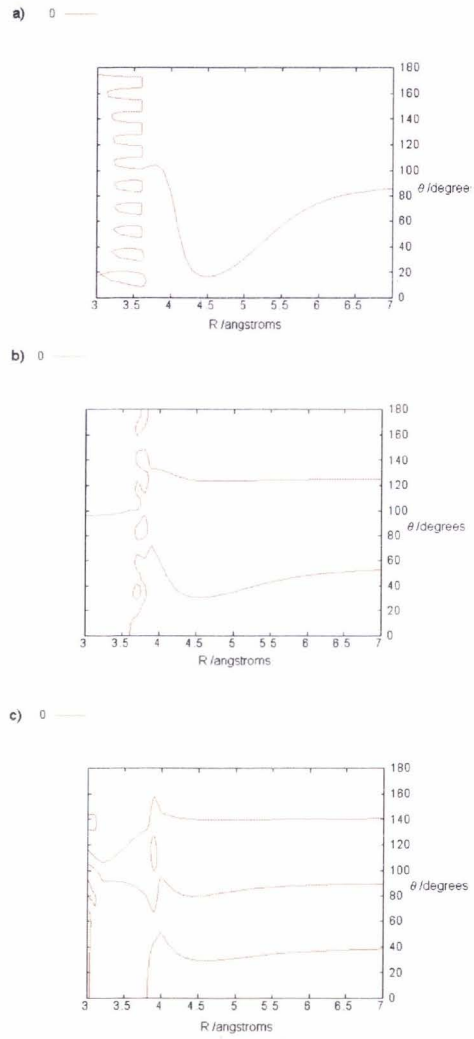

Fig. 38. Nodal surface of the angular wavefunction.

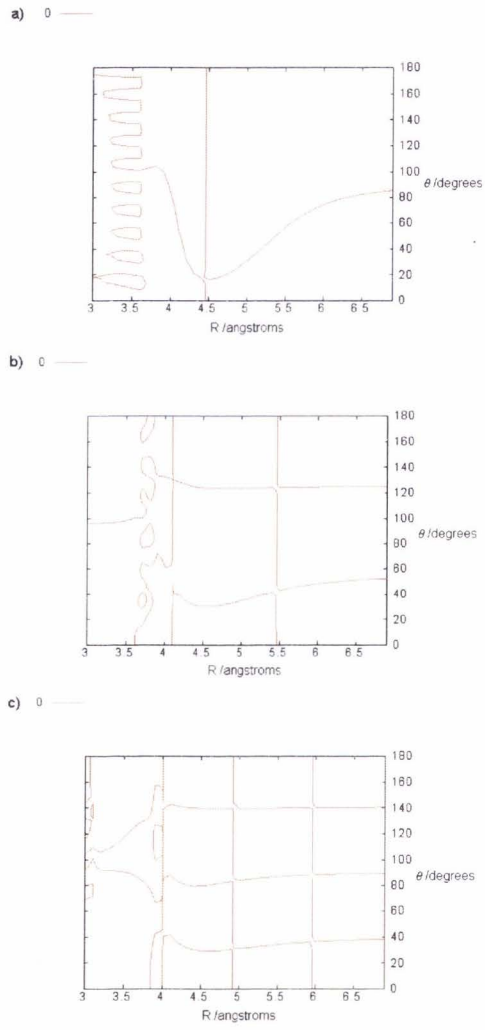

Fig. 39. Nodal surface of the modified wavefunction.

3.5. Conclusion

Use of the adiabatic separation to determine trial wavefunctions that can be used in Diffusion Monte Carlo to calculate excited state energies and wavefunctions has been shown here to be an attractive alternative to other methods currently used to calculate trial wavefunctions such as Variational Monte Carlo. This is the first time the adiabatic approximation has been applied to an extremely radially floppy complex such as He-HCN.

As shown here, the angular wavefunction calculated using the adiabatic separation is a reasonable approximation to the true wavefunction that can be calculated at a computational cost significantly less than Variational Monte Carlo. The form of this wavefunction can be improved by multiplying it by the radial wavefunction found by solving the radial equation. This calculation is minimally expensive and greatly improves the form of the approximate wavefunction. This improvement in form is important because it can speed up the DMC calculation while decreasing the error associated with calculation of excited states using DMC.

References

[1] S. Bratoz, M. Martin, J. Chem. Phys. 42 (1965) 1051.

[2] S. Holmgren, M. Waldman, W. Klemperer, J. Chem. Phys. 67 (1977), 4414.

[3] W. Press, S. Teukolsky, W. Vetterling, B. Flannery, Numerical Recipes in Fortran, Cambridge University Press, Cambridge, 1992.

[4] K. Atkins, J. Hutson, J. Chem. Phys. 105 (1996) 440.

[5] S. Drucker, F. Tao, W. Klemperer, J. Phys. Chem. 99 (1995), 2646.

[6] K. Tang, J. Toennies, J. Chem. Phys. 80 (1984), 3726.

CHAPTER 4

SUMMARY

Ultra-cold droplets of helium are interesting for at least two reasons. First, as finite sized superfluids, they offer the opportunity to study superfluidity on a microscopic level. This is important because it could lead to a more complete and fundamental description of superfluidity. Second, helium droplets in a molecular beam could be used to obtain high-quality spectroscopic data for molecules that are currently too large to study using spectroscopy because their many modes of motion lead to spectra that are impossible to interpret. Helium droplets are ideal matrices for spectroscopy because they are very cold and interact very weakly with other molecules. Thus, a better understanding of the behavior and properties of helium droplets will lead to significant advances in both chemistry and physics.

An important part of the study of helium clusters is the behavior of molecules with well understood properties inside the clusters. A molecule inside a helium cluster can be used as a probe that relays information about the helium cluster environment back to the macroscopic world. Experimental studies of these systems have revealed some fascinating results about the properties of helium clusters. One of these results was the apparent free rotation of the molecule in the droplet of helium. This free rotation is a result of the superfluid nature of the helium cluster. To better understand the properties of helium droplets, a detailed study of molecular rotation inside the cluster is necessary.

Any theoretical study of these systems requires the use of approximate methods because analytical solutions are not possible. A particularly powerful numerical method that can be used for many body calculations is the so-called Diffusion Monte Carlo method, which can calculate numerically exact ground state energies

and wavefunctions. To calculate excited states using Diffusion Monte Carlo, an approximation—the fixed-node approximation—is necessary. The error associated with this approximation is directly dependent on how well the nodes of the wavefunction are known. This makes the calculation of approximate wavefunctions that can accurately predict nodal structure a critical component of Diffusion Monte Carlo studies.

Typically, the many-body wavefunction for Diffusion Monte Carlo is written as a product of the helium-impurity molecule dimer wavefunctions, i.e., for a helium cluster consisting of n helium atoms, the wavefunction is written as

$$\Psi = \prod_{n} \psi_{He-I},$$

where ψ_{He-I} is the helium-impurity molecule wavefunction and Ψ is the total wavefunction. Thus, a form for the dimer wavefunction is first necessary. One of the most common methods used to calculate approximate wavefunctions for Diffusion Monte Carlo studies is the so-called Variational Monte Carlo method. This method provides wavefunctions that are accurate enough to use for Diffusion Monte Carlo; however, they are computationally costly. Thus, a method that can calculate approximate wavefunctions at a reduced computational cost is highly desirable.

A method that has been used to calculate approximate wavefunctions for van der Waals complexes in the past is the adiabatic approximation. In general, this approximation decouples one component of the system from another based on the large difference between the amplitudes of the two components. For example, the adiabatic approximation for molecules uses the fact that nuclei move much more slowly than electrons to decouple the nuclear motion from the electronic. In the case of van der Waals complexes, the approximation is based on the large difference in the radial and angular motions of the complex.

The approximation has never before been applied to the van der Waals complexes of helium because, unlike other complexes, those of helium do not have highly localized radial motion. As the essence of the adiabatic approximation is the large difference in the relative motions of the angular and radial components of the van der Waals complex, whether it is valid for complexes that have large radial and angular motions could not be determined without testing it on a radially floppy complex. The results presented here show for the first time that the adiabatic approximation is valid for He-HCN and other radially floppy complexes.

By showing that the adiabatic approximation is appropriate for helium van der Waals complexes, this thesis shows that the adiabatic approximation can be used to calculate approximate wavefunctions for these systems. The major application of the method is expected to be in the approximation of helium-impurity molecule dimer wavefunctions. These approximate wavefunctions can then be used to construct wavefunctions for Diffusion Monte Carlo calculations of excited state helium clusters. The code for the program written to calculate these wavefunctions is given in Appendix B.

APPENDICES

Appendix A. The Fourier Grid Hamiltonian Method

The radial wave function, $\chi(R)$, is found by solving the "radial" Schrödinger equation,

$$\left[\frac{\hbar^2}{2\mu}\nabla^2 + U(R)\right]\chi(R) = [E_R]\,\chi(R). \tag{1}$$

In this equation, the potential, $U(R)$, is the effective radial potential, which is represented by the eigenvalues that are found by diagonalizing the approximate Hamiltonian matrix. To solve this equation, the Fourier Grid Hamiltonian method developed by Balint-Kurti was employed [1, 2].

The Fourier Grid Hamiltonian method utilizes the fact that the Hamiltonian is comprised of a kinetic and a potential term and that these two terms are best treated in different representations. The kinetic operator is most easily treated in the momentum representation, where it is diagonal, while the potential operator is easiest to handle in the coordinate representation in which it is diagonal. The essence of the method is to express each of these terms in the representations that are easiest to use and then connect them using the Fourier transform,

$$\langle k\,|x\rangle = \frac{1}{\sqrt{2\pi}}e^{-ikx}. \tag{2}$$

This method allows bound state eigenvalues and eigenfunctions for a one-dimensional Schrödinger equation to be easily found.

References

[1] C. Marston, G. Balint-Kurti, J. Chem. Phys 91 (1989), 3571.

[2] G. Balint-Kurti, R. Dixon, C. Marston, Internat. Rev. Phys. Chem. 11 (1992) 317.

Appendix B. Fortran 90 Program Used to Calculate He-HCN Trial Wave Function

```
PROGRAM Adiabatic_Approximation_for_He-HCN
!C This program calculates a trial wave function for the van der Waals
!C molecule He-HCN using an adiabatic separation of angular and radial
!C motion. A description of the theory behing the adiabatic method is
!C given in Holmgren, Waldman, and Klemperer, J. Chem. Phys. 67 (1977)
!C 4414. The angular wave function calculated using the method can be
!C made to be a better approximation if it is multiplied by the radial
!C wave function. Thus, the final wave function this program calculates
!C is a product of the radial wave function and the angular wave function.
!C The radial wavefunction was calculated here using the Fast Fourier
!C Grid Hamiltonian method developed by Balint-Kurti and Marston. The
!C potential for He-HCN was taken from a paper by Atkins and Hutson.
!C Many of the subroutines used were found in Numerical Recipes in FORTRAN,
!C 2nd edition by Press, Teukolsky, Vetterling, and Flannery, Cambridge
!C Press, Cambridge. The files that are output and what they correspond
!C to are listed in the program. Most of the calculations in the program
!C are done in atomic units although there were some necessary conversions
!C betweeen subroutines that were necessary.
!C –Dan Ward
!C
INTEGER size !size=size of matrix
INTEGER np,uu,zz,pp,vv,ff
INTEGER rinit,rfinal !initial and final values of r in angstroms
INTEGER i,j,ii,jj,n,nn,intr,kk,npoint,qq,q,intangle,intscale,intscale2
INTEGER xxmin,xxmax !xxmin and xxmax provide a way to use a spline routine
PARAMETER (size=20)
PARAMETER (np=size)
PARAMETER (npoint=20)
PARAMETER (intscale=100)
PARAMETER (intscale2=100)
PARAMETER (rinit=3*intscale)
PARAMETER (rfinal=7*intscale)
PARAMETER (uu=rfinal-rinit+1)
PARAMETER(xxmin=3*intscale2)
PARAMETER(xxmax=7*intscale2)
PARAMETER (vv=4)
PARAMETER (ff=rfinal-rinit+1) !(ff=xxmax-xxmin+1)
!These are the integer variables from the Fast Fourier Grid Method Program
(FFGM)
!*************************************************************************
INTEGER NX,ITEST,NFACT1,NFACT2,IJD,IERR,NWRIT,NPRIN
PARAMETER (nx=xxmax-xxmin,NWRIT=4,NPRIN=1) !xxmax-xxmin
```

```
!*************************************************************************
REAL*8 a(size,size),d(np),v(np,np),moe,e(np),pot,cn,cni,w(npoint),x(npoint)
REAL*8 dc !dc=kronecker delta(i,j)
REAL*8 b0,bigJ,angle,pi,mu,mHCN,mHe,twopi
REAL*8 threejterm,threejtermi,hbar
REAL*8 aa,bb,funct,THREEJ,plgndr,term0,potex,potexi
REAL*8 expandthreej,expandthreeji,psiq,psi,scale
REAL*8 xa(uu),ya(uu),y2(uu),yp1,ypn,xx
REAL*8 scale2,xa1(uu),ya1(uu),xa2(uu),ya2(uu),xa3(uu),ya3(uu)
REAL*8 aaa,bbb,ccc,ddd,psiq1,psi1,psiq2,psi2,psiq3,psi3
!These are the real variables from the Fast Fourier Grid Method Program
(FFGM)
!*************************************************************************
REAL*8
R0,WCH(NX),ZR(NX,NX),FV1(NX),FV2(NX),AR(NX,NX),R1,RMIN,RMAX,ZL,DX,CON
REAL*8
CONST2,DARG,RATIO,VVV,NEWX,PSQ,XXAA(UU),YYAA(UU),YYP1,YYPN,YY2(UU
REAL*8 eee,fff,ggg,hhh,IXA(NX)
PARAMETER (R0=1.d0) !7329D0)
!*************************************************************************
EXTERNAL funct,pot,plgndr,ocspotential,HCNpotential
!***Set values of parameters (in atomic units).
b0=6.698d-6 !6.698d-6 Hartree = 1.47 wavenumber
mHe=7.29629343490d3
mHCN=4.92646549378d4
pi=2.d0*dasin(1.d0)
mu=6.35508047048d3
aa=0.d0
bb=pi
twopi=2.d0*pi
hbar= 1.d0
scale=1.d0*intscale
scale2=1.d0*intscale2
!***Convert from degrees to radians
do intangle=1,ff
ang=180.d0/(rfinal-rinit)
angledeg=intangle*ang-ang
angle=angledeg*pi/180.d0
write(6,*) 'theta= ',angledeg,'degrees'
do intr=rinit,rfinal
moe=(intr/scale)/.529d0 !moe is the value of r in atomic units
pp=intr-rinit+1
psi=0.d0
```

```
psi1=0.d0
psi2=0.d0
psi3=0.d0
do ii=1,np
i=ii-1
do jj=1,np
j=jj-1
cn=0.d0
threejterm=0.d0
leg=0.d0
potex=0.d0
expandthreej=0.d0
!************This segment calculates cn and threej^
2**********************************
do nn=1,15
n=nn-1
expandthreeji=0.d0
cni=0.d0
threejtermi=0.d0
potexi=0.d0
!***calculate three j symbols
threejtermi=(THREEJ(i,n,j))**2
!***Integrate to find the expansion coefficients***
call gauleg(aa,bb,x,w,npoint) !* This is the integration of
!* potential(r,theta)*
do kk=1,npoint !* LegendreP[n,Cos[theta]]
cni=cni+w(kk)*funct(x(kk),moe,n) !*to find the expansion coeff.
enddo
!*************************************************
potexi=cni*plgndr(n,0,angle)
potex=potex+potexi
expandthreeji=cni*threejtermi
expandthreej=expandthreej+expandthreeji
enddo
expandthreej=expandthreej*4.55635d-6 !converts cn from cm-1 to hartree
! *3.16683d-6 kelvin
!***This segment calculates the matrix
elements*******************************
if (i.ne.j) then !This "if" loop calculates dc=kroneckerdelta(i,j)
dc=0.d0
else
dc=1.d0
endif
```

```
bigJ=0.d0
term0=dsqrt(4.d0*i*j+2.d0*i+2.d0*j+1.d0)
a(ii,jj)= b0*((j**2)+j)*dc + &
term0*expandthreej + &
((0.5d0)/(mu*((moe)**2)))* &
((hbar**2)*((bigJ**2)+bigJ+(j**2)+j) )*dc
expandthreej=0.d0
enddo
enddo
!***Diagonalize the matrix
call jacobi(a,size,np,d,v,nrot) !Jacobi diagonalizes the matrix, a,
!and then puts the eigenvalues in the array d.
!The eigenvectors (in the legendre polynomial
!basis) are in the array v.
!***Convert eigenvalues (radial potential from a.u. to wave #
do j=1,np
e(j)=d(j)*219474.d0 !This converts from Eh to wave#
enddo
!***Calculate angular wave functions
do qq=1,np
q=qq-1
psiq=v(qq,1)*plgndr(q,0,angle) !psi is the ground state angular
psi=psi+psiq !wave function.
psiq1=v(qq,2)*plgndr(q,0,angle) !psi1 is the first excited state
psi1=psi1+psiq1 !angular wave function.
psiq2=v(qq,3)*plgndr(q,0,angle) !psi2 is the first excited state
psi2=psi2+psiq2 !angular wave function.
psiq3=v(qq,4)*plgndr(q,0,angle) !psi3 is the first excited state
psi3=psi3+psiq3 !angular wave function.
enddo
!***get x and y values for wavefunction spline
xa(pp)=moe*.529d0
ya(pp)=psi
xa1(pp)=moe*.529d0
ya1(pp)=psi1
xa2(pp)=moe*.529d0
ya2(pp)=psi2
xa3(pp)=moe*.529d0
ya3(pp)=psi3
!***This segment puts the radial potential into an array for the FFGH method.
XXAA(pp)=moe
YYAA(pp)=d(1)
!*************************************************************************
```

```
!***Write to file
write(10,*) moe*.529d0, angledeg, psi
write(11,*) moe*.529d0, angledeg, psi1
write(12,*) moe*.529d0, angledeg, psi2
write(13,*) moe*.529d0, angledeg, psi3
write(99,*) moe*0.529,angledeg,HCNpotential(moe*.529d0,angle)
write(98,*) moe*.529,angledeg,potex
!***Reset values of angular wave functions
psi=0.d0
psi1=0.d0
psi2=0.d0
psi3=0.d0
enddo !end of r loop
!**********BEGINNING OF FFGM*********************************************
!C....TEST THE EFFECTIVE POTENTIAL
DO J=xxmin,xxmax !200
R1=dfloat(J)/scale2/.529d0
call SUB(R1,xxaa,yyaa,uu,poten)
!write(6,*) r1*.529d0,poten*2.19474d5
WRITE(30,*) r1*.529d0,POTEN*2.19474D5
poten=0.d0
!call morsepot(r1,VVV)
!write(6,*) ' r1 ',' VVV '
!write(31,*) r1,VVV
ENDDO
!C....TEST THAT NX IS EVEN
ITEST=MOD(NX,2)
IF(ITEST.NE.0) THEN
WRITE(6,*) '**** NX MUST BE EVEN-FATAL ERROR ****'
STOP
ENDIF
!C....SET UP GRID
WRITE(6,*)'GRID PARAMETERS:'
WRITE(6,*)' NUMBER OF GRID POINTS = ',NX
RMIN=xxmin/scale2/0.529d0 !rinit/scale/.529d0 !xxmin/100.d0/.529d0
RMAX=xxmax/scale2/0.529d0 !rfinal/scale/.529d0 !xxmax/100.d0/.529d0
ZL=(RMAX-RMIN)
WRITE(6,*)' GRID LENGTH = ',ZL
DX=ZL/dfloat(NX)
WRITE(6,*) ' GRID SPACINGS = ',DX
!C....COMPUTE CONSTANTS
PSQ=PI*PI
CONST1=PSQ/(MU*(ZL**2))
```

```
      NFACT1=(NX-1)*(NX-2)
      NFACT2=(NX-2)/2
      CONST2=CONST1*(dfloat(NFACT1)/6.D0+1.D0+dfloat(NFACT2))
      DARG=PI/dfloat(NX)
      !C....NOW COMPUTE HAMILTONIAN MATRIX
      NEWX=RMIN
      DO I=1,NX
      ixa(I)=NEWX
      DO J=1,I
      IJD=(I-J)
      IF(IJD.EQ.0) THEN
      AR(I,J)=CONST2
      ELSE
      RATIO=1.D0/DSIN(DARG*dfloat(IJD))
      AR(I,J)=((-1)**IJD)*CONST1*(RATIO**2)
      ENDIF
      ENDDO
      !C....FIND THE POTENTIAL VALUE AT X
      call SUB(newx,xxaa,yyaa,uu,VVV)
      !call morsepot(newx,VVV)
      !C....ADD THE POTENTIAL VALUE WHEN THE KRONECKER DELTA
FUNCTION
      !C....EQUALS ONE, I.E. WHEN I AND J ARE EQUAL
      AR(I,I)=AR(I,I)+VVV
      NEWX=NEWX+DX
      ENDDO
      !C....NOW FILL OUT HAMILTONIAN MATRIX
      DO I=1,NX
      DO J=1,I
      AR(J,I)=AR(I,J)
      ENDDO
      ENDDO
      !C....NOW CALL EIGENVALUE SOLVER
      CALL FLUSH(6)
      CALL RS(NX,NX,AR,WCH,NPRIN,ZR,FV1,FV2,IERR)
      !C....PRINT OUT E'VALUES AND E'VECTORS
      WRITE(6,12)
      12 FORMAT(/)
      WRITE(6,*)' THE FIRST',NWRIT,' ENERGY LEVELS FOR He-HCN MOLECULE
,
      DO I=1,NWRIT
      WRITE(6,*)' ENERGY LEVEL NO. ',I,'E-VAL(1/cm)=',WCH(I)*2.19474D5
      !PAUSE
```

```
ENDDO
WRITE(6,12)
WRITE(6,*)' THE CORRESPONDING EIGENFUNCTIONS ARE:'
DO I=1,NWRIT
WRITE(6,*)' ENERGY LEVEL NO. ',I,' EIGENVALUE= ',WCH(I)
IF (NPRIN.EQ.1) THEN
DO J=1,NX
WRITE(6,*)' R = ',ixa(J)*.529,' WAVEFUNCTION=',ZR(J,I)
WRITE(20+i-1,*) ixa(J)*.529d0,ZR(J,I)
ENDDO
ENDIF
ENDDO
!***This segment calculates the modified wave functions**********************
do i=1,pp-1
write(75,*) xa(i),angledeg,ya(i)*zr(i,1)
write(81,*) xa1(i),angledeg,ya1(i)*zr(i,2)
write(82,*) xa2(i),angledeg,ya2(i)*zr(i,3)
write(83,*) xa3(i),angledeg,ya3(i)*zr(i,4)
enddo
!*****************************************************************************
write(10,*) !ground state angular wave function=psi
write(11,*) !first excited state angular wave function =psi1
write(12,*) !second excited state angular wave function =psi2
write(13,*) !third excited state angular wave function =psi3
write(75,*) !modified ground state wave function=
!(ground state angular wf=psi)*(ground state radial wf)
write(81,*) !modified first excited state wave function=
!(1st excited state angular wf=psi1)*(1st excited state radial wf)
write(82,*) !modified second excited state wave function=
!(2nd excited state angular wf=psi2)*(2nd excited state radial wf)
write(83,*) !modified third excited state wave function=
!(3rd excited state angular wf=psi3)*(3rd excited state radial wf)
write(98,*) !He-HCN potential expansion
write(99,*) !He-HCN potential
enddo !end of angle loop
!*****************************************************************************
!*This program outputs the following files: *
!* 1. Angular Wave Function (fort.10) *
!* 2. Potential (fort.99) *
!* 3. Potential Expansion (fort.98) *
!* 4. Radial Potential (fort.30) *
!* 5. Radial Wave Function (fort.20) *
!* 6. Trial Wave Function–Radial Wave Function * Angular Wave Function
```

```
(fort.75) *
!*************************************************************************
END
REAL*8 FUNCTION funct(theta,rr,o) !funct with r in Angstroms
REAL*8 r,theta,functi,rr
INTEGER o
r=rr*0.529d0
functi=plgndr(o,0,theta)*HCNpotential(r,theta)
funct=functi*(dsin(theta)*(2.d0*o+1.d0)/2.d0)
RETURN
END
REAL*8 FUNCTION pot(r,theta)
REAL*8 r,theta,alpha,rm,eps,p0r,p1r,p2r,p3r,p0a,p1a,p2a,p3a,K,G,L,Va,Vr
REAL*8 term1,term2,term3,term4,term5 !,terma,termb
alpha=13.5d0
eps=202.d0
rm=3.805d0
p0r=1.d0
p1r=0.650d0 !0.35d0
p2r=0.919d0 !0.65d0 !
p3r=0.d0
p0a=1.d0
p1a=0.313d0 !0.35d0 !
p2a=0.400d0 !0.09d0 !
p3a=0.d0
K=eps*((6.d0/alpha)/(1.d0-(6.d0/alpha)))
G=alpha/rm
L=(-eps/(1.d0-(6.d0/alpha)))*rm**6
term1=(K*dEXP(13.5d0-G*R))
term2=(1.d0+P1R*dCOS(theta)+P2R*(1.5d0*(dCOS(theta)**2)-0.5d0))
term3=(L/(r**6))
term4=1.d0+P1A*dCOS(theta)
term5=P2A*(1.5d0*(dCOS(theta)**2)-0.5d0)
!POT=term1*term2+term3*(term4+term5)
Va= -eps*(alpha/(alpha-6.d0))*((rm/r)**6)*(p0a+p1a*(rm/r)*dcos(theta)+ &
p2a*(-0.5d0+(3.d0*dcos(theta)**2)/2.d0)+p3a*(((-3.d0*dcos(theta))/2.d0) + &
(5.d0*dcos(theta)**2)/2.d0))
Vr=
eps*(6.d0/(alpha-6.d0))*(dexp(alpha*(1.d0-(r/rm))))*(p0r+p1r*(dcos(theta))+
&
p2r*(-0.5d0+(3.d0*dcos(theta)**2)/2.d0)+p3r*(((-3.d0*dcos(theta))/2.d0) + &
(5.d0*dcos(theta)**2)/2.d0))
POT=Va+Vr
```

```
RETURN
END
REAL*8 FUNCTION HCNpotential(R,theta)
!He-HCN potential from Atkins and Hutson in JCP vol. 105, p. 440 (1996)
!Distances enter in a.u.; angles in radians and energies leave in cm^-1
real*8
R,theta,alpha,mu,thetabig,beta,eps0,eps1,eps2,eps3,eps,rm,rm0,rm1,rm2,rm3, &
c6,c60,c62,c7,c71,c73,d6,d7,vfix,d8,d8prime,c8,A,vrep,vind,vdisp, &
b1,b2,b3,b4,b5,b6,b7,b8,d6rm,d7rm,d8rm,vindrm
alpha=1.383d0
mu=1.174d0
thetabig=1.777d0
c60=13.067d0
c62=1.864d0
c71=9.939d0
c73=9.401d0
beta=3.90115d0
eps0=24.825139d0
eps1=3.402d0
eps2=0.3853d0
eps3=0.854d0
rm0=3.7151d0
rm1=0.200d0
rm2=0.4582d0
rm3=0.071d0
eps=eps0+eps1*dcos(theta)+eps2*(-0.5d0 + (3.d0*dcos(theta)**2)/2.d0)+ &
eps3*((-3.d0*dcos(theta))/2.d0 + (5.d0*dcos(theta)**3)/2.d0)
rm=rm0+rm1*dcos(theta)+rm2*(-0.5d0 + (3.d0*dcos(theta)**2)/2.d0)+ &
rm3*((-3.d0*dcos(theta))/2.d0 + (5.d0*dcos(theta)**3)/2.d0)
vindrm=-(alpha*(mu**2)*(1.d0+(-0.5d0 +
(3.d0*dcos(theta)**2)/2.d0))*rm**(-6))- &
(6*alpha*mu*thetabig*((dcos(theta))**3)*rm**(-7))
c6=c60+c62*(-0.5d0 + (3.d0*dcos(theta)**2)/2.d0)
c7=c71*dcos(theta)+c73*((dcos(theta))**3)
b1=beta
b2=(beta**2)/2.d0
b3=(beta**3)/6.d0
b4=(beta**4)/24.d0
b5=(beta**5)/120.d0
b6=(beta**6)/720.d0
b7=(beta**7)/5040.d0
b8=(beta**8)/40320.d0
```

```
      d6rm=1-(dexp(-beta*rm))*(1+(b1*rm)+(b2*rm**2)+(b3*rm**3)+(b4*rm**4)+
     &
      (b5*rm**5)+(b6*rm**6))
      d7rm=1-(dexp(-beta*rm))*(1+(b1*rm)+(b2*rm**2)+(b3*rm**3)+(b4*rm**4)+
     &
      (b5*rm**5)+(b6*rm**6)+(b7*rm**7))
      d8rm=1-(dexp(-beta*rm))*(1+(b1*rm)+(b2*rm**2)+(b3*rm**3)+(b4*rm**4)+
     &
      (b5*rm**5)+(b6*rm**6)+(b7*rm**7)+(b8*rm**8))
      d8prime=beta*(d7rm-d8rm)
      vfix=vindrm-(c6*d6rm*rm**(-6))-(c7*d7rm*rm**(-7))
      c8=((beta*eps*rm**8)+(beta*vfix*rm**8)+(vfix*rm**8))/((8.d0*d8prime/rm) &
      -(beta*d8prime)-(8.d0*d8rm/rm)+(beta*d8rm))
      A=(c8*d8rm*(rm**(-8))*dexp(beta*rm))-(eps*dexp(beta*rm))-(vfix*dexp &
      (beta*rm))
      d6=1-(dexp(-beta*R))*(1+(b1*R)+(b2*R**2)+(b3*R**3)+(b4*R**4)+(b5*R**5)
     &
      +(b6*R**6))
      d7=1-(dexp(-beta*R))*(1+(b1*R)+(b2*R**2)+(b3*R**3)+(b4*R**4)+(b5*R**5)
     &
      +(b6*R**6)+(b7*R**7))
      d8=1-(dexp(-beta*R))*(1+(b1*R)+(b2*R**2)+(b3*R**3)+(b4*R**4)+(b5*R**5)
     &
      +(b6*R**6)+(b7*R**7)+(b8*R**8))
      vrep=A*dexp(-beta*R)
      vind=-alpha*(mu**2)*(0.5d0 + (3.d0*dcos(theta)**2)/2.d0)*(R**(-6))- &
      6*alpha*mu*thetabig*((dcos(theta))**3)*(R**(-7))
      vdisp=-(c6*d6*(R**(-6)))-(c7*d7*(R**(-7)))-(c8*d8*(R**(-8)))
      HCNpotential=(vrep+vind+vdisp)
      RETURN
      END
      REAL*8 FUNCTION ocspotential(r,theta)
      implicit real*8(a-h,o-z)
      ! MMSV fit to Higgins/Klemperer Potential. JCP vol. 110, p. 1383 (1999)
      ! Distances come in in a.u., angles in radians and
      ! energies exit in cm-1.
      ! All internal calculations are done in Angstroms/cm-1
      rlim=12.25d0
      c6= -5.518337819784241d6 - &
      2.815526194265673d7*dcos(theta) - &
      1.222022540841727d6* &
      dcos(theta)**2 + &
      9.28433189859905d7*dcos(theta)**3 - &
```

```
5.147503485810399d7* &
dcos(theta)**4 - &
2.371545691135376d8* &
dcos(theta)**5 - &
4.298021781056592d7* &
dcos(theta)**6 + &
3.356904207752721d8* &
dcos(theta)**7 + &
3.711123257316639d8* &
dcos(theta)**8 - &
2.187342071289152d8* &
dcos(theta)**9 - &
4.779521200196144d8* &
dcos(theta)**10 + &
5.369489863099884d7* &
dcos(theta)**11 + &
2.052905607469811d8*dcos(theta)**12
c7=8.11427571093064d7 + &
6.915953104421285d8*dcos(theta) - &
3.302620601496621d6* &
dcos(theta)**2 - &
2.469508814138928d9* &
dcos(theta)**3 + &
7.862550115104378d8* &
dcos(theta)**4 + &
6.917741424386369d9* &
dcos(theta)**5 + &
2.361213150132689d9* &
dcos(theta)**6 - &
1.056687114337185d10* &
dcos(theta)**7 - &
1.131235010642686d10* &
dcos(theta)**8 + &
7.644808382222111d9* &
dcos(theta)**9 + &
1.340802869471866d10* &
dcos(theta)**10 - &
2.110569758270181d9* &
dcos(theta)**11 - &
5.555319238436844d9*dcos(theta)**12
c8=1.772439907199721d8 - &
3.640726069925961d9*dcos(theta) + &
3.75432581972346d8*dcos(theta)**2 + &
```

```
1.455420403849449d10*&
dcos(theta)**3 + &
1.000003378656614d9*&
dcos(theta)**4 - &
4.617315309215618d10* &
dcos(theta)**5 - &
2.797291174372391d10* &
dcos(theta)**6 + &
7.628291799349615d10* &
dcos(theta)**7 + &
8.72010434775106d10* &
dcos(theta)**8 - &
6.021553008035978d10* &
dcos(theta)**9 - &
9.29303148210638d10* &
dcos(theta)**10 + &
1.798370591378436d10* &
dcos(theta)**11 + &
3.658316840255752d10*dcos(theta)**12
eps= 26.56028761259177d0 - &
66.09411254231051d0*dcos(theta) + &
113.7818762706877d0*dcos(theta)**2 + &
253.9842073871831d0*dcos(theta)**3 - &
861.492606088799d0*dcos(theta)**4 - &
372.9912295962823d0*dcos(theta)**5 + &
2377.068086446698d0*dcos(theta)**6 + &
205.2388974075886d0*dcos(theta)**7 - &
3252.98701566141d0*dcos(theta)**8 + &
12.69015656350256d0*dcos(theta)**9 + &
2231.436969104492d0*dcos(theta)**10 - &
31.65780133613179d0*dcos(theta)**11 - &
607.0829551078825d0*dcos(theta)**12
rm=6.92914426071697d0 + &
3.120064333769215d0*dcos(theta) + &
0.7266538203334347d0*dcos(theta)**2 - &
9.33193775575554d0*dcos(theta)**3 + &
12.48406119957126d0*dcos(theta)**4 + &
9.27208587287942d0*dcos(theta)**5 - &
30.3186120249881d0*dcos(theta)**6 + &
0.1118890543344939d0*dcos(theta)**7 + &
33.45333098260939d0*dcos(theta)**8 - &
6.833790407369015d0*dcos(theta)**9 - &
18.5896707305704d0*dcos(theta)**10 + &
```

```
3.376275567894582d0*dcos(theta)**11 + &
4.162194568737485d0*dcos(theta)**12
alpha1=0.840448904141149d0 - &
0.3277295205015458d0*dcos(theta) + &
0.4753915908998801d0*dcos(theta)**2 + &
2.024536820689881d0*dcos(theta)**3 - &
3.3499513477514d0*dcos(theta)**4 - &
6.291793710168604d0*dcos(theta)**5 + &
9.84880899105184d0*dcos(theta)**6 + &
10.26389067511479d0*dcos(theta)**7 - &
13.62883386586445d0*dcos(theta)**8 - &
8.36423058246549d0*dcos(theta)**9 + &
9.18459266695318d0*dcos(theta)**10 + &
2.668208793886965d0*dcos(theta)**11 - &
2.404330940098348d0*dcos(theta)**12
alpha2= 0.818336538913902d0 - &
0.2992044234243529d0*dcos(theta) + &
0.4697371433769803d0*dcos(theta)**2 + &
1.665773063318903d0*dcos(theta)**3 - &
3.089373409988408d0*dcos(theta)**4 - &
4.735292584383614d0*dcos(theta)**5 + &
8.62422346889475d0*dcos(theta)**6 + &
7.247704748785174d0*dcos(theta)**7 - &
11.55703899826573d0*dcos(theta)**8 - &
5.664708797189565d0*dcos(theta)**9 + &
7.619667615634005d0*dcos(theta)**10 + &
1.764938759546422d0*dcos(theta)**11 - &
1.952200596603219d0*dcos(theta)**12
! Region I of MMSV potential
if(r.lt.rm) then
vv1=dexp(-2.d0*alpha1*(r-rm))
vv2=dexp(-alpha1*(r-rm))
vv=eps*(vv1-2.d0*vv2)
endif
rtest=rm+dlog(2.d0)/alpha2
! Region II of MMSV potential
if(r.gt.rm.and.r.lt.rtest) then
vv1=dexp(-2.d0*alpha2*(r-rm))
vv2=dexp(-alpha2*(r-rm))
vv=eps*(vv1-2.d0*vv2)
endif
x1=rtest/rm
x2=rlim/rm
```

```
x=r/rm
! Region III of MMSV potential
if(x.ge.x1.and.x.lt.x2) then
dx=x2-x1
term1=dexp(-rm*alpha2*(x1-1.d0))
beta1=eps*term1*(term1-2.d0)
term2=1.d0/(rm**2*x2**2)
term3=-(c8/(rm**8*x2**8)) - &
c7/(rm**7*x2**7) - &
c6/(rm**6*x2**6)
beta2=(term3-beta1)/dx
term4=-2.d0*rm*alpha2*eps*term1*(term1-1.d0)
beta3=-(term4-beta2)/dx
term5=(8*c8)/(rm**8*x2**9) + &
(7*c7)/(rm**7*x2**8) + &
(6*c6)/(rm**6*x2**7)
beta4=(term5-dx*beta3-beta2)/dx**2
term6=x-x1
term7=x-x2
vv=beta1+term6*(beta2+term7*(beta3+term6*beta4))
vv=vv
endif
! Region IV of MMSV potential
if(r.ge.rlim) then
vv=(-c6/r**6-c7/r**7-c8/r**8)
endif
ocspotential=vv
return
end
SUBROUTINE gauleg(x1,x2,x,w,n)
INTEGER n
REAL*8 x1,x2,x(n),w(n),EPS
PARAMETER (EPS=3.d-16)
INTEGER i,j,m
REAL*8 p1,p2,p3,pp,xl,xm,z,z1
m=(n+1)/2
xm=0.5d0*(x2+x1)
xl=0.5d0*(x2-x1)
do 12 i=1,m
z=cos(3.141592654d0*(i-.25d0)/(n+.5d0))
1 continue
p1=1.d0
p2=0.d0
```

```
do 11 j=1,n
p3=p2
p2=p1
p1=((2.d0*j-1.d0)*z*p2-(j-1.d0)*p3)/j
11 continue
pp=n*(z*p1-p2)/(z*z-1.d0)
z1=z
z=z1-p1/pp
if(abs(z-z1).gt.EPS)goto 1
x(i)=xm-xl*z
x(n+1-i)=xm+xl*z
w(i)=2.d0*xl/((1.d0-z*z)*pp*pp)
w(n+1-i)=w(i)
12 continue
return
END
FUNCTION plgndr(l,m,xx)
INTEGER l,m,i,ll
REAL*8 plgndr
REAL*8 x,xx
REAL*8 fact,pll,pmm,pmmp1,somx2
x=dcos(xx)
if(m.lt.0.or.m.gt.l.or.dabs(x).gt.1.d0) then
!pause'bad arguments in plgndr'
endif
pmm=1.d0
if(m.gt.0) then
somx2=dsqrt((1.d0-x)*(1.d0+x))
fact=1.d0
do 11 i=1,m
pmm=-pmm*fact*somx2
fact=fact+2.d0
11 continue
endif
if(l.eq.m) then
plgndr=pmm
else
pmmp1=x*(2.d0*m+1)*pmm
if(l.eq.m+1) then
plgndr=pmmp1
else
do 12 ll=m+2,l
pll=(x*(2.d0*ll-1)*pmmp1-(ll+m-1.d0)*pmm)/(ll-m)
```

```
pmm=pmmp1
pmmp1=pll
12 continue
plgndr=pll
endif
endif
return
END
SUBROUTINE jacobi(a,n,np,d,v,nrot)
INTEGER n,np,nrot,NMAX
REAL*8 a(np,np),d(np),v(np,np)
PARAMETER (NMAX=500)
INTEGER i,ip,iq,j
REAL*8 c,g,h,s,sm,t,tau,theta,tresh,b(NMAX),z(NMAX)
do 12 ip=1,n
do 11 iq=1,n
v(ip,iq)=0.d0
11 continue
v(ip,ip)=1.d0
12 continue
do 13 ip=1,n
b(ip)=a(ip,ip)
d(ip)=b(ip)
z(ip)=0.d0
13 continue
nrot=0
do 24 i=1,50
sm=0.d0
do 15 ip=1,n-1
do 14 iq=ip+1,n
sm=sm+dabs(a(ip,iq))
14 continue
15 continue
if(sm.eq.0.d0)return
if(i.lt.4)then
tresh=0.2d0*sm/n**2
else
tresh=0.d0
endif
do 22 ip=1,n-1
do 21 iq=ip+1,n
g=100.d0*dabs(a(ip,iq))
if((i.gt.4).and.(dabs(d(ip))+g.eq.dabs(d(ip))).and. &
```

```
(dabs(d(iq))+g.eq.dabs(d(iq))))then
a(ip,iq)=0.d0
else if(dabs(a(ip,iq)).gt.tresh)then
h=d(iq)-d(ip)
if(dabs(h)+g.eq.dabs(h))then
t=a(ip,iq)/h
else
theta=0.5d0*h/a(ip,iq)
t=1.d0/(dabs(theta)+dsqrt(1.d0+theta**2))
if(theta.lt.0.d0)t=-t
endif
c=1.d0/dsqrt(1+t**2)
s=t*c
tau=s/(1.d0+c)
h=t*a(ip,iq)
z(ip)=z(ip)-h
z(iq)=z(iq)+h
d(ip)=d(ip)-h
d(iq)=d(iq)+h
a(ip,iq)=0.d0
do 16 j=1,ip-1
g=a(j,ip)
h=a(j,iq)
a(j,ip)=g-s*(h+g*tau)
a(j,iq)=h+s*(g-h*tau)
16 continue
do 17 j=ip+1,iq-1
g=a(ip,j)
h=a(j,iq)
a(ip,j)=g-s*(h+g*tau)
a(j,iq)=h+s*(g-h*tau)
17 continue
do 18 j=iq+1,n
g=a(ip,j)
h=a(iq,j)
a(ip,j)=g-s*(h+g*tau)
a(iq,j)=h+s*(g-h*tau)
18 continue
do 19 j=1,n
g=v(j,ip)
h=v(j,iq)
v(j,ip)=g-s*(h+g*tau)
v(j,iq)=h+s*(g-h*tau)
```

```
      19 continue
      nrot=nrot+1
      endif
      21 continue
      22 continue
      do 23 ip=1,n
      b(ip)=b(ip)+z(ip)
      d(ip)=b(ip)
      z(ip)=0.d0
      23 continue
      24 continue
      !pause 'too many iterations in jacobi'
      return
      END
      REAL*8 FUNCTION THREEJ (J1,J2,J3)
      !IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      !C
      !C COMPUTATION OF SPECIAL WIGNER 3J COEFFICIENT WITH
      !C VANISHING PROJECTIONS. SEE EDMONDS, P. 50.
      !C
      !C THIS VERSION EVALUATES BINOM AND PARITY IN-LINE
      !C SHOULD IMPROVE EFFICIENCY, ESPECIALLY ON CRAY;
      !C ALSO GIVES IMPROVEMENT ON AMDAHL (SG: 20 DEC 92)
      !C
      !C STATEMENT FUNCTION FOR DELTA ASSOCIATED W/ RACAH AND
    SIXJ SYMBOLS
      !C DELTA(I,J,K)= SQRT(1.D0/ ( BINOM(I+J+K+1,I+J-K) *
      !C 1 BINOM(K+K+1,I-J+K) * DBLE(K+J-I+1) ) )
      !C
      INTEGER J1,J2,J3, I1,I2,I3,I4,I5,I6,N,M,NM,MNM,FN
      REAL*8 SIGN,F,C,B,B1,B2,B3,B4,DELTA
      I1=J1+J2+J3
      IF (I1-2*(I1/2).NE.0) GO TO 8
      1 I2=J1-J2+J3
      IF (I2.lt.0.d0) goto 8
      if (i2.eq.0.d0) goto 2
      if (i2.gt.0.d0) goto 2
      2 I3=J1+J2-J3
      IF (I3.lt.0.d0) goto 8
      if (i3.eq.0.d0) goto 3
      if (i3.gt.0.d0) goto 3
      3 I4=-J1+J2+J3
      IF (I4.lt.0.d0) goto 8
```

```
if (i4.eq.0.d0) goto 4
if (i4.gt.0.d0) goto 4
4 I5=I1/2
I6=I2/2
SIGN=1.D0
IF (I5-2*(I5/2).NE.0) SIGN=-SIGN
!C 7 THREEJ=SIGN*DELTA(J1,J2,J3)*BINOM(I5,J1)*BINOM(J1,I6)
!C B1,B2 ARE BINOM ASSOCIATED W/ DELTA
N=J1+J2+J3+1
M=J1+J2-J3
NM = N-M
MNM = MIN(NM,M)
IF(MNM.LE.0) THEN
B1=1.D0
ELSE
FN = N+1
F = 0.D0
B = 1.D0
DO I = 1,MNM
F = F+1.D0
C = (FN-F)*B
B = C/F
enddo
B1 = B
ENDIF
N=J3+J3+1
M=J1-J2+J3
NM = N-M
MNM = MIN(NM,M)
IF(MNM.LE.0) THEN
B2=1.D0
ELSE
FN = N+1
F = 0.D0
B = 1.D0
DO I = 1,MNM
F = F+1.D0
C = (FN-F)*B
B = C/F
enddo
B2 = B
ENDIF
DELTA=dSQRT(1.D0/(B1*B2*(J3+J2-J1+1)))
```

```
!C B3=BINOM(I5,J1), B4=BINOM(J1,I6)
N=I5
M=J1
NM = N-M
MNM = MIN(NM,M)
IF(MNM.LE.0) THEN
B3=1.D0
ELSE
FN = N+1
F = 0.D0
B = 1.D0
DO I = 1,MNM
F = F+1.D0
C = (FN-F)*B
B = C/F
enddo
B3 = B
ENDIF
N=J1
M=I6
NM = N-M
MNM = MIN(NM,M)
IF(MNM.LE.0) THEN
B4=1.D0
ELSE
FN = N+1
F = 0.D0
B = 1.D0
DO I = 1,MNM
F = F+1.D0
C = (FN-F)*B
B = C/F
enddo
B4 = B
ENDIF
THREEJ=SIGN*DELTA*B3*B4
RETURN
8 THREEJ=0.D0
RETURN
END
SUBROUTINE RS(NM,N,A,W,MATZ,Z,FV1,FV2,IERR)
IMPLICIT real*8(A-H,O-Z)
DIMENSION A(NM,N),W(N),Z(NM,N),FV1(N),FV2(N)
```

```
      !C ***********************************************************
      !C THIS SUBROUTINE CALLS THE RECOMMENDED SEQUENCE OF
      !C SUBROUTINES FROM THE EIGENSYSTEM SUBROUTINE PACKAGE
(EISPACK)
      !C TO FIND THE EIGENVALUES AND EIGENVECTORS (IF DESIRED)
      !C OF A REAL SYMMETRIC MATRIX.
      !C
      !C ON INPUT :
      !C
      !C N,M MUST BE SET TO THE ROW DIMENSION OF THE TWO-DIMENSIONAL
      !C ARRAY PARAMETERS AS DECLARED IN THE CALLING PROGRAM
      !C DIMENSION STATEMENT,
      !C N IS THE ORDER OF THE MATRIX A,
      !C A CONTAINS THE REAL SYMMETRIC MATRIX,
      !C MATZ IS AN INTEGER VARIABLE SET EQUAL TO ZERO IF
      !C ONLY EIGENVALUES ARE DESIRED, OTHERWISE IT IS SET TO
      !C ANY NON-ZERO INTEGER FOR BOTH EIGENVALUES AND EIGEN-
VECTORS.
      !C
      !C ON OUTPUT :
      !C
      !C W CONTAINS THE EIGENVALUES IN ASCENDING ORDER,
      !C Z CONTAINS THE EIGENVECTORS IF MATZ IS NOT ZERO,
      !C IERR IS AN INTEGER OUTPUT VARIABLE SET EQUAL TO AN
      !C ERROR COMPLETION CODE DESCRIBED IN SECTION 2B OF THE
      !C DOCUMENTATION. THE NORMAL COMPLETION CODE IS ZERO,
      !C FV1 AND FV2 ARE TEMPORARY STORAGE ARRAYS.
      !C
      !C QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GAR-
BOW,
      !C APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORA-
TORY
      !C ***********************************************************
      IF (N .LE. NM) GO TO 10
      IERR = 10 * N
      GO TO 50
      !C
   10 IF (MATZ .NE. 0) GO TO 20
      !C ********** FIND EIGENVALUES ONLY **********
      CALL TRED1(NM,N,A,W,FV1,FV2)
      CALL TQLRAT(N,W,FV2,IERR)
      GO TO 50
      !C ********** FIND BOTH EIGENVALUES AND EIGENVECTORS **********
```

```
   20 CALL TRED2(NM,N,A,W,FV1,Z)
   CALL TQL2(NM,N,W,FV1,Z,IERR)
   50 RETURN
   !C ********** LAST CARD OF RS **********
   END
   !C
   !C
   SUBROUTINE TRED1(NM,N,A,D,E,E2)
   IMPLICIT real*8(A-H,O-Z)
   DIMENSION A(NM,N),D(N),E(N),E2(N)
   !C ******************************************************************
   !C THIS SUBROUTINE IS A TRANSLATION OF THE ALGOL PROCEDURE
TRED1,
   !C NUM. MATH. 11, 181-195(1968) BY mArTIN, REINSCH, AND WILKIN-
SON.
   !C HANDBOOK FOR AUTO. COMP., VOL.II-LINEAR ALGEBRA, 212-226(1971).
   !C THIS SUBROUTINE REDUCES A REAL SYMMETRIC MATRIX
   !C TO A SYMMETRIC TRIDIAGONAL MATRIX USING
   !C ORTHOGONAL SIMILARITY TRANSFORMATIONS.
   !C
   !C ON INPUT :
   !C
   !C NM MUST BE SET TO THE ROW DIMENSION OF TWO-DIMENSIONAL
   !C ARRAY PARAMETERS AS DECLARED IN THE CALLING PROGRAM
   !C DIMENSION STATEMENT,
   !C
   !C N IS THE ORDER OF THE MATRIX,
   !C A CONTAINS THE REAL SYMMETRIC INPUT MATRIX. ONLY THE
   !C LOWER TRIANGLE OF THE MATRIX NEED BE SUPPLIED.
   !C
   !C ON OUTPUT :
   !C
   !C A CONTAINS INFORMATION ABOUT THE ORTHOGONAL TRANS-
   !C FORMATIONS USED IN THE REDUCTION IN ITS STRICT LOWER
   !C TRIANGLE. THE FULL UPPER TRIANGLE OF A IS UNALTERED,
   !C
   !C D CONTAINS THE DIAGONAL ELEMENTS OF THE TRIDIAGONAL
MATRIX,
   !C
   !C E CONTAINS THE SUBDIAGONAL ELEMENTS OF THE TRIDIAGONAL
   !C MATRIX IN ITS LAST N-1 POSITIONS. E(1) IS SET TO ZERO,
   !C
```

```
!C E2 CONTAINS THE SQUARES OF THE CORRESPONDING ELEMENTS
OF E.
!C E2 MAY COINCIDE WITH E IF THE SQUARES ARE NOT NEEDED.
!C
!C QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GAR-
BOW,
!C APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORA-
TORY
!C ****************************************************************
!C
DO I = 1, N
D(I) = A(I,I)
enddo
!C ********** FOR I=N STEP -1 UNTIL 1 DO - **********
DO 300 II = 1, N
I = N + 1 - II
L = I - 1
H = 0.0D0
SCALE = 0.0D0
IF (L .LT. 1) GO TO 130
!C ********** SCALE ROW (ALGOL TOL THEN NOT NEEDED) **********
DO K = 1, L
SCALE = SCALE + DABS(A(I,K))
enddo
!C
IF (SCALE .NE. 0.0D0) GO TO 140
130 E(I) = 0.0D0
E2(I) = 0.0D0
GO TO 290
!C
140 DO 150 K = 1, L
A(I,K) = A(I,K) / SCALE
H = H + A(I,K) * A(I,K)
150 CONTINUE
!C
E2(I) = SCALE * SCALE * H
F = A(I,L)
G = -DSIGN(DSQRT(H),F)
E(I) = SCALE * G
H = H - F * G
A(I,L) = F - G
IF (L .EQ. 1) GO TO 270
F = 0.0D0
```

```
!C
DO 240 J = 1, L
G = 0.0D0
!C ********** FORM ELEMENT OF A*U **********
DO K = 1, J
G = G + A(J,K) * A(I,K)
enddo
!C
JP1 = J + 1
IF (L .LT. JP1) GO TO 220
!C
DO K = JP1, L
G = G + A(K,J) * A(I,K)
enddo
!C ********** FORM ELEMENT OF P **********
220 E(J) = G / H
F = F + E(J) * A(I,J)
240 CONTINUE
!C
H = F / (H + H)
!C ********** FORM REDUCED A **********
DO 260 J = 1, L
F = A(I,J)
G = E(J) - H * F
E(J) = G
!C
DO K = 1, J
A(J,K) = A(J,K) - F * E(K) - G * A(I,K)
enddo
260 CONTINUE
!C
270 DO K = 1, L
A(I,K) = SCALE * A(I,K)
enddo
!C
290 H = D(I)
D(I) = A(I,I)
A(I,I) = H
300 CONTINUE
!C
RETURN
!C ********** LAST CARD OF TRED1 **********
END
```

```
      !C
      !C
      SUBROUTINE TRED2(NM,N,A,D,E,Z)
      IMPLICIT real*8(A-H,O-Z)
      DIMENSION A(NM,N),D(N),E(N),Z(NM,N)
      !C ***********************************************************
      !C THIS SUBROUTINE IS A TRANSLATION OF THE ALGOL PROCEDURE
TRED2,
      !C NUM. MATH. 11, 181-195(1968) BY MARTIN, REINSCH, AND WILKIN-
SON.
      !C HANDBOOK FOR AUTO. COMP., VOL.II-LINEAR ALGEBRA, 212-226(1971).
      !C
      !C THIS SUBROUTINE REDUCES A REAL SYMMETRIC MATRIX TO A
      !C SYMMETRIC TRIDIAGONAL MATRIX USING AND ACCUMULATING
      !C ORTHOGONAL SIMILARITY TRANSFORMATIONS.
      !C
      !C ON INPUT :
      !C
      !C NM MUST BE SET TO THE ROW DIMENSION OF TWO-DIMENSIONAL
      !C ARRAY PARAMETERS AS DECLARED IN THE CALLING PROGRAM
      !C DIMENSION STATEMENT,
      !C
      !C N IS THE ORDER OF THE MATRIX,
      !C
      !C A CONTAINS THE REAL SYMMETRIC INPUT MATRIX. ONLY THE
      !C LOWER TRIANGLE OF THE MATRIX NEED BE SUPPLIED.
      !C
      !C ON OUTPUT :
      !C
      !C D CONTAINS THE DIAGONAL ELEMENTS OF THE TRIDIAGONAL
MATRIX,
      !C
      !C E CONTAINS THE SUBDIAGONAL ELEMENTS OF THE TRIDIAGONAL
      !C MATRIX IN ITS LAST N-1 POSITIONS. E(1) IS SET TO ZERO,
      !C
      !C Z CONTAINS THE ORTHOGONAL TRANSFORMATION MATRIX
      !C PRODUCED IN THE REDUCTION,
      !C
      !C A AND Z MAY COINCIDE. IF DISTINCT, A IS UNALTERED.
      !C
      !C QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GAR-
BOW,
```

```
!C APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORA-
TORY
!C ************************************************************
DO 100 I = 1, N
DO J = 1, I
Z(I,J) = A(I,J)
enddo
100 CONTINUE
!C
IF (N .EQ. 1) GO TO 320
!C ********** FOR I=N STEP -1 UNTIL 2 DO - **********
DO 300 II = 2, N
I = N + 2 - II
L = I - 1
H = 0.0D0
SCALE = 0.0D0
IF (L .LT. 2) GO TO 130
!C ********** SCALE ROW (ALGOL TOL THEN NOT NEEDED) **********
DO K = 1, L
SCALE = SCALE + DABS(Z(I,K))
enddo
!C
IF (SCALE .NE. 0.0D0) GO TO 140
130 E(I) = Z(I,L)
GO TO 290
!C
140 DO 150 K = 1, L
Z(I,K) = Z(I,K) / SCALE
H = H + Z(I,K) * Z(I,K)
150 CONTINUE
!C
F = Z(I,L)
G = -DSIGN(DSQRT(H),F)
E(I) = SCALE * G
H = H - F * G
Z(I,L) = F - G
F = 0.0D0
!C
DO 240 J = 1, L
Z(J,I) = Z(I,J) / H
G = 0.0D0
!C ********** FORM ELEMENT OF A*U **********
DO K = 1, J
```

```
G = G + Z(J,K) * Z(I,K)
enddo
!C
JP1 = J + 1
IF (L .LT. JP1) GO TO 220
!C
DO K = JP1, L
G = G + Z(K,J) * Z(I,K)
enddo
!C ********** FORM ELEMENT OF P **********
220 E(J) = G / H
F = F + E(J) * Z(I,J)
240 CONTINUE
!C
HH = F / (H + H)
!C ********** FORM REDUCED A **********
DO 260 J = 1, L
F = Z(I,J)
G = E(J) - HH * F
E(J) = G
!C
DO K = 1, J
Z(J,K) = Z(J,K) - F * E(K) - G * Z(I,K)
enddo
260 CONTINUE
!C
290 D(I) = H
300 CONTINUE
!C
320 D(1) = 0.0D0
E(1) = 0.0D0
!C ********** ACCUMULATION OF TRANSFORMATION MATRICES **********
DO 500 I = 1, N
L = I - 1
IF (D(I) .EQ. 0.0D0) GO TO 380
!C
DO 360 J = 1, L
G = 0.0D0
!C
DO K = 1, L
G = G + Z(I,K) * Z(K,J)
enddo
!C
```

```
      DO K = 1, L
      Z(K,J) = Z(K,J) - G * Z(K,I)
      enddo
      360 CONTINUE
      !C
      380 D(I) = Z(I,I)
      Z(I,I) = 1.0D0
      IF (L .LT. 1) GO TO 500
      !C
      DO 400 J = 1, L
      Z(I,J) = 0.0D0
      Z(J,I) = 0.0D0
      400 CONTINUE
      !C
      500 CONTINUE
      !C
      RETURN
      !C ********** LAST CARD OF TRED2 **********
      END
      !C
      !C
      SUBROUTINE TQLRAT(N,D,E2,IERR)
      IMPLICIT real*8(A-H,O-Z)
      DIMENSION D(N),E2(N)
      REAL*8 MACHEP
      !C ******************************************************************
      !C THIS SUBROUTINE IS A TRANSLATION OF THE ALGOL PROCEDURE
TQLRAT,
      !C ALGORITHM 464, COMM. ACM 16, 689(1973) BY REINSCH.
      !C
      !C THIS SUBROUTINE FINDS THE EIGENVALUES OF A SYMMETRIC
      !C TRIDIAGONAL MATRIX BY THE RATIONAL QL METHOD.
      !C
      !C ON INPUT :
      !C
      !C N IS THE ORDER OF THE MATRIX,
      !C
      !C D CONTAINS THE DIAGONAL ELEMENTS OF THE INPUT MATRIX'
      !C
      !C E2 CONTAINS THE SQUARES OF THE SUBDIAGONAL ELEMENTS OF
THE
      !C INPUT MATRIX IN ITS LAST N-1 POSITIONS. E2(1) IS ARBITRARY.
      !C
```

```
!C ON OUTPUT :
!C
!C D CONTAINS THE EIGENVALUES IN ASCENDING ORDER. IF AN
!C ERROR EXIT IS MADE, THE EIGENVALUES ARE CORRECT AND
!C ORDERED FOR INDICES 1,2,...IERR-1, BUT MAY NOT BE
!C THE SMALLEST EIGENVALUES,
!C
!C IERR IS SET TO
!C ZERO FOR NORMAL RETURN,
!C J IF THE J-TH EIGENVALUE HAS NOT BEEN
!C DETERMINED AFTER 30 ITERATIONS.
!C
!C QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GARBOW,
!C APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORATORY
!C ************************************************************
!C
!C ********** MACHEP IS A MACHINE DEPENDENT PARAMETER SPECIFYING
!C THE RELATIVE PRECISION OF FLOATING POINT ARITHMETIC.
!C
!C
MACHEP = 2.D0**(-26)
!C
IERR = 0
IF (N .EQ. 1) GO TO 1001
!C
DO I = 2, N
E2(I-1) = E2(I)
enddo
!C
F = 0.0D0
B = 0.0D0
C = 0.0D0
E2(N) = 0.0D0
!C
DO 290 L = 1, N
J = 0
H = MACHEP * (DABS(D(L)) + DSQRT(E2(L)))
IF (B .GT. H) GO TO 105
B = H
C = B * B
```

```
!C ********** LOOK FOR SMALL SQUARED SUB-DIAGONAL ELEMENT
**********
105 DO 110 M = L, N
IF (E2(M) .LE. C) GO TO 120
!C ********** E2(N) IS ALWAYS ZERO, SO THERE IS NO EXIT
!C THROUGH THE BOTTOM OF THE LOOP **********
110 CONTINUE
WRITE(6,*)' **** FATAL ERROR IN TQLRAT **** '
WRITE(6,*)' **** FALLEN THROUGH BOTTOM OF LOOP 110 *** '
STOP
!C
120 IF (M .EQ. L) GO TO 210
130 IF (J .EQ. 30) GO TO 1000
J = J + 1
!C ********** FORM SHIFT **********
L1 = L + 1
S = DSQRT(E2(L))
G = D(L)
P = (D(L1) - G) / (2.0D0 * S)
R = DSQRT(P*P+1.0D0)
D(L) = S / (P + DSIGN(R,P))
H = G - D(L)
!C
DO I = L1, N
D(I) = D(I) - H
enddo
!C
F = F + H
!C ********** RATIONAL QL TRANSFORMATION **********
G = D(M)
IF (G .EQ. 0.0D0) G = B
H = G
S = 0.0D0
MML = M - L
!C ********** FOR I=M-1 STEP -1 UNTIL L DO – **********
DO 200 II = 1, MML
I = M - II
P = G * H
R = P + E2(I)
E2(I+1) = S * R
S = E2(I) / R
D(I+1) = H + S * (H + D(I))
G = D(I) - E2(I) / G
```

```
      IF (G .EQ. 0.0D0) G = B
      H = G * P / R
      200 CONTINUE
      !C
      E2(L) = S * G
      D(L) = H
      !C ********** GUARD AGAINST UNDERFLOW IN CONVERGENCE TEST
**********
      IF (H .EQ. 0.0D0) GO TO 210
      IF (DABS(E2(L)) .LE.DABS(C/H)) GO TO 210
      E2(L) = H * E2(L)
      IF (E2(L) .NE. 0.0D0) GO TO 130
      210 P = D(L) + F
      !C ********** ORDER EIGENVALUES **********
      IF (L .EQ. 1) GO TO 250
      !C ********** FOR I=L STEP -1 UNTIL 2 DO -- *********
      DO 230 II = 2, L
      I = L + 2 - II
      IF (P .GE. D(I-1)) GO TO 270
      D(I) = D(I-1)
      230 CONTINUE
      !C
      250 I = 1
      270 D(I) = P
      290 CONTINUE
      !C
      GO TO 1001
      !C ********** SET ERROR -- NO CONVERGENCE TO AN
      !C EIGENVALUE AFTER 30 ITERATIONS **********
      1000 IERR = L
      1001 RETURN
      !C ********** LAST CARD OF TQLRAT **********
      END
      !C
      !C
      SUBROUTINE TQL2(NM,N,D,E,Z,IERR)
      IMPLICIT real*8(A-H,O-Z)
      DIMENSION D(N),E(N),Z(NM,N)
      REAL*8 MACHEP
      !C ************************************************************
      !C THIS SUBROUTINE IS A TRANSLATION OF THE ALGOL PROCEDURE
TQL2,
      !C NUM. MATH. 11, 293-306(1968) BY BOWDLER, MARTIN, REINSCH, AND
```

```
!C WILKINSON.
!C HANDBOOK FOR AUTO. COMP., VOL.II-LINEAR ALGEBRA, 227-240(1971).
!C
!C THIS SUBROUTINE FINDS THE EIGENVALUES AND EIGENVECTORS
!C OF A SYMMETRIC TRIDIAGONAL MATRIX BY THE QL METHOD.
!C THE EIGENVECTORS OF A FULL SYMMETRIC MATRIX CAN ALSO
!C BE FOUND IF TRED2 HAS BEEN USED TO REDUCE THIS
!C FULL MATRIX TO TRIDIAGONAL FORM.
!C
!C ON INPUT :
!C
!C NM MUST BE SET TO THE ROW DIMENSION OF TWO-DIMENSIONAL
!C ARRAY PARAMETERS AS DECLARED IN THE CALLING PROGRAM
!C DIMENSION STATEMENT,
!C
!C N IS THE ORDER OF THE MATRIX,
!C
!C D CONTAINS THE DIAGONAL ELEMENTS OF THE INPUT MATRIX,
!C
!C E CONTAINS THE SUBDIAGONAL ELEMENTS OF THE INPUT MA-
TRIX
!C IN ITS LAST N-1 POSITIONS. E(1) IS ARBITRARY
!C
!C Z CONTAINS THE TRANSFORMATION MATRIX PRODUCED IN THE
!C REDUCTION BY TRED2, IF PERFORMED. IF THE EIGENVECTORS
!C OF THE TRIDIAGONAL MATRIX ARE DESIRED, Z MUST CONTAIN
!C THE IDENTITY MATRIX.
!C
!C ON OUTPUT :
!C
!C D CONTAINS THE EIGENVALUES IN ASCENDING ORDER. IF AN
!C ERROR EXIT IS MADE, THE EIGENVALUES ARE CORRECT BUT
!C UNORDERED FOR INDICES 1,2,...,IERR-1,
!C
!C E HAS BEEN DESTROYED,
!C
!C Z CONTAINS ORTHONORMAL EIGENVECTORS OF THE SYMMETRIC
!C TRIDIAGONAL (OR FULL) MATRIX. IF AN ERROR EXIT IS MADE,
!C Z CONTAINS THE EIGENVECTORS ASSOCIATED WITH THE STORED
!C EIGENVALUES,
!C
!C IERR IS SET TO
!C ZERO FOR NORMAL RETURN,
```

```
!C J IF THE J-TH EIGENVALUE HAS NOT BEEN
!C DETERMINED AFTER 30 ITERATIONS.
!C
!C QUESTIONS AND COMMENTS SHOULD BE DIRECTED TO B. S. GARBOW,
!C APPLIED MATHEMATICS DIVISION, ARGONNE NATIONAL LABORATORY
!C
!C ************************************************************
!C ********** MACHEP IS A MACHINE DEPENDENT PARAMETER SPECIFYING
!C THE RELATIVE PRECISION OF FLOATING POINT ARITHMETIC.
MACHEP = 2.D0**(-26)
!C
IERR = 0
IF (N .EQ. 1) GO TO 1001
!C
DO I = 2, N
E(I-1) = E(I)
enddo
!C
F = 0.0D0
B = 0.0D0
E(N) = 0.0D0
!C
DO 240 L = 1, N
J = 0
H = MACHEP * (DABS(D(L)) + DABS(E(L)))
IF (B .LT. H) B = H
!C ********** LOOK FOR SMALL SUB-DIAGONAL ELEMENT **********
DO 110 M = L, N
IF (DABS(E(M)) .LE. B) GO TO 120
!C ********** E(N) IS ALWAYS ZERO, SO THERE IS NO EXIT
!C THROUGH THE BOTTOM OF THE LOOP **********
110 CONTINUE
120 IF (M .EQ. L) GO TO 220
130 IF (J .EQ. 30) GO TO 1000
J = J + 1
!C ********** FORM SHIFT **********
L1 = L + 1
G = D(L)
P = (D(L1) - G) / (2.0D0 * E(L))
R = DSQRT(P*P+1.0D0)
```

```
D(L) = E(L) / (P + DSIGN(R,P))
H = G - D(L)
DO I = L1, N
D(I) = D(I) - H
enddo
!C
F = F + H
!C ********** QL TRANSFORMATION **********
P = D(M)
C = 1.0D0
S = 0.0D0
MML = M - L
!C ********** FOR I=M-1 STEP -1 UNTIL L DO – **********
DO 200 II = 1, MML
I = M - II
G = C * E(I)
H = C * P
IF (DABS(P) .LT. DABS(E(I))) GO TO 150
C = E(I) / P
R = DSQRT(C*C+1.0D0)
E(I+1) = S * P * R
S = C / R
C = 1.0D0 / R
GO TO 160
150 C = P / E(I)
R = DSQRT(C*C+1.0D0)
E(I+1) = S * E(I) * R
S = 1.0D0 / R
C = C * S
160 P = C * D(I) - S * G
D(I+1) = H + S * (C * G + S * D(I))
!C ********** FORM VECTOR **********
DO 180 K = 1, N
H = Z(K,I+1)
Z(K,I+1) = S * Z(K,I) + C * H
Z(K,I) = C * Z(K,I) - S * H
180 CONTINUE
200 CONTINUE
E(L) = S * P
D(L) = C * P
IF (DABS(E(L)) .GT. B) GO TO 130
220 D(L) = D(L) + F
240 CONTINUE
```

```
!C ********** ORDER EIGENVALUES AND EIGENVECTORS **********
DO 300 II = 2, N
I = II - 1
K = I
P = D(I)
!C
DO 260 J = II, N
IF (D(J) .GE. P) GO TO 260
K = J
P = D(J)
260 CONTINUE
!C
IF (K .EQ. I) GO TO 300
D(K) = D(I)
D(I) = P
DO 280 J = 1, N
P = Z(J,I)
Z(J,I) = Z(J,K)
Z(J,K) = P
280 CONTINUE
300 CONTINUE
GO TO 1001
!C ********** SET ERROR - NO CONVERGENCE TO AN
!C EIGENVALUE AFTER 30 ITERATIONS **********
1000 IERR = L
1001 RETURN
!C ********** LAST CARD OF TQL2 **********
END
SUBROUTINE splint(xa,ya,y2a,n,x,y)
INTEGER n
REAL*8 x,y,xa(n),y2a(n),ya(n)
INTEGER k,khi,klo
REAL a,b,h
klo=1
khi=n
1 if (khi-klo.gt.1) then
k=(khi+klo)/2
if(xa(k).gt.x)then
khi=k
else
klo=k
endif
goto 1
```

```
endif
h=xa(khi)-xa(klo)
if (h.eq.0.d0) then
write(6,*) 'bad xa input in splint'
stop
endif
a=(xa(khi)-x)/h
b=(x-xa(klo))/h
y=a*ya(klo)+b*ya(khi)+((a**3-a)*y2a(klo)+(b**3-b)*y2a(khi))*(h**2)/6.
return
END
SUBROUTINE spline(x,y,n,yp1,ypn,y2)
INTEGER n,NMAX
REAL*8 yp1,ypn,x(n),y(n),y2(n)
PARAMETER (NMAX=500)
INTEGER i,k
REAL*8 p,qn,sig,un,u(NMAX)
if(yp1.gt..99e30) then
y2(1)=0.d0
u(1)=0.d0
else
y2(1)=-0.5d0
u(1)=(3./(x(2)-x(1)))*((y(2)-y(1))/(x(2)-x(1))-yp1)
endif
do i=2,n-1
sig=(x(i)-x(i-1))/(x(i+1)-x(i-1))
p=sig*y2(i-1)+2.d0
y2(i)=(sig-1.d0)/p
u(i)=(6.d0*((y(i+1)-y(i))/(x(i+1)-x(i))-(y(i)-y(i-1))/(x(i)- &
x(i-1)))/(x(i+1)-x(i-1))-sig*u(i-1))/p
enddo
if(ypn.gt..99e30) then
qn=0.d0
un=0.d0
else
qn=0.5d0
un=(3.d0/(x(n)-x(n-1)))*(ypn-(y(n)-y(n-1))/(x(n)-x(n-1)))
endif
y2(n)=(un-qn*u(n-1))/(qn*y2(n-1)+1.d0)
do k=n-1,1,-1
y2(k)=y2(k)*y2(k+1)+u(k)
enddo
return
```

```
END
SUBROUTINE SUB(x,xa,ya,n,V)
INTEGER n
REAL*8 x,xa(n),ya(n),V,a,b,yp1,ypn,y2(n)
!if (x.lt.2.d0) then
! b=(dlog(ya(4))-dlog(ya(5)))/(xa(4)-xa(5))
! a=dexp(0.5d0*dlog(ya(4))+0.5d0*dlog(ya(5))- &
! 0.5d0*b*xa(4)-0.5d0*b*xa(5))
! V=a*dexp(b*x)
!elseif (x.gt.8.d0) then
! b=(dlog(ya(n-5))-dlog(ya(n-4)))/(xa(n-5)-xa(n-4))
! a=dexp(0.5d0*dlog(ya(n-5))+0.5d0*dlog(ya(n-4))- &
! 0.5d0*b*xa(n-5)-0.5d0*b*xa(n-4))
! V=a*dexp(b*x)
!else
yp1=(ya(3)-ya(1))/(xa(3)-xa(1))
ypn=(ya(n)-ya(n-3))/(xa(n)-xa(n-3))
call spline(xa,ya,n,yp1,ypn,y2)
call splint(xa,ya,y2,n,x,V)
!endif
RETURN
END
SUBROUTINE morsepot(x,V)
REAL*8 x,V,De,Re,a
De=60.d0
a=1.d0
Re=3.d0
V=De*(1-dexp(-a*(x-Re)))**2
RETURN
END
SUBROUTINE gauher(x,w,n)
INTEGER n,MAXIT
REAL*8 w(n),x(n),EPS,PIM4
PARAMETER (EPS=3.D-14,PIM4=.7511255444649425D0,MAXIT=10)
INTEGER i,its,j,m
real*8 p1,p2,p3,pp,z,z1
m=(n+1)/2
do 13 i=1,m
if(i.eq.1)then
z=sqrt(float(2*n+1))-1.85575*(2*n+1)**(-.16667)
else if(i.eq.2)then
z=z-1.14*n**.426/z
else if (i.eq.3)then
```

```
z=1.86*z-.86*x(1)
else if (i.eq.4)then
z=1.91*z-.91*x(2)
else
z=2.*z-x(i-2)
endif
do 12 its=1,MAXIT
p1=PIM4
p2=0.d0
do 11 j=1,n
p3=p2
p2=p1
p1=z*sqrt(2.d0/j)*p2-sqrt(dble(j-1)/dble(j))*p3
11 continue
pp=sqrt(2.d0*n)*p2
z1=z
z=z1-p1/pp
if(abs(z-z1).le.EPS)goto 1
12 continue
pause 'too many iterations in gauher'
1 x(i)=z
x(n+1-i)=-z
w(i)=2.d0/(pp*pp)
w(n+1-i)=w(i)
13 continue
return
END
```