Theoretical and Simulation Tools for Electron Transfer and Chain Reactions in Single Walled Carbon Nanotubes

by

Nitish Nair

B.E. (Hons) Chemical Engineering
Birla Institute of Technology and Science (Pilani), 2004
M.S. Chemical and Biomolecular Engineering
University of Illinois at Urbana-Champaign, 2006

SUBMITTED TO THE DEPARTMENT OF CHEMICAL ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY IN CHEMICAL ENGINEERING
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

MAY 2009
[JUNE]

Signature of Author: ____________________________
Department of Chemical Engineering
May 13, 2009

Certified by: ____________________________
Michael S. Strano
Associate Professor of Chemical Engineering
Thesis Supervisor

Accepted by: ____________________________
William M. Deen
Professor of Chemical Engineering
Chairperson, Department Committee for Graduate Students

Theoretical and Simulation Tools for Electron Transfer and Chain Reactions in Single Walled Carbon Nanotubes

by

Nitish Nair

Submitted to the Department of Chemical Engineering
on May 15, 2009, in Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

Abstract

Single walled carbon nanotubes (SWNT) are cylindrical sheets of graphene whose electronic structures and diameters are determined by their chiralities. Current synthetic methods produce batches of nanotubes containing a variety of electronic properties. The separation of these mixtures into semiconductors and metals will greatly enhance their utility for nanoelectronic devices. Progress has been achieved in this area at the laboratory scale through chemical and density-based handles for viable separation. While pristine SWNT of a certain electronic type are in great demand, nanotubes functionalised with specific molecules have applications in the detection of biological analytes, gas sensing and nanoenergetics. An ongoing area of interest is the concept of nanostructure-guided chain reactions, wherein the superior thermal conductivity of a nanotube is used to anisotropically enhance the combustion velocity of the energetic moiety covalently attached to its sidewall. A molecular assembly with this property has the potential to act as a nanoscale thruster and a possible source of pulsed power. This thesis therefore tackles the two research problems stated above: (a) Understand the separation of metallic/semiconducting nanotubes from a modelling perspective to gauge the phenomena from a molecular standpoint; (b) Develop a simple coarse-grained model that describes the characteristics of one-dimensional nanoenergetic materials and extracts the properties of the thermally propagated reaction wave.

The chemical route of metal/semiconductor separation involves the preferential reaction of a diazonium salt (4-hydroxybenzenediazonium tetrafluoroborate) with metallic nanotubes. Previous experimental work has identified electron transfer from the nanotube to the diazonium molecule as the source of this selectivity. We have used this insight to develop a rate model that extracts rate constants from photoabsorption spectra. This necessitated the deconvolution of the UV-vis-nIR absorption spectra of single-walled carbon nanotubes, recorded subsequent to the reactions, into individual contributions – a complicated procedure because nanotube transition energies are closely spaced. An algorithm has been presented in this work to convert spectral data to diazonium surface coverage on representative SWNT, which were then fitted using adsorption/reaction-based rate equations. The selective reaction of a metallic nanotube with diazonium causes an increase in the density of the latter, which can be harnessed to separate it from the unreacted – and hence, less dense – semiconductors via ultracentrifugation. It is already possible to use a centrifugal field to sort surfactant-suspended carbon nanotubes by diameter in a density gradient. Through a hydrodynamic model, we seek to understand the effects of surfactant adsorption, diazonium reaction and cylindrical geometry on the density of a single walled carbon nanotube.

In order to study the propagation of energetic reactions in a nanotube scaffold, we have modelled the latter by a one-dimensional chain of oscillators. The characteristics of thermally initiated chain reactions – e.g., wave velocity and modes of propagation– will be explored with a simple molecular dynamics model in conjunction with a Monte Carlo kernel that simulates the stochastic nature of the system. The contribution of the force field to the properties of the reaction has also been analysed through the use of harmonic and anharmonic interactions between the set of oscillators. This conceptual system helps us in formulating design parameters for the fabrication of actual nanostructures in the laboratory.

Thesis Supervisor: Michael S. Strano
Title: Associate Professor of Chemical Engineering

Acknowledgements

I am indebted to my advisor, Dr. Michael S. Strano, for his guidance, infectious enthusiasm for research and insightful comments at the right junctures, which inevitably simplified the task at hand. Over the course of the doctorate, he always took my innate fear of experiments into account and allowed me to tackle purely computational problems. I would also like to express my gratitude to Dr. Richard D. Braatz for taking an active interest in my work and for detailed discussions on key topics, academic or otherwise. I thank Dr. William H. Green Jr. and Dr. Bernhardt L. Trout for serving on my thesis committee and providing useful comments with respect to my projects and postdoctoral life.

I would have been able to accomplish precious little had it not been for the continual advice, data and friendship provided by my group mates. Special mention must be made of Dr. Monica Usrey and Dr. Woo-Jae Kim for providing high-quality experimental results that formed the bases of multiple papers; Dr. Jong Hyun Choi, Dr. Paul Barone and Esther Jeng for conversations ranging from basketball to religion and politics, not to mention research; Chang Young Lee for carefully proofreading my manuscripts and for being a patient mentor in the early stages of my academic career; Saurabh Tejwani and Dr. Naresh Chennamsetty for explaining the outlines of Monte Carlo simulations and thereby releasing me from a rut. I would be remiss if I did not acknowledge the invaluable comradeship of Dr. Ketan Bhatt, Dr. Ankur Agarwal, Sanketh Shetty, Nitish Korula and Amit Kumar over the last 5 years and beyond.

Academic *moksha* would have forever remained out of reach without the soothing effects of south Indian filter coffee, Scandinavian heavy metal, and the support – moral and financial – of my family. Finally, the doctoral experience deserves a vigorous nod for accepting a raw product from the undergraduate reactor and fashioning him, via suitable downstream processes, into an entity capable of analytical thought.

Table of Contents

List of Figures

List of Tables

1. Introduction

Single walled carbon nanotubes (SWNT) were discovered in 1991 by Iijima and co-workers.[1] They can be conceived of as cylindrically rolled sheets of graphene. The direction of rolling is quantified by a pair of indices, *n* and *m*; the ordered pair, (*n*,*m*), represents a unique nanotube and also determines many of its electronic and structural properties.[2-4] Over the last fifteen years, carbon nanotubes have occupied a preeminent position in the field of nanotechnology. Although a vast number of properties of nanotubes have been studied, they are far from being thoroughly understood. While SWNT have the potential to be used as nanosensors,[5-10] their most popular application lies in the area of nanoelectronics, where they are touted as the successors to silicon technology. The current path of miniaturization will inevitably lead to atomic scale features in processors. It is possible that SWNT will have an enormous role to play when that time comes.

1.1 (*n*,*m*) Characterization of SWNT

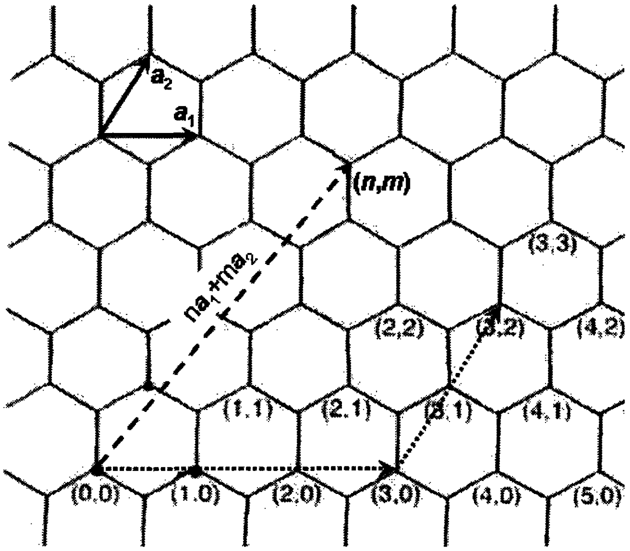


Figure 1.1. Two-dimensional coordinate system on a graphene lattice specified by the unit vectors a_1 and a_2. Also shown is the path to reach the (3,2) node on the sheet, along with the chiral vector that yields a generic (*n*,*m*) nanotube upon rolling the graphene lattice.

Graphite has a three-dimensional lattice composed of sheets of hexagonally bonded carbon atoms. A single such sheet is termed graphene, whose two-dimensional coordinate system defined by the unit vectors $\boldsymbol{a_1}$ and $\boldsymbol{a_2}$ is shown in Figure 1.1. With the origin placed the point (0,0), we can specify a so-called chiral vector (dashed arrow), $C_h=n\boldsymbol{a_1}+m\boldsymbol{a_2}$, along which the sheet may be rolled to give a generic (n,m) single walled carbon nanotube. As a numerical example, consider the location of the (3,2) lattice node and the path taken to reach it (dotted arrow). The appropriate chiral vector $3\boldsymbol{a_1}+2\boldsymbol{a_2}$ is generated by first taking 3 steps along the $\boldsymbol{a_1}$ direction and subsequently 2 steps along $\boldsymbol{a_2}$. Rolling the graphene sheet along this vector will yield the (3,2) nanotube. The (n,m) indices encapsulate structural and electronic information about the corresponding nanotube. The diameter of the tube is given by[3,11]

$$d_{(n,m)} = \frac{\sqrt{3}a_{CC}}{\pi}\sqrt{n^2+mn+m^2} \tag{1.1}$$

while the angle made by C_h with the vector $\boldsymbol{a_1}$ – or the chiral angle, $\theta_{(n,m)}$ – is

$$\theta_{(n,m)} = cos^{-1}\left(\frac{\vec{a}_1\cdot\vec{C}_h}{|\vec{a}_1||\vec{C}_h|}\right) = cos^{-1}\left(\frac{n+m/2}{\sqrt{n^2+mn+m^2}}\right) \tag{1.2}$$

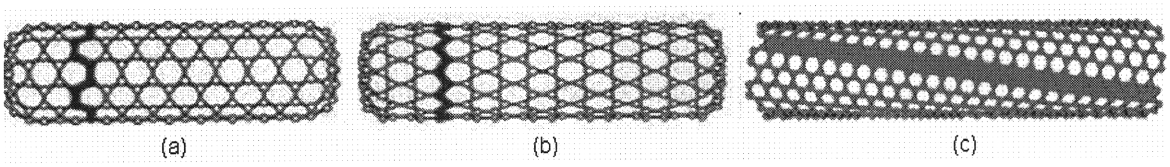


Figure 1.2. Three classes of nanotubes based on the values assumed by the chiral angle: (a) Armchair, (b) Zigzag, (c) Chiral.

Figure 1.2 shows the three classes of nanotubes that exist based on the chiral angle. The possibilities are listed in Table 1.1:

Chiral Angle (rad)	**Type of Nanotube**
$\theta_{(n,m)} = \pi/6$ (Fig. 1.2a)	Armchair
$\theta_{(n,m)} = 0$ (Fig. 1.2b)	Zigzag
$0<\theta_{(n,m)}<\pi/6$ (Fig. 1.2c)	Chiral

Table 1.1. Values of the chiral angle, $\theta_{(n,m)}$, that yield the major types of single walled carbon nanotubes

The (n,m) indices also determine whether a given nanotube is metallic or semiconducting. If $(n-m)$ is divisible by 3, the resulting tube is a metal, while it is a semiconductor in the converse case.[3] A typical SWNT manufacture process produces a variety of chiralities, and it is of utmost importance to achieve a scalable method of separating the metals from the semiconductors for various applications. The literature outlines several separation schemes but we will focus on two in this work that may ultimately be linked in a cascade to provide distinct metallic and semiconducting fractions.

1.2 Motivation for Selective Reaction

All the current methods of SWNT production – high pressure CO decomposition (HiPco),[12] arc discharge,[13] laser ablation,[14] CoMoCAT[15] and chemical vapor deposition (CVD)[16] – yield mixtures of semiconducting and metallic nanotubes. For them to have any significant utility in the electronics industry, it is essential that the mixture be separated into individual semiconductors and metals, since the presence of the latter will render any nanotube-based field effect transistor useless. Significant progress towards this end was made by Strano et al. in 2003,[17] where a diazonium salt was used to react preferentially with metallic nanotubes. The first evidence of separation was shown by

Arnold et al.,[18] wherein a density gradient method was used to separate nanotubes by diameter and by electronic type.

The functionalisation of nanotubes with different moieties tunes their electronic properties and is critical to their performance as nanoscale sensors. Functionalisation of a nanotube is generally caused by noncovalent adsorption of the analyte on the SWNT surface or the formation of a covalent bond between the analyte and the carbon atoms on the SWNT or a combination of both. Noncovalent adsorption is mostly mediated by electron transfer from the nanotube to the analyte or vice versa,[19,20] while covalent functionalisation results in the localisation of the electron near the covalent bond.[21,22] Both processes can be detected through the use of UV-vis-nIR absorption spectroscopy; they result in the depletion of delocalised electrons (or unoccupied valence states) in the nanotube, which directly leads to a decrease in the peaks that comprise the absorption spectrum.

1.2.1 Density of States of Carbon Nanotubes

The density of states (DOS) per unit volume as a function of frequency (ω) for any three-dimensional structure is given by[23]

$$D(\omega)=\frac{1}{8\pi^3}\int\frac{dS_\omega}{\nabla_k\omega(k)} \tag{1.3.1}$$

In the above equation, dS_ω is an area element on the surface in k-space corresponding to the selected frequency, ω. The denominator in the integrand is the group velocity evaluated from the gradient (with respect to the wavevector k) of the electron dispersion relation.

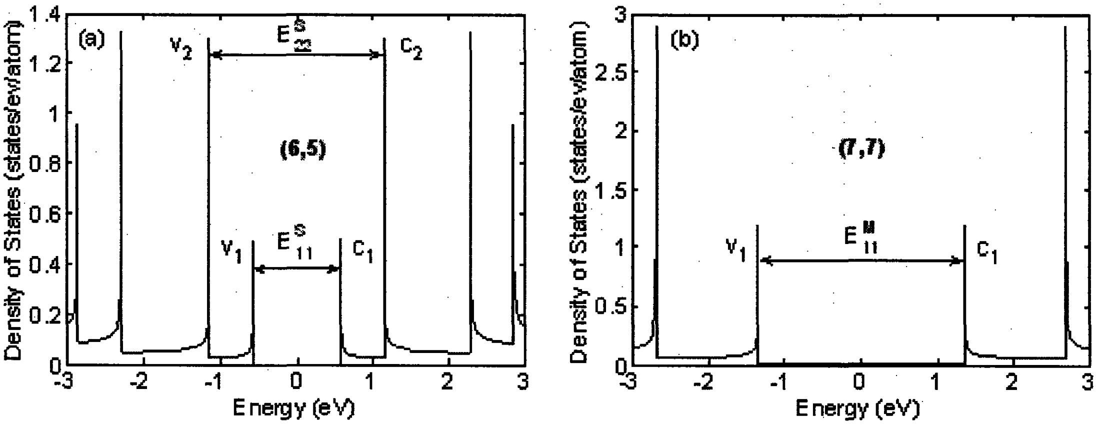


Figure 1.3. Density of states for (a) semiconductor – (6,5), (b) metal – (7,7), and the electronic transitions that give rise to the peaks observed in the absorption spectrum.

For a one-dimensional system such as a carbon nanotube, it is possible to simplify the complex integral in equation 1.3.1 to a simple summation:[24]

$$D(E)=\frac{2}{l}\sum_{i} dk\,\delta\left(k-k_i\right)\left|\frac{\partial\varepsilon}{\partial k}\right|^{-1} \qquad (1.3.2)$$

where E is the energy being probed, l is the length of the 1D Brillouin zone, and k_i is root of the equation E-$\varepsilon(k)$=0. Equation 1.3.2 was used to calculate the DOS of a semiconducting – (6,5) – and a metallic nanotube – (7,7) – as shown in Figure 1.3. The extremely narrow peaks representing large electron densities are termed Van Hove singularities and are a standard feature of one-dimensional systems. The electron population is clearly zero in the energy range -0.57 eV<E<0.57 eV for the (6,5) nanotube. On the contrary, a finite DOS is present for the metallic tube between the first valence band (v_1) and the first conduction band (c_1).

An absorption spectrum characteristic of surfactant-suspended single walled carbon nanotubes is depicted in Figure 1.4. The various peaks seen in the spectrum

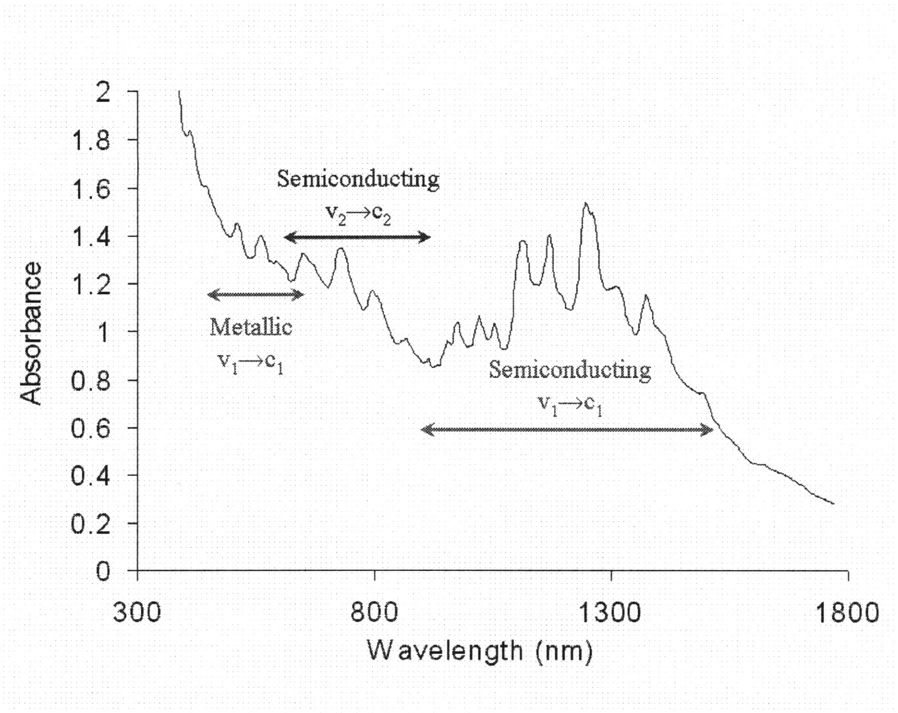


Figure 1.4. Typical photoabsorption spectrum of a surfactant-suspended solution of nanotubes in D_2O showing the electronic transitions (Figure 1.3) that contribute to the peaks observed.

correspond to electronic transitions taking place between the different Van Hove singularities in Figure 1.3. For instance, a transition from v_1 to c_1 in the (6,5) nanotube leads to a peak in the red region in Figure 1.4. More specifically, the spectral peak at the wavelength 974 nm represents the above case.[25] The metallic and higher-order semiconducting transitions (i.e., $v_2 \rightarrow c_2$) can be explained in the same manner.

1.2.2 Quantifying SWNT Absorption Spectra

Raman spectroscopy,[26,27] photoabsorption[28,29] and spectrofluorimetry[25,30] have been used to estimate the composition of nanotube samples. Early optical absorption studies on bundled nanotubes prepared by laser vapourisation and the electric arc method show three broad SWNT-related peaks at approximately 0.75 eV, 1.3 eV and 1.8 eV.[28,29,31] They were assigned as the first (E_{11}^{S}) and second (E_{22}^{S}) inter-band electronic transitions of semiconducting nanotubes, and the first transition for metallic nanotubes

(E_{11}^{M}), respectively (Figures 1.3-4). The aggregation of the nanotubes due to strongly attractive van der Waals forces destroys fine structure in the spectra.[32]

Highly-resolved absorption and photoluminescence (PL) spectra were obtained after SWNT were individually suspended in solution by surfactant adsorption.[30] These samples enabled the spectral assignment of metallic and semiconducting species.[30,33,34] The assignment was further refined when only one of the metallic branches of the transitions was observed optically.[33] Spectrofluorimetry has been used to estimate the relative concentrations of semiconducting nanotubes in a sample.[30,35] Since metallic nanotubes and bundles of nanotubes do not fluoresce, it is not possible to detect them using fluorescence studies. The absorption spectrum typically has a distinct, yet convoluted, metallic region. Additionally, in past work, we have shown that spectrofluorimetric data should be interpreted with caution due to the effects of sonication-induced changes[36,37] and nanotube lengths[36] on the quantum yields. The results could also be skewed by the fact that spectrofluorimetry cannot detect quenched, damaged or defective nanotubes, which do not fluoresce at all.[38]

Previous work focused on using tight-binding theory[39], density functional theory[39] and excitonic calculations with the Bethe-Salpeter equation[40] for plotting the absorption spectra of individual nanotubes. The more complex case of fitting the absorption spectrum of sodium dodecyl sulfate (SDS)-suspended nanotubes has been performed by two methods: (1) expressing the absorption coefficient of each nanotube as a function of its joint density of states (JDOS) after broadening the singularities in the JDOS profile with Gaussians.[41] (2) using Voigt profiles to approximate the spectral contribution of each nanotube, although this method was restricted to the E_{11}^{S} region of the spectrum.[42]

1.2.3 Rate Law and Electron Transfer Mechanism

It has been previously shown that the reaction of SWNT with diazonium follows a two-step process[43]: a (*n*,*m*)-selective adsorption step followed by a reaction that results in a covalent bond. The adsorption step is mediated by electron transfer from the nanotube[19,44] to the diazonium molecule, while the covalent bond results in the localisation of electrons in its vicinity and the formation of an impurity state at the Fermi level.[20,21] Both processes lead to the lowering of peaks in the absorption spectrum. The reactions of carbon nanotubes with 4-hydroxybenzenediazonium salt have been studied at length and analysed for the rate constants, which were then linked to the density of states of individual SWNT.[45]

The adsorption step has been assumed to be rate-limiting. In the semibatch reactor used, the diazonium concentration is never allowed to accumulate. Due to these quasi-steady state conditions, it is difficult to obtain dynamic information about the reaction; however, in a sample containing N nanotubes, we can reliably obtain the rate constants of N-1 nanotubes relative to the N^{th} nanotube, which yields a method of quantifying their reactivities in relation to one another.

The relative rate constants extracted from the data have been compared with those predicted by Marcus theory[46,47] and Gerischer-Marcus theory.[48,49] Marcus theory yields rate constants for the transfer of electrons between donor and acceptor molecules in terms of the free energy of the electron transfer step and the reorganization energy, which is the energy required to alter the atomic configuration of the reacting species so that isoenergetic electron transfer can take place in accordance with the Franck-Condon principle.[46,50] Gerischer-Marcus theory provides rate constants based on the convolution of the densities of states (DOS) of the acceptor and donor species.[48,49] Although this theory has been used for estimating rate constants of reactions at semiconductor[48,49] and SWNT[51,52] electrodes, in this work it has been used to examine trends in reactivities for nanotubes suspended in solution.

1.2.4 Hydrodynamic Model for Ultracentrifuge

The centrifuge-based separation of CoMoCAT SWNT by diameter and electronic structure was recently achieved by suspension with bile salts such as sodium cholate (SC).[44] The extent of adsorption of the surfactant on nanotubes of varying chiralities is not uniform and creates assemblies with different buoyant densities. This facilitates their separation by isopycnic centrifugation, where particles migrate to regions of like density. It is important to understand these phenomena at the molecular level so that further improvements to the process can be made.

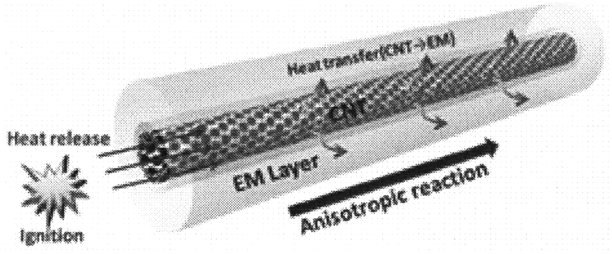
The sedimentation of solute molecules in a solvent during ultracentrifugation is modelled by the Lamm equation.[53] The temporal and spatial distribution of particles in the centrifuge tube is determined by competing diffusive and sedimentation fluxes. Analytical solutions[31] based on certain simplifications are available but involve the calculation of functions that are too complex to fit to experimental data. Numerical solutions have been performed for the sedimentation of molecules in dynamic density gradients and compressible media, among other cases.

The theory of the ultracentrifuge and the numerical analysis of the Lamm equation can describe the motion of surfactant-suspended SWNT in a density gradient, and estimate the number of surfactant molecules adsorbed per unit length of the nanotube. The latter determines the density of the SWNT-surfactant assembly. The sedimentation of the gradient material and the solvent compressibility due to the high speed of rotation will also be considered in our calculations.

The diazonium reaction yields reacted metallic nanotubes that are far denser than the semiconductors. A hydrodynamic model that describes the motion of pristine surfactant-suspended carbon nanotubes[54] in a centrifugal field was used to elucidate the effect of chemical functionalisation with 4-hydroxybenzene diazonium salt on carbon nanotubes.[55]

1.3 Motivation for Nanostructure-Guided Reactions

Phonon confinement in one dimensional nanotube and nanowire systems results in thermal conductivities that often exceed those of conventional materials by orders of magnitude. We show theoretically that a chemical reaction represented as a nonlinear source term in Fourier's law decomposes to a directed reaction wave of amplified velocity confined almost exclusively to an annular region around a nanotube thermal waveguide. The spatially confined wave solution is experimentally realised for carbon nanotubes functionalised with a 7 nm shell of trinitramine (Figure 1.5). Anisotropic reaction velocities along the tube length are ~100 times the unamplified value. The resulting materials are unique in their ability to direct the reaction along only the longitudinal direction specified by the nanotube, creating nanoscale thruster materials that far exceed the intrinsic impulse (>0.3 kN·s/kg) and specific impulse (>5.5 s/μg) of any system demonstrated to date. Control of the rate and direction of energy release in this manner may lead to more efficient chemical energy conversion processes and



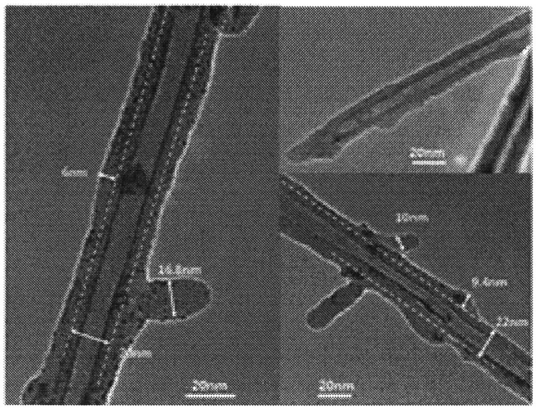


Figure 1.5. Schematic of a SWNT wrapped with an energetic material (EM). To the left is a transmission electron microscopy image of trinitramine-SWNT synthesised by wet impregnation.

devices. To this end, laboratory experiments have also demonstrate that the reaction wave produces a corresponding thermopower wave that can potentially be used to power nanoscale devices.

1.3.1 Experimental and Computational Conceptualization

We seek to harness the one-dimensionality and thermal conductivity of SWNT in modelling and simulating chain reactions of energetic molecules (EM) that have been covalently attached to the nanotube sidewall. A continuum model that describes the use of SWNT as thermal conduits[56] has been framed but here we aim to track the directed thermal transfer process at the molecular scale. There is a wealth of experimental data on the reactions of metallic and semiconducting nanotubes with diazonium salts bearing OH, Cl, and NO_2 functional groups.[17,43,45,55] The same techniques can be used to functionalise SWNT with reactive molecules whose thermal decomposition reactions are highly exothermic. Molecular analogs of TNT and RDX may be covalently bonded to the nanotube surface for this purpose. We have experimentally explored laser-ignited reactions and the concomitant thermopower generation in RDX-coated carbon nanotube arrays. The superior thermal conduction of the nanotube backbone leads to a higher reaction velocity as compared to a pure RDX crystal. This amplified anisotropic velocity could aid in the creation of nanothrusters that exceed the specific impulse of any system demonstrated to date. An electrical wave is generated in the same direction as the thermal wave, which could supply nanoscale electronic devices with extremely high power densities. The current theoretical study is confined to the thermal propagation in a coarse-grained one-dimensional lattice using harmonic and anharmonic force fields.

Force-fields allowing *n*-dimensional ($n \geq 1$) motion of the beads in a chain have been developed for fracture studies,[57,58] elastic deformations and self-assembly phenomena,[59] and simulations of the nanomechanics of single walled carbon nanotubes.[60] The Zhigilei model[60] includes a potential corresponding to breathing modes that describes the internal degrees of freedom of the structure;[61-63] however, their effect, along with that of torsion and the associated coupling terms, were seen to be minimal in the case of isolated nanotubes.[60] In this work, we have used the Buehler model[59] as a simple representation of a carbon nanotube. The bond-stretch component of the total potential energy uses a bilinear model derived from a harmonic potential, which accounts for nonlinearities up to a limit.

1.3.2 Anharmonic Interactions between Oscillators

Anharmonic potentials have been shown to produce solitons[64,65] and breathers[66] in crystal lattices. Breathers, in particular, adversely affect the thermal conduction properties of the crystal due to the localisation of energy.[67,68] The mobility of high-energy fluctuations in a one-dimensional lattice depends on the type of nonlinearity present in the force-field. The fluctuations may result in localised high-frequency oscillations in a lattice with an onsite potential characterized by the hard φ^4 term,[68] which leads to the so-called "diagonal anharmonicity"[69] – the interactions between neighbouring beads is

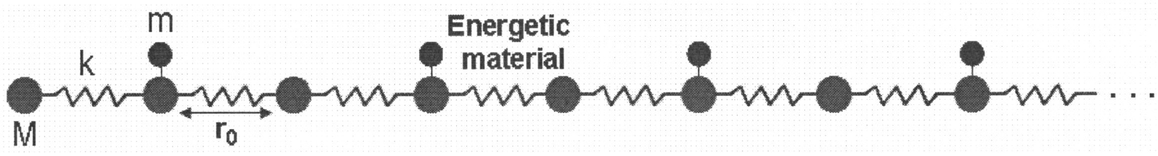


Figure 1.6. Coarse-grained form of a carbon nanotube: red beads of mass M represent successive sections of the tube, each consisting of ~163 C atoms and connected by springs of force constant k. Energetic molecules of mass m (blue) are loaded at specified intervals.

purely harmonic, while the anharmonicities are inherent within each oscillator, as represented by the onsite potential. The mobility of high-frequency oscillations is greatly enhanced through the use of an "off-diagonal" nonlinearity[70] such as that encapsulated in the classic Fermi-Pasta-Ulam β (FPU-β) lattice.[71] In this case, the inter-bead interactions are themselves anharmonic. From a mathematical standpoint, the anharmonicity may be either "diagonal" or "off-diagonal" depending on the positions of nonlinear terms in the force-constant matrix of the system of oscillators.

1.3.3 Computational Approach

Each bead in the lattice is loaded with energetic molecules at a specified density (Figure 1.6). An excitation applied at one end of the chain is expected to raise the local temperature at the reaction sites, leading to the decomposition of the group. The one-dimensionality can be used to channel the energy released during combustion and facilitate subsequent reactions further down the length of the array of oscillators. We seek to identify conditions under which sustained chain reactions will be feasible. The following system parameters are varied: (a) activation energy of the reaction, E_a; (b) fraction of the reaction enthalpy that is converted into the kinetic energy (KE) of the reacted fragments, α. In the simulations, we have allowed α to vary from 0 to 1. An upper bound on α is necessary according to the second law of thermodynamics but this has not been implemented in the current work. Zones of stability of the nanostructure as well as the velocity of the reaction wave have been mapped out. A Fourier analysis of the time-series of the kinetic energy of the lattice and that of each bead was performed to determine the modes that were responsible for the propagation of a disturbance through the lattice. The effect of loading of the EM has also been studied for pristine and defect-laden chains.

1.4 Thesis Overview

The first part of this work deals with the quantitative analysis of the photoabsorption spectrum of carbon nanotubes and its subsequent application in the analysis of SWNT-diazonium reactions. Chapter 2 describes in detail the deconvolution of the SWNT absorption spectrum into spectral contributions from representative nanotubes. Chapter 3 applies the deconvolution procedure to the series of steady state absorption spectra that were recorded during the SWNT-diazonium reaction, and yields relative rate constants for particular nanotubes. These values obtained from fits of the experimental data are then compared with predictions from standard electron transfer theories. A hydrodynamic model for the motion of surfactant-coated nanotubes in a centrifugal field is presented in Chapter 4. It is validated with fits of two independent sets of laboratory data for the diameter-based fractionation of nanotube decants. The model is then extended to the case where nanotubes reacted with diazonium are separated from unreacted species using ultracentrifugation.

The second half of the thesis focuses on the concept of energetic chain reactions guided along a one-dimensional channel, such as a single walled carbon nanotube. Chapter 5 contains the detailed derivation of the model incorporating energy and momentum transfer from the energetic material to the 1D scaffold. The reactivities of anharmonic and harmonic lattices have been compared along with the determination of the modes that contribute to signal propagation in each chain. Lastly, Chapter 6 outlines the conclusions of this research project and proposes future thrusts.

2. Deconvolution of the Absorption Spectra of Carbon Nanotubes

2.1 Introduction

Single walled carbon nanotubes (SWNT) absorb electromagnetic radiation over a broad range of wavelengths – from the ultraviolet (~400 nm) to the near-infrared (~1500 nm). Photoabsorption spectroscopy of solutions of surfactant-suspended nanotubes yield highly resolved peaks that are characteristic of monodisperse moieties. Each spectral peak either corresponds to an electronic transition in a single nanotube or is a convolution of several SWNT whose transitions are closely spaced in the energy domain. Covalent reactions or noncovalent interactions that involve electron transfer to or from a nanotube result in a suppression of certain absorption peaks. A quantitative analysis of the spectrum should result in an estimate of the population of reactant molecules on the nanotube surface. This is a highly complicated procedure due to the sheer number of nanotubes present in solution. The algorithm presented in this chapter seeks to reduce the complexity of the problem and extract enough information so as to create plots of surface coverage on the nanotube sidewall for further analyses.

2.2 Preparation of SWNT Decant

HiPco nanotubes (Rice University, Reactor Run 107) were suspended in 1 wt. % SDS (Sigma Aldrich) in deuterated water (D_2O, Sigma Aldrich) according to a previously published protocol.[30] A deuterated aqueous solution was used to eliminate water absorption features above 1400 nm in the UV-Vis-nIR photoabsorption spectra. The preparation of DNA-wrapped SWNTs has been specified in past work.[72-74]

2.3 Model Development

2.3.1 Background Subtraction

The absorbance at a certain wavelength, $A(\lambda)$, is:

$$A(\lambda) = A_{bkg}(\lambda) + \sum_{(n,m)} A_{(n,m)}(\lambda) \tag{2.1}$$

$A_{bkg}(\lambda)$ is the background due to colloidal graphite and π-plasmon absorption.[75-77] $\sum_{(n,m)} A_{(n,m)}(\lambda)$ is the total absorbance of all the nanotubes at wavelength λ. The following functional form is assumed for the background, with b and k as empirical fit parameters.[78]

$$A_{bkg}(\lambda) = k/\lambda^{b} \tag{2.2}$$

The absorbance after background subtraction is:

$$A_{sub}(\lambda) = A(\lambda) - A_{bkg}(\lambda) \tag{2.3}$$

The spectrum obtained after complete functionalisation of the nanotubes, leaving no resolved absorption peaks, is expected to be solely due to the background. Figure 2.1

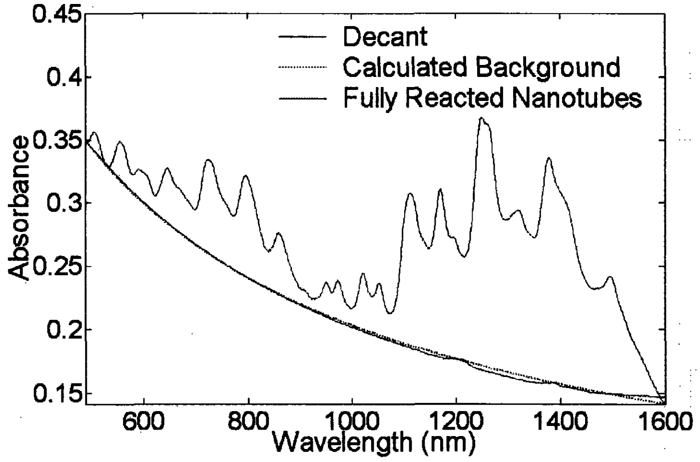


Figure 2.1. Comparison of the calculated background for the absorption spectrum of the given SWNT decant with the spectrum of completely functionalised nanotubes.

shows that the regressed background matches the recovered baseline after functionalisation, thus validating the functional form used.

2.3.2 Spectral Fitting of Optical Transitions

The Franck-Condon principle states that each electronic state has associated vibronic states.[79-81] Environmental[5-8,38] and excitonic effects[82-84] on the nanotube create intermediate electronic states. The major electronic transition is accompanied by minor ones, thereby broadening the absorption lineshape.

The comparison of the fits for 9 absorption spectra with Gaussian, Lorentzian and Voigt profiles (Appendix A, Fig. A1) showed that the latter two gave much lower errors than the Gaussian fits. The Voigt lineshape was chosen over the Lorentzian, since it represents a combination of natural line broadening due to the uncertainty in particle lifetimes, and Doppler-broadening due to a distribution of particle velocities.[81] The spectral contributions of the nanotubes from 490 nm - 1600 nm (2.53 eV – 0.775 eV) are therefore represented as Voigt lineshapes in energy space with peak areas $C_N = \left\{C_{(n,m)}^{ii}\right\}$, peak centers $\left\{E_{ii}^{(n,m)}\right\}$, and full widths at half maxima (FWHM) $\left\{\Gamma_{V,ii}\right\}$, where ii represents the transition, i.e., E_{11}^{M}, E_{22}^{S} or E_{11}^{S}. The transition energies are known from the spectral assignment.[25,33,85] Three regions are identified for HiPco SWNT: 400 nm – 600 nm for E_{11}^{M}, 550 nm – 900 nm and 800 nm – 1600 nm for E_{22}^{S} and E_{11}^{S}, respectively.[30] Unique values of $\Gamma_{V,ii}$ are assumed for the three regimes. The lineshape for a nanotube as a function of energy is:

$$P_{(n,m)}^{ii}\left(E\right) = \int_{-\infty}^{\infty} G_{(n,m)}^{ii}\left(E'\right) L_{(n,m)}^{ii}\left(E - E'\right) dE' \qquad (2.4.1)$$

The Voigt profile is a convolution of Gaussian ($G_{(n,m)}^{ii}(E)$) and Lorentzian ($L_{(n,m)}^{ii}(E)$) lineshapes, which represent Doppler and natural broadening of the spectral line, respectively. $G_{(n,m)}^{ii}(E)$ is centred at zero, while $L_{(n,m)}^{ii}(E)$ is centred at $E_{ii}^{(n,m)}$.

$$G_{(n,m)}^{ii}(E) = \frac{1}{\sqrt{2\pi}\sigma_{ii}} \exp\left[-\left(\frac{E}{\sqrt{2}\sigma_{ii}} \right)^2 \right] \tag{2.4.2}$$

$$\Gamma_{G,ii} = 2\sqrt{2\ln 2}\sigma_{ii} \tag{2.4.3}$$

$$L_{(n,m)}^{ii}(E) = \frac{\Gamma_{L,ii}/2\pi}{\left(E - E_{ii}^{(n,m)}\right)^2 + \Gamma_{L,ii}^2/4} \tag{2.4.4}$$

where $\Gamma_{G,ii}$ and $\Gamma_{L,ii}$ are the FWHMs of the Gaussian and Lorentzian respectively, and σ_{ii} is the standard deviation for the Gaussian. A relation between $\Gamma_{V,ii}$, $\Gamma_{G,ii}$ and $\Gamma_{L,ii}$ is given by[86]:

$$\Gamma_{V,ii} = 0.5346\Gamma_{L,ii} + \sqrt{0.2166\Gamma_{L,ii}^2 + \Gamma_{G,ii}^2} \tag{2.5.1}$$

Assuming $\Gamma_{G,ii} \approx \Gamma_{L,ii}$ to reduce the number of fit parameters, we get:

$$\Gamma_{G,ii} \approx \Gamma_{L,ii} \approx \frac{\Gamma_{V,ii}}{1.6376} \tag{2.5.2}$$

For the E_{11}^{S}, E_{22}^{S} and E_{11}^{M} lineshapes, the pairs $\left(E_{11}^{(n,m)}, \Gamma_{V,11}\right)$, $\left(E_{22}^{(n,m)}, \Gamma_{V,22}\right)$ and $\left(E_{M}^{(n,m)}, \Gamma_{V,M}\right)$, respectively, are substituted in Equations 2.4.1-4 and 2.5.1-2. The lineshape, $P_{(n,m)}^{ii}(E)$, can be expressed as a function of the wavelength, λ, by using the relation $E = \frac{hc}{\lambda}$, where h is Planck's constant and c is the velocity of light. After background subtraction, the overall absorbance at λ is:

$$A_{sub}(\lambda)=\sum_{(n,m)} C^{11}_{(n,m)}P^{11}_{(n,m)}(\lambda)+C^{22}_{(n,m)}P^{22}_{(n,m)}(\lambda)+C^{M}_{(n,m)}P^{M}_{(n,m)}(\lambda) \tag{2.6}$$

2.3.3 Assumptions

The deconvolution procedure is predicated on two major assumptions. A more detailed knowledge of each of the following aspects will enable the code to be more robust and rigorous.

(a) FWHM. The FWHMs ($\Gamma_{V,11}$, $\Gamma_{V,22}$, $\Gamma_{V,M}$) are held constant for all nanotubes in a particular spectral region, although a variation of the linewidths with diameter, band structure or energy is more likely. Due to the lack of information at present on the actual dependence of the FWHMs on any of the above parameters, constant peak widths have been assumed.

(b) Weighting Scheme. Each absorption peak is assumed to correspond to an individual Voigt function, which envelopes a certain number of nanotube transition energies. The set of these peak areas, C_G, contains the fitting parameters. The peak areas of the enveloped nanotubes are approximated as fractions of the peak area of the parent Voigt profile. Given a peak energy, $p_k(E)$, it is assumed that only the nanotubes with transition energies within the flanking valleys, $v_k(E)$ and $v_{k+1}(E)$, contribute to the peak in question. These contributions are assumed to depend on the proximity of their respective transition energies to $p_k(E)$, with the weighting factor of the j^{th} nanotube under the k^{th} parent Voigt profile being:

$$\gamma_{jk}=\exp\left(-\frac{\left|E^{j}_{ii}-p_k(E)\right|}{v_{k+1}(E)-v_k(E)}\right) \tag{2.7}$$

These weights are normalised so that the parent Voigt lineshape is a sum of the individual nanotube Voigt lineshapes:

$$\omega_{jk} = \frac{\gamma_{jk}}{\sum_j \gamma_{jk}} \tag{2.8}$$

Approximate values for the spectral contribution of each (*n,m*) species, C_N, from C_G are computed with the following transformation:

$$C_N = \Omega C_G \tag{2.9}$$

with the matrix Ω containing the normalised weights ω_{jk}. Implicit in the weighting scheme is the assumption that the sample population is continuous and non-zero over the spectral range.

In each group of nanotubes with similar transition energies, the weighting scheme approximates the effect of the absorption coefficient, which determines the (*n,m*) dependence of the probability of absorption of radiation at different energies; however, an explicit (*n,m*) variation of the absorption coefficient has not been included.

2.3.4 Confidence Intervals

The 95% confidence intervals for the parameter estimates were computed with the Statistics Toolbox in MATLAB® in conjunction with a code that used Maximum Likelihood estimation.[87-89] We fit approximately 23 parameters - corresponding to 23 peaks - in a typical SDS-SWNT absorption spectrum. Tightly bound confidence intervals guarantee the uniqueness of the fit, although some compromise has to be made with regard to the quality of the fit. We note that it is possible to obtain a perfect fit by floating the peak areas, peak-centers and linewidths; however, such a fit will not be unique and the corresponding confidence intervals will be large, implying large uncertainties in the parameter estimates. This work is the first to address spectral uniqueness for SWNT deconvolution.

The peak areas are proportional to the oscillator strengths of the individual nanotubes in the sample.[79] 57 nanotubes (39 semiconductors and 18 metals) have been considered for the fit. Spectral deconvolution in this case is difficult because information content on specific nanotubes is distributed non-uniformly. The E_{11}^{S} region is more informative than the E_{22}^{S} or the E_{11}^{M} because it covers a larger energy range and contains more well-defined peaks.

2.4 Algorithm for Deconvolution

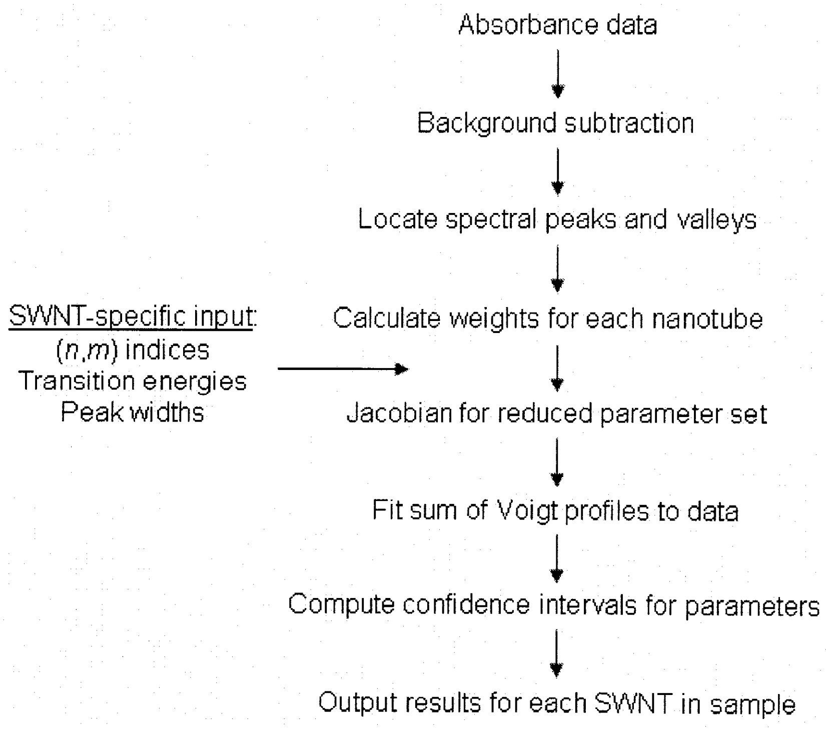


Figure 2.2. Flowchart showing the outline of the program to deconvolute an absorption spectrum of a nanotube solution into its component peak areas.

As outlined in Figure 2.2, the deconvolution algorithm first reads in the spectral data containing the absorbance as a function of wavelength, and subtracts the background due to graphitic impurities (Figure 2.1). The peaks and valleys are located in the subtracted spectrum for the calculation of weights that are assigned to each nanotube as specified in equations 2.7-8. The transition wavelengths and peak widths for each SWNT are provided beforehand, since they are known quantities. From a dataset of ~57 tubes, i.e., 57 fit parameters, it is possible to reduce the parameter space to ~23 with the help of the weighting scheme. The coefficient matrix is then calculated using the transformation in equation 2.9, which is ultimately fed to a nonlinear least-squares optimisation routine to fit the sum of the Voigt profiles to the experimental spectrum. Once the tolerance has reached an acceptable value, the confidence intervals for the 23 fit parameters are computed and are stored along with the computed peak areas for each tube in the starting sample.

2.5 Results of the Fitting Algorithm

The deconvolution method was tested on the absorption spectra of the following: (1) unfunctionalised, selectively functionalised and completely functionalised SDS-suspended HiPco SWNT, and (2) fractionated samples of DNA-SWNT. After obtaining reasonable fits and tight confidence intervals, the algorithm was used to analyse absorption spectra of selective reactions of SWNT with diazonium reagents.[38,90]

2.5.1 Analysis of SDS-SWNT Spectra

Approximate values for $\Gamma_{V,11}$ (29.86 meV), $\Gamma_{V,22}$ (57.96 meV) and $\Gamma_{V,M}$ (93.42 meV) were obtained from fits of the absorption spectra of DNA-HiPco SWNT separated by diameter using ion-exchange chromatography (Appendix A, Fig. A2).[72-74] The weighting scheme relies heavily on the experimentally determined transition wavelengths as obtained for the SDS-SWNT system.[25,33,34,85] In the case of DNA-SWNT absorption spectra, the transition wavelength of each nanotube is red-shifted to a different extent,[73] thus adversely affecting the weighting scheme. The DNA-SWNT spectra we used had fewer nanotubes and were less complex than the SDS-SWNT spectra. Consequently, each nanotube peak could be fitted individually, without the assignment of specific weights. A value of 25 meV for $\Gamma_{V,11}$ provided a better agreement for SDS-SWNT spectra. This could be due to surfactant effects, which might affect the E_{11}^{S} more than the E_{22}^{S} or the E_{11}^{M} due to greater electron screening in the latter two cases.[91] It also agrees with data reported in literature.[30,42,92]

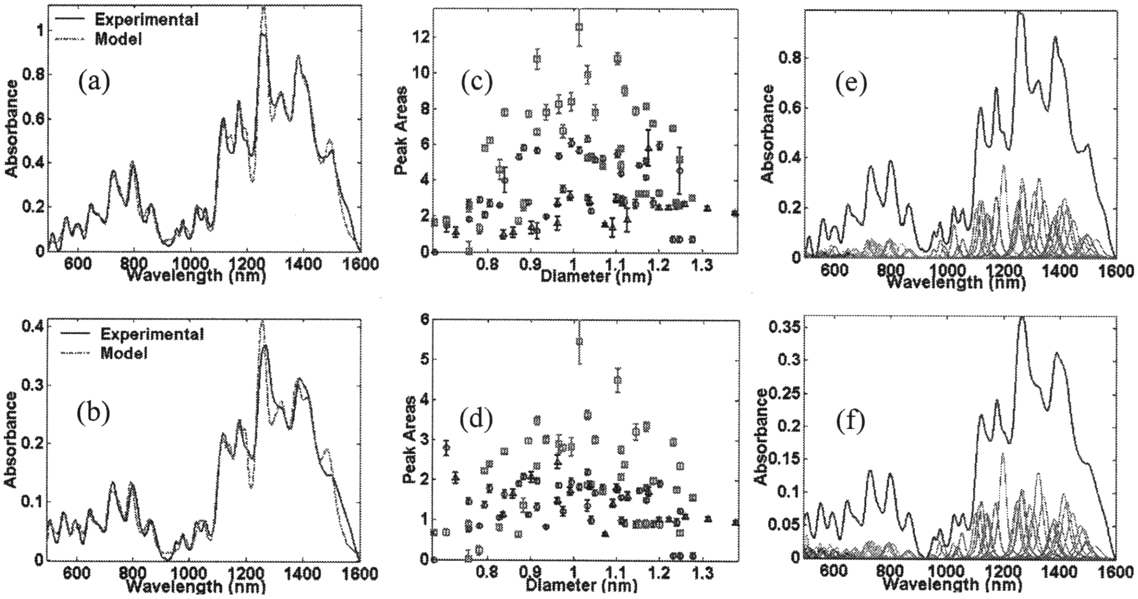


Figure 2.3. Fits of SDS-suspended HiPco nanotubes. (a) and (b) The solid blue line denotes the actual spectrum, and the dotted red line represents the fit. (c) and (d) Confidence intervals corresponding to the 2 spectra, with the red squares denoting the E_{11} peaks, blue circles denoting the E_{22} peaks and the black triangles denoting the metallic peaks. (e) and (f) The deconvoluted peaks for the spectra in Figures 2.3(a)-(b) respectively.

Figures 2.3(a)-(b) show the fits of the absorption data of two different SDS-HiPco SWNT samples. The associated diameter distributions of the calculated peak areas are shown in Figures 2.3(c)-(d) respectively. The red squares denote the E_{11}^{S} peak areas, the blue circles denote the E_{22}^{S} peak areas and the black triangles represent the E_{11}^{M} peak areas. It should be noted that the parameter estimates are tightly bound, as demostrated by the small confidence intervals. Hence, despite the complexity of the spectrum, the obtained parameters are meaningful within the accuracy of the stated assumptions. Figures 2.3(e)-(f) show the deconvoluted E_{11}^{M}, E_{22}^{S} and E_{11}^{S} peaks for the spectra in Figures 2.3(a)-(b) respectively.

Figures 2.4(a)-(b) show three-dimensional plots of the E_{11}^{S} peak areas versus diameter and chiral angle for the deconvoluted spectra. The plots are qualitatively similar to the smoothed surface plot of fluorescence intensity as a function of nanotube diameter and chiral angle, as reported by Bachilo et al[25]; however, Figures 2.4(a)-(b) do not show the reported trend with chiral angle, wherein the quantum yields are greater for near-armchair chiralities and nanotubes from the $(n-m)\bmod 3=-1$ family.[25,84] This could be because photoabsorption is an ensemble measurement of nanotube concentrations, while photoluminescence is able to probe single nanotubes, and hence, extract chirality and diameter dependencies simultaneously. We note that the (n,m) dependence of the quantum yield is not known, and does not affect the absorption spectrum.

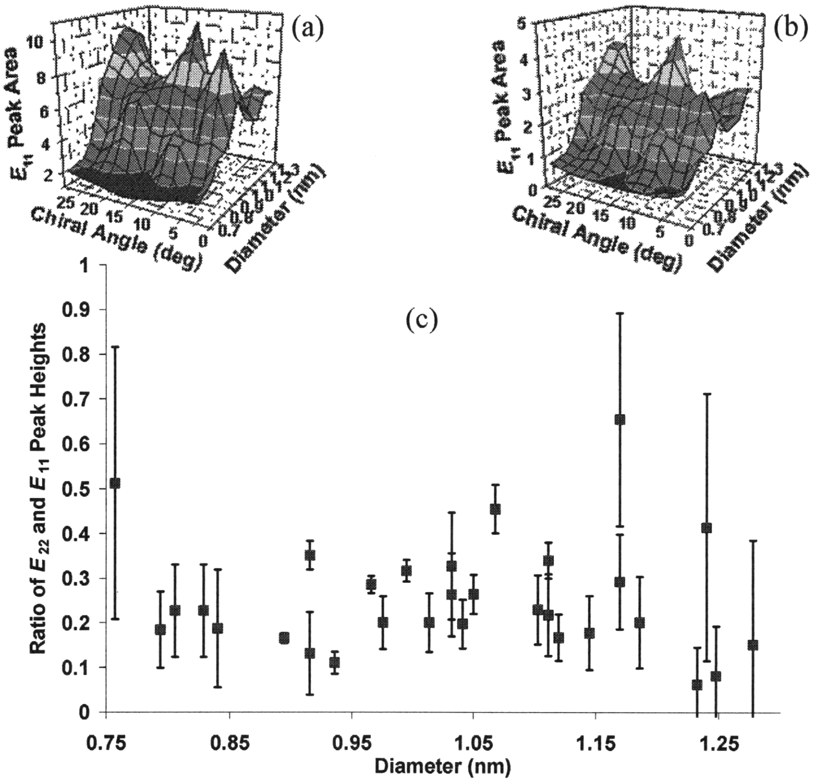


Figure 2.4. (a) 3-D plot of E_{11} Peak Area vs. Diameter and Chiral Angle for the spectrum in Figure 2.3(a). (b) E_{11} Peak Area vs. Diameter and Chiral Angle for the spectrum in Figure 2.3(b). The plots in (a) and (b) were compared with the fluorescence plots in Bachilo et al. (c) Ratios of E_{22} and E_{11} peak heights for 30 semiconducting nanotubes were obtained from fits of 9 spectra.

The E_{22}^S peak height of a particular nanotube is expected to be lower than its corresponding E_{11}^S peak height, simply because the E_{11}^S transition is energetically more probable than the E_{22}^S transition. To investigate this, the ratios of the E_{22}^S and E_{11}^S peak heights for nine SDS-HiPco SWNT spectra were calculated. Out of the 35 semiconductors considered, it was found that 30 of them had ratios which were less than unity. The results have been plotted, with error bars, versus diameter in Figure 2.4(c). With the knowledge of such links between individual E_{22}^S and E_{11}^S peaks, future modifications to the algorithm could involve the generation of the E_{22}^S portion of the absorption spectrum entirely from the E_{11}^{S} region, thus reducing the number of fit parameters. Specifically, knowledge of these individual ratios would permit the relaxation of the diameter continuity constraint.

2.5.2 Analysis of SWNT-Diazonium Reactions

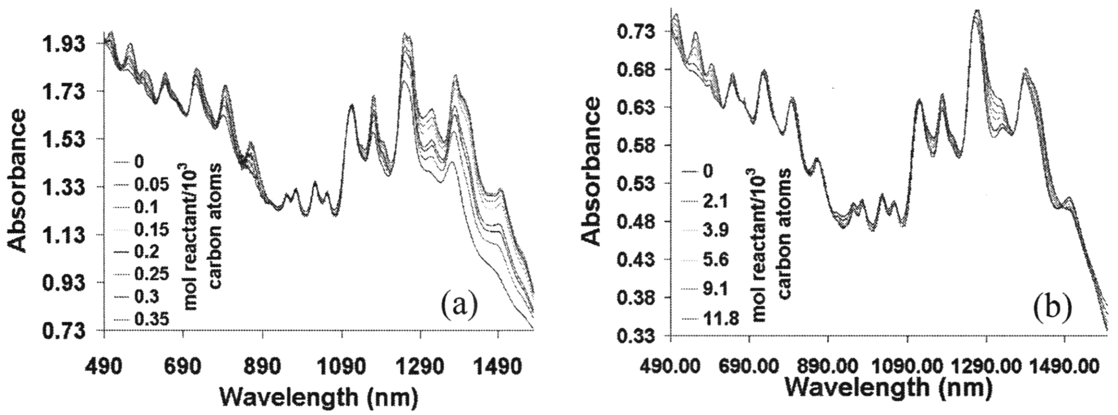


Figure 2.5. Two different sets of reaction spectra that were analyzed with the algorithm. (a) SWNT reaction carried out in a flow-through setup with 4 hydroxybenzene diazonium salt, as described in the experimental section. (b) SWNT reaction setup with 4 chlorobenzene diazonium salt, as described in Strano et. al.

Figure 2.5(a) shows the steady-state photoabsorption spectra for the reaction setup described in the experimental section. Figure 2.5(b) shows data from Strano et al, where the reagent used was 4-chlorobenzene diazonium salt.[93] The algorithm was used to analyse both data sets as a demonstration of its utility.

Under selective reaction conditions, metallic tubes react with the diazonium compound first, followed by large-diameter semiconducting tubes and finally by small-diameter semiconducting tubes.[90,93] This progression is clearly seen in the absorption spectra in Figures 2.5(a)-(b). Nanotubes with similar band gaps will show similar trends in reactivity. According to our algorithm, the spectral contributions of nanotubes that are grouped under a certain absorption peak will decay at the same rate, since they are simply fractions of the total peak area, as derived in equation 2.9. It is difficult to obtain reactivity information on all the metallic nanotubes, since absorption features at wavelengths above 630 nm have considerable contributions from the E_{22}^S peaks of semiconducting nanotubes. The low energy (E_{11}^S) region is easier to analyse, since it is composed purely of E_{11}^S peaks. Hence, as far as estimating rate constants for nanotube reactions is concerned, it is expected that more information will be available for semiconducting nanotubes.

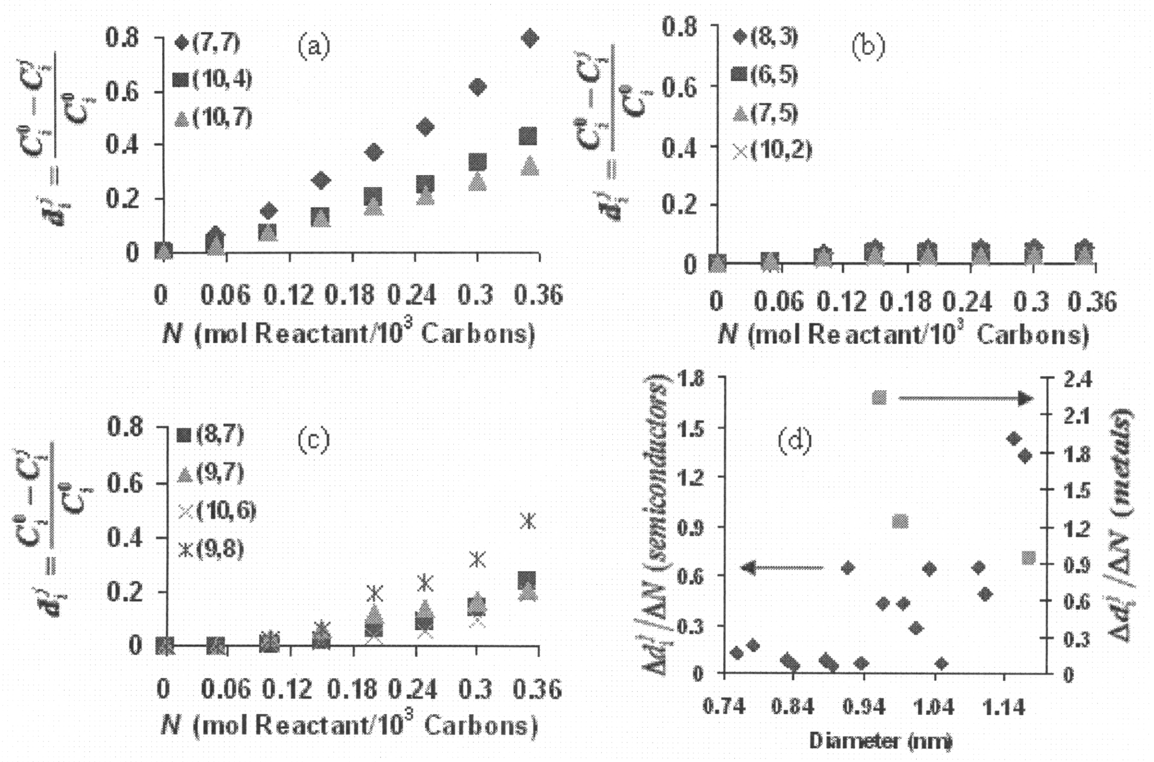


Figure 2.6. Analysis of reaction data shown in Figure 2.5(a), The degree of functionalisation, d_i^j, represents the decrease in peak area of the i^{th} nanotube, normalised to its starting peak area, subsequent to the j^{th} addition of diazonium salt. N denotes the concentration diazonium reagent in the solution. The figures depict the effect of diazonium reaction on: (a) Metallic SWNTs. (b) Small-diameter semiconducting SWNTs. (c) Large-diameter semiconducting SWNTs. (d) Slopes ($\Delta d_i^j/\Delta N$), as measures of reactivity, were calculated from the reaction data for 17 semiconducting nanotubes (blue diamonds) and 3 metallic nanotubes (orange squares). Reactivity increases with diameter for the semiconductors.

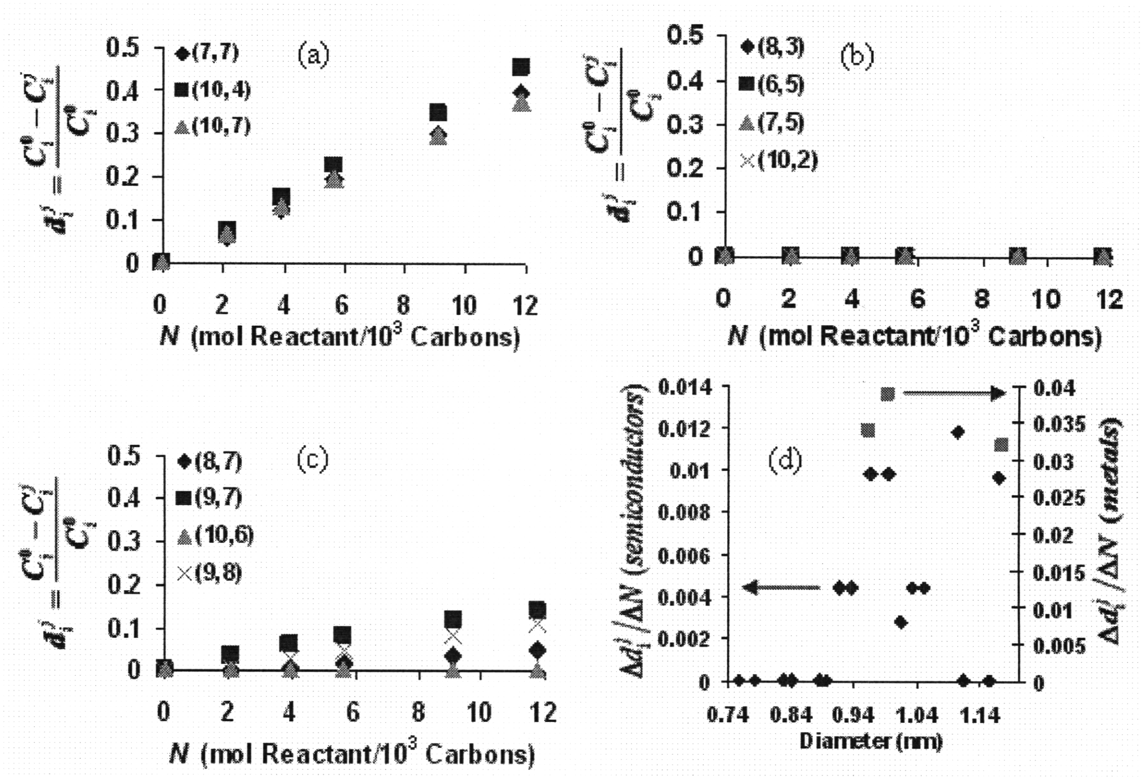


Figure 2.7. Analysis of reaction data shown in Figure 2.5(b), where $d_i^{\ j}$ is the degree of functionalisation; it represents the decrease in peak area of the i^{th} nanotube, normalised to its starting peak area, subsequent to the j^{th} addition of diazonium salt. N denotes the concentration diazonium reagent in the solution. The figures depict the effect of diazonium reaction on: (a) Metallic SWNTs. (b) Small-diameter semiconducting SWNTs. (c) Large-diameter semiconducting SWNTs. (d) Slopes ($\Delta d_i^{\ j}/\Delta N$), as measures of reactivity, were calculated from the reaction data for 17 semiconducting nanotubes (blue diamonds) and 3 metallic nanotubes (orange squares), and plotted against diameter. A distinct trend is observed for the semiconductors, where reactivity increases with diameter.

We have measured the peak areas of 3 metallic tubes, 4 small-diameter and 4 large-diameter semiconductors with increasing diazonium addition for both the data sets considered. Figures 2.6(a)-(d) and 2.7(a)-(d) show the results of the analysis. The degree of functionalisation, d_i^j, for the i^{th} nanotube after the j^{th} addition is:

$$d_i^j = \frac{C_i^0 - C_i^j}{C_i^0} \tag{2.10}$$

C_i^0 and C_i^j are the peak areas at the zeroth and j^{th} addition steps respectively. It can be seen that the degree of functionalisation for metallic nanotubes increases linearly with diazonium addition (Figures 2.6(a) and 2.7(a)), while the small-diameter semiconductors remain almost unaffected (Figures 2.6(b) and 2.7(b)). The large-diameter semiconductors (Figures 2.6(c) and 2.7(c)) are more reactive than the small-diameter semiconductors but not to the same extent as the metals. In past work, we have shown that the reaction follows a two-step mechanism.[38] The first step (rate constant = k_1) is a selective, non-covalent adsorption of the diazonium on the SWNT. It depends on the density of states (DOS) at the Fermi level.[90] The second step is presumably a non-selective reaction in which a covalent bond is formed. A higher DOS gives a larger value of k_1, which yields a stronger linear relationship between d_i^j and diazonium concentration. This explains the greater slopes for the plots of the metallic and large-diameter semiconducting tubes as compared to the small-diameter semiconductors. Slopes for the d_i^j versus diazonium concentration data - $\frac{\Delta d_i^j}{\Delta N}$, where N is the concentration of diazonium salt in solution - were calculated for 17 semiconducting nanotubes and 3 metallic nanotubes, and the results have been shown in Figures 2.6(d) and 2.7(d). The blue diamonds represent semiconducting nanotubes and the orange squares denote metallic nanotubes, and they have been referred to different axes for clarity. Since more data is available for the semiconductors, we can see a definite trend wherein the slopes increase (i.e., reactivities increase) with increasing diameters. We note that our algorithm can be used to extract structure-reactivity relationships for a range of (n,m) nanotubes for various chemistries – an important step in understanding chemical reactivity in carbon nanotube systems. Subsequent work will attempt to establish a more concrete link between the band structure of individual nanotubes and their respective degrees of functionalisation.

3. Structure-Reactivity Relationship for Carbon Nanotubes

3.1 Introduction

Diazonium salts have been observed to react preferentially with metallic carbon nanotubes followed by large- and small-diameter semiconductors respectively. This reactivity trend is counterintuitive from a pyramidalisation perspective, according to which smaller nanotubes should be highly reactive due to the curvature-induced strain experienced by the carbon bonds. It was later discovered that electron transfer from the nanotube to the diazonium molecule was the step that attributed selectivity to the process.[43] Metals, having a finite electron population at the Fermi level, are clearly more reactive from this point of view. Additionally, the semiconducting band gap is inversely proportional to nanotube diameter, which implies that electron transfer has a higher probability of occurring in larger tubes. In this chapter, we have developed a rate model that explains the experimental results, and have also linked the reactivities of different nanotubes to their electronic band structures.

3.2 Experimental Section

3.2.1 Reactant Preparation

HiPco SWNT (Rice University, HPR 107.1) were suspended in D_2O with 1 wt. % sodium dodecyl sulfate (SDS, Sigma-Aldrich). Ultrasonication was followed by ultra-centrifugation to individually suspend the SWNT following a previously published protocol.[30] The final concentration of SWNT in solution was approximately 9.296 wt%. The D_2O was used to eliminate the contribution of water to the UV-vis-nIR absorption spectrum in the 1400 nm - 1770 nm wavelength range. 4-hydroxybenzene diazonium tertrafluoroborate was chosen as the reagent instead of 4-chlorobenzene diazonium tertrafluoroborate, which was used in a previous study,[17] because the hydroxyl group aids the electrophoretic separation of nanotube mixtures. The 4-hydroxybenzene diazonium salt was prepared by the reaction of nitrosonium tetrafluoroborate ($NOBF_4$, Sigma-Aldrich) and 4-aminophenol (HO-C_6H_4-NH_2, Sigma-Aldrich). Both reagents were dissolved in acetonitrile (Sigma-Aldrich) in a nitrogen environment. 4-aminophenol solution was slowly added to the nitrosonium tetrafluoroborate solution at -20°C (dry ice/acetone bath) for the reaction to proceed. The resultant diazonium salt was precipitated with the addition of diethyl ether, filtered and dried under N_2 for 24 hours. The diazonium salt was stored at -20°C and dissolved into D_2O before the reaction.

3.2.2 Functionalisation Procedure

The SWNT-diazonium reaction was performed at pH 5.5 by injecting the diazonium salt solution with a syringe pump (Cole-Parmer) into a semibatch reactor containing the SWNT/SDS suspension (Figure 3.1). The total volume of the diazonium solution (500 μl) was added at an injection rate of 20.83 μl/h into a reactor volume of 5 ml under various diazonium concentrations. The reactor was well-stirred throughout the reaction time of 24 hours. The conversion of functionalised SWNT was controlled by varying the concentration of diazonium salt from 0 to 0.217 mol diazonium/mol carbon, at a reaction temperature of 45°C. The influence of illumination by room light on the reaction rate and selectivity was also investigated at the diazonium concentration where the reaction selectivity for metallic SWNT was maximised. The reacted SWNT solutions

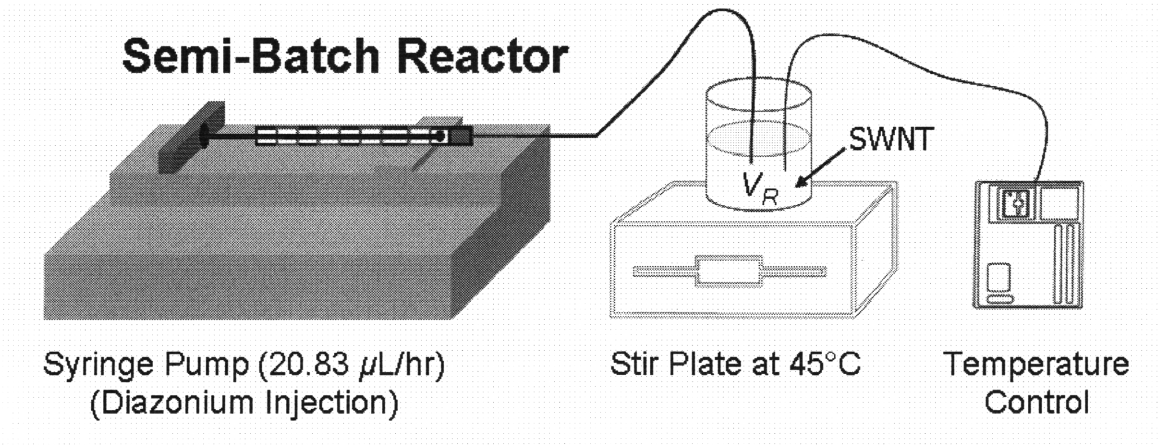


Figure 3.1. Semibatch reactor used for the SWNT-diazonium reaction. The diazonium solution is injected into the reactor (volume = V_R) over a 24-hour period. The reaction mixture is constantly stirred and maintained at 45°C.

were characterized by UV-vis-nIR absorption spectroscopy (Shimadzu UV-310PC) to investigate the extent of reaction and selectivity.

3.3 Model Development

A rate model for the reaction between the diazonium reagent and the nanotube sample was developed using an adsorption/reaction scheme. Fitting the model to steady state absorption data yielded rate constants relative to the (11,5) nanotube, which had the highest rate constant. In order to describe the trend in reactivities with band gap, the rate constants were calculated relative to the (11,5) tube by using Marcus theory and Gerischer-Marcus theory. The use of both formalisms is predicated on the assumption that the adsorption step, which is mediated by charge transfer, is rate limiting.[43]

3.3.1 Formulation of Rate Equations

Consider the adsorption/reaction scheme[43] in Figure 3.2 for each (n,m) nanotube:

$$D+\theta_{(n,m)} \xrightarrow{k_A^{(n,m)}} A\theta_{(n,m)} \xrightarrow{k_R^{(n,m)}} P\theta_{(n,m)}$$

Here, D denotes the diazonium molecule, while $\theta_{(n,m)}$, $A\theta_{(n,m)}$ and $P\theta_{(n,m)}$, respectively, refer to the vacant sites on the nanotube, the sites occupied by the adsorption intermediate and the sites occupied by the reaction product. The adsorption rate constant is $k_A^{(n,m)}$, and the reaction rate constant is $k_R^{(n,m)}$.

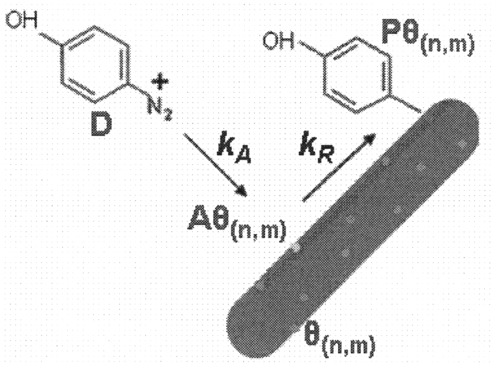


Figure 3.2. Schematic representation of the SWNT-diazonium adsorption-reaction scheme. The first step is a non-covalent electron-transfer mediated adsorption with rate constant k_A, while the second step is a covalent surface reaction with rate constant k_R. Each dot on the tube denotes one of the following types of sites: adsorption (yellow), reaction (green) or vacant (blue)

An overall mass balance on the reactor gives its volume (V_R) as a function of the initial volume (V_0), volumetric flow rate (v_0) and time (t), assuming that the density of the solution remains constant.

$$V_R = V_0 + \nu_0 t \tag{3.1}$$

A site balance carried out on the nanotube surface at any instant of time requires that the total number of sites ($N_{T_{(n,m)}}$) on a particular nanotube be the sum of the number of vacant sites ($N_{\theta_{(n,m)}}$) and those occupied by the adsorbed ($N_{A\theta_{(n,m)}}$) and reacted species ($N_{P\theta_{(n,m)}}$).

$$N_{T_{(n,m)}} = N_{\theta_{(n,m)}} + N_{A\theta_{(n,m)}} + N_{P\theta_{(n,m)}} \tag{3.2}$$

In the above equation, $N_{\theta_{(n,m)}}$, $N_{A\theta_{(n,m)}}$ and $N_{P\theta_{(n,m)}}$ vary with time, while $N_{T_{(n,m)}}$ remains invariant. Once the reagent has entered the reactor, it reacts with the nanotubes to different extents, depending on whether they are metallic or semiconducting. The balance for the number of moles of diazonium (N_D) yields

$$\frac{dN_D}{dt} = F_{D_0} - \sum_{(n,m)} k_A^{(n,m)} \frac{N_D N_{\theta_{(n,m)}}}{V_R} \tag{3.3.1}$$

where F_{D_0} denotes the molar flow rate of diazonium into the reactor. On substituting the value for $N_{\theta_{(n,m)}}$ from the site balance in equation 3.2, we get

$$\frac{dN_D}{dt} = F_{D_0} - \frac{N_D}{V_R} \sum_{(n,m)} k_A^{(n,m)} \left[N_{T_{(n,m)}} - N_{A\theta_{(n,m)}} - N_{P\theta_{(n,m)}} \right] \tag{3.3.2}$$

In order to obtain an expression for $N_{A\theta_{(n,m)}}$, we use the pseudo-steady state approximation, wherein the adsorbed species is consumed as soon as it is generated.

$$\frac{dN_{A\theta_{(n,m)}}}{dt} = \frac{k_A^{(n,m)}}{V_R} N_D N_{\theta_{(n,m)}} - k_R^{(n,m)} N_{A\theta_{(n,m)}} = 0 \tag{3.4.1}$$

Once again, invoking the site balance, we get the following relation:

$$N_{A\theta_{(n,m)}} = \frac{k_A^{(n,m)} N_D \left(N_{T_{(n,m)}} - N_{P\theta_{(n,m)}} \right)}{k_R^{(n,m)} V_R + k_A^{(n,m)} N_D} \tag{3.4.2}$$

The rate of product formation is given by

$$\frac{dN_{P\theta_{(n,m)}}}{dt} = k_R^{(n,m)} N_{A\theta_{(n,m)}} = k_R^{(n,m)} \frac{k_A^{(n,m)} N_D \left(N_{T_{(n,m)}} - N_{P\theta_{(n,m)}} \right)}{k_R^{(n,m)} V_R + k_A^{(n,m)} N_D} \tag{3.5.1}$$

Defining the surface coverage on a (*n*,*m*) nanotube as $\gamma_{(n,m)} = \frac{N_{P\theta_{(n,m)}}}{N_{T_{(n,m)}}}$, equation 3.5.1 can be modified to read as

$$\frac{d\gamma_{(n,m)}}{dt} = \frac{k_R^{(n,m)} k_A^{(n,m)} N_D \left(1-\gamma_{(n,m)}\right)}{k_R^{(n,m)} V_R + k_A^{(n,m)} N_D} \tag{3.5.2}$$

Equation 3.3.2 for diazonium can now be rewritten as

$$\frac{dN_D}{dt} = F_{D_0} - \frac{N_D}{V_R} \sum_{(n,m)} k_A^{(n,m)} N_{T_{(n,m)}} \left(1-\gamma_{(n,m)}\right) \frac{k_R^{(n,m)} V_R}{k_R^{(n,m)} V_R + k_A^{(n,m)} N_D} \tag{3.6}$$

At this point, we use the assumption that the adsorption step is rate-limiting (i.e., $k_R^{(n,m)} >> k_A^{(n,m)}$) to simplify the mole balances.

$$\frac{dN_D}{dt} = F_{D_0} - \frac{N_D}{V_R} \sum_{(n,m)} k_A^{(n,m)} N_{T_{(n,m)}} \left(1-\gamma_{(n,m)}\right) \tag{3.7.1}$$

$$\frac{d\gamma_{(n,m)}}{dt} = \frac{k_A^{(n,m)}}{V_R} N_D \left(1-\gamma_{(n,m)}\right) \tag{3.7.2}$$

Equation 3.7.2 is the general form for all nanotubes that participate in the reaction. In order to reduce the set of unknown parameters, the total number of sites on each nanotube ($N_{T_{(n,m)}}$) is expressed as a fraction ($\alpha_{(n,m)}$) of the total number of sites present in solution (N_T).

$$N_{T_{(n,m)}} = \alpha_{(n,m)} N_T \tag{3.8.1}$$

$$\alpha_{(n,m)} = \frac{A^0_{(n,m)}}{\sum_{(n,m)} A^0_{(n,m)}} \tag{3.8.2}$$

The deconvolution of the absorption spectrum of an unreacted SWNT solution produces peak areas of the spectral profiles corresponding to each nanotube.[94] The peak area of a nanotube in the unreacted decant solution ($A^0_{(n,m)}$) is assumed to be proportional to the total number of available reaction sites on that nanotube, since increasing the surface coverage decreases the peak area.. When normalised to the total area, the ratio is assumed to give the fraction of sites for a particular nanotube.

3.3.2 Time Scale for the Reaction

The Damköhler Number (*Da*) compares the characteristic times for reaction and convection.[95] When evaluated at the initial conditions, *Da* gives an estimate of the time scale for the reaction of diazonium.

$$Da = \frac{-r_{D_0} V_R}{F_{D_0}} \tag{3.9.1}$$

where r_D is the rate of depletion of diazonium. Initially, all the sites on each nanotube are available for adsorption.

$$Da = \frac{\sum_{(n,m)} k_A^{(n,m)} \frac{N_D N_{T_{(n,m)}}}{V_R}}{C_{D_0} v_0} \tag{3.9.2}$$

The numerator in equation 3.9.2 can be simplified via the following assumptions: (a) Initially, $N_D \approx C_{D_0} V_I$, where C_{D_0} is the concentration of diazonium in the syringe, and V_I is the volume of diazonium solution injected into the reactor until that instant. (b) The reactor volume changes negligibly initially, i.e. $V_0 >> V_I$, and hence, $V_R \approx V_0$. (c) Only the nanotube with the greatest adsorption rate constant (k_A^M) contributes significantly to the summation.

Applying these assumptions leads to the following expression for *Da*.

$$Da = \frac{V_I k_A^M N_{T_M}}{V_0 v_0} = \frac{V_I / v_0}{V_0 / k_A^M N_{T_M}} = \frac{\tau_F}{\tau_R} \tag{3.9.3}$$

In equation 3.9.3, N_{T_M} is the total number of sites on the nanotube with the greatest rate constant, τ_F is the amount of time spent by the reagent in the reactor, and τ_R provides a time scale for the diazonium reaction. Since τ_R was calculated by considering only the most reactive nanotube, it is an estimate of the instant at which the diazonium begins to react. Using equation 3.8.2 for the nanotube with the maximum rate, the final form of τ_R can be expressed as

$$\tau_R = \frac{V_0}{k_A^M N_{T_M}} = \frac{V_0}{k_A^M \alpha_M N_T} \tag{3.9.4}$$

3.3.3 Nondimensional Analysis of Rate Equations

Equation 3.7.1 for the diazonium mole balance can be nondimensionalised by introducing a dimensionless time ($t^* = \frac{t}{\tau_R}$) and a dimensionless molar amount of diazonium ($\phi_D = \frac{N_D}{N_T}$). The reason for the latter normalisation is that the total number of sites that the diazonium reagent can possibly occupy is N_T. The resulting nondimensional differential equation is

$$\frac{d\phi_D}{dt^*} = \frac{F_{D_0} \tau_R}{N_T} - \phi_D \frac{V_0}{V_R} \sum_{(n,m)} \frac{k_{(n,m)}}{k_M} \frac{\alpha_{(n,m)} \left(1 - \gamma_{(n,m)}\right)}{\alpha_M} \tag{3.10.1}$$

where $k_{(n,m)} = k_A^{(n,m)} N_T$ and $k_M = k_A^M N_T$. Equation 3.7.2 for the coverage on a (*n*,*m*) nanotube can be also nondimensionalised in the same manner.

$$\frac{d\gamma_{(n,m)}}{dt^*} = \phi_D \frac{k_{(n,m)}}{k_M} \frac{V_0}{V_R} \frac{\left(1-\gamma_{(n,m)}\right)}{\alpha_M} \tag{3.10.2}$$

The fit parameters are $k_{(n,m)}$ and N_T. The nondimensional analysis naturally gives the reactivity of each nanotube relative to that of the most reactive nanotube, which is expected to be a metal. It is these ratios that have been reported in this work.

3.3.4 Fermi Level Calculation

The position of the Fermi level (E_F) relative to vacuum gives the work function (W). If vacuum is taken to be zero, we have

$$E_F = -W \tag{3.11}$$

In the case of SWNT, Okazaki et al. reported an inverse dependence of the work function on diameter.[96] Suzuki et al. ruled out significant diameter- and chirality-based differences in the work functions of metallic and semiconducting nanotubes.[97] More recent work has shown that nanotubes with diameters greater than 0.9 nm have work functions that asymptotically converge to the graphene limit (4.6 eV), while those with diameters less than 0.9 nm have diameter- and chirality-dependent work functions.[98,99] Since the diameter range of the nanotubes that we have considered for the reaction analysis extend from 0.757 nm – 1.375 nm, with most of the diameters exceeding 0.9 nm, we have assumed the absence of chirality and diameter effects on the work function. From the tabulated work functions of several nanotubes,[99] we have chosen W_{SWNT}= 4.45 eV as a representative value, thus giving E_F^{SWNT} =-4.45 eV.

The half-wave potentials ($E_{1/2}$) of various diazonium salts have been estimated using polarography.[100] We have assigned a value of 0.35 eV (SCE) to the redox potential of 4-hydroxybenzene diazonium salt. This assumes that it is at least as reactive with SWNT as 4-chlorobenzenediazonium salt, for which $E_{1/2}$ is available. The redox potential, referred to vacuum, is equivalent to the Fermi level of the redox species in solution.[101,102] The conversion from one scale to another can be performed as follows[49]:

$$E_F^D = -4.7 - V_{SCE} \tag{3.12}$$

which yields E_F^D=-5.05 eV.

E_F^{SWNT} can be fixed at the center of the gap,[98,99] or more generally, at the zero energy in the DOS of nanotubes. The redox level of diazonium relative to the zero of the nanotube DOS becomes -0.60 eV.

3.3.5 Marcus Theory

The rate constant for an electron transfer reaction (k_{ET}) is a product of an attempt frequency (ν_n), a tunneling factor (κ_{el}) and a nuclear factor(κ_n).[46,47,103]

$$k_{ET} = \nu_n \kappa_{el} \kappa_n \tag{3.13.1}$$

Tunneling can be neglected ($\kappa_{el} \approx 1$) if the diazonium molecule is physically close to the nanotube during electron donation, i.e., the reaction is assumed to be adiabatic. The nuclear factor is quantified by Marcus theory in terms of the reorganization energy (λ) and the free energy of the reaction (ΔG^0),[46,47] giving the following rate constant for a (*n,m*) nanotube:

$$k_{ET}^{(n,m)} = \nu_n \, exp\left(-\frac{\left(\lambda + \Delta G^0_{(n,m)}\right)^2}{4\lambda kT} \right) \tag{3.13.2}$$

The electrochemical driving force, $\Delta G^0_{(n,m)}$, is linked to the band gap as follows:[104,105]

$$\Delta G^{0}_{(n,m)} = -\left(\left[E_F^{SWNT} - \frac{E_g^{(n,m)}}{2}\right] - E_F^{D}\right) \qquad (3.13.3)$$

where E_F^{SWNT} is the nanotube Fermi level and $E_g^{(n,m)}$ is the band gap, which is zero for metallic tubes. The relative rate constant for a *(n,m)* nanotube can be obtained by normalising $k_{ET}^{(n,m)}$ with respect to the maximum rate constant.

3.3.6 Gerischer-Marcus Theory

Gerischer-Marcus theory has been used to describe the kinetics of electrochemical reactions at SWNT electrodes.[51,52] We extend this formalism to electron transfer reactions in solution by considering each suspended nanotube as an electrode. The rate constant for a certain nanotube depends on the convolution of its DOS ($D_{(n,m)}(E)$) and the distribution of unoccupied redox states in solution ($W_{ox}^{(n,m)}(E)$)[48,49] as shown in Figure 3.3.

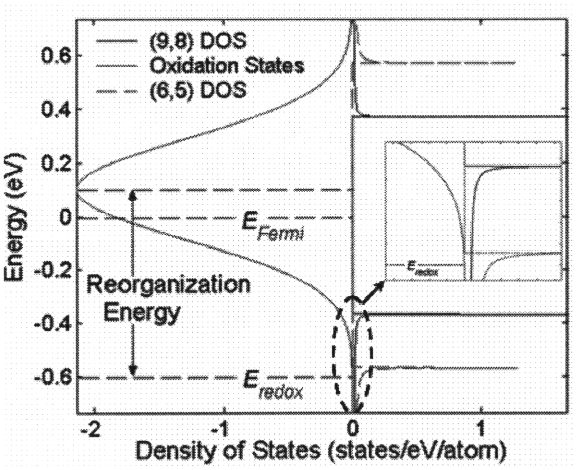


Figure 3.3. Comparison of the density of unoccupied oxidation states of the diazonium molecule (red) with the density of states (DOS) of the (6,5) – green – and (9,8) – blue – nanotubes. The extent of overlap between the DOS of the electron-withdrawing reagent and the nanotube (inset) determines the rate constant of the reaction.

$$k_{ET}^{(n,m)} = \nu_n \int_{E_F^D}^{E_F^{SWNT}} \varepsilon_{ox}(E) D_{(n,m)}(E) W_{ox}^{(n,m)}(E) dE \tag{3.14.1}$$

$$W_{ox}^{(n,m)}(E) = \frac{1}{\sqrt{4\pi\lambda kT}} exp\left(-\frac{\left(\lambda + \Delta G_{(n,m)}^{0}\right)^2}{4\lambda kT} \right) \tag{3.14.2}$$

The band structures of different carbon nanotubes were computed from tight-binding theory with the third nearest neighbour approximation.[4,106] The densities of states were calculated by the general expression provided by White and Mintmire for a 1D system.[24] The tunneling term, κ_{el}, is sometimes extracted from the proportionality function, $\varepsilon_{ox}(E)$, and included in the integral prefactor.[49] Assuming that ε_{ox} is independent of energy, and that ν_n and ε_{ox} are not nanotube-specific, it follows that they cancel out when we calculate the relative rate constants.

3.3.7 Selectivity of Reagent towards Metals

The preference of diazonium towards metallic or semiconducting nanotubes can be measured by defining the reaction selectivity as the ratio of the total surface coverage for metallic SWNT (Γ_{met}) to that of the semiconductors (Γ_{sc}). We have compared the selectivity predicted by the rate model (S_p) to that obtained from the experimental data (S_e).

$$S_p = \frac{\Gamma_{met}}{\Gamma_{sc}} \tag{3.15.1}$$

Let M denote the set of all values of (n,m) corresponding to metallic SWNT, i.e., $(n\text{-}m)$ is divisible by 3. The following expressions can be derived for Γ_{met} and Γ_{sc} by using the definition of surface coverage, $\gamma_{(n,m)} = \dfrac{N_{P\theta_{(n,m)}}}{N_{T_{(n,m)}}}$, and equation 3.8.1, which relates the total number of sites on each nanotube ($N_{T_{(n,m)}}$) to the total number of sites in solution (N_T).

$$\Gamma_{met} = \frac{\sum\limits_{(n,m)\in M} N_{P\theta_{(n,m)}}}{\sum\limits_{(n,m)\in M} N_{T_{(n,m)}}} = \frac{\sum\limits_{(n,m)\in M} \gamma_{(n,m)}\alpha_{(n,m)}}{\sum\limits_{(n,m)\in M} \alpha_{(n,m)}} \tag{3.15.2}$$

$$\Gamma_{sc} = \frac{\sum\limits_{(n,m)\notin M} N_{P\theta_{(n,m)}}}{\sum\limits_{(n,m)\notin M} N_{T_{(n,m)}}} = \frac{\sum\limits_{(n,m)\notin M} \gamma_{(n,m)}\alpha_{(n,m)}}{\sum\limits_{(n,m)\notin M} \alpha_{(n,m)}} \tag{3.15.3}$$

The experimental reaction selectivity (S_e) is obtained by taking the ratio of the overall degrees of functionalisation for metallic (δ_{met}) and semiconducting SWNT (δ_{sc}).

$$S_e = \frac{\delta_{met}}{\delta_{sc}} \tag{3.16.1}$$

All metallic SWNT are grouped into one unit, and all the semiconductors into a second. For each group, the degree of functionalisation is defined as

$$\delta_{met,sc} = \frac{A_{initial}^{met,sc} - A_{unreacted}^{met,sc}}{A_{initial}^{met,sc}} \tag{3.16.2}$$

where $A_{initial}^{met,sc}$ is the area under the metallic or semiconducting region of the absorption spectrum of the starting SWNT solution, and $A_{unreacted}^{met,sc}$ is the area under the corresponding region after reaction.

3.4 Results and Discussion

Reactions of SWNT with 4-hydroxybenzene diazonium salt were performed at various diazonium concentrations and temperatures, with and without illumination by visible light. The UV-vis-nIR absorption spectra were collected for each sample after the reaction. The spectral contributions of carbonaceous species and unreacted diazonium salt were eliminated by background subtraction.

3.4.1 Functionalisation of Nanotubes

Figures 3.4(a)-(b) show UV-vis-nIR absorption spectra of deuterated SWNT solutions reacted with increasing concentrations of 4-hydroxybenzene diazonium at 45°C, with and without illumination. The reagent concentrations have been expressed as moles of diazonium per moles of carbon (D/C). The absorption features represent Van Hove transitions of each *(n,m)* SWNT at different wavelengths. The first Van Hove transitions of metallic species (E_{11}^{M}) appear between 440 nm and 645 nm, while the first (E_{11}^{S}) and second (E_{22}^{S}) Van Hove transitions of semiconducting species appear between 830 nm – 1600 nm and 600 nm – 800 nm, respectively. When the SWNT are covalently functionalised by diazonium salts, their absorption peaks diminish because electrons are localised by the formation of a covalent bond.[21,22,43] This enables the monitoring of the extent of adsorption/reaction for each *(n,m)* nanotube.

Figure 3.4(a) shows that there are no significant changes in the absorption features of all individual SWNT until 0.019 D/C diazonium solution is added. When the concentration of added diazonium solution exceeds 0.027 D/C, notable changes in the absorption features are observed. The E_{11}^{M} peak intensities start to decrease, while those representing the E_{11}^{S} and E_{22}^{S} transitions of the semiconducting species show little change. This indicates that metallic SWNT react with the diazonium salt before the semiconducting SWNT. This trend becomes more prominent at the diazonium

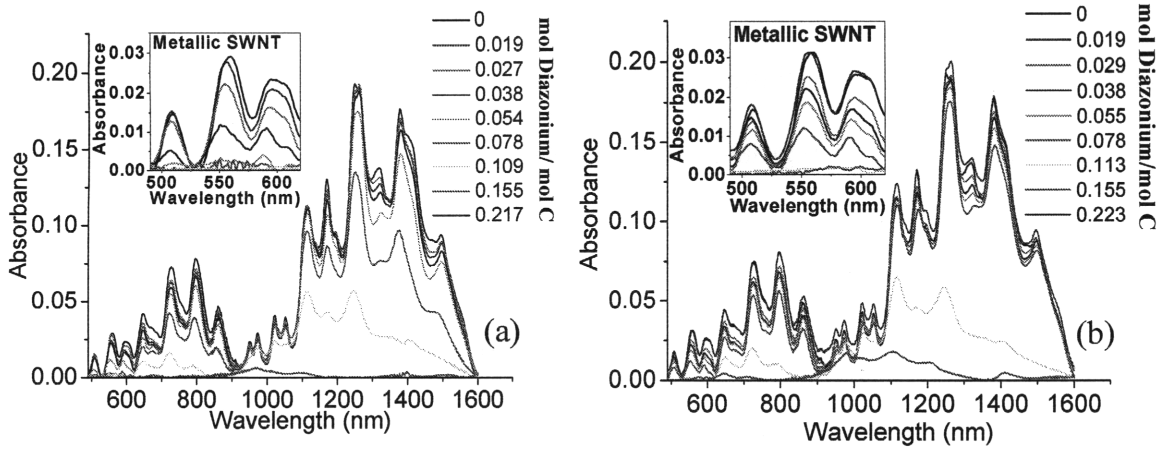


Figure 3.4. Photoabsorption spectra of SWNT reacted with 4-hydroxybenzene diazonium salt (a) under illumination, and (b) in the dark. Diazonium concentrations, normalised to the number of carbons, are listed in the figures. For each of the listed concentrations, 500 μl of 4-hydroxybenzene diazonium salt dissolved in D_2O was injected into 5 ml of SWNT solution over the course of 24 hours, at 45°C and a pH of 5.5. The insets in (a) and (b) show the decay of the metallic peaks with increasing diazonium content.

concentration of 0.054 D/C, where half the metallic SWNT have reacted, while most of the semiconducting SWNT still remain unreacted. At 0.078 D/C, most of the metallic SWNT have been functionalised, and semiconducting SWNT with large diameters (0.93 nm ~ 1.25 nm) begin to react. At high concentrations of diazonium solution (>0.155 D/C) all nanotubes react, regardless of electronic structure, resulting in the complete decay of all absorption features.

The reactivities of SWNT in the absence of illumination show a different trend, as shown in the SWNT absorption spectra in Figure 3.4(b). At a concentration of 0.078 D/C, significant amounts of metallic SWNT still remain unreacted, in contrast to the illuminated reaction. The effect of illumination on the extents of reaction of SWNT may originate from the different reactivities of the diazonium intermediate radical, whose form changes upon illumination.[107] The diazonium is known to form two types of radicals: the diazenyl radical ($\mathrm{Ar}-\mathrm{N}=\mathrm{N}\cdot$), and the aryl radical ($\mathrm{Ar}\cdot$), where Ar denotes an aromatic group. The latter is known to be the major intermediate when the diazonium is exposed to illumination. Based on the reaction results in Figures 3.4(a)-(b), the presence of excess aryl radicals, in addition to diazenyl radicals, leads to greater extents of reaction in the presence of light.

The above results confirm that electronic structure selective chemistry on nanotubes can be performed with 4-hydroxybenezene diazonium. The preference of diazonium towards metallic or semiconducting SWNT can be gauged through the selectivity parameter, which we have defined as the ratio of the total extent of reaction of metallic SWNT to that of the semiconductors. The effects of diazonium concentration and illumination on the selectivity are explored in the following section.

3.4.2 Approximation during Deconvolution

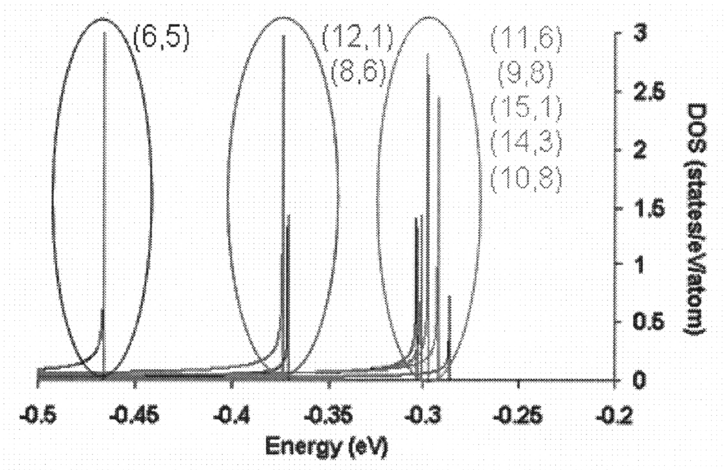


Figure 3.5. Density of states (valence band only) of certain semiconducting single walled carbon nanotubes. As tube diameter increases (left to right), a grouping of nanotubes occurs about a common band gap energy. Species in the same group (as defined by the ovals) are assigned the same rate constant assuming that they have similar reactivities with respect to electron transfer reactions.

Spectral lineshapes corresponding to at least 56 nanotubes (18 metals + 38 semiconductors) constitute the absorption spectrum of SDS-suspended HiPco SWNT. Strictly speaking, in equations 3.10.1-2, 56 values of $k_{(n,m)}$ should be used as fit parameters; however, each absorption peak is composed of nanotubes with similar transition energies. We used a simple weighting scheme to approximate the spectral contribution of each (*n,m*) species to the parent peak. The upshot in reaction analysis is that the absorption peaks of all nanotube comprising the parent peak decrease at the same rate with increasing diazonium content. In other words, nanotubes with comparable transition energies are observed to have similar reactivities. This concept is further bolstered by Figure 3.5, which shows the density of states for specific semiconducting nanotubes. A grouping of tubes is seen to occur around certain band gap energies. The size of the group is inversely proportional to nanotube diameter, which increases from left to right in Figure 3.5. SWNT in a particular group are assigned the same rate constant, since we assume that they behave similarly during the electron transfer step.

Across the wavelength range considered (490 nm – 1600 nm), we have examined the effect of diazonium on 17 representative nanotubes (4 metals + 13 semiconductors) whose transition energies lay closest to the corresponding absorption peak. The depletion of these peaks is due to the presence of adsorbed and covalently bonded diazonium molecules on each nanotube.[43] Therefore, the surface coverage introduced in equations 3.5.1-2 ($\gamma_{(n,m)}$) can also be interpreted as

$$\gamma_{(n,m)} = \frac{A^{0}_{(n,m)} - A^{N}_{(n,m)}}{A^{0}_{(n,m)}} \tag{3.17}$$

where $A^{0}_{(n,m)}$ is the peak area of a (*n,m*) nanotube in the unreacted decant, and $A^{N}_{(n,m)}$ is the depleted peak area of the same nanotube at the completion of the N^{th} addition of diazonium.

3.4.3 Fits of Absorption Data

The (*n*,*m*)-specific fits in Figures 3.6 and 3.7 show fits for the following in the presence and absence of illumination: metals ((7,7) and (11,5)), small-diameter semiconductors ((6,5) and (10,2)) and large-diameter semiconductors ((9,8) and (10,3)). The fits for the less reactive semiconducting nanotubes in Figures 3.6c-d and 3.7c-d show a cusp at large diazonium concentrations. This is because when higher amounts of diazonium are present in solution, the more reactive nanotubes quickly react to completion leaving a considerable fraction of the reagent to react with the less reactive semiconductors. This sudden glut in diazonium molecules competes for the limited semiconducting sites still available and causes a steep rise in the calculated and observed surface coverage. Once the latter reaches the maximum possible value – 1 – it levels off and remains fixed at larger quantities of diazonium used. The poor fits for the (9,8) nanotube in Figures 3.6f and 3.7f indicate that the model faces difficulties in describing the adsorption phenomena on large-diameter semiconductors whose reactivities lie in between those of the metals and small-diameter semiconductors. A nonlinear dependence of the adsorption rate on the diazonium concentration (i.e., exponent ≥ 2) was seen to capture the S-shape of the isotherm. The total number of sites in solution (N_T) was computed as 0.00574 mmol and 0.00684 mmol for the light and dark reactions, respectively. The approximate concentration of SWNT in the starting solution was 9.296 wt%. In the 5 ml reactor volume, this translates to 0.0347 mmol of carbon atoms. The total number of sites computed is at least an order of magnitude lower than the number of carbons present in solution, which means that not all the carbon atoms are reactive sites. Once it was known that the (11,5) nanotube had the highest rate constant ($k_{(11,5)}$), the fitting procedure was repeated by keeping $k_{(11,5)}$ invariant and determining the rate constants of the other nanotubes relative to it. The gradient in rate constants between the various reacting species causes the metal → large-diameter semiconductor → small-diameter semiconductor progression. The precise metering of the diazonium reagent into the reactor ensures that this progression is maintained.

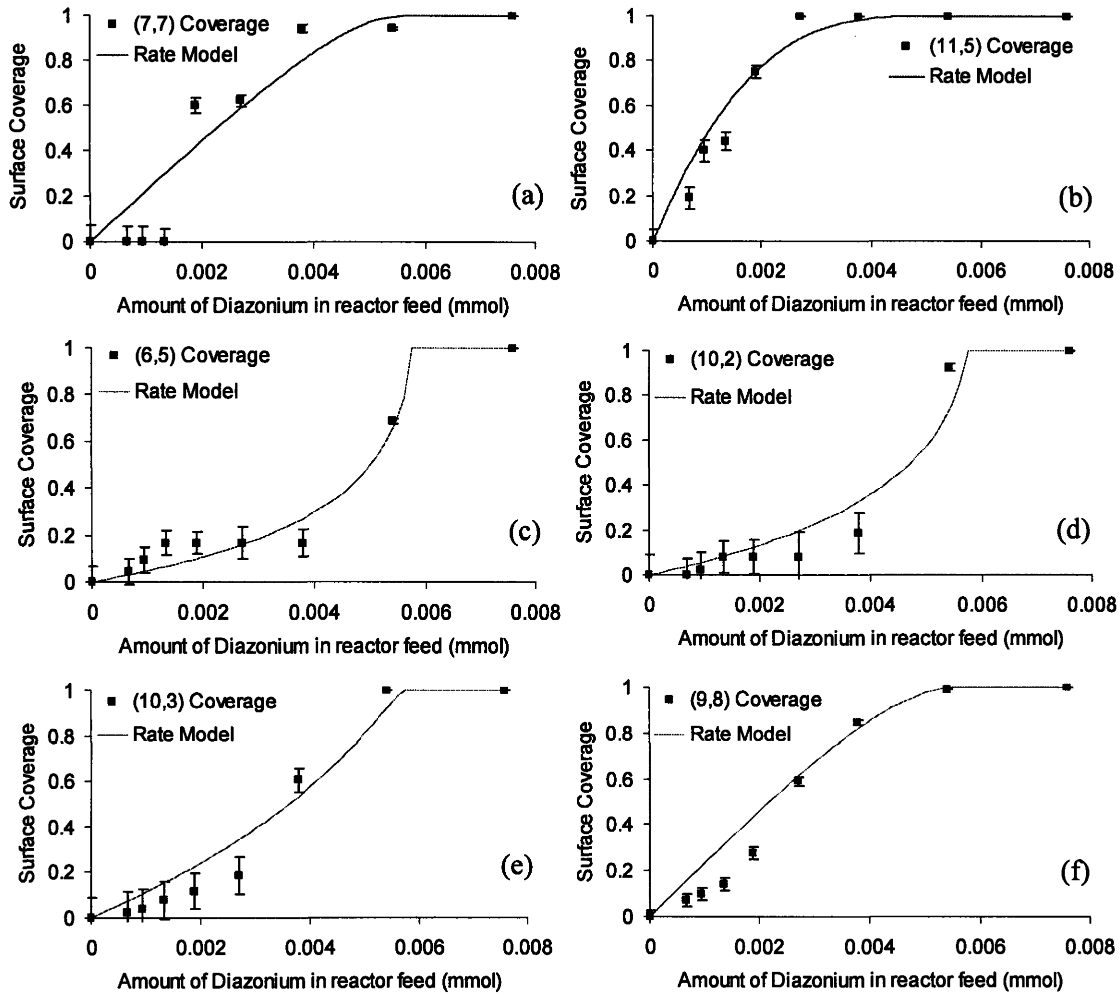


Figure 3.6. Surface coverage ($\gamma_{(n,m)}$) as a function of the total amount of diazonium fed into the 5 ml reactor for the light reaction for representative metallic and semiconducting nanotubes. The black squares denote the surface coverage data obtained from the deconvolution of the absorption spectra of the SWNT-diazonium reactions, and the red lines represent the fits predicted by the adsorption-based rate model.

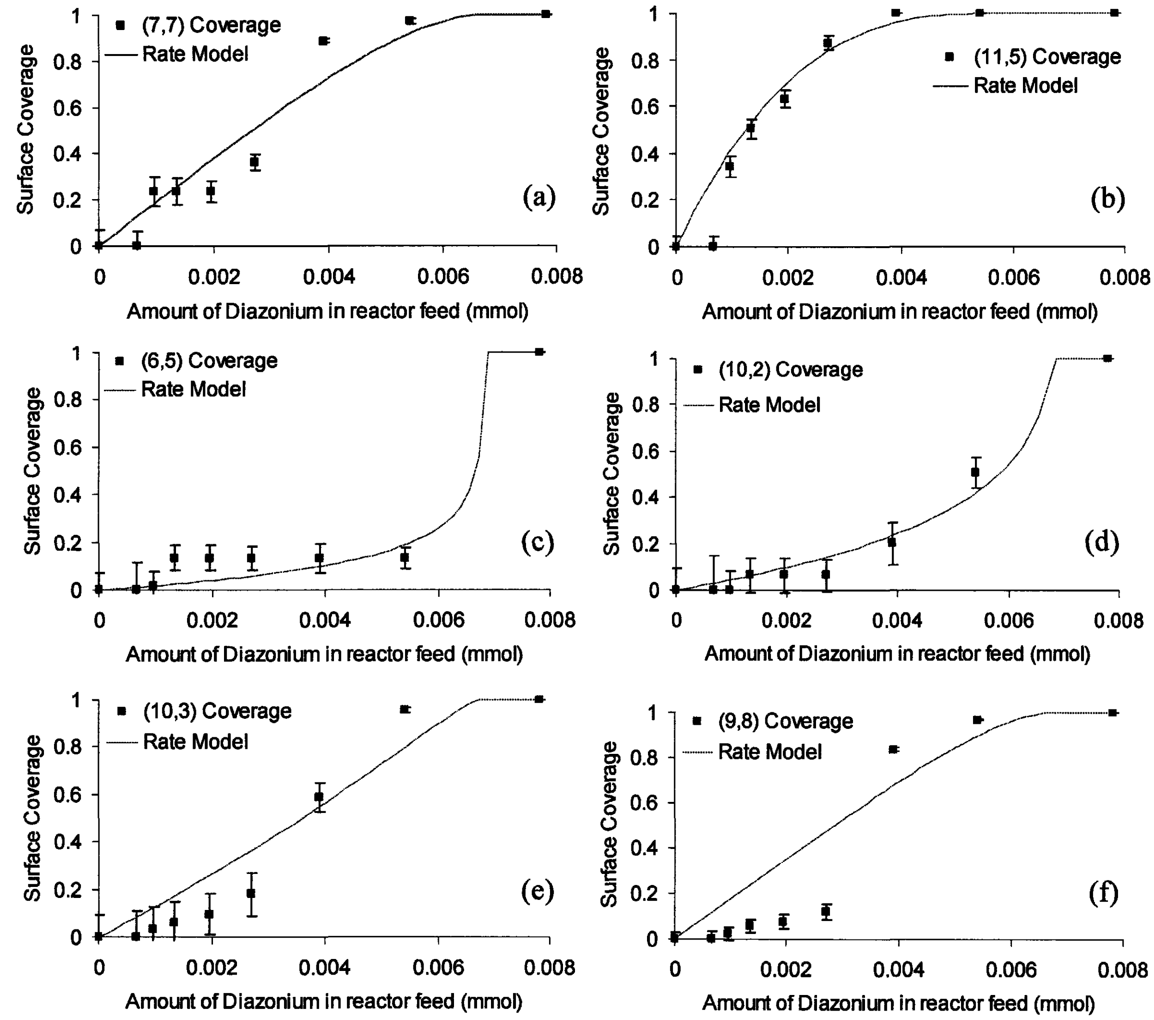


Figure 3.7. Surface coverage ($\gamma_{(n,m)}$) as a function of the total amount of diazonium fed into the 5 ml reactor for the dark reaction for representative metallic and semiconducting nanotubes. The black squares denote the surface coverage data obtained from the deconvolution of the absorption spectra of the SWNT-diazonium reactions, and the red lines represent the fits predicted by the adsorption-based rate model.

3.4.4 Structure-Reactivity

The electronic structure-reactivity relationship for nanotubes can be explained by using the Marcus and Gerischer-Marcus formalisms to fit the relative rate constant data. The comparison between the computed ($k_{(n,m)}$) and theoretical ($k_{ET}^{(n,m)}$) relative rate constants for semiconducting SWNT is shown in Figures 3.8(a)-(b), along with the

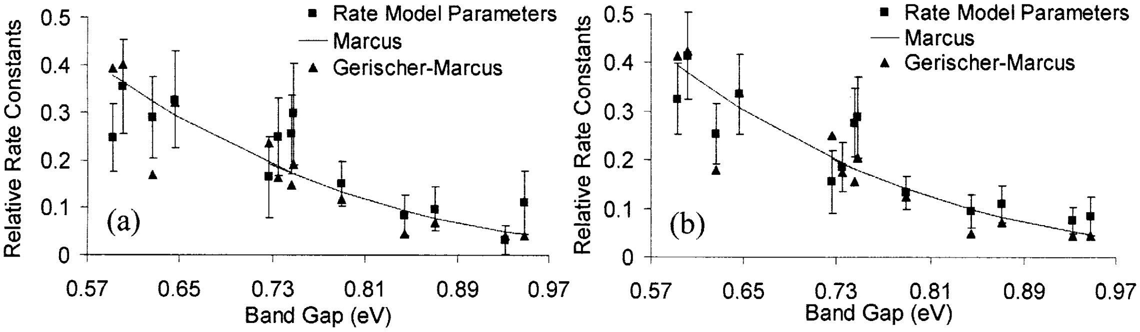


Figure 3.8. Comparison of the relative rate constants (fit parameters obtained from the surface coverage data) for the semiconductors, along with their associated 95% confidence intervals, with the predictions of the Marcus and Gerischer-Marcus theories for the (a) light, and (b) dark. Both theories predict the expected dependence of relative rate constants on band gap.

associated 95% confidence intervals for the former. The sole fit parameter, the reorganization energy (λ), was estimated as 0.54 eV by Marcus theory and 0.71 eV by Gerischer-Marcus theory. These values lie between the observed bounds for λ (0.5 eV and 1 eV).[49] Despite the scatter, the expected trend is obtained for the semiconductors: large-diameter semiconductors have higher relative rate constants than their small-diameter counterparts. Metallic nanotubes have higher rate constants than the semiconductors due to the finite DOS at the Fermi level, which is conducive to electron transfer. Although this is predicted by the Marcus and Gerischer-Marcus theories, the model and data values do not agree as well for the metallic nanotubes (Appendix A, Table A1). This difference could be attributed to the reaction step, which we have assumed as non-rate limiting.

The relative rate constants have been correlated with the band gaps instead of the SWNT diameters. This is because the band gaps play a more significant role in determining the values of the relative rate constants. The electrochemical driving force defined in equation 3.13.3 depends on the position of the first Van Hove singularity in semiconducting nanotubes, relative to the redox potential of diazonium. The larger the band gap, the smaller the driving force, and hence, the smaller the rate constant. The rate constant according to Gerischer-Marcus theory (equation 3.14.1) depends on the convolution of the densities of states of the reacting species. The larger the band gap, the larger the region where the SWNT DOS is zero, which leads to a smaller overlap between the densities of states of the nanotube and the diazonium molecule. In a SWNT-diazonium reaction mixture where different (*n*,*m*) nanotubes are present, the charge transfer mediated adsorption step could be rate-limiting because a spectrum of energy states is available for electron donation; this, along with the metal → large-diameter semiconductor → small-diameter semiconductor progression observed experimentally, also rationalise the dependence of the relative rate constants on the band gap.

3.4.5 Reaction Selectivity in the Light and Dark

The reaction selectivity has been defined as the ratio of the overall degrees of functionalisation of metallic (δ_{met}) and semiconducting SWNT (δ_{sc}). The effect of illumination on δ_{met} and δ_{sc}, as shown in Figure 3.9(a), is to increase the extents of reaction and the rates for metals and semiconductors at all concentrations of the reagent. We expect the enhanced conversion to lower the selectivity in the light when compared to the dark. This is evident in Figures 3.9(b)-(c), which depict the experimentally observed (S_e) and theoretical selectivities (S_p), respectively, for each input of diazonium. Although a similar trend is seen in both figures, there are discrepancies in the magnitudes of S_e and S_p. This can be attributed to the relatively low quality of the fit for large-diameter semiconductors at low diazonium concentrations in the dark (Figure 3.4(e)-(f)). Due to an overestimation of the surface coverage of large-diameter semiconducting SWNT in the dark reaction, the rate model underestimates its selectivity. Consequently, the difference

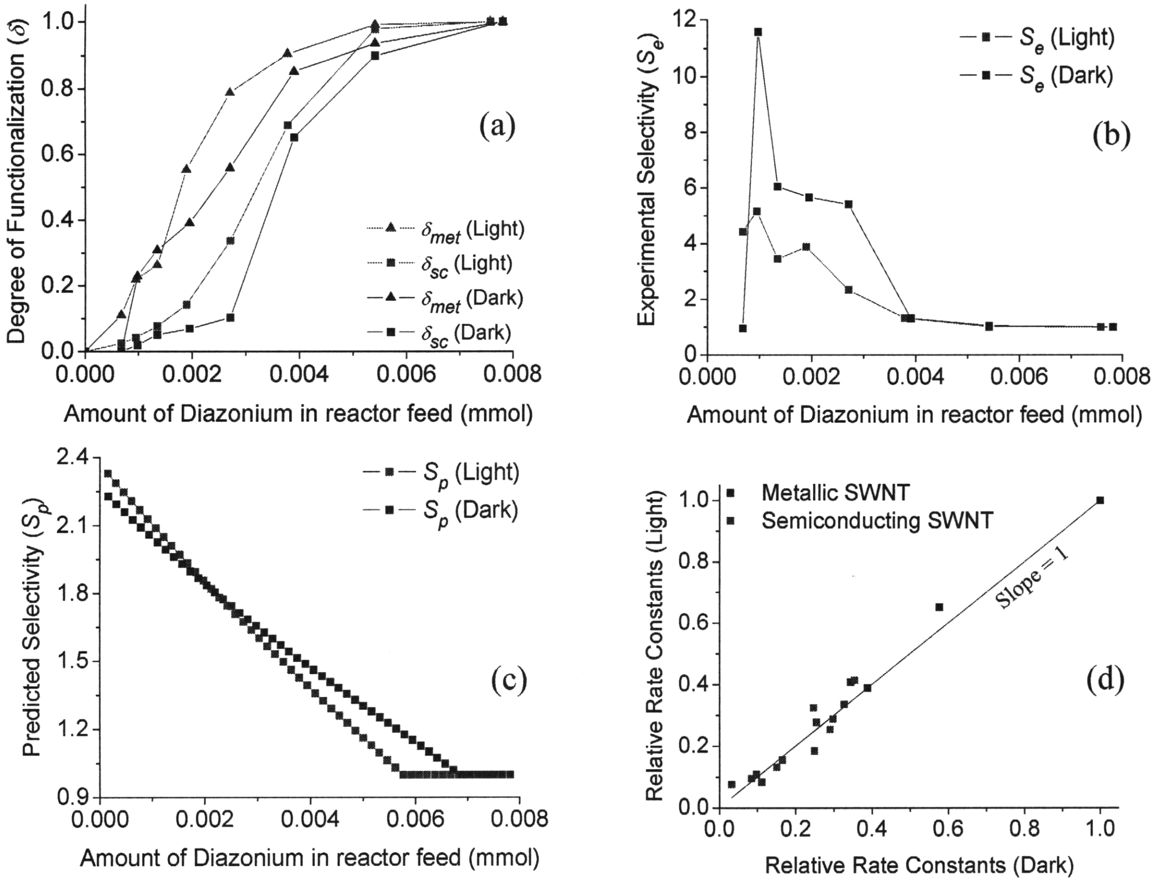


Figure 3.9. (a) Degrees of functionalisation of metallic (δ_{met}) and semiconducting (δ_{sc}) SWNT in the light and dark. (b), (c) Experimental and theoretical selectivities, respectively, as functions of the total amount of diazonium fed. The light reaction has a lower selectivity due to higher extents of reaction for all nanotubes. The rate model overestimates δ_{sc} for large-diameter semiconductors at low concentrations of diazonium in the dark reaction, thus leading to a mismatch with the observed trend. (d) A comparison of relative rate constants for the light and dark shows that they are similar. Illumination does not preferentially enhance the rate constant of a particular (*n*,*m*) species over another.

in light and dark selectivities in the low reagent concentration regime (Figure 3.9(c)) is much lower than reality (Figure 3.9(b)).

We stated previously that illumination raised the overall reaction rate, thus resulting in greater conversion. An increase in the rate can either be due to higher rate constants for the nanotubes, or a higher activity of diazonium in solution. A parity plot between the light and dark relative rate constants in Figure 3.9(d) shows that they are similar, with limited dispersion (standard deviation = 0.0417) about a line with unit slope.

With the inclusion of confidence intervals (Appendix A, Fig. A3), we see that the relative rate constants for the two cases are not statistically different from each other, thus leading to two possibilities: (a) the absolute rate constants do not change at all in the presence of light, or (b) a uniform increase in the rate constants of all the nanotubes occurs in such a way as to maintain the same relative ratios. Illumination causes the diazonium to decompose into aryl radicals, the concentrations of which are negligible in the dark. The presence of excess reactive radicals - $Ar-N=N\cdot$ and $Ar\cdot$ - increases the activity of diazonium in solution, and hence, the reaction rate. All these factors contribute to greater degrees of functionalisation for each (*n*,*m*) SWNT in the presence of illumination, thereby decreasing the selectivity.

4. Nanotube Transport in a Centrifugal Field

4.1 Introduction

The successful suspension of single walled carbon nanotubes (SWNT) in water was made possible by the use of the surfactant sodium dodecyl sulfate (SDS), and ultracentrifugation to remove bundles from the sample.[30] The centrifuge-based separation of nanotubes by diameter and electronic structure was achieved by suspension with bile salts such as sodium cholate (SC).[44] Previous efforts involved DNA as the suspension agent.[108] The non-uniform extent of adsorption of the surfactant on nanotubes of varying chiralities facilitates their separation by density-based centrifugation.

The sedimentation of solute molecules in a solvent during ultracentrifugation is modelled by the Lamm equation.[53] Under certain conditions, analytical solutions of the latter are available[109-115] but use functions that are too complex to fit to experimental data. Computer simulations reduce the number of assumptions and account for nonlinearities that would otherwise prevent closed-form answers. The first numerical solution of the Lamm equation was performed using finite element methods by Claverie et al.[116]. Later work extended the scope of this technique to the sedimentation of molecules in dynamic density gradients[117] and compressible media,[118] among other cases.[119,120]

In this work, we seek to use the theory of the centrifuge[110,121] and the numerical analysis of the Lamm equation to describe the motion of surfactant-suspended SWNT – both reacted and unreacted – in a density gradient, and estimate the number of surfactant molecules adsorbed per unit length of the nanotube. The latter determines the density of the SWNT-surfactant assembly, and although we cannot discern how it depends on chirality, we now have a bound for the number of species on the SWNT surface. It has been noted that this technique of separating suspended nanotubes with the centrifuge is purely density-based[44] and does not depend on their velocities in the gradient. We have accounted for the sedimentation of the gradient material and the solvent compressibility due to the high speed of rotation, since both factors determine the final positions of the enriched fractions.

4.2 Experimental Section

4.2.1 Preparation of Nanotube Solutions

Two types of nanotubes were used in this study: CoMoCAT SWNT (Southwest Nanotechonologies, Inc.) for separation by diameter, and HiPco SWNT (HPR 162.3, Rice University) for separation by electronic type. Suspension of SWNT in water with 2 w/v% SC was performed by ultrasonication and ultracentrifugation to obtain individual nanotubes as described in the literature.[44] In both cases, the concentration of SWNT was adjusted to 1 mg/ml. SWNT solution was first homogenised for 1 hr at 65,000 rpm (T18 basic ultra-turrax, IKA) and then sonicated for another hour with a 1/8" probe tip at 10 W (130 W ultrasonic processor, Cole-Parmer). The resulting solution was centrifuged for 1 hr at 22°C and 32,000 rpm to eliminate bundles and non-suspended nanotubes (Optima L-100XP centrifuge, Beckman Coulter). SWNT prepared in this way were used in the separation steps described below.

4.2.2 Conditions for Diameter Separation

Separation of CoMoCAT SWNT by diameter in a density gradient was performed as described in the literature.[44] The gradient was made using a non-ionic medium, OptiPrep (60 w/v% iodixanol, Sigma-Aldrich), in a linear density gradient maker (SG30, Hoefer, Inc). The resulting density of the solution ranged from 7.5 to 22.5 w/v%, with a total volume of 8 ml (Figure 4.1a). One ml of CoMoCAT SWNT (density = 20 w/v%) was injected at the bottom of the gradient and centrifuged for 22.26 hr at 22°C and 32,000 rpm using a swinging bucket SW 32.1 Ti rotor (Beckman Coulter). After centrifugation, SWNT samples were fractionated at every 150 μl in a fraction recovery system (Beckman Coulter), and characterized by UV-vis-nIR absorption spectroscopy (Shimadzu UV-310PC absorption spectrometer) to investigate the extent of enrichment of specific SWNT diameters.

4.2.3 Conditions for Electronic Type Separation

The separation of HiPco SWNT into metallic or semiconducting fractions was performed according to the protocol[44] to separate laser-ablation-grown nanotubes. Surfactant mixtures of SDS and SC in a weight ratio of 3:2 for metallic SWNT enrichment and 1:4 for semiconducting SWNT separation were utilised.[2] The density gradient also contained the same mixture of SDS and SC. The solution density ranged from 20 to 35 w/v% and 15 to 30 w/v% for the metallic and semiconducting SWNT separations, respectively. One ml of HiPco SWNT, having the same ratios of surfactant mixtures by the addition of 2 w/v% SDS in water, was injected at the bottom of each density gradient. The density of the injected SWNT solution was 32.5 w/v% and 27.5 w/v% in the 3:2 and 1:4 cases, respectively. All the other conditions were similar to the diameter separation described above.

4.2.4 Selective Reactions for Density Enhancement

The SWNT-diazonium reaction was performed at pH 5.5 by injecting the diazonium salt solution with a syringe pump (Cole-Parmer) into a semibatch reactor containing the SWNT/SDS suspension. The total volume of the 4-hydroxybenzenediazonium solution (500 μl) was added at an injection rate of 20.83 μl/h into a reactor volume of 10 ml under various diazonium concentrations (0.42, 0.84 and 1.68 mM). The reactor was well-stirred throughout the reaction time of 24 hours. The conversion of functionalised SWNT was controlled by varying the concentration of diazonium salt at a reaction temperature of 45°C.

The separation of reacted tubes from the unreacted ones based on the difference in their densities was carried out with the following arrangement of density media (OptiPrep, 60% w/v iodixanol, Sigma-Aldrich) in a centrifuge tube (Figure 4.1b):[55]

(a) 3 mL layer of 60% w/v iodixanol at the base of the centrifuge tube, which is called the "stop-layer", the purpose of which is to prevent nanotubes from reaching the base.

(b) 7 mL layer of 30% w/v iodixanol in which nanotube motion takes place under the cenitrifugal field.

(c) The remainder of the centrifuge tube was filled with water.

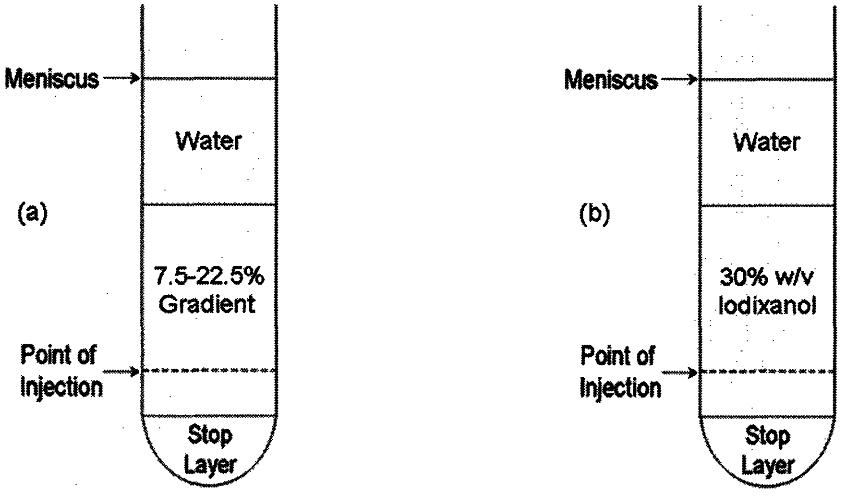


Figure 4.1. Schematic of the ordering of layers of different densities in a centrifuge tube for the (a) diameter-based separation of a nanotube solution, and (b) separation of reacted nanotubes from their unreacted counterparts. The starting solution is injected at the location shown.

A 1 mL solution of nanotubes with a concentration of 32.5% w/v was injected deep into the second layer and centrifugation was performed for 22 hours at 32000 rpm using a swinging bucket rotor (SW 32.1 Ti, Beckman Coulter). After centrifugation, 150 μL fractions of SWNT samples were characterized by UV-vis-nIR spectroscopy (Shimadzu UV-310PC absorption spectrometer).

4.3 Hydrodynamic Model Development

4.3.1 Derivation of the Lamm Equation

The sector-shaped[53,110,114,121] differential element (EFGH) in Figure 4.2 has a volume equal to $h*dr*r\theta$, where h is the height of the element, dr is its radial thickness, r is its distance from the origin (O), and θ is the angle swept by the rotor arm (OB) in a time interval dt. Within that time frame, the accumulation of the solute in the volume element is given by:

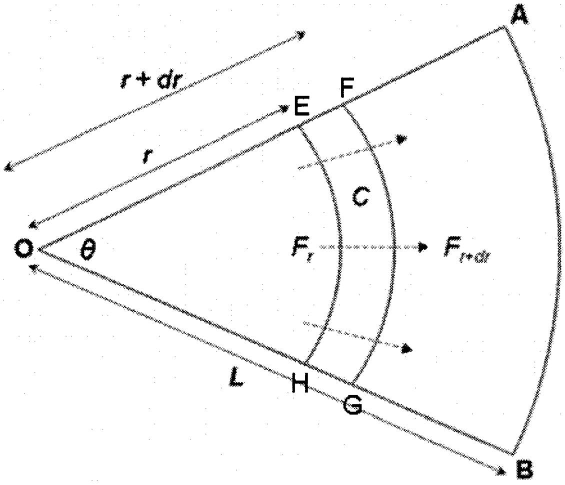


Figure 4.2. Sector-shaped cell showing a differential element across which mass transport of the solute takes place in a centrifugal field. The Lamm equation is obtained by framing a shell balance using the flow variables shown.

$$\left[C(r,t+dt)-C(r,t)\right]h\,dr\,r\theta=\left[J\,r\theta\big|_r-J\,r\theta\big|_{r+dr}\right]h\,dt \tag{4.1.1}$$

where $C(r,t)$ is the spatiotemporal distribution of the solute and J is its flux at a radial distance r from O. Dividing equation 4.1.1 throughout by $h*dr*r\theta*dt$ yields

$$\frac{\partial C}{\partial t}=-\frac{1}{r}\frac{\partial(rJ)}{\partial r} \tag{4.1.2}$$

The solute flux can be described in terms of the diffusive and sedimentation components, the latter being due to the centrifugal field applied.

$$J=-D\frac{\partial C}{\partial r}+u\,C \tag{4.1.3}$$

where D is the diffusion coefficient of the solute, and u is the magnitude of its velocity.

A solute particle experiences a net acceleration due to the competing effects of the centrifugal, buoyancy and hydrodynamic drag forces:

$$m_s a_s=m_s\omega^2 r-m_L\omega^2 r-f\,u \tag{4.1.4}$$

In the above equation, m_s is the mass of the solute and m_L is that of the displaced solvent which provides an upward thrust on the sedimenting particle. The drag coefficient is f, while the angular velocity of the rotor is ω. Assuming that the particle reaches a terminal velocity, i.e., the velocity field reaches a steady state, we get:

$$u = \frac{m_s \omega^2 r}{f}\left(1 - \frac{m_L}{m_s}\right) = \frac{m_s \omega^2 r}{f}\left(1 - \frac{\rho_L V_L}{\rho_s V_s}\right) \tag{4.1.5}$$

Here, V_L is the volume of the solvent that is displaced by the solute, and is trivially equal to the volume of the solute itself, V_s by Archimedes' Principle. The final expression that results for the steady state velocity is:

$$u = \frac{m_s \omega^2 r}{f}\left(1 - \frac{\rho_L}{\rho_s}\right) \tag{4.1.6}$$

Equations 4.1.2-6 in concert give us the Lamm equation:

$$\frac{\partial C}{\partial t} = \frac{D}{r}\frac{\partial}{\partial r}\left(r\frac{\partial C}{\partial r}\right) - \frac{1}{r}\frac{\partial}{\partial r}(ruC) \tag{4.1.7}$$

with the assumption that D does not vary spatially.

4.3.2 SWNT Concentration Profiles

The one-dimensional Lamm equation derived in the previous section can be used to calculate the concentration distribution ($C_{(n,m)}$) of a (n,m) nanotube as a function of its diffusion coefficient ($D_{(n,m)}$) and sedimentation velocity ($u_{(n,m)}$):

$$\frac{\partial C_{(n,m)}}{\partial t} = \frac{D_{(n,m)}}{r}\frac{\partial}{\partial r}\left(r\frac{\partial C_{(n,m)}}{\partial r}\right) - \frac{1}{r}\frac{\partial}{\partial r}\left(rC_{(n,m)}u_{(n,m)}\right) \tag{4.2.1}$$

The above PDE is solved with a finite volume method, using a first order upwind scheme for the sedimentation term.[122] The velocity in terms of the SWNT mass ($M_{(n,m)}$), angular velocity of the rotor (ω), friction coefficient ($f_{(n,m)}$), solution density (ρ_s) and particle density ($\rho_{(n,m)}$) is

$$u_{(n,m)} = \frac{M_{(n,m)}\omega^2 r}{f_{(n,m)}}\left[1 - \frac{\rho_s}{\rho_{(n,m)}}\right] \tag{4.2.2}$$

We define velocity explicitly instead of including it in the sedimentation coefficient, which is not constant for each SWNT and varies radially due to the non-uniform density.[123-125] The orientationally-averaged friction coefficient, $f_{(n,m)}$, is obtained by approximating the nanotube as a circular cylinder composed of a stack of rings:[126-130]

$$f_{(n,m)} = \frac{3\pi\eta L_{(n,m)}}{\ln\left[\frac{L_{(n,m)}}{d_{(n,m)}}\right] + 0.32} \tag{4.2.3}$$

where $L_{(n,m)}$ and $d_{(n,m)}$ are the SWNT length and diameter, respectively, and η is the viscosity of the aqueous solution, which is assumed to change negligibly with pressure.[131] Supposing that the nanotubes in solution do not interact with each other, the *(n,m)* frictional and diffusion coefficients can be related by the Einstein-Smoluchowski equation:[132,133]

$$f_{(n,m)} = \frac{kT}{D_{(n,m)}} \tag{4.2.4}$$

The diffusion coefficients are taken to be independent of the solute concentration. At high values of the latter, the Lamm equation ceases to be applicable.[134]

4.3.3 Solution of the Lamm Equation by Finite Volume Method

The computational domain is discretised into equally-spaced cells for the solution of the Lamm equation by the Finite Volume Method (Figure 4.2).[122] Each cell i has an average solute concentration C_i, radial coordinate r_i, left (right) boundary $r_{i-1/2}$ ($r_{i+1/2}$) with its associated mass flux $J_{i-1/2}$ ($J_{i+1/2}$). The benefit of the method of finite volumes over finite differences is that the former assures mass conservation at the smallest level of discretisation. For this reason, it is easier to start with the version of the Lamm equation that uses flux variables, i.e., equation 4.1.2.

$$\frac{\partial C(r,t)}{\partial t}=-\frac{1}{r}\frac{\partial(rJ)}{\partial r} \tag{4.1.2}$$

The definition of the flux in terms of the concentration and velocity will be implemented at the very end with the first order upwind scheme[122] to ensure that the solution algorithm remains stable throughout. Zero flux boundary conditions are applied at each end, $r_{1/2}$ and $r_{N+1/2}$, while the initial condition for the solute concentration is a pulse centered around r_0 shown in the insert to Figure 4.3. Consider the cell i. Equation 4.1.2 can be integrated as shown below within this volume between the radial limits $r_{i-1/2}$ and $r_{i+1/2}$:

$$\frac{1}{hr_i\Delta r}\int_{r_{i-1/2}}^{r_{i+1/2}}\frac{\partial C}{\partial t}h\,r\,dr=\frac{\partial}{\partial t}\left(\frac{1}{hr_i\Delta r}\int_{r_{i-1/2}}^{r_{i+1/2}}C\,h\,r\,dr\right)=-\frac{1}{hr_i\Delta r}\int_{r_{i-1/2}}^{r_{i+1/2}}\frac{1}{r}\frac{\partial(rJ)}{\partial r}h\,r\,dr \tag{4.3.1}$$

where h, r_i and Δr have already been defined. The denominator is simply the volume of the chosen cell. Defining $\bar{C}_i=\frac{1}{hr_i\Delta r}\int_{r_{i-1/2}}^{r_{i+1/2}}C\,h\,r\,dr$ as the spatially averaged concentration in cell i, the simple integration on the RHS gives

$$\frac{\partial\bar{C}_i}{\partial t}=\frac{1}{r_i\Delta r}\left[rJ\big|_{r_{i-1/2}}-rJ\big|_{r_{i+1/2}}\right] \tag{4.3.2}$$

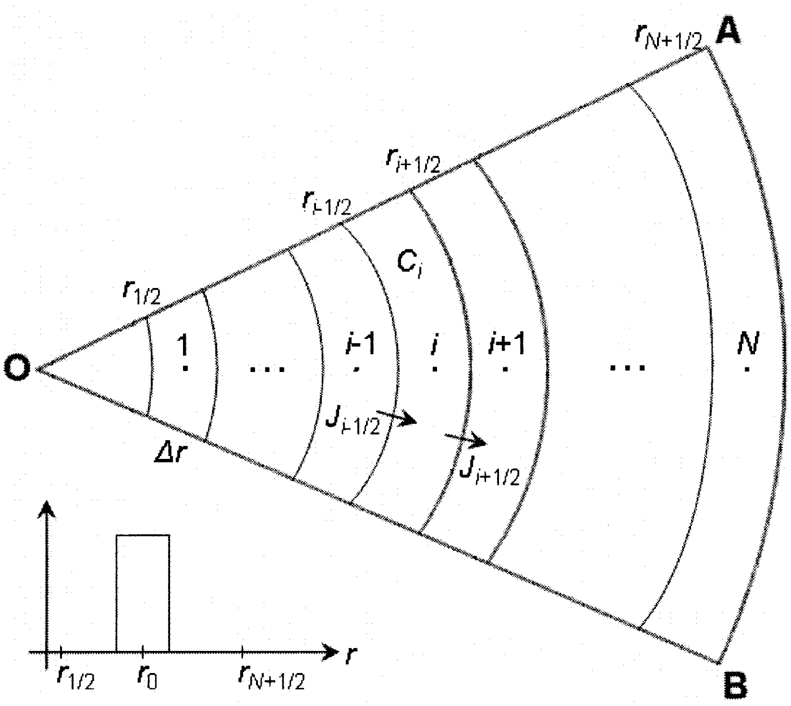


Figure 4.3. Discretisation of sector-shaped cell for the solution of the Lamm equation by the Finite Volume Method. The inset shows the initial concentration distribution of the solute centered at a radial distance r_0 from the origin O.

The next step is to average both sides of equation 4.3.2 over a time interval Δt.

$$\int_t^{t+\Delta t} \frac{\partial \bar{C}_i}{\partial t}\,dt = \frac{1}{r_i \Delta r}\int_t^{t+\Delta t}\left[rJ\big|_{r_{i-1/2}} - rJ\big|_{r_{i+1/2}}\right]dt = \frac{1}{r_i\Delta r}\left[r_{i-1/2}\int_t^{t+\Delta t} J_{i-1/2}\,dt - r_{i+1/2}\int_t^{t+\Delta t} J_{i+1/2}\,dt\right] \tag{4.3.3}$$

Using $\bar{J} = \frac{1}{\Delta t}\int_t^{t+\Delta t} J\,dt$ as the definition of a time-averaged flux across a cell boundary, we may rewrite equation 4.3.3 as:

$$\bar{C}_i\left(t+\Delta t\right) - \bar{C}_i\left(t\right) = -\frac{\Delta t}{r_i\Delta r}\left[r\bar{J}\big|_{i+1/2} - r\bar{J}\big|_{i-1/2}\right] \tag{4.3.4}$$

Therefore, the solute concentration at any time step $t+\Delta t$ in terms of the known quantities from the previous instant is:

$$\bar{C}_i\left(t+\Delta t\right) = \bar{C}_i\left(t\right) - \frac{\Delta t}{r_i\Delta r}\left[r\bar{J}\big|_{i+1/2} - r\bar{J}\big|_{i-1/2}\right] \tag{4.3.5}$$

Equation 4.1.3 states the dependence of the flux on the local concentration gradient and the velocity of the solute.

$$J = -D\frac{\partial C}{\partial r} + u\,C \tag{4.1.3}$$

For the direction of flow shown in Figure 4.3, equation 4.1.3 may be discretised into the left- and right-hand mass fluxes using the first order upwind scheme:[122]

$$\bar{J}_{i+1/2} = -\frac{D}{\Delta r}\left(\bar{C}_{i+1} - \bar{C}_i\right) + u_{i+1/2}\,\bar{C}_i \tag{4.4.1}$$

$$\bar{J}_{i-1/2} = -\frac{D}{\Delta r}\left(\bar{C}_i - \bar{C}_{i-1}\right) + u_{i-1/2}\,\bar{C}_{i-1} \tag{4.4.2}$$

The velocity at a boundary is estimated from equation 4.1.6. Equations 4.4.1-2 are then substituted into 4.3.5 and applied to each node $1 \leq i \leq N$ in the computation domain. This results in a system of algebraic equations in terms of the unknown concentrations at each point. It may be succinctly expressed in matrix form:

$$c^{n+1} = M\,c^n \tag{4.5}$$

where c^n (c^{n+1}) is the vector of concentrations at the time instant n (n+1), and M is the coefficient matrix. A forward time-stepping algorithm along with the stated initial and boundary conditions can be used to calculate the concentration distribution in the centrifuge time at any instant.

4.3.4 Density of Surfactant-SWNT Assembly

The density of the surfactant-nanotube assembly ($\rho_{(n,m)}$) can be described by a single parameter, namely the number of surfactant molecules adsorbed per unit length of the SWNT (n_s). For a specific (n,m) species, the number of carbons per nanometer (n_c)

can be estimated,[3,135] and when combined with n_s, yields the total mass per unit length ($M_{(n,m)}$):

$$M_{(n,m)} = \sum_j n_{s,j} M_{s,j} + n_c M_c + n_f M_f + n_{sol} M_{sol} \tag{4.6.1}$$

where M_s (M_c) is the molecular (atomic) weight of surfactant (carbon), and the summation is carried out over all the surfactants in solution. The last pair of terms is applicable when the interior of the nanotube is filled with either n_f molecules (per nm) of a fluid with molecular weight M_f or n_{sol} molecules (per nm) of the solvent with molecular weight M_{sol}. The general formula for the volume per unit length of the assembly is

$$V_{(n,m)} = \sum_j n_{s,j} V_{s,j} + \frac{\pi}{4} d_{eff}^2 - n_{sol} V_{sol} \tag{4.6.2}$$

$$d_{eff} = d_{(n,m)} + 2r_c \tag{4.6.3}$$

where r_c is the van der Waals radius of a carbon atom in aromatic molecules (~1.72 Å).[136] The third term in equation 4.6.2 is a correction when the interior of the SWNT is accessible to the solvent whose molar volume is V_{sol}. In this work, we have assumed that the pores in the SWNT lattice are devoid of any fluid, i.e., $n_f = n_{sol} = 0$. Additionally, the nanotubes are presumed to be uncapped and lacking catalyst particles. Values of the anhydrous molar volumes of the surfactants under study (V_s)[137-139] were found in the literature. Anhydrous surfactant molecules impart little, if any, buoyancy to the nanotubes. We have therefore accounted for one hydration shell while calculating the apparent molar volumes of SC (~613 ml/mol) and SDS (~403 ml/mol) using approximate hydration numbers for each.[140,141] Finally, the buoyant density of a generic nanotube is

$$\rho_{(n,m)} = \frac{M_{(n,m)}}{V_{(n,m)}} \tag{4.6.4}$$

4.3.5 Dynamic Density Gradient

The concentration of the density gradient material, iodixanol, changes in space and time due to sedimentation, leading to a variation in the solution density (ρ_s).[117] Consequently, the nanotube density cannot be estimated from its final position in the gradient. The iodixanol molecules settle according to equation 4.2.1, with the subscript (*n,m*) replaced by *I*. The motion of the gradient in the centrifugal field is assumed to be independent of the co-solute (SWNT).[142] Instead of solving the partial differential equations for the nanotubes and iodixanol simultaneously, a look-up table containing the sedimentation profiles of the gradient at different times was created beforehand. During the computation of the SWNT trajectories, the instantaneous density gradient was obtained by interpolation from the tabulated values. The diffusion coefficient of iodixanol (D_I) was calculated by comparing the theoretically predicted gradient profile with data collected by Arnold et al[44] (Figure 4.4), and was found to be ~2.5×10^{-10} m^2/s.

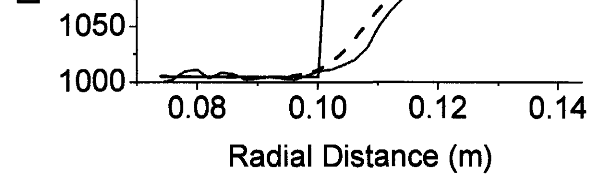


Figure 4.4. The original density gradient (blue) changes after centrifuging for 12 hr (red) due to the sedimentation of Iodixanol. The solution of the Lamm Equation predicts the final density profile (black), and a comparison with the experimental data gives the diffusion coefficient of Iodixanol (D_I) as ~2.5×10^{-10} m^2/s.

4.3.6 Solvent Compressibility

The compressibility of water (κ) is so low (4.6×10^{-10} Pa^{-1}) that it is considered incompressible at ordinary pressures. However, the tremendous forces generated by centrifugation lead to pressures that are high enough to affect the solution density. The radial dependence of pressure (P) is given by[118]

$$dP = \rho_s \omega^2 r dr \tag{4.7.1}$$

Combining equation 4.7.1 with the definition of compressibility,

$$d\rho_s = \kappa \rho_s dP \tag{4.7.2}$$

a relation between the densities of successive cells in the computation domain can be obtained.

$$\rho_{s,i+1} = \frac{\rho_{s,i}}{1-\alpha\rho_{s,i}} \tag{4.7.3}$$

where $\alpha = \frac{\kappa\omega^2}{2}\left(r_{i+1}^2 - r_i^2\right)$. In this pair of equations, i denotes the i^{th} cell in the discretised space. The density in the first cell ($\rho_{s,1}$) is simply that of water at atmospheric pressure. We have considered the compressibility of the iodixanol-water solution to be the same as that of pure water.

4.3.7 Estimation of Time Scale

The length-dependent time scale for each (n,m) entity ($\tau_L^{(n,m)}$) is determined by the interplay of the hydrodynamic drag and the centrifugal force.

$$\tau_L^{(n,m)} = \frac{f_{(n,m)}}{M_{(n,m)} L_{(n,m)} \omega^2} \tag{4.8}$$

In the denominator of equation 4.8, the SWNT length ($L_{(n,m)}$) appears because the total mass of the nanotube system is used to calculate the centrifugal force. For the CoMoCAT sample, a Gaussian length distribution was used with parameters that were obtained for DNA-SWNT separated by centrifugation,[108] while for HiPco, a log-normal abundance profile was assumed.[143] Centrifugal forces have a smaller effect on short nanotubes as compared to longer ones, since their masses are lower. In the density gradient, the former travel slower but diffuse to a greater extent than the latter.

4.4 Results and Discussion

4.4.1 Assumptions in the Model

Monolayer coverage of the surfactant on the SWNT is assumed while calculating the density of the assembly. In the presence of a co-surfactant, both species are thought to adsorb at different sites on the nanotube so that the total volume equals the sum of the component volumes. A conclusive visualisation of SDS adsorbed on SWNT is yet to be articulated. SDS has been shown to form cylindrical micelles,[30] beads,[144] hemimicelles[145] and more recently, random, structure-less features[146] on SWNT surfaces. We have adopted the last formulation in determining the total surfactant volume. The estimation of n_s is contingent on the assumption that the SWNT is pristine, and can be constructed by a simple axial translation of the unit cell. End effects have been neglected. The use of a single parameter (n_s) for a separated fraction implies the uniform adsorption of surfactants on the corresponding nanotube surface. In other words, each (*n*,*m*) species has a single density, not a range.

Diffusion plays a secondary role while the nanotubes are moving through the gradient in a centrifugal field. However, its effect will be greater during the start-up and shut-down phases of the centrifuge. We have neglected these regimes in the computation. It is not clear whether the density gradient is linear as soon as it is injected into the centrifuge tube. The evolution of the gradient with time has been calculated by approximating linearity at the start of the run.

The critical micelle concentration (CMC) and solubility of a surfactant are affected by pressure.[147,148] A change in the CMC can be neglected due to the excess quantities of surfactants used to suspend SWNT. The solubility of SDS in water drops drastically above 1000 atm.[147] Calculations indicate that pressures in the region of interest – the density gradient – do not exceed this value (Appendix B, Fig. B1). A similar analysis could not be performed for SC due to the lack of information regarding the effect of pressure on its solubility in water.

4.4.2 Separation using Sodium Cholate

The diameter-based separation of CoMoCAT nanotubes using centrifugation is conceptually the easiest to describe mathematically, since only one surfactant (2 w/v% SC) is involved. The model has been applied to data generated during the course of this work and in the literature.[44] We use relative SWNT concentrations as defined by Arnold et al.,[44] wherein the (*n*,*m*) concentrations are normalised to the corresponding maxima. The two datasets differ in the rotor configuration, centrifuge tube dimensions and centrifugation time. The final SWNT positions (Figures 4.5a,b) and gradient profiles[46] differ in both cases. However, the result that we seek – n_s, number of SC molecules adsorbed per nm – should be similar, as is evident in Table 4.1, which shows n_s with its

95% confidence limits.[88] The radial SWNT concentration curves from the Arnold dataset are approximate and this could adversely affect the comparison. In the original work,[44] the distribution of each fraction was given in terms of densities. The spatial equivalent had to be extracted from the predicted density gradient after 12 hours of centrifugation.

Earlier measurements of SDS coverage on SWNT yielded 2-3 molecules/ nm^2.[149,150] The surface coverage of sodium dodecylbenzene sulfate was at least 8 times higher than that of SDS.[151] Such a high packing density might probably be due to adsorption along the tube followed by the 'tails-on' configuration.[151,152] Our estimates of SC adsorption on an areal basis are listed in Table 4.1. The values for SC surface coverage are lower than those of SDS because of the larger molar volume of the former. The diameter and length of the cholic acid molecule are roughly 0.6 nm and 1.32 nm, respectively.[153] It is plausible that at least 2 cholate molecules can adsorb onto the SWNT surface within a linear distance of 1 nm.

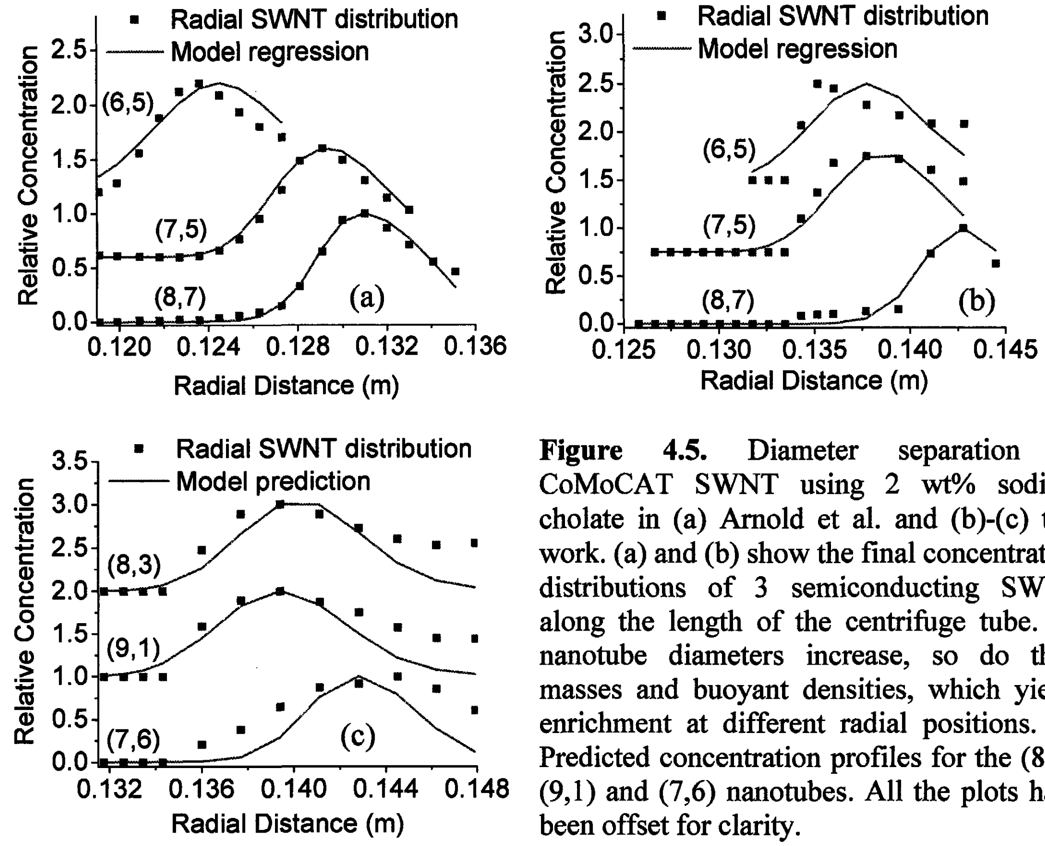


Figure 4.5. Diameter separation of CoMoCAT SWNT using 2 wt% sodium cholate in (a) Arnold et al. and (b)-(c) this work. (a) and (b) show the final concentration distributions of 3 semiconducting SWNT along the length of the centrifuge tube. As nanotube diameters increase, so do their masses and buoyant densities, which yields enrichment at different radial positions. (c) Predicted concentration profiles for the (8,3), (9,1) and (7,6) nanotubes. All the plots have been offset for clarity.

SWNT	n_s (molec/nm)	Lower Limit	Upper Limit	Areal Basis (molec/nm^2)	Density (kg/m^3)
(6,5)	2.1229	2.1038	2.1601	0.6138	1060.3
	2.0955	2.0611	2.1294	0.6058	1063.6
(7,5)	2.0003	1.9920	2.0074	0.5428	1085.2
	2.1382	2.1247	2.1560	0.5802	1068.7
(8,7)	2.0076	1.9988	2.0120	0.4644	1094.9
	2.0753	2.0638	2.0865	0.4801	1087.4

Table 4.1. Comparison of the fit parameter (n_s) for 3 semiconducting SWNT in 2 datasets involving sodium cholate, and the corresponding buoyant densities: Arnold et al. (blue) and the present work (black). The fifth column shows the nominal value of n_s on an areal basis. Relative to the corresponding numbers for SDS, SC has much lower coverage values, probably due to its larger size.

We have also collected data for different run times of the 2 w/v% SC-SWNT sample – 12 hours and 44.53 hours – in order to check the veracity of the coverage estimates (Appendix B, Fig. B2). The final SWNT concentration profiles were situated close to the point of injection. In the case of the 12 hr sample, this was because the nanotubes had not been given sufficient time to move upwards. For the 44.53 hr sample, the extended run time meant that the SWNT had reached their respective isopycnic points and were carried downwards by the sedimenting density gradient. Either way, it was not possible to obtain data that were as clean as the 22.26 hr sample, since the bulkier tubes at the bottom have high absorbances and artificially raise the concentrations of the SWNT of interest. The approximate profiles for the (6,5) and (7,5) nanotubes were extracted and fitted. The values of n_s for the 12 hr (44.53 hr) runs are 1.815 (1.84) and 1.832 (1.89) molecules/nm, respectively, which are not identical to those in Table 4.1 but are still within 5-10% of the corresponding 22.26 hr values. Ideally, the same number of adsorbed SC molecules should be able to describe the 12, 22.26 and 44.53 hr datasets. However, the optimal value of the fit parameter, n_s, is very sensitive to factors such as the spatial location of the fractions in the centrifuge tube, the starting position of the SWNT sample and its spread after being injected into the gradient. In addition, as mentioned previously, the proximity of the SWNT fractions to the bulk sample also skews the concentration profiles. We note that the values reported in this work are simply statistical averages of the SC population on the nanotube surface. There is a constant exchange of SC molecules from the nanotube to the solution due to the thermodynamic equilibrium that exists between the adsorbed, micellar and free surfactant phases.[154]

A more reliable test of n_s is against the concentration profiles of the (9,1), (8,3) and (7,6) nanotubes (Figure 4.5c). Since the fitted estimates of n_s are clustered around 2 molec/nm, the profiles for these 3 tubes were obtained by fixing n_s at 2.04 molec/nm. The predictions are not perfect, although they roughly capture the SWNT positions after 22.26 hours of centrifugation. Note that the experimental concentration profiles for individual SWNT are extracted from the absorption spectra, and are therefore affected by the convolution with neighbouring spectral peaks.[155]

4.4.3 Separation using SDS-SC Mixtures

Tuning the SDS:SC weight ratio has resulted in the enrichment of semiconductors (surfactant ratio = 1:4) and metals (surfactant ratio = 3:2), most prominently in the case of laser-ablation-grown SWNT.[44] HiPco samples have metallic and semiconducting species of comparable diameters. It is difficult to discern electronic separation in the case of HiPco nanotubes for 2 reasons: (i) tracking the absorption peak of a specific metal as an indicator of enrichment is not feasible due to the presence of semiconducting E_{22} peaks in the same spectral region, (ii) metallic fractions in the centrifuged samples may also contain semiconductors with similar diameters. We have focused on the (6,5), (7,5) and (8,7) semiconducting nanotubes so that information obtained from the CoMoCAT fits can be used for the HiPco data. In other words, the n_s estimates from the single surfactant (SC) case provide a foundation for examining the dual surfactant (SDS-SC) scenario.

The fits of the 3:2 data (Figure 4.6a) show a diameter-based separation for semiconducting SWNT, although their buoyant densities are observed to be higher than

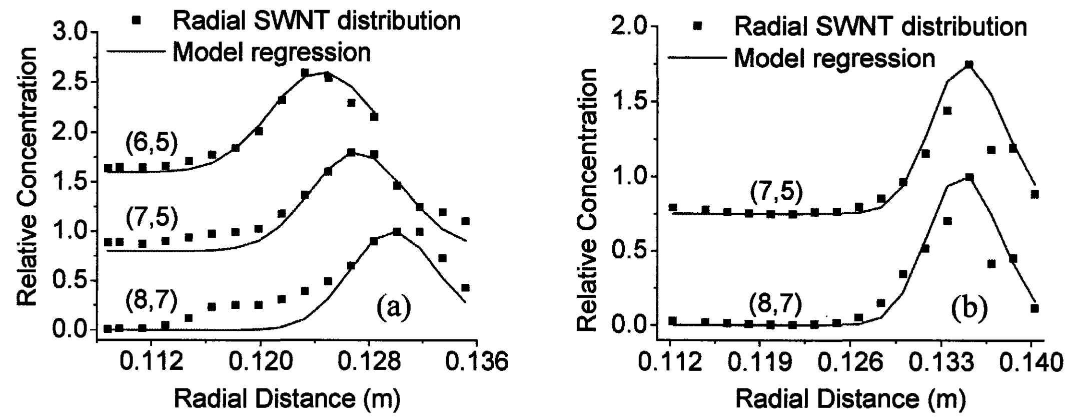


Figure 4.6. Concentration profiles for HiPco semiconducting SWNT with (a) 3:2 SDS-SC weight ratio, and (b) 1:4 weight ratio. The fits assume that the total number of surfactant molecules adsorbed on the SWNT is conserved. SDS is postulated to replace the already existing cholate, whose concentration on the nanotube surface is given by the analysis of the CoMoCAT diameter-separation data. The plots have been offset to improve clarity.

SWNT	$\{n_{SDS}, n_{SC}\}$ (molec/nm)	Density (kg/m^3)
(6,5)	{0.6629,1.4326}	1094.9
	—	—
(7,5)	{0.8715,1.2667}	1109
	{1.0643,1.0739}	1119.1
(8,7)	{.8545,1.2208}	1124.4
	{0.7639,1.3113}	1120.2

Table 4.2. Fit parameters for different SDS:SC weight ratios – 3:2 (blue) and 1:4 (black) – along with the buoyant densities of the HiPco SWNT-surfactant assemblies for each case.

when SC alone is used (Table 4.2). The desorption of SC from the nanotube surface and its replacement by SDS, which has a lower molar volume, leads to an increase in density. The concentration profile for each semiconductor was fitted by assuming a 1:1 molecular displacement of SC by SDS, with the initial SC coverage already known from the CoMoCAT fits. While this may not be an accurate representation of reality, it does yield a lower bound for the SDS population on the nanotube. The analysis reveals the progressively greater uptake of SDS molecules (i.e., greater SC desorption) as SWNT diameter increases (Table 4.2). This is consistent with the higher activation energies required by SC to desorb from most small-diameter nanotubes.[156,157] The ability of the 3:2 weight ratio to separate metals from semiconductors can be explained by the stronger interaction of sodium cholate with the former. Apart from the fact that cholate suspends more nanotubes due to its naphthenic structure,[158] the proximity of the hydroxyl groups in the cholate molecule to the nanotube surface results in electronic polarisation towards the SWNT.[159] Metallic nanotubes have delocalised electrons at the Fermi level, and are more receptive to induced charges. Thus, metallic SWNT have lower densities in the 3:2 case because of minimal SC desorption as compared to the semiconductors. In the case of large-diameter metals, it is also plausible that SDS adsorbs onto vacant areas on the surface and further reduces the density.

The 1:4 surfactant ratio leads to the enrichment of the (6,5) nanotube in the HiPco- and CoMoCAT-based runs (Figure 4.6b); however, the distribution of the (6,5) is too broad to be accurately described by our model, and so, we have restricted parameter estimation to the (7,5) and (8,7) nanotubes (Table 4.2). Here, as expected, we observe that the increase in density can be accounted for by the adsorption of SDS in place of SC. Given the smaller amount of SDS present in the 1:4 surfactant solution as compared to 3:2, it is expected that its population on the SWNT surface will be correspondingly lower, which certainly is the case for the (8,7) nanotube (0.764 molec/nm versus 0.855 molec/nm). By this logic, there is an apparent discrepancy for the (7,5) species. At this juncture, we would like to emphasise that the 1:1 replacement of SC by SDS was assumed because of the lack of information involving any other adsorption scheme. It is entirely possible that not all the cholate molecules will be replaced by SDS, which still raises the density of the nanotube, while conforming to the approximate diameter dependence of SC desorption.[156,157]

4.4.3 Extension of Model to Reacted Nanotubes

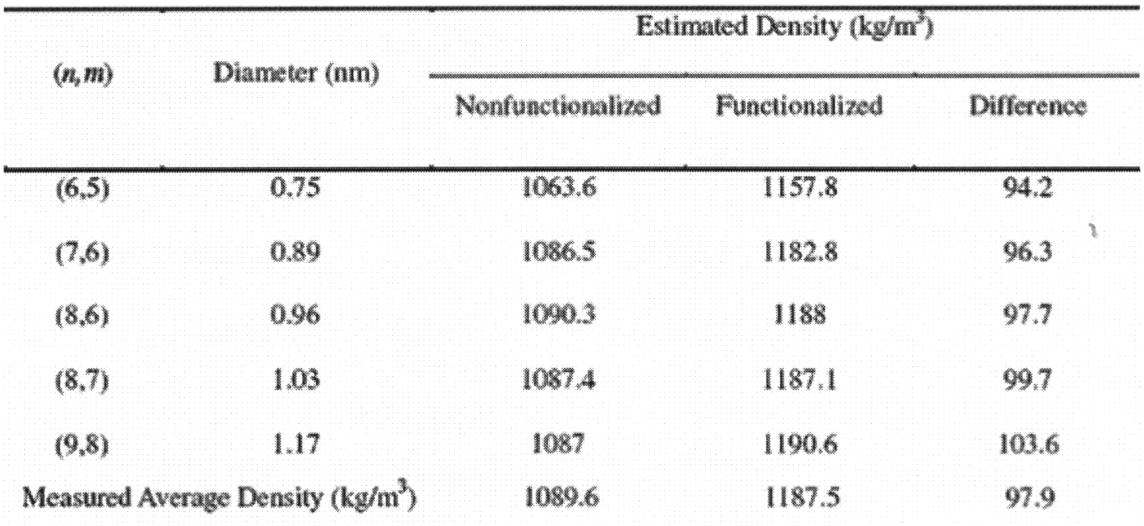
Equations 4.6.1-2 were used to calculate the densities of unreacted surfactant-coated nanotubes. The simple modification in equations 4.9.1-2 can estimate the densities of SWNT reacted with diazonium groups.[55] Assuming a uniform loading (n_d~1 diazonium group per 10 C atoms) of 4-hydroxybenzenediazonium molecules on the nanotube surface, the mass and volume per length can be modified.

$$M_{(n,m)} = n_c M_c + n_s M_s + n_d M_d \tag{4.9.1}$$

$$V_{(n,m)} = n_s V_s + \frac{\pi}{4} d_{eff}^2 + n_d V_d \tag{4.9.2}$$

where M_d (V_d) is the molecular weight (molar volume) of the functional moiety, and the other variables retain their original meanings. The value of n_s is assumed to remain the same as in the unreacted case (~2 molecules/nm), while the diazonium molecules attached to the nanotube are not assigned a hydration layer. The predicted values of pristine and reacted semiconducting SWNT are shown in the first 5 rows of Table 4.3. A glance at columns 3 and 4 indicates that a solvent whose density lies in between the reacted and unreacted densities will promote their motion in opposite directions upon centrifugation. The solvent density was fixed at 1164 kg/m^3 (30% w/v iodixanol, Figure 4.1b). Figure 4.7 shows the initial (red) and final (black) density profiles in the centrifuge tube. Two distinct bands of nanotubes were formed after the 22-hour centrifugation step (Figure 4.7). As shown in the last row of Table 4.3, the upper (unreacted) and lower (reacted) bands have densities of ~1090 kg/m^3 and ~1188 kg/m^3 respectively. These values match our predictions and therefore lend physical significance to the model for the estimation of

SWNT density. The analysis using a heavier functional group (4-nitrobenzenediazonium) also yielded similar a correspondence between model and experiment.[55]



(n,m)	Diameter (nm)	Estimated Density (kg/m^3)		
		Nonfunctionalized	Functionalized	Difference
(6,5)	0.75	1063.6	1157.8	94.2
(7,6)	0.89	1086.5	1182.8	96.3
(8,6)	0.96	1090.3	1188	97.7
(8,7)	1.03	1087.4	1187.1	99.7
(9,8)	1.17	1087	1190.6	103.6
Measured Average Density (kg/m^3)		1089.6	1187.5	97.9

Table 4.3. Comparison of calculated and experimental densities for unreacted and reacted semiconducting SWNT. The last row shows the average densities of the unreacted and reacted fractions of nanotubes obtained after ultracentrifuge-based separation.

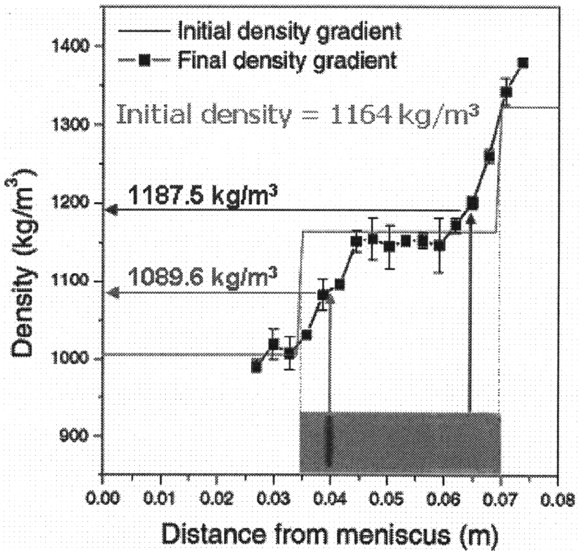


Figure 4.7. Density measurement for functionalised and unfunctionalised nanotubes. The initial density profile (red) used a solvent density of 1164 kg/m^3 based on the predictions in Table 4.3. The final densities (black) were measured after 22 hours of centrifugation. The inset shows a picture of the centrifuged 0.84 mM reaction sample with 2 distinct bands representing the unreacted (~1090 kg/m^3) and reacted (~1188 kg/m^3) nanotubes.

5. One-Dimensional Nanostructure-Guided Chain Reactions

5.1 Introduction

One-dimensional chains with different potential functions have been extensively studied with respect to their dynamical properties,[71] shock waves[160-162], bond dissociation,[163] energy relaxation[164,165] and pulse propagation.[166,167] We use this simple platform to model sustained chain reactions that propagate in a single dimension, as along the backbone of a carbon nanotube. Energetic groups bonded to the lattice react such that the energy released is coupled back into the nanostructure and directed along the backbone to propagate the reaction. The hypothesis is that the reduced dimensionality of the nanostructure should be able to guide and accelerate the reactions along a preferred orientation. Apart from their utility as thermal interface materials[168,169], the quasi one-dimensional nature of SWNT also holds prospects for applications as waveguides. Past work has focused on confined propagation of x-rays[170] and thermal neutrons[171] through carbon nanotubes.

In order to determine the characteristics of reactions with positive energy feedback, we have represented a carbon nanotube as a linear chain of oscillators connected by springs. Each oscillator is allowed to move in 3 dimensions, while thermal conduction occurs only along the chain. The bead-spring model that was proposed by Buehler[59] accounts for stretching and bending contributions to the overall force field. The entire setup is assumed to be isolated from its surroundings. Each bead in the lattice is loaded with energetic molecules (EM) at a specified density. An excitation applied at one end of the chain is expected to raise the local temperature at the reaction sites leading to the decomposition of the group. The one-dimensionality of the nanostructure channels the energy released during combustion and facilitates subsequent reactions.

5.2 Model Development

5.2.1 Types of Lattices and Force Field Components

The Hamiltonian (H) for a system of N oscillators as a function of the individual momenta (p_i), masses (m_i) and position vectors ($\vec{r}_i$) is:

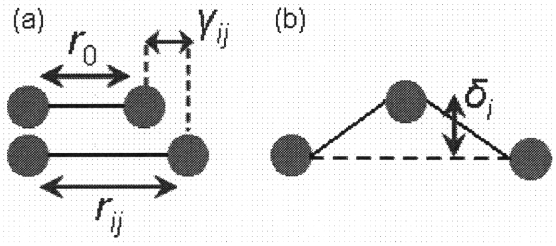


Figure 5.1. (a) Component of the force field that involves the stretching of the bond about the equilibrium distance r_0; (b) Displacement of the i^{th} bead from its equilibrium position (δ_i) which leads to a finite onsite potential.

$$H=\sum_{i=1}^{N}\frac{p_i^2}{2m_i}+\frac{1}{2}\left[V_s\left(\left|\vec{r}_{i+1}-\vec{r}_i\right|-r_0\right)+V_s\left(\left|\vec{r}_i-\vec{r}_{i-1}\right|-r_0\right)\right]+V_b\left(\vec{r}_i,\vec{r}_{i+1},\vec{r}_{i+2}\right)+U\left(\left|\vec{r}_i-\vec{r}_{i0}\right|\right) \quad (5.1.1)$$

where $V_s(\gamma_{ij})$ is the contribution to the total potential due to bond stretching, and γ_{ij} is the deviation of the i-j bond length from its equilibrium value, r_0 (Figure 5.1); V_b is the bond angle component of the overall force field in terms of the position vectors of the beads at locations i, i+1 and i+2; $U(\delta_i)$ is the onsite potential, with δ_i denoting the displacement of the i^{th} oscillator from its equilibrium position, r_{i0} (Figure 5.1). The effect of the weak interactions between nonbonded beads can be safely ignored, since the cutoff distance for the 6-12 Lennard-Jones potential is less than the distance between a bead and its third nearest neighbour.[59]

The analytical forms of $V(\gamma_{ij})$ and $U(\delta_i)$ change according to the type of lattice used. Non-dimensionalisation of equation 5.1.1 yields a time scale (t_{sc}) that depends on the tensile force constant (k_s) and the total mass of the bead (i.e., with the full complement of reactive groups).

$$t_{sc} = \sqrt{\frac{m_b + n_0 m_e}{k_s}} \tag{5.1.2}$$

In the above equation, m_b (m_e) denotes the mass of a single bead (reactive group) and n_0 is the total number of energetic groups loaded on each bead.

The details of the harmonic potential along with the associated parameters have already been outlined in previous papers by Buehler et al.[57-59]

$$V\left(\gamma_{ij}\right) = \frac{k_s}{2}\gamma_{ij}^2 \tag{5.2.1}$$

The FPU-β lattice has a quartic term in addition to the harmonic portion, which leads to the nonlinearities in inter-bead interactions and the aforementioned off-diagonal effect. The onsite potential, $U(\delta_i)$ is zero in this case.

$$V_{ij}\left(\gamma_{ij}\right) = \frac{k_s}{2}\gamma_{ij}^2 + \frac{k_s'}{4}\gamma_{ij}^4 \tag{5.2.2}$$

We have assumed that $k_s' = k_s$. The nonlinear term plays a greater role at higher deformations. As the energy input into the lattice increases, so does the frequency of oscillation, which results in a pulse propagating at a higher speed and lower dispersion than in a harmonic chain.[166,167]

The φ^4 lattice retains the pure harmonic interactions between beads (equation 5.2.2) but includes an external contribution in the form of a non-zero onsite potential

$$U\left(\delta_i\right) = \frac{k_1}{2}\delta_i^2 + \frac{k_2}{2}\delta_i^4 \tag{5.2.3}$$

which represents the effect of a substrate or the environment of the chain on the motion of solely the i^{th} bead. The nonlinearity is embedded within the oscillator itself, thus resulting in diagonal anharmonicity. Again, we set $k_1=k_2=k_s$.

Since the beads are allowed to move in all 3 dimensions, a deviation of the *ijk* bond angle as shown in Figure 5.2a from its equilibrium value (π) also contributes to the overall potential in the form of V_b.

$$V_b\left(\theta_{ijk}\right)=\frac{k_b}{2}\left(\theta_{ijk}-\pi\right)^2 \tag{5.2.4}$$

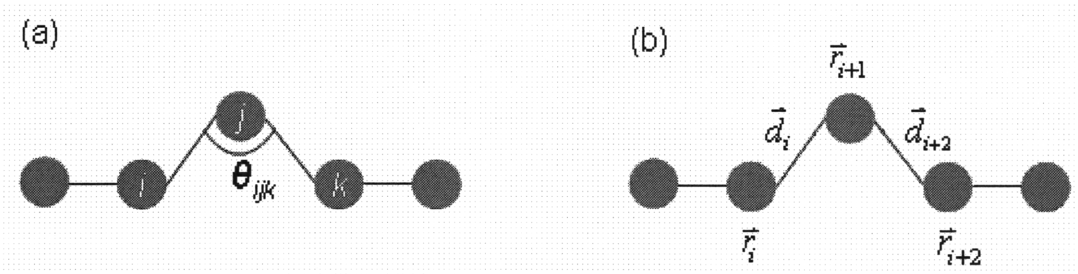


Figure 5.2. (a) Deviation of the *ijk* bond angle (θ_{ijk}) from its natural value, π, adds a bending component to the overall inter-bead potential; (b) Position vectors that determine the magnitude of the bond angle formed by the beads at the locations *i*, *i*+1 and *i*+2.

Let the bond angle determined by the beads at locations *i*, *i*+1 and *i*+2 be denoted as θ. Its magnitude can be computed as a function of the respective position vectors of the above beads - $\vec{r}_i, \vec{r}_{i+1}, \vec{r}_{i+2}$.[172]

$$\theta=cos^{-1}\left(\frac{\vec{d}_i\cdot\vec{d}_{i+2}}{\left|\vec{d}_i\right|\left|\vec{d}_{i+2}\right|}\right) \tag{5.2.5}$$

where $\vec{d}_i=\vec{r}_i-\vec{r}_{i+1}$ and $\vec{d}_{i+2}=\vec{r}_{i+2}-\vec{r}_{i+1}$ as shown in Figure 5.2b.

5.2.2 Calculation of Reaction Times

Reactions at the molecular scale are stochastic in nature, being subject to fluctuations in temperature and pressure. In order to predict the time at which one will occur and its position in the chain of oscillators, we use Gillespie's First-Reaction Method.[173] With the help of the Arrhenius formula, zeroth-order rates for each bead (k_i) are calculated in terms of the activation energy (E_a) and the kinetic energy of the bead in question ($E_{k,i}$):

$$k_i\left(s^{-1}\right)=10^{13}\,exp\left(-\frac{E_a}{E_{k,i}}\right) \tag{5.3.1}$$

The time at which a reaction will occur at the i^{th} bead is obtained by sampling the governing probability distribution for a zero-th order reaction:[173]

$$P_i(\tau)\,d\tau=k_i\,exp(-k_i\tau)\,d\tau \tag{5.3.2}$$

The reaction times (τ) thus calculated for each bead are sorted in ascending order and the bead with the smallest time is chosen. The reaction location, μ, and time, τ_μ, can therefore be estimated together.

5.2.3 Reaction Acceptance Criterion

Once a possible reaction site has been located by Gillespie's First-Reaction Method[173,174], it remains to be seen whether it can be accepted or not. The metric used in our algorithm compares the computed reaction occurrence time (τ_μ) with the time scale of energy dissipation from an excited bead to the quiescent lattice. This implies that the reaction may be accepted if a bead remains in an excited state for a sufficiently long period. To estimate the dissipation time scale, a bead in the center of the chain was excited and its KE was tracked as a function of time. The energy correlation function was computed using equation 5.4 from this time series for the FPU-β and the φ^4 lattices (Figure 5.3).

$$C(\tau)=\frac{\sum_{i=1}^{T-\tau}\left(E_i-\bar{E}\right)\left(E_{i+\tau}-\bar{E}\right)}{\sum_{i=1}^{T}\left(E_i-\bar{E}\right)^2} \tag{5.4}$$

where T is the total number of time steps, τ is the correlation time, E_i is the kinetic energy of the excited bead at the i^{th} instant and $\bar{E}$ is the time-averaged KE of the bead.

Panel (i) shows that dissipation occurs within one oscillation period (i.e., t_{sc}) for the FPU lattice, which is the also the norm for harmonic chains.[69] Anharmonicities in the potential function cause the dissipation to occur over longer time scales,[68] which is obvious for the φ^4 lattice in panel (ii), where at least 3 lattice time steps are required to achieve dissipation. Using the information in Figure 5.3, we can execute a reaction if $\tau_\mu < t_{sc}$ (FPU-β or harmonic) or $\tau_\mu < 3t_{sc}$ (φ^4). Despite the greater leeway given to the φ^4 lattice as compared to the FPU-β, we shall see that it has a lower reactivity than the latter for the same energetic load.

When multiple energetic molecules are loaded on a single bead, the coarse-grained nature of the system prevents us from ascertaining the time at which each group will react. In other words, the computed reaction time – τ_μ – applies to the entire node-reactive group assembly. We introduce stochasticity by permitting n energetic molecules on a bead to react if $n\tau_\mu < mt_{sc}$, where m=1 or 3 depending on the lattice considered.

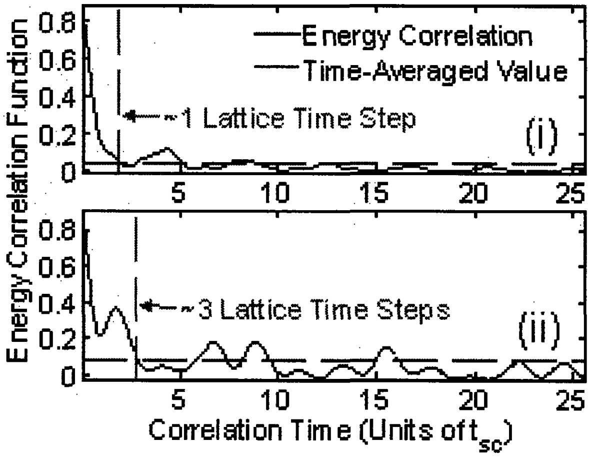


Figure 5.3. Estimation of the reaction acceptance criterion for (i) FPU lattice, (ii) φ^4 lattice: The 50th bead in the lattice is excited and the decay of the energy correlation function is observed. The bead attains a steady energetic level after 1 (3) oscillation period of the lattice for the FPU (φ^4) case. The reaction is accepted if the computed time required for it is less than t_{sc}.

5.2.4 Energy Conservation during Reaction

The chemical energy released during the reaction (ΔH) is deposited in the kinetic component of the local bead. The fraction, α, determines the amount of enthalpy that is converted to the KE of the reaction products, the remainder being lost as heat to the surroundings. Before a reaction at a bead j, the total energy (TE) of the entire lattice is:

$$TE = E_p + E_k - N\left(n_0 \Delta H\right) \tag{5.5.1}$$

where E_p and E_k are the total potential (PE) and kinetic (KE) energies of the lattice, N is the number of energetic molecules (EM), n_0 is the EM loading on each bead and ΔH (<0) is the enthalpy of reaction of a single EM; the last term gives the total energy stored in the reactive bonds. After n_r reactions have taken place at the location j, TE becomes

$$TE = E_p' + E_k' - \left(Nn_0 - n_r\right)\Delta H + Q \tag{5.5.2}$$

E_p' (E_k') is the new PE (KE) of the lattice, and Q is the heat lost to the surroundings. The part of the reaction enthalpy that has not been irreversibly lost has partitioned into the PE and KE of the affected bead. We shall now probe each term in the above equations for greater clarity.

a) We assume that the change in the lattice PE in the instant after the reaction is negligible because the change in conformation is highly localised. This is plausible when the number of beads is large. It therefore follows that $E'_p \approx E_p$.

b) The lattice KE prior to the reaction is given by $E_k = \frac{1}{2}\sum_{i=1}^{N}(m_b + n_0 m_e)u_i^2$, where m_b (m_e) is the mass of the bead (EM). For the purpose of this derivation, we know that the next n_r reactions will occur at j, so the KE can also be written as

$$E_k = \left[\frac{1}{2}\sum_{i \neq j}(m_b + n_0 m_e)u_i^2\right] + \left[\frac{1}{2}(m_b + n_0 m_e)u_j^2\right] \tag{5.5.3}$$

The first sum is over the beads that do not react, while the second term corresponds to the bead that does. After n_r reactions have occurred, the total KE of the lattice is

$$E'_k = \left[\frac{1}{2}\sum_{i \neq j}(m_b + n_0 m_e)u_i^2\right] + \left[\frac{1}{2}\left(m_b v_{b,j}^2 + n_r m_e v_e^2\right)\right] \tag{5.4.4}$$

The first sum extends over the unreacted beads with the assumption that their velocities are not altered immediately after the reaction. In other words, only the velocity of the bead j will change. The second term is a sum of the final KE of the reacted bead and the centre of mass of the decomposed EM. Since the summation terms cancel out, the change in the total KE of the lattice due to the reaction is

$$E'_k - E_k = \frac{1}{2}\left[m_b v_{b,j}^2 + n_r m_e v_e^2 - (m_b + n_0 m_e)u_j^2\right] \tag{5.5.5}$$

c) The amount of energy lost irreversibly to the surroundings as heat can be parameterised as a fraction of the reaction enthalpy. If we assume that a portion α of the energy released – $n_r\Delta H$ – is converted to useful work, $Q = (1-\alpha)n_r\Delta H$.

Equating the total energy before and after the reaction, we get

$$E_p + E_k + Nn_0\Delta H = E'_p + E'_k + (Nn_0 - n_r)\Delta H + Q \tag{5.5.6}$$

On substituting the expressions for each term that have been derived above along with the concomitant assumptions, the result of energy conservation at the j^{th} bead is:

$$2\alpha n_r\Delta H = m_b v_b^2 + n_r m_e v_e^2 - (m_b + n_0 m_e)u^2 \tag{5.5.7}$$

In the above equation, the first two terms represent the final kinetic energies of these species. We only consider the centre of mass (CM) of the latter set in the analysis to simplify matters. It follows that m_e (v_e) is the mass (velocity) of the CM of the reaction products of the molecule. The last term denotes the initial KE of the unreacted bead, where u is the velocity of the bead prior to the reaction. The RHS of the energy balance can be expressed in terms of the components of the kinetic energies in the x, y and z dimensions:

$$2\alpha n_r\Delta H = m_b\left(v_{bx}^2 + v_{by}^2 + v_{bz}^2\right) + n_r m_e\left(v_{ex}^2 + v_{ey}^2 + v_{ez}^2\right) - (m_b + n_0 m_e)\left(u_x^2 + u_y^2 + u_z^2\right) \tag{5.5.8}$$

Grouping the x, y and z terms separately, we get

$$2\alpha n_r\Delta H = \sum_{j=x,y,z}\left[m_b v_{b,j}^2 + n_r m_e v_{e,j}^2 - (m_b + n_0 m_e)u_j^2\right] = \sum_{j=x,y,z} E_{k,j} \tag{5.5.9}$$

where the RHS is simply the sum of KE along each of the 3 spatial dimensions.

The LHS in equation 5.5.7 is the total energy available for useful work after a reaction. It may be partitioned into the corresponding spatial contributions thus:

$$\varepsilon_j = \beta_j\left(2\alpha n_r\Delta H\right) \tag{5.5.10}$$

where β_j is the fraction of the reaction enthalpy available to dimension j ($j = x, y, z$). With the concepts developed above, the total energy balance can be split into dimension-wise portions with the appropriate velocity components.

$$\varepsilon_j = m_b v_{b,j}^2 + n_r m_e v_{e,j}^2 - \left(m_b + n_0 m_e\right) u_j^2 \tag{5.5.11}$$

During a given time step, all the quantities in equation 5.5.11 are known except for the final velocities of the reaction fragments – $v_{b,j}$ and $v_{e,j}$. The conservation of momentum in each spatial dimension ($j = x, y, z$) gives us a second relationship between the product velocities.

$$\left(m_b + n_0 m_e\right) u_j = m_b v_{b,j} + n_r m_e v_{e,j} \tag{5.6}$$

Equations 5.5.11 and 5.6 can be solved simultaneously for the unknown velocities of the reacted bead and the CM of the decomposed energetic molecule. The bead velocity after the reaction has occurred is:

$$v_{b,j} = u_j \pm \sqrt{\frac{\varepsilon_j}{\gamma\left(\gamma + 1\right)}} \tag{5.7}$$

where $\gamma = \dfrac{m_b}{n_r m_e} + \dfrac{n_0}{n_r}$. The choice of the sign is governed by the conservation of momentum. As an example, assume that the centre of mass of the decomposed energetic molecule moves in the $+x$ direction immediately after the reaction. If the unreacted bead initially moves along $+x$, its reaction product will experience a thrust in the opposite direction and the minus sign is used in equation 5.7. The converse holds if the unreacted bead initially moves along $-x$. The same rule applies for the y and z dimensions. In this way, the energy released during the reaction can be converted to the KE of the affected segment of the chain.

5.2.5 Evaluation of the Parameter β

The parameter β_j was introduced in the previous section to facilitate the energy conversion mechanism. An exact expression for it can be derived by assuming that the direction cosines, $\cos\theta_j = \frac{u_j}{|\bar{u}|}$, remain invariant in the instants preceding and succeeding the reaction. Here, θ_j is the angle made by the velocity vector, $\bar{u}$, with the j-axis, where j stands for either x, y or z. We suppose that during an infinitesimal time interval after the reaction, the centre of mass of the decomposed EM moves along the same trajectory as the unreacted bead prior to the reaction. It is evident from equation 5.7 that the final velocity components of the reacted bead are functions of ε_j, which itself depends on β_j (equation 5.5.10). We can therefore state that $v_{b,j} \sim \sqrt{\beta_j}$. Using the aforementioned assumption regarding the invariance of the direction cosines, we get

$$\frac{v_{b,j}}{|\bar{v}_b|} = \frac{u_j}{|\bar{u}|} = \cos\theta_j \tag{5.8.1}$$

Since each component of the final velocity vector (v_b) is known in terms of β_j, Eq 5.8.1 can be expressed as

$$\frac{\sqrt{\beta_j}}{\sqrt{\beta_x + \beta_y + \beta_z}} = \cos\theta_j \tag{5.8.2}$$

By definition, the denominator equals 1, and so,

$$\beta_j = \cos^2\theta_j \quad [j = x, y, z] \tag{5.8.3}$$

The direction cosines are obviously known before the reaction and hence the corresponding values of β can be calculated. Equation 5.8.3 has the following implication: if, for instance, the angle made by trajectory of the unreacted bead with the

x-axis is smaller than those made with the other two, there is a greater likelihood of the energy released augmenting the x-component of the reacted bead's velocity.

5.3 Molecular Dynamics Algorithm

Figure 5.4 shows the flow of logic behind the calculation of the properties of the reaction wave. The system properties – masses of the bead and the energetic molecules, load on each bead, force constants, simulation time – are established at the very beginning. The temperature of the system is raised to 300 K via the technique of velocity scaling.[175] The reaction wave is initiated by exciting the first bead in the chain. The derivative of the Hamiltonian of the system (equation 5.1.1) with respect to the coordinates of each bead gives the net force acting on it, which can be used in Newton's Second Law to determine the positions and velocities for the next time step. Before that is carried out, it is important to determine whether subsequent reactions are possible following the excitation at time $t = 0$. The temperature at each lattice node is found from its kinetic energy,[175,176] which is then used in estimating the local Arrhenius rate constant for a zero-th order reaction (equation 5.3.1). The predicted reaction times and positions are sorted in ascending order according to Gillespie's scheme[173,174] and the criterion tailored to the nature of the lattice is invoked to determine whether the reaction may be accepted or not.

In the case of a successful reaction, the appropriate portion of the enthalpy, as fixed by α is transferred to the kinetic energy of the bead using the energy conversion mechanism that has been described in sections 5.2.4-5. Newton's Second Law is finally used with the information from the reaction kernel to compute the new positions of each bead in the chain. Due to the probabilistic nature of the system, multiple runs of the code are necessary to generate statistically significant estimates for the reaction wave velocity and other characteristics.

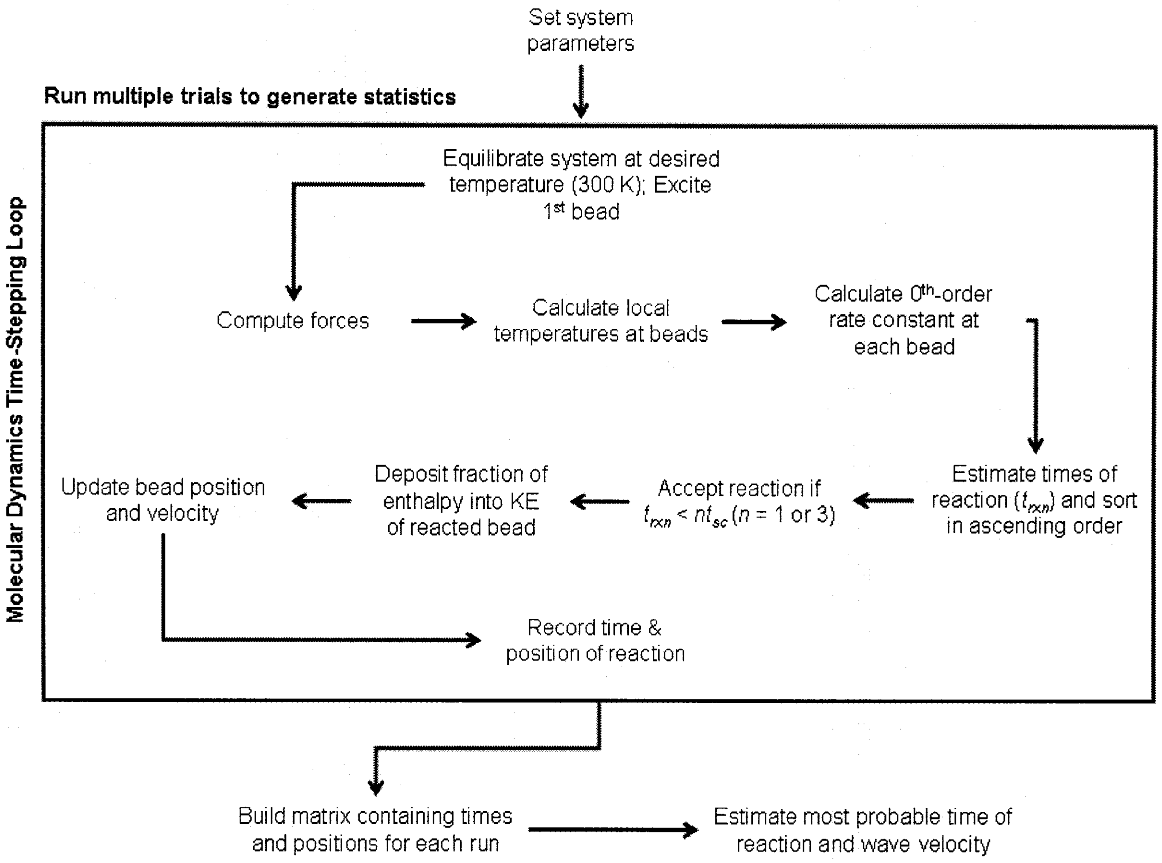


Figure 5.4. Flowsheet depicting the hybrid molecular dynamics/Monte Carlo algorithm used in estimating the properties of one-dimensional chain reactions along a nanostructure waveguide.

5.4 Thermal Conductivity of the Lattice

5.4.1 Constant Temperature Molecular Dynamics

The thermal conductivity of a system may be measured either by equilibrium or nonequilibrium molecular dynamics.[177] In both cases, some form of temperature control is needed: equilibrium MD (EMD) requires that the entire system be at a fixed temperature with minimum deviation from the set point; in non equilibrium MD (NEMD), hot and cold baths are placed at the two ends of the conductor and the heat current due to the ensuing temperature gradient is measured. The coupling of the system to an external bath was first achieved by the Andersen algorithm[178] for temperature and/or pressure control. Here, particles within the system were subjected to stochastic collisions with species in the bath according to a preset collision frequency, i.e., a coupling constant. If a certain particle has been selected to undergo a collision, its new velocity is then drawn from a Boltzmann distribution corresponding to the desired temperature, T. This technique is not suitable for the purpose of computing the thermal conductivity, since it leads to a sudden decorrelation of particle velocities and an artificial drop in the velocity autocorrelation function.[175] The Nosé-Hoover thermostat[179-181] is far more suitable for the intended application.

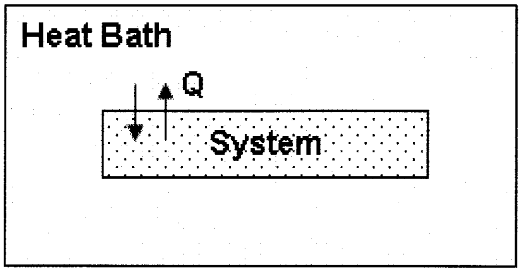


Figure 5.5. Schematic of the extended system – the heat bath and the system under study linked by the coupling parameter, Q.

According to the Nosé-Hoover scheme, the system is once again coupled to the external heat bath via a coupling constant Q (Figure 5.5). Unlike the Andersen thermostat, the interactions of the bath with the system are not random collisions but evolve deterministically in time as specified by an additional coordinate – s – and an extended Hamiltonian as developed by Nosé[179,180] and later simplified by Hoover with the introduction of the thermodynamic friction coefficient, ξ.[175,181]

$$H_{Nosé}=\sum_{i=1}^{N}\frac{p_i^2}{2m_i}+U\left(r_N\right)+\frac{\xi^2 Q}{2}+3Nk_BT\,ln\left(s\right)=H_0+\frac{\xi^2 Q}{2}+3Nk_BT\,ln\left(s\right) \qquad (5.9.1)$$

Equation 5.9.1 shows the addition of two terms to the traditional Hamiltonian, H_0, consisting of the total kinetic and potential energies of the system. The dynamic evolution of the artificial coordinate s is described by:

$$\frac{d\,ln\left(s\right)}{dt}=\xi \qquad (5.9.2)$$

where ξ is obtained by solving the ordinary differential equation

$$\frac{d\xi}{dt}=\frac{\sum_i \frac{p_i^2}{m_i}-3Nk_BT}{Q} \qquad (5.9.3)$$

Newton's Second Law for each component of the system is then modified in terms of the newly added connection to the external heat bath:

$$\dot{p}_i=-\frac{\partial U\left(r_N\right)}{\partial r_i}-\xi p_i \qquad (5.9.4)$$

The second term in the equation above acts to constrain the particle velocity according to its interaction with the heat bath. The system defined by $H_{Nosé}$ is a NVE ensemble, and the total energy in equation 5.9.1 is therefore conserved. On the contrary, H_0 represents a NVT ensemble where energy is allowed to flow across the system-bath interface. The dynamics of the extended system may be easily solved using a centered-difference algorithm first conceived by Holian et al.[182] The value of Q should be chosen carefully so as to avoid large fluctuations in the temperature. A small value of Q implies a low inertia of the heat bath. From equation 5.9.2-3, we see that a low Q-value leads to a larger rate of change of the bath variable s, which ultimately causes greater fluctuations in the velocity of the particle via equation 5.9.4.[175] The converse occurs for large values of Q. The optimum Q can be found by matching it to the characteristic frequency of the system,[183] or $1/t_{sc}$ from equation 5.1.2, where t_{sc} is the lattice time scale.

5.4.2 Calculation of Thermal Conductivity

The thermal conductivity (κ) of a one-dimensional lattice may be found using either equilibrium or nonequilibrium molecular dynamics. Both routes require the calculation of the heat current in the axial direction (J_z), i.e., along the path of thermal conduction.

$$J_z(t) = \sum_{i=1}^{N} \left[v_{z,i} E_i + \frac{1}{2} \sum_{j \neq i} r_{ij} \left(\vec{F}_{ij} \cdot \vec{v}_i \right) \right] \tag{5.10}$$

where $v_{z,i}$ is the z-velocity of the i^{th} bead, E_i is its total energy, r_{ij} is the distance between it and its nearest neighbours, and F_{ij} is the force of interaction between them.[177,184,185] The essence of equation 5.10 is the transfer of energy as heat through the motion of the bead (captured in the first term) and its interactions with the immediate neighbours (encapsulated in the second term). The EMD path of the estimation of κ involves the use of the Green-Kubo formalism.[177,184,185]

$$\kappa = \frac{1}{V k_B T^2} \int_0^{\infty} \langle J_z(0) J_z(t) \rangle \, dt \tag{5.11}$$

In the above equation, V is the volume of the system and T is its temperature, which explains the need for the thermostat outlined previously. The integrand is the heat current autocorrelation function (HCACF). Equation 5.11 is derived from linear response theory and uses the fluctuation-dissipation theorem, according to which the effect of an external field on a system may be linked to the decay of fluctuations to an equilibrium state.[186]

An alternative, and conceptually easier, approach to compute κ would be to use NEMD by establishing a temperature gradient across the 1D lattice and using Fourier's Law.[185,187-189]

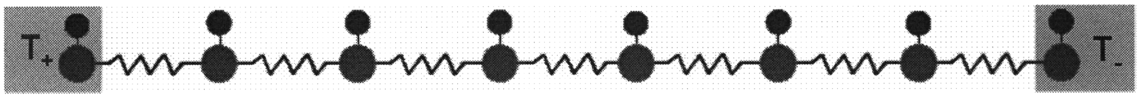


Figure 5.6. Setup to calculate the thermal conductivity using nonequilibrium molecular dynamics. The first and last beads are connected to hot (T_+) and cold baths (T_-), respectively.

The first step is to connect the first and last beads in the chain to hot and cold temperature baths respectively (Figure 5.6). The Nosé-Hoover equations are applied to just these beads with a simple modification to equations 5.9.3-4:

$$\frac{d\xi_{+,-}}{dt}=\frac{\frac{p_{1,N}^{2}}{m_{1,N}}-3k_BT_{+,-}}{Q} \tag{5.12.1}$$

$$\dot{p}_{1,N}=-\frac{\partial U\left(r_N\right)}{\partial r_{1,N}}-\xi_{+,-}p_{1,N} \tag{5.12.2}$$

with $\xi_{+,-}$ equal to zero for all the interior beads.

The establishment of a steady temperature gradient across the lattice allows us to calculate the axial heat current using equation 5.10. The thermal conductivity is then given by Fourier's Law:[177]

$$\kappa V=-\frac{J_z}{dT/dz} \tag{5.13}$$

5.5 Results and Discussion

5.5.1 Parametric Study of Reactions

The number of beads in the chain (*N*) was fixed at 100. We perform the analysis for loads of 1-4 reactive groups/bead. The heat of decomposition of TNT (2732.48 kJ/kg)[190] was used as a representative value for ΔH in the simulation. The lattice, initially equilibrated at 300 K, was allowed to evolve in time over a period covering ~7ps.

The first step was to ascertain the stability of the nanomaterial as a function of the system parameters E_a and α. After equilibration, the system was allowed to evolve in time and reaction events were recorded. The average conversions from 100 trials have been plotted in Figures 5.7a-d for different loads as a function of the 2 parameters. It is observed that there is a threshold of stability with some spontaneous reactions occurring when E_a<25 kcal/mol. Approximately 80% of the reactive groups present react without an external stimulus for E_a<20 kcal/mol; the obvious conclusion is that at low activation barriers, the reaction sites are energetic enough to obviate the need for a prior excitation. Most of the species react haphazardly at room temperature itself thereby making the assembly inherently unstable. This test establishes a lower limit for the selection of molecules in the design of such structures.

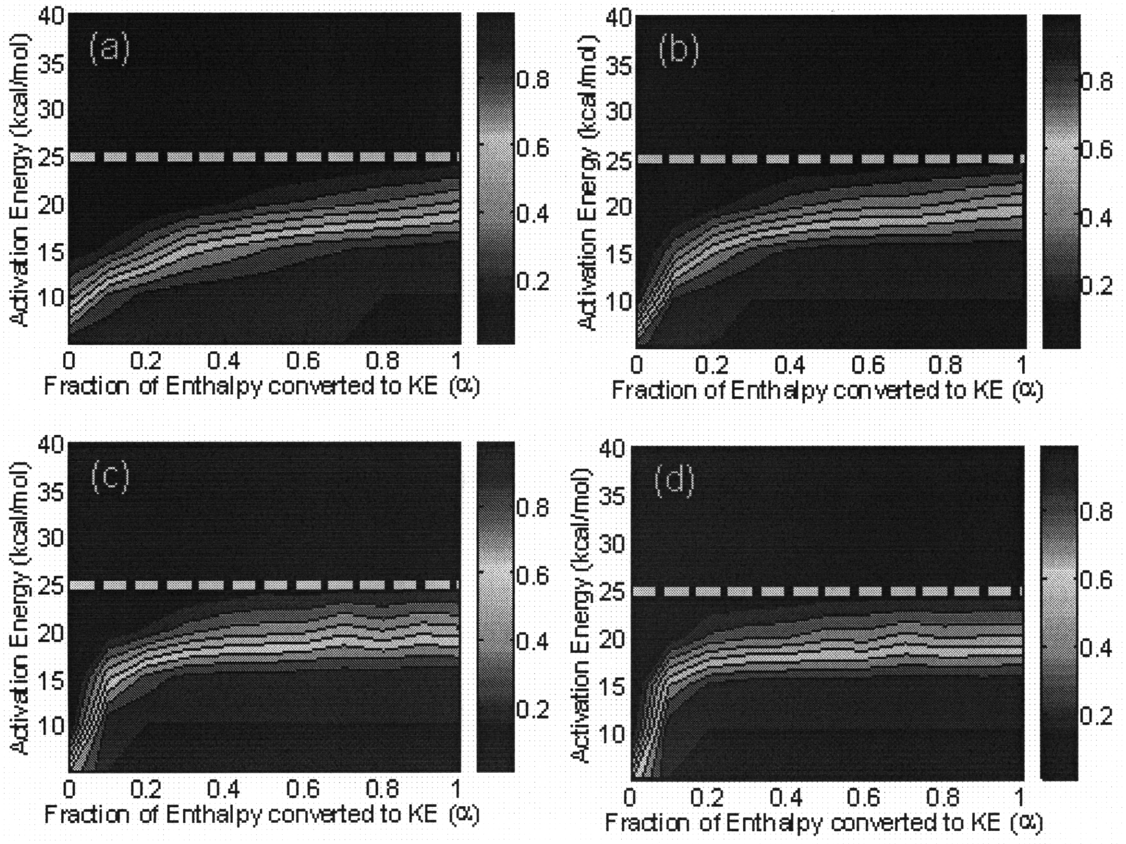


Figure 5.7. Stability plots for 4 loadings: (a) 1 EM/bead, (b) 2 EM/bead, (c) 3 EM/bead, and (d) 4 EM/bead. Each system was equilibrated at 300 K and allowed to evolve in time without any initial excitation. Average conversions have been plotted as a function of E_a and α. The nanostructures seem to be unstable for $E_a < 25$ kcal/mol (yellow dotted line), since room temperature is probably sufficient to set off the reactions.

The effect of an excitation at time $t = 0$ was studied by permitting all the reactive groups at the first node to react unconditionally and tracking the progress of the reaction wave with time. Figure 5.8 shows the total conversion as a function of the system parameters for (a)-(c) harmonic, FPU and φ^4 lattices at 3 reactive groups/bead, and (d)-(f) the respective lattices at 4 reactive groups/bead. While the conversion plots for the harmonic and FPU systems are almost identical for the loads used, the φ^4 lattice exhibits a comparatively lower conversion at each value of α. This can be attributed to the poor thermal conduction characteristics of the hard φ^4 potential.[68] Nonlinear 1D lattices can support a temperature gradient, as has been proved by Non-Equilibrium Molecular

Dynamics simulations.[67,189] The approximate thermal conductivity of each chain can be found by calculating the ratio of the time-averaged heat current to the applied temperature gradient.[185,187-189] The thermal conductivity of the φ^4 lattice in our study is lower than that of the FPU chain by a factor of ~14 (Appendix C, Fig. C1). The difference between the harmonic and FPU lattices is minimal due to the relatively small displacement of the beads from their equilibrium positions. As a result, the quartic term does not play a significant role under the current energetic loadings. The use of a considerably higher reaction enthalpy will be explored later to elicit the difference in the speed of the reaction wave.

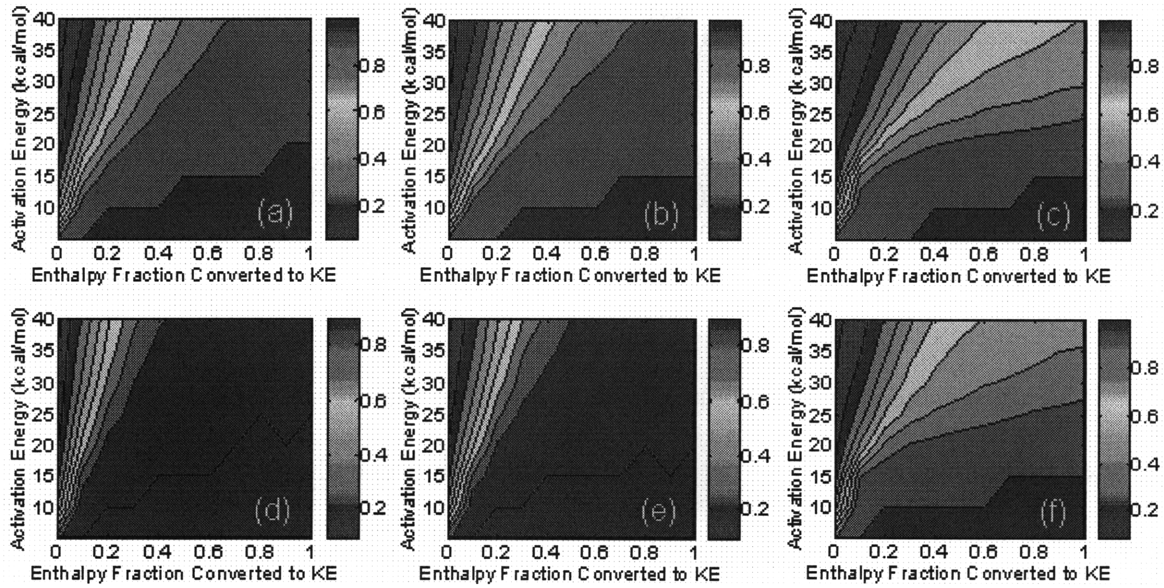


Figure 5.8. Conversion plots for (a) harmonic, (b) FPU and (c) φ^4 lattices at 3 EM/bead; (d)-(f) conversion plots for the respective cases at 4 EM/bead after equilibration at 300 K and allowing the first bead to react unconditionally, thus releasing energy for subsequent reactions. While the harmonic and FPU lattices have similar degrees of conversion, the φ^4 lattice lags behind them at both loads.

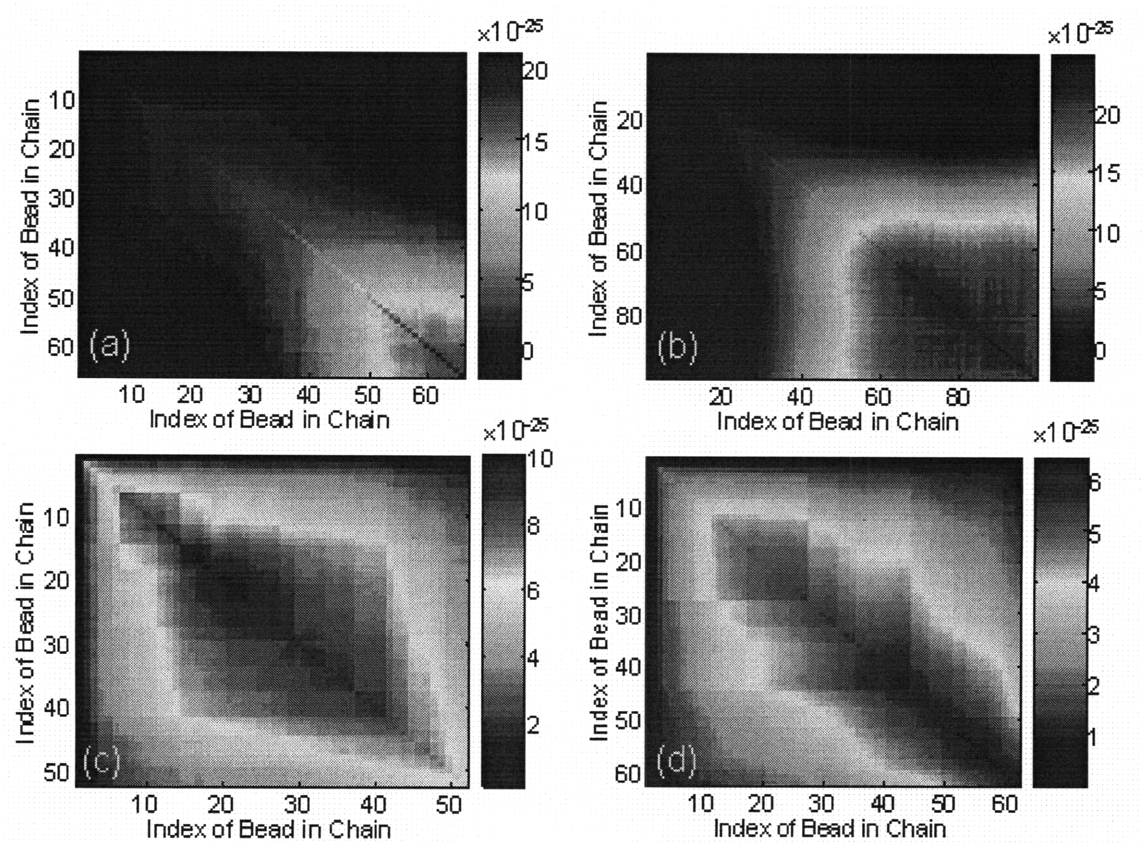


Figure 5.9. Covariance maps in units of sec^2 for the (a)-(b) harmonic, and (c)-(d) φ^4 lattices with loads of (a),(c) 3 EM/bead, and (b), (d) 4 EM/bead (E_a=35 kcal/mol, α=0.7). Contrary to the harmonic chain, the uncertainty in reaction times decreases with increasing reactive group loading for the φ^4 lattice.

The covariance maps in Figure 5.9 show the effect of a reaction event at position *i* on sites further along the chain as a function of loading and the post-reaction energy available to the lattice. The steadily increasing pixel intensities down the length of the chain – along the main diagonal – indicate progressively greater correlations between neighbouring reactive sites. For the harmonic lattice at loads of 3 and 4 reactive groups/bead, we see in Figures 5.9a-b, respectively, that the variance in reaction times increases with loading. This is because an energetic pulse undergoes dispersion while traversing a harmonic chain.[166,167] Consequently, the reaction front progressively broadens and subjects the beads ahead to constant excitation, thus enabling them to react before the front has passed them by. Identical reaction conditions for the φ^4 lattice present a contrary set of results (Figures 5.9c-d) wherein the variance in reaction times decreases with increasing loads and is also lower than the corresponding values for the harmonic system. In order to explain this phenomenon, we examined the change in the phonon density of states (DOS) of the φ^4 lattice with reactions at loads of 3-7 and 12 reactive groups/bead (Appendix C, Fig. C2). A band gap partitions the spectrum into low- and high-frequency regions. The ratio of the area under the high-frequency portion to the total spectral area was recorded as a function of loading of the energetic molecules. The total contribution of the higher modes is directly proportional to the reactive group loading. In the next sub-section, we shall demonstrate that only signals with frequencies lying in the higher end of the spectrum have a finite probability of traveling through the φ^4 lattice. Therefore, energetic reactions involving a sparser coverage of reactive groups excite the low-frequency modes to a greater extent, which have a limited capacity to propagate in the φ^4 chain. This decreases the efficiency of information transfer in the lattice and consequently raises the uncertainty in reaction times.

5.5.2 Fourier Analysis of Lattice KE

In order to understand why the conversion in the φ^4 lattice is less than the FPU and harmonic cases, it is important to analyse the DOS of each lattice (Figure 5.10a, blue) by taking the Fourier transform of the lattice KE.[191] Additionally, we have mapped out the range of frequencies that support signal transduction in each lattice (Figure 5.10a, green) for a load of 3 reactive groups/bead. This range was determined by a technique similar to that used by Bowman et al. to obtain normal modes without a Hessian using Driven Molecular Dynamics.[192]

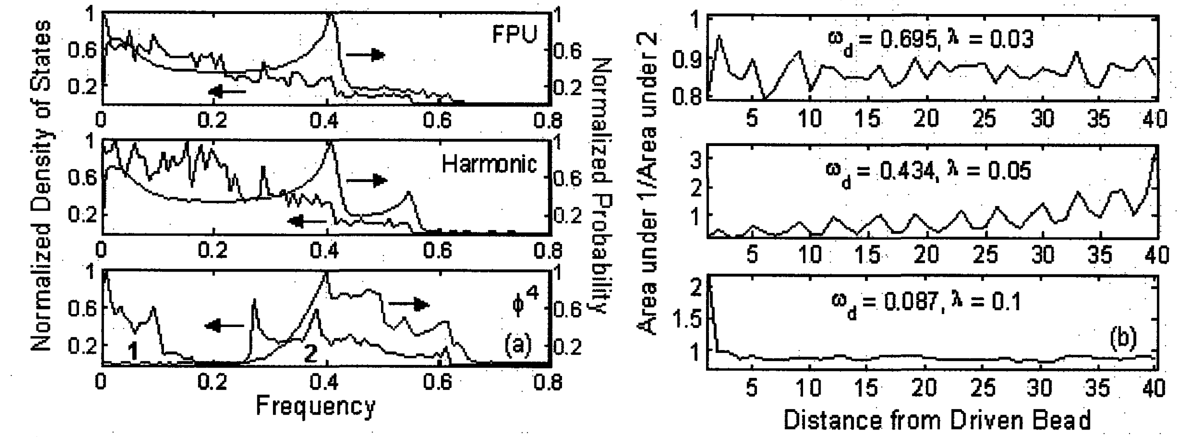


Figure 5.10. (a) Superposition of the DOS (blue) for the 3 lattices with the probability (green) of a signal applied at the 50th bead being propagated down the chain. Signals spanning the permitted frequency range have a finite probability of penetrating the FPU and harmonic lattices. Only high-frequency signals will propagate in the φ^4 case. (b) Ratio of the contribution of lower modes (area 1 in 5.10a) to their higher counterparts (area 2 in 5.10a) at different driving frequencies (ω_d) and their respective optimal amplitudes (λ). Energy transfer from higher to lower modes is evident in the central panel, which corresponds to a signal that has a high likelihood of propagation.

The 50th bead in the chain was driven at various frequencies and the total kinetic energy of 40 beads on either side of the point of application of the signal was recorded. It is clear that viable propagation is possible at all the permitted frequencies ($0 \leq \omega \leq 0.6$) for the FPU and harmonic chains; however, only high-frequency signals ($0.3 \leq \omega \leq 0.65$) have non-zero probabilities of traveling through the φ^4 lattice. The effects of specific driving frequencies (ω_d) are depicted in Figure 5.10b for the φ^4 chain. Each frequency, ω_d, has an optimal amplitude, λ,[192] (Appendix C, Fig. C3) whose value is also shown in each panel. The ratio of the areas under the regions labeled "1" and "2" in the bottom panel of Figure 5.10a has been plotted against the distance from the 50th bead in Figure 5.10b. This ratio is a measure of the relative contributions of the low- and high-frequency modes to signal propagation. The top and bottom panels in Figure 5.10b show a value close to 1 for the ratio, which remains almost constant with distance from the driven bead, thus implying that no energy transfer occurs from the higher to the lower modes. This is not surprising, since the corresponding ω_d lie in the forbidden range (i.e., $\omega_d<0.3$ and $\omega_d>0.65$). The central panel uses a driving frequency that lies in the permitted range; despite the noise, a distinct increase is observed in the ratio with distance, indicating that the role of the lower modes gradually grows in importance as one moves away from the driven bead. A transfer of energy takes place from the higher modes to the lower modes. In other words, the allowed high-frequency phonons from the point of excitation disintegrate into their low-frequency counterparts, which are responsible for the successful propagation of the wave.

The 50th bead in a φ^4 chain was energetically excited and its DOS has been plotted in Figure 5.11 along with those of its immediate neighbours. The enhanced energy state of node 50 raises the phonon population in the low- and high-frequency regions of the spectrum, including a spillover into the band gap. From the results presented in Figures 5.10a-b, it is now certain that the increased DOS among frequencies less than $\omega=0.3$ do not aid in energy propagation. When compared to the FPU and harmonic lattices where all frequencies are allowed, only the upper half of the spectral range in the φ^4 chain is capable of transmitting signals. Oscillations with $\omega>0.65$ become highly localised and possibly lead to the formation of breathers. This reduced capacity of the φ^4 chain to support signals of all permitted frequencies explains the poor conduction, and hence, lower chemical conversion when compared to the other 2 systems.

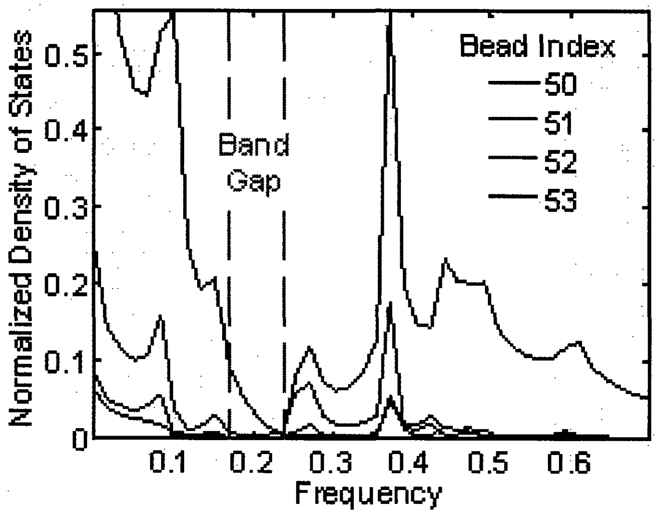


Figure 5.11. DOS of 4 beads after the energetic excitation of the 50^{th} node. Although the population of the lower modes (ω<0.25) of the latter have been raised, their contribution to propagation is small.

It is also of interest to determine whether a sustained reaction wave is facilitated by the activation of certain modes of oscillation in the harmonic lattice. Figure 5.12a shows the power spectra for 0 (orange) and 3 (blue) reactive groups/bead calculated at 300 K. The absence of modes in the frequency range 0.55< ω<0.66 for the loaded chain (inset) is due to the fact that its component beads are heavier than the unloaded case, and the oscillation frequency is inversely proportional to mass. A lattice undergoing chain reactions will have a mix of completely reacted (i.e., lighter) and unreacted (i.e., heavier) beads. The above frequency range will play an important role in these circumstances especially since a rise in temperature increases the phonon population at the band-edge (Figure 5.12b). Additionally, the central panel in Figure 5.10a shows that signals with frequencies $\omega \leq 0.4$ have a greater likelihood of penetrating a harmonic lattice loaded with 3 reactive groups/bead. It is evident that energy propagation in this lattice is carried out by the low-frequency portion of the spectrum. Modes above ω=0.57 do not contribute to propagation at all but we will show that they are capable of inducing reactions.

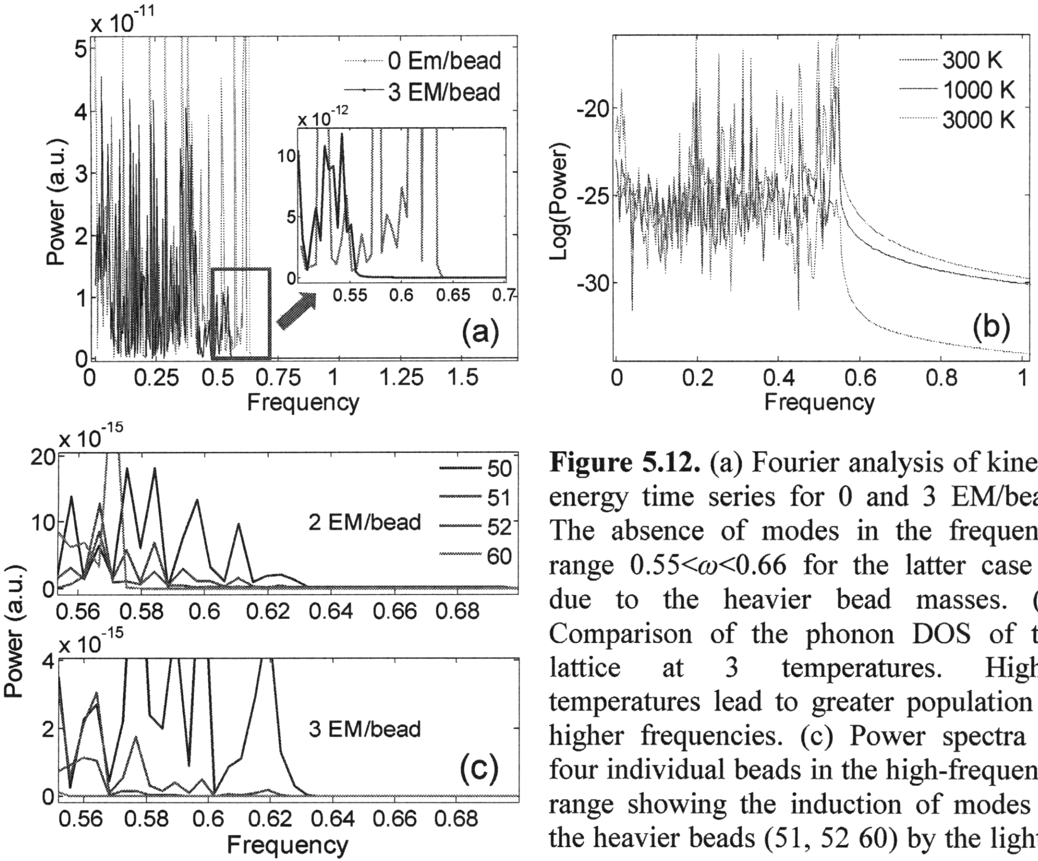


Figure 5.12. (a) Fourier analysis of kinetic energy time series for 0 and 3 EM/bead. The absence of modes in the frequency range $0.55<\omega<0.66$ for the latter case is due to the heavier bead masses. (b) Comparison of the phonon DOS of the lattice at 3 temperatures. Higher temperatures lead to greater population at higher frequencies. (c) Power spectra of four individual beads in the high-frequency range showing the induction of modes in the heavier beads (51, 52 60) by the lighter one (50).

The results in Figure 5.12c were computed by partitioning the 100-bead chain into 2 halves: beads 1-50 had no reactive groups loaded, and beads 51-100 were equally loaded with a finite number of molecules. The chain was thus split into light and heavy segments, the former being akin to the portion that lies behind the reaction front. The kinetic energies of each bead were recorded over 2048 time steps for numerical convenience and averaged over 100 independent runs. No reactions were executed during this timeframe, since the object was to determine the coupling between beads of different masses at the interface. The power spectra in Figure 5.12c correspond to two reactive group loads and have been plotted for a light bead (50) and 3 heavy beads (51, 52, 60). In the cases of 2 and 3 reactive groups/bead, it is evident that the coupling between the light and heavy beads is conducive to the appearance of modes beyond the band-edge (i.e., ω>0.57) of the latter. The effect is most prominent for the heavy bead that is directly connected to the lighter one, and progressively decays further into the loaded portion of the chain. We postulate that the presence of these localised high-frequency modes in the heavier – or unreacted – beads causes them to react as soon as the preceding bead has reacted. A comparison of the two panels in Figure 5.12c also shows that it is easier to induce higher modes in the case of lower EM loads (i.e., 2 versus 3 reactive groups/bead) simply because of the difference in bead mass, and hence, inertia.

5.5.3 Reaction Velocity Calculation

It is possible to extract the velocity of the reaction wave and the associated confidence intervals from these simulated datasets in the specific case of directional energy transfer. Reaction events with probabilities of occurrence below 0.90 were neglected during this calculation. Figures 5.13a-c show the velocity calculations for 3 reactive groups/bead for the φ^4, FPU and harmonic lattices, respectively, with E_a=35 kcal/mol and α=0.7. The most probable reaction times at *n* beads can be easily fit with a straight line, the reciprocal slope of which is the velocity of the reaction wave. In each plot, the green and red lines represent fits to the reaction times at beads 1-3 and 4-13, respectively. This was necessitated by the presence of a distinct start-up zone for the Φ^4 lattice (Figure 5.13a), where the velocity computed from the first 3 beads (2.72 km/s) is a third of the bulk velocity computed from the next 10 beads. The FPU and harmonic lattices show a comparatively negligible difference in velocities between the 2 zones (Figures 5.13b,c). The contrast between the initial and bulk velocities is clearly presented in Figure 5.13d for two different loads of the energetic material. The φ^4 velocities

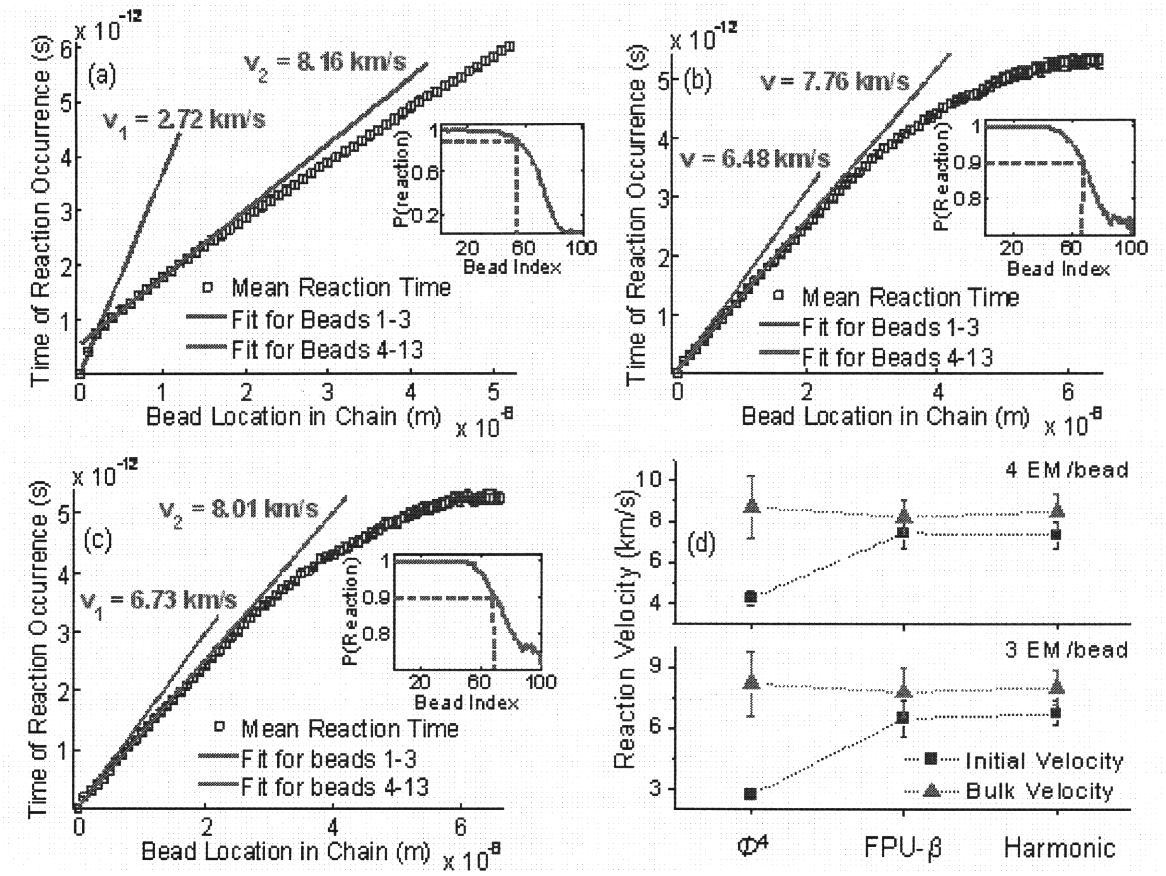


Figure 5.13. Comparison of velocities for (a) φ^4, (b) FPU and (c) harmonic lattices for 3 EM/bead (E_a=35 kcal/mol, α=0.7). A distinct start-up regime is observed in (a). An analysis of the FPU and harmonic cases indicates that a similar start-up zone is almost negligible; indeed, almost similar velocities are predicted in each situation, which is depicted most clearly in (d) for each lattice at two different loadings. The insets in (a)-(c) show the probability cutoffs (0.90) that decided the number of viable reactions.

predicted from beads 4-13 lie in the same range as the corresponding values for the other lattices within the 95% confidence intervals;[87,88] however, the reaction wave definitely undergoes an activation phase, which depends on the energetic load used. Furthermore, the difference between the two zones in the φ^4 lattice decreases as the loading is increased from 4-7 reactive groups/bead (Appendix C, Fig. C4).

The predicted reaction times in the FPU and harmonic chains deviate considerably from the linear fit (Figures 5.13b-c, red lines) as the distance from the ignition point increases. This observation points towards an increase in the wave velocity due to the energy injected into the lattice at each reacted node. The φ^4 chain, on the other hand, adheres to the linear reaction trajectory to a greater extent (Figure 5.13a) even at higher energetic loads (Appendix C, Fig. C4). We have already seen that the variance in reaction times is less than the FPU and harmonic cases. The φ^4 lattice therefore offers a greater degree of control over the reaction trend although the overall conversion lags behind the other systems.

The reaction characteristics of the FPU-β and harmonic lattices are almost identical for the 3 EM/bead (Figure 5.13b,c) and 4 EM/bead[25] scenarios. Previous work by Sarmiento et al.[166] has proved that pulse propagation in a hard anharmonic lattice occurs at speeds that exceed those in the harmonic variant. Since the energetic loads used thus far have been unable to distinguish between the two lattices, we increased the reaction enthalpy by a factor of 20 while keeping the load at 2 groups/bead. The corresponding velocity calculations are displayed in Figures 5.14a-b. Linear fits using the first 10 reaction times indeed show that the FPU-β has a marginally larger velocity (14.22 ± 0.28 km/s) than the harmonic chain (12.88 ± 0.57 km/s), thus showing that large deformations are required to distinguish between the 2 lattices.

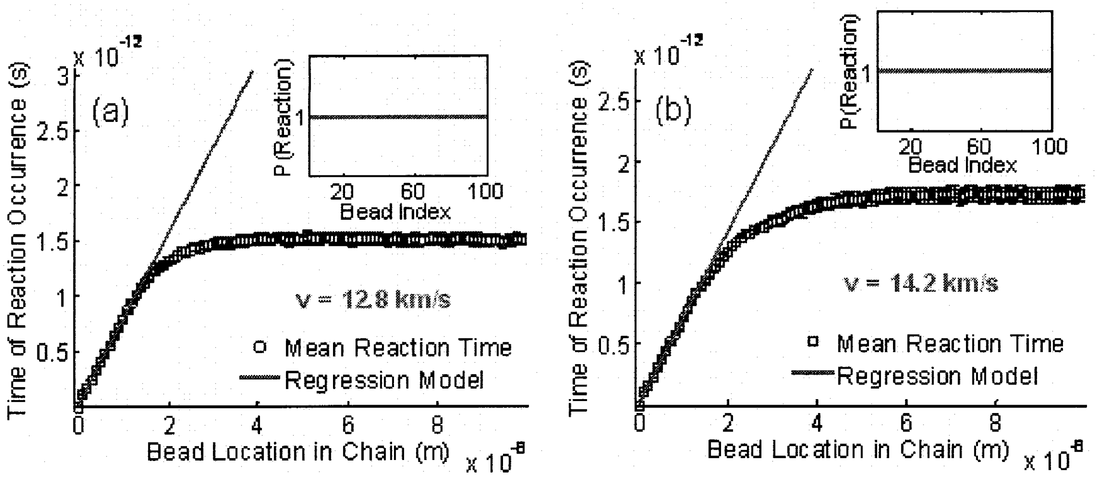


Figure 5.14. The reaction velocities for the (a) harmonic and (b) FPU lattices have been calculated for 2 EM/bead (E_a=35 kcal/mol, α=0.7) but with 20 times the reaction enthalpy as in the standard cases. As has been shown in the literature, increased energy input leads to greater wave velocities in hard potentials – (a) – compared to the harmonic – (b). The inset in each plot shows the variation in reaction probability with bead location.

5.5.4 Effect of Defects

Defects were introduced in the chain of oscillators by lowering the bond-stretch and bending force constants by a factor of 50 at select locations: 320 and 650 Å (Figures 5.15a-b. The simulations were run for the 3 reactive groups/bead case and can therefore be compared to Figure 5.8a. The contour plot in Figures 5.15a shows a distinct reduction in conversion with the presence of defects in comparison to the pristine chains in Figure 5.8a. It is interesting to note that the average conversion at E_a=40 kcal/mol is around 70% even at α=1, whereas it was ~1 for the non-defective chains in Figure 5.8a under similar circumstances. In addition, Figure 5.15b shows that the very first defect in both cases drastically reduces the probability of reactions further down the length of the chain (inset). The reaction wave is disrupted by the defect at 320 Å and all subsequent reactions shown in light blue in the primary plot are comparatively rare events. The use of free boundary conditions at both ends of the chain results in more energetic beads at the extremes as compared to the central portion of the lattice (Appendix C, Fig. C5) and sometimes reactions at the end of the chain appear more likely than those in the center. Thus, defects may serve as a practical consideration making it difficult to experimentally realize these chain reactions in practice.

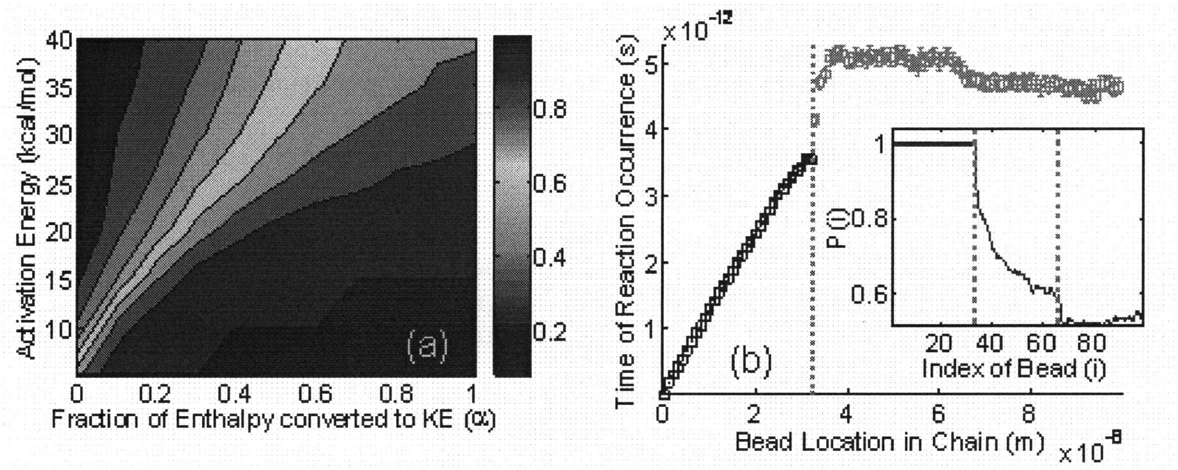


Figure 5.15. Conversion plot (3 EM/bead) with the introduction of 2 defects along the length of the chain. Comparing the contour plot to Fig. 5.8a, a distinct reduction in reactivity is observed due to poor thermal transport. (b) The path of the reaction wave is impeded at the very first defect (orange dashed line), with the subsequent beads (light blue) having drastically reduced probabilities of reacting, as depicted in the inset.

6. Conclusions and Future Directions

In this work, we have analysed the interactions – covalent and non-covalent – of single walled carbon nanotubes (SWNT) with diazonium groups and surfactants, and their utility with respect to metal/semiconductor separation. A structure-reactivity relationship for reactions of HiPco SWNT with 4-hydroxybenzene diazonium salt has been presented. Subsequent deconvolution of the reaction spectra yielded surface coverage of the diazonium on the nanotubes as a function of the total molar amount of diazonium added. A generic algorithm was developed to fit SDS-suspended HiPco SWNT, and DNA-suspended HiPco SWNT. The spectral profile of each nanotube has been approximated as a Voigt lineshape with peak widths that were estimated from the fits of DNA-suspended HiPco SWNT. Due to the clustering of nanotubes with similar transition energies, a weighting scheme was devised to apportion the contributions to the parent spectral peak. This scheme does not account for the true absorption cross section of different SWNT. It is therefore entirely possible that the spectral contributions of some nanotubes are either overestimated or not accounted for completely. The deconvolution procedure is a starting attempt at trying to understand the absorption spectrum of monodispersed nanotubes and obtaining quantitative information related to nanotube reactions. Peak widths that do not vary with nanotube chirality adversely affect the fit. It would be instructive to derive the exact dependence of the full width at half maximum for different nanotubes either via theory or experiment.

A rate model for the steady state data that considered the adsorption step as rate-limiting was used to extract rate constants normalised to the (11,5) nanotube. Along with values predicted from electron transfer theories, the metal > large-semiconductor > small-semiconductor reactivity trend has been explained. The non-covalent adsorption was assumed to be the rate-limiting step, which could explain the relatively poor fits of the surface coverage data for the large-diameter semiconductors. We are only able to estimate the relative rate constants for a solution containing a large variety of nanotubes. The use of highly pure samples containing a single chiral entity would yield absolute rate constants that would have greater utility while designing experiments. The extension of the model to other diazonium-based functional groups would enable the creation of a structure-reactivity map that would have an additional utility vis-à-vis density-based separation.

A hydrodynamic model to describe the separation of surfactant-suspended single walled carbon nanotubes using centrifugation has been developed. Parameter estimates for the number of sodium cholate molecules adsorbed per unit nanotube length were obtained by fitting data in the literature as well as our own. The model has veen validated by its application to the separation of reacted nanotubes from unreacted ones. However, the analysis of concentration profiles with our mathematical formulation does not adequately distinguish between surfactant adsorption on nanotubes of different diameters. The use of surfactant mixtures for metal/semiconductor separation is still not entirely understood and more fundamental work needs to be carried out in this area.

The last part of this thesis dealt with a conceptual system, the experimental realisation of which is an ongoing research area. We have performed a parametric study of energetic chain reactions propagating through harmonic and anharmonic lattices. The mean conversion for a 100-node lattice was investigated as a function of these handles. At loads of 3-4 energetic molecules/bead (E_a=35 kcal/mol, α=0.7), the FPU and harmonic lattices behaved similarly with reaction velocities ranging between 8-8.5 km/sec, which are lower than the speed of sound in the lattice(~14 km/s). The φ^4 lattice exhibited lower conversions along with the formation of a start-up zone where the reaction wave velocity was at least half of the bulk value at the aforementioned loads. Fourier analyses of the kinetic energy traces of the φ^4 lattice revealed that only high-frequency (i.e., high energy) excitations led to viable wave propagation, which explains the prominence of the start-up zone at lower loadings of the energetic molecules. The bead-spring model is a one-dimensional analog for a carbon nanotube but suffers from the main drawback of highly coarse-grained systems: the absence of realistic vibrational modes. A more extensive theoretical study could use the same principles in the atomistic case to uncover the true flavor of this nanostructure.

7. Bibliography

1 S. Iijima, Nature **354**, 56 (1991).

2 M. S. Dresselhaus, G. Dresselhaus, and P. C. Eklund, *Science of Fullerences and Carbon Nanotubes*. (Academic Press, San Diego, 1996).

3 R. Saito, G. Dresselhaus, and M. S. Dresselhaus, *Physical Properties of Carbon Nanotubes*. (Imperial College Press, London, 1998).

4 S. Reich, C. Thomsen, and J. Maultzsch, *Carbon Nanotubes*. (Wiley-VCH, Weinheim, 2004).

5 P. Barone, S. Baik, D. A. Heller, and M. S. Strano, Nat. Mater. **4**, 86 (2005).

6 D. A. Heller, S. Baik, T. E. Eurell, and M. S. Strano, Adv. Mater. **17**, 2793 (2005).

7 D. A. Heller, E. S. Jeng, T. K. Yeung, B. M. Martinez, A. E. Moll, J. B. Gastala, and M. S. Strano, Science **311**, 508 (2006).

8 E. S. Jeng, A. E. Moll, A. C. Roy, J. B. Gastala, and M. S. Strano, Nano Lett. **6** (3), 371 (2006).

9 C. Y. Lee, S. Baik, J. Zhang, R. I. Masel, and M. S. Strano, J. Phys. Chem. B **110**, 11055 (2006).

10 C. Y. Lee and M. S. Strano, Langmuir **21**, 5192 (2005).

11 M. S. Dresselhaus, G. Dresselhaus, and P. C. Eklund, *Science of Fullerenes and Carbon Nanotubes*. (Academic Press, San Diego, 1996).

12 P. Nikolaev, M. J. Bronikowski, R. K. Bradley, F. Rohmund, D. T. Colbert, K. A. Smith, and R. E. Smalley, Chem. Phys. Lett. **313**, 91 (1999).

13 C. Journet, W. K. Maser, P. Bernier, and A. Loiseau, Nature **388**, 756 (1997).

14 T. Guo, P. Nikolaev, A. Thess, D. T. Colbert, and R. E. Smalley, Chem. Phys. Lett. **243**, 49 (1995).

15 B. Kitiyanan, W. E. Alvarez, J. H. Harwell, and D. E. Resasco, J. Catal. **204**, 129 (2001).

16 A. Cassell, J. Raymakers, J. Kong, and H. Dai, J. Phys. Chem. B. **103**, 6484 (1999).

17 M. S. Strano, C. A. Dyke, M. L. Usrey, P. W. Barone, M. J. Allen, H. W. Shan, C. Kittrell, R. H. Hauge, J. M. Tour, and R. E. Smalley, Science **301** (5639), 1519 (2003).

18 M. S. Arnold, A. A. Green, J. F. Hulvat, S. I. Stupp, and M. C. Hersam, Nat. Nanotechnol. **1**, 60 (2006).

19 J. Lu, S. Nagase, X. Zhang, D. Wang, M. Ni, Y. Maeda, T. Wakahara, T. Nakahodo, T. Tsuchiya, T. Akasaka, Z. Gao, D. Yu, H. Ye, W. N. Mei, and Y. Zhou, J. Am. Chem. Soc. **128**, 5114 (2006).

20 J. Zhao and J. P. Lu, Appl. Phys. Lett. **82**, 3746 (2003).

21 J. Zhao, H. Park, J. Han, and J. Lu, J. Phys. Chem. B **108**, 4227 (2004).

22 H. Park, J. Zhao, and J. P. Lu, Nanotechnology **16**, 635 (2005).

23 C. Kittel, *Introduction to Solid State Physics*, 7 ed. (John Wiley and Sons, New York, 1996).

24 J. W. Mintmire and C. T. White, Phys. Rev. Lett. **81**, 2506 (1998).

25 S. M. Bachilo, M. S. Strano, C. Kittrell, R. H. Hauge, R. E. Smalley, and R. B. Weisman, Science **298** (5602), 2361 (2002).

26 A. Jorio, A. P. Santos, H. B. Ribeiro, C. Fantini, M. Souza, J. P. M. Vieira, A. Furtado, J. Jiang, R. Saito, L. Balzano, D. E. Resasco, and M. A. Pimenta, Phys. Rev. B. **72**, 075207 (2005).

27 A. Jorio, C. Fantini, M. A. Pimenta, D. A. Heller, M. S. Strano, M. S. Dresselhaus, Y. Oyama, J. Jiang, and R. Saito, Appl. Phys. Lett. **88**, 023109 (2006).

28 O. Jost, A. A. Gorbunov, W. Pompe, T. Pichler, R. Friedlein, M. Knupfer, M. Reibold, H. D. Bauer, L. Dunsch, M. S. Golden, and J. Fink, Appl. Phys. Lett. **75** (15), 2217 (1999).

29 X. Liu, T. Pichler, M. Knupfer, M. S. Golden, J. Fink, H. Kataura, and Y. Achiba, Phys. Rev. B. **66** (4) (2002).

30 M. J. O'Connell, S. M. Bachilo, C. B. Huffman, V. C. Moore, M. S. Strano, E. H. Haroz, K. L. Rialon, P. J. Boul, W. H. Noon, C. Kittrell, J. P. Ma, R. H. Hauge, R. B. Weisman, and R. E. Smalley, Science **297** (5581), 593 (2002).

31 H. Kataura, Y. Kumazawa, Y. Maniwa, I. Umezu, S. Suzuki, Y. Ohtsuka, and Y. Achiba, Synth. Met. **103** (1-3), 2555 (1999).

32 D. A. Heller, P. W. Barone, J. P. Swanson, R. M. Mayrhofer, and M. S. Strano, J. Phys. Chem. B. **108** (22), 6905 (2004).

33 M. S. Strano, S. K. Doorn, E. H. Haroz, C. Kittrell, R. H. Hauge, and R. E. Smalley, Nano Lett. **3** (8), 1091 (2003).

34 R. B. Weisman and S. M. Bachilo, Nano Lett. **3** (9), 1235 (2003).

35 S. M. Bachilo, L. Balzano, J. E. Herrera, F. Pompeo, D. E. Resasco, and R. B. Weisman, J. Am. Chem. Soc. **125**, 11186 (2003).

36 D. A. Heller, R. M. Mayrhofer, S. Baik, Y. V. Grinkova, M. L. Usrey, and M. S. Strano, J. .Am. Chem. Soc. **126**, 14567 (2004).

37 D. A. Heller, P. W. Barone, and M. S. Strano, Carbon **43**, 651 (2004).

38 M. L. Usrey, E. S. Lippmann, and M. S. Strano, J. Am. Chem. Soc. **127**, 16129 (2005).

39 Y. Zhao, X. J. Wang, C. C. Ma, and G. H. Chen, Chem. Phys. Lett. **387** (1-3), 149 (2004).

40 V. Perebeinos, J. Tersoff, and P. Avouris, Phys. Rev. Lett. **92** (25) (2004).

41 A. Hagen and T. Hertel, Nano Lett. **3** (3), 383 (2003).

42 A. Hagen, G. Moos, V. Talalaev, and T. Hertel, Appl. Phys. A. **78**, 1137 (2004).

43 M. L. Usrey, E. S. Lippmann, and M. S. Strano, J. Am. Chem. Soc. **127**, 16129 (2005).

44 M. S. Arnold, A. A. Green, J. F. Hulvat, S. I. Stupp, and M. C. Hersam, Nat. Nanotech. **1**, 60 (2006).

45 N. Nair, W.-J. Kim, M. L. Usrey, and M. S. Strano, J. .Am. Chem. Soc. **129**, 3946 (2007).

46 R. A. Marcus, J. Chem. Phys. **24**, 966 (1956).

47 J. R. Bolton and M. D. Archer, in *Electron Transfer in Inorganic, Organic and Biological Systems*, edited by G. McLendon (American Chemical Society, Washington, D.C., 1991), pp. 7.

48 H. Gerischer, in *Physical Chemistry: An Advanced Treatise*, edited by H. Eyring (Academic Press, Inc, New York, 1970), Vol. 9A, pp. 463.

49 A. Bard and L. R. Faulkner, *Electrochemical Methods, Fundamentals and Applications*, 2 ed. (John Wiley and Sons, New York, 2001).

50 B. J. Zwolinski, R. J. Marcus, and H. Eyring, Chem. Rev. **55**, 157 (1955).

51 I. Heller, J. Kong, H. A. Heering, K. A. Williams, S. G. Lemay, and C. Dekker, Nano Lett. **5**, 137 (2005).

52 I. Heller, J. Kong, K. A. Williams, C. Dekker, and S. G. Lemay, J. Am. Chem. Soc. **128**, 7353 (2006).

53 O. Lamm, Ark. Mat. Astr. Fys. **21B** (1929).

54 N. Nair, W.-J. Kim, R. D. Braatz, and M. S. Strano, Langmuir **24**, 1790 (2008).

55 W.-J. Kim, N. Nair, C. Y. Lee, and M. S. Strano, J. Phys.Chem. C. **112**, 7326 (2008).

56 J. T. Abrahamson, N. Nair, and M. S. Strano, Nanotechnology **19**, 195701 (2008).

57 M. J. Buehler, F. F. Abraham, and H. Gao, Nature **426**, 141 (2003).

58 M. J. Buehler and H. Gao, Nature **439**, 307 (2006).

59 M. J. Buehler, J. Mater. Res. **21**, 2855 (2006).

60 L. V. Zhigilei, C. Wei, and D. Srivastava, Phys. Rev. B. **71**, 165417 (2005).

61 L. V. Zhigilei, P. B. Kodali, and B. J. Garrison, J. Phys.Chem. B. **101**, 2028 (1997).

62 Y. Yingling and B. J. Garrison, J. Phys.Chem. B. **108**, 1815 (2004).

63 M. Prasad, P. F. Conforti, and B. J. Garrison, J. Chem. Phys. **127**, 084705 (2007).

64 A. M. Kosevich, *The Crystal Lattice: Phonons, Solitons, Dislocations, Superlattices*. (Wiley, Weinheim, 2005).

65 A. C. Newell, *Solitons in Mathematics and Physics*. (Society for Industrial and Applied Mathematics, Philadelphia, 1985).

66 J. L. Marin, S. Aubry, and L. M. Floria, Physica D **113**, 283 (1998).

67 G. P. Tsironis, B. A. R., A. V. Savin, and A. V. Zolotaryuk, Phys. Rev. E. **60**, 6610 (1999).

68 G. P. Tsironis and S. Aubry, Phys. Rev. Lett. **77**, 5225 (1996).

69 R. Reigada, A. H. Romero, A. Sarmiento, and K. Lindenberg, J. Chem. Phys. **111**, 1373 (1999).

70 R. Reigada, A. Sarmiento, A. H. Romero, J. M. Sancho, and K. Lindenberg, J. Chem. Phys. **112**, 10615 (2000).

71 E. Fermi, J. Pasta, and S. Ulam, Report No. LA 1940, 1955.

72 M. Zheng, A. Jagota, M. S. Strano, A. P. Santos, P. Barone, S. G. Chou, B. A. Diner, M. S. Dresselhaus, R. S. McLean, G. B. Onoa, G. G. Samsonidze, E. D. Semke, M. Usrey, and D. J. Walls, Science **302** (5650), 1545 (2003).

73 M. S. Strano, M. Zheng, A. Jagota, G. B. Onoa, D. A. Heller, P. W. Barone, and M. L. Usrey, Nano Lett. **4** (4), 543 (2004).

74 M. Zheng, A. Jagota, E. D. Semke, B. A. Diner, R. S. McLean, S. R. Lustig, R. E. Richardson, and N. G. Tassi, Nat. Mater. **2** (5), 338 (2003).

75 B. J. Landi, H. J. Ruf, C. M. Evans, C. D. Cress, and R. P. Raffaelle, J. Phys. Chem. B. **109** (20), 9952 (2005).

76 M. F. Lin and K. W. K. Shung, Phys. Rev. B. **50** (23), 17744 (1994).

77 M. F. Lin, Phys. Rev. B. **62** (19), 13153 (2000).

78 A. G. Ryabenko, T. V. Dorofeeva, and G. I. Zvereva, Carbon **42** (8-9), 1523 (2004).

79 H. S. Gold, C. E. Rechsteiner, and R. P. Buck, Anal. Chem. **48** (11), 1540 (1976).

80 R. P. Bauman, *Absorption Spectroscopy*. (John Wiley and Sons, New York, 1962).

81 R. Eisberg and R. Resnick, *Quantum Physics of Atoms, Molecule, Solids, Nucliei, and Particles*. (John Wiley and Sons, New York, 1974).

82 F. Wang, G. Dukovic, L. E. Brus, and T. F. Heinz, Science **308**, 838 (2005).

83 J. Maultzsch, R. Pomeraenke, S. Reich, E. Chang, D. Prezzi, A. Ruini, E. Molinari, M. S. Strano, C. Thomsen, and C. Lienau, Phys. Rev. B. **72**, 241402 (2005).

84 S. Reich, C. Thomsen, and J. Robertson, Phys. Rev. Lett. **95**, 077402 (2005).

85 C. Fantini, A. Jorio, M. Souza, M. S. Strano, M. S. Dresselhaus, and M. A. Pimenta, Phys. Rev. Lett. **93** (14) (2004).

86 J. Olivero and R. L. Longbothum, J. Quant. Spectrosc. Radiat. Transfer **17**, 233 (1977).

87 R. Gunawan, M. Y. L. Jung, E. G. Seebauer, and R. D. Braatz, AlChE J. **49** (8), 2114 (2003).

88 J. V. Beck and K. J. Arnold, *Parameter Estimation in Engineering and Science*. (John Wiley and Sons, Inc., New York, 1977).

89 Z. K. Nagy and R. D. Braatz, IEEE Trans. Control Syst. Technol. **11** (5), 694 (2003).

90 M. S. Strano, C. A. Dyke, M. L. Usrey, P. W. Barone, M. J. Allen, H. Shan, C. Kittrell, R. H. Hauge, J. M. Tour, and R. E. Smalley, Science **301** (12), 1519 (2003).

91 H. Huang, H. Kajiura, R. Maruyama, K. Kadano, and K. Noda, J. Phys.Chem. B. **110** (10), 4686 (2006).

92 A. Hartschuh, H. N. Pedrosa, L. Novotny, and T. D. Krauss, Science **301** (5638), 1354 (2003).

93 M. S. Strano, J. Am. Chem. Soc. **125**, 16148 (2003).

94 N. Nair, M. L. Usrey, W. Kim, R. D. Braatz, and M. S. Strano, Anal. Chem. **78**, 7689 (2006).

95 H. S. Fogler, *Elements of Chemical Reaction Engineering*, 3 ed. (Prentice Hall, Inc., New Jersey, 1999).

96 K. Okazaki, Y. Nakato, and K. Murakoshi, Phys. Rev. B **68**, 035434 (2003).

97 S. Suzuki, Y. Watanabe, and Y. Homma, Appl. Phys. Lett. **85**, 127 (2004).

98 B. Shan and K. Cho, Phys. Rev. Lett. **94**, 236602 (2005).

99 V. Barone, J. E. Peralta, J. Uddin, and G. E. Scuseria, J. Chem. Phys. **124**, 024709 (2006).

100 R. M. Elofson and F. F. Gadallah, J. Org. Chem. **34**, 854 (1968).

101 H. Reiss, J. Phys. Chem. **89**, 3783 (1985).

102 H. Gerischer and W. Ekardt, Appl. Phys. Lett. **43**, 393 (1983).

103 N. S. Lewis, Annu. Rev. Phys. Chem. **42**, 543 (1991).

104 M. S. Strano, C. B. Huffman, V. C. Moore, M. J. O'Connell, E. H. Haroz, J. Hubbard, M. Miller, K. L. Rialon, C. Kittrell, S. Ramesh, R. H. Hauge, and R. E. Smalley, J. Phys. Chem. B **107**, 6979 (2003).

105 M. J. O'Connell, E. E. Eibergen, and S. K. Doorn, Nat. Mater. **4**, 412 (2005).

106 S. Reich, J. Maultzsch, and C. Thomsen, Phys. Rev. B **66**, 035412 (2002).

107 J. G. a. J. A. Murphy, J. Chem. Soc., Chem. Commun., 24 (1992).

108 M. S. Arnold, S. I. Stupp, and M. C. Hersam, Nano Lett. **5**, 713 (2005).

109 H. Faxen, Ark. Mat. Astr. Fys. **21B**, 1 (1929).

110 H. Fujita, *Mathematical Theory of Sedimentation Analysis*. (Academic Press, New York, 1962).

111 L. A. Holladay, Biophys. Chem. **10**, 187 (1979).

112 J. S. Philo, Biophys. J. **72**, 435 (1996).

113 J. Behlke and O. Ristau, Biophys. J. **72**, 428 (1997).

114 W. J. Archibald, Phys. Rev. **53**, 746 (1938).

115 W. J. Archibald, J. Appl. Phys. **18**, 362 (1946).

116 J. Claverie, H. Dreux, and R. Cohen, Biopolymers **14**, 1685 (1975).

117 P. Schuck, Biophys. Chem. **108**, 187 (2004).

118 P. Schuck, Biophys. Chem. **108**, 201 (2004).

119 P. Schuck, Biophys. J. **74**, 466 (1998).

120 P. Schuck, Biophys. J. **75**, 1503 (1998).

121 T. Svedberg and K. O. Pedersen, *The Ultracentrifuge*. (Oxford University Press, Oxford, 1940).

122 R. J. Leveque, *Finite Volume Methods for Hyperbolic Problems*. (Cambridge University Press, Cambridge, 2002).

123 V. N. Schumaker and J. Rosenbloom, Biochemistry **4**, 1005 (1965).

124 V. N. Schumaker and J. Rosenbloom, Biochemistry **6**, 1149 (1967).

125 M. Dishon, G. H. Weiss, and D. A. Yphantis, Biopolymers **10**, 2095 (1971).

126 V. Bloomfield, W. O. Dalton, and K. E. Van Holde, Biopolymers **5**, 135 (1967).

127 J. G. De La Torre and V. A. Bloomfield, Biopolymers **16**, 1747 (1977).

128 J. A. McCammon and J. M. Deutch, Biopolymers **15**, 1397 (1976).

129 K. L. Planken, Undergraduate Thesis, Utrecht University, 2003.

130 M. M. Tirado and J. G. De La Torre, J. Chem. Phys **71**, 2581 (1979).

131 K. E. Bett and J. B. Cappi, Nature **207**, 620 (1965).

132 R. Zwanzig and M. Bixon, Phys. Rev. A **2**, 2005 (1970).

133 M. A. Islam, Phys. Scr. **70**, 120 (2004).

134 T. Mashimo, Phys. Rev. A **38**, 4149 (1988).

135 I. Cabria, J. W. Mintmire, and C. T. White, Phys. Rev. B **67**, 121406 (2003).

136 A. I. Kitaigorodskii, Organicheskaya Kristallokhimiya, 558 (1955).

137 T. S. Brun, H. Hoiland, and E. Vikingstad, J. Colloid Interface Sci. **63**, 89 (1977).

138 G. H. Szabo and H. Hoiland, Langmuir **14**, 5539 (1998).

139 W. B. Smith and G. D. Barnard, Can. J. Chem. **59**, 1602 (1981).

140 J. B. Hayter and J. Penfold, Colloid Polym Sci **261**, 1022 (1983).

141 P. Messina, M. A. Morini, and P. C. Schulz, Colloid Polym Sci **281**, 695 (2003).

142 W. K. Sartory, H. B. Halsall, and J. P. Breillatt, Biophys. Chem. **5**, 107 (1976).

143 M. L. Usrey, N. Nair, D. E. Agnew, C. F. Pina, and M. S. Strano, Langmuir **23**, 7768 (2007).

144 M. F. Islam, E. Rojas, D. M. Bergey, A. t. Johnson, and A. G. Yodh, Nano Lett. **3**, 269 (2003).

145 C. Richard, F. Balavoine, P. Schultz, T. W. Ebbesen, and C. Mioskowski, Science **300**, 775 (2003).

146 K. Yurekli, C. A. Mitchell, and R. Krishnamoorti, J. Am. Chem. Soc. **126**, 9902 (2004).

147 M. Tanaka, S. Kaneshina, T. Tomida, K. Noda, and K. Aoki, J. Colloid Interface Sci. **44**, 525 (1973).

148 S. D. Hamann, J. Phys. Chem. **66**, 1359 (1962).

149 N. Grossiord, P. van der Schoot, J. Meuldijkm, and C. E. Koning, Langmuir **23**, 3646 (2007).

150 M. S. Strano, C. M. Moore, M. K. Miller, M. J. Allen, E. H. Haroz, C. Kittrell, R. H. Hauge, and R. E. Smalley, J. Nanosci. Nanotechnol. **3**, 81 (2003).

151 O. Matarredona, H. Rhoads, Z. Li, J. H. Harwell, L. Balzano, and D. E. Resasco, J. Phys. Chem. B **107**, 13357 (2003).

152 J. F. Cardenas and M. Glerup, Nanotechnol. **17**, 5212 (2006).

153 G. Sugihara and M. Tanaka, Hyomen **16**, 537 (1978).

154 H. Wang, W. Zhou, D. L. Ho, K. I. Winey, J. E. Fischer, C. J. Glinka, and E. K. Hobbie, Nano Lett. **4**, 1789 (2004).

155 N. Nair, M. L. Usrey, W.-J. Kim, R. D. Braatz, and M. S. Strano, Anal. Chem. **78**, 7689 (2006).

156 T. J. McDonald, C. Engtrakul, M. Jones, G. Rumbles, and M. J. Heben, J. Phys. Chem. B **110**, 25339 (2006).

157 T. J. McDonald, C. Engtrakul, M. Jones, J. L. Blackburn, G. Rumbles, and M. J. Heben, Proc. of SPIE **5929**, 59290W1 (2005).

158 Y. Tan and D. E. Resasco, J. Phys. Chem. B **109**, 14454 (2005).

159 J. L. Blackburn, C. Engtrakul, T. J. McDonald, A. C. Dillon, and M. J. Heben, J. Phys. Chem. B **110**, 25551 (2006).

160 G. K. Straub, B. L. Holian, and R. G. Petschek, Phys. Rev. B. **19**, 4049 (1979).

161 G. Stoltz, Nonlinearity **18**, 1967 (2005).

162 B. L. Holian and G. K. Straub, Phys. Rev. B. **18**, 1593 (1978).

163 J. N. Stember and G. S. Ezra, Chem. Phys. **337**, 11 (2007).

164 R. Reigada, A. Sarmiento, and K. Lindenberg, Phys. Rev. E. **64**, 066608 (2001).

165 R. Reigada, A. Sarmiento, and K. Lindenberg, Physica A **305**, 467 (2002).

166 A. Sarmiento, R. Reigada, A. H. Romero, and K. Lindenberg, Phys. Rev. E. **60**, 5317 (1999).

167 A. Rosas and K. Lindenberg, Phys. Rev. E. **69**, 016615 (2004).

168 X. J. Hu, A. A. Padilla, J. Xu, T. S. Fisher, and K. Goodson, J. Heat. Trans. **128**, 1109 (2006).

169 S. T. Huxtable, D. G. Cahill, S. Shenogin, L. Xue, R. Ozisik, P. W. Barone, M. L. Usrey, M. S. Strano, G. Siddons, M. Shim, and P. Keblinski, Nat. Mater. **2**, 731 (2003).

170 P. A. Childs and A. G. O' Neill, Physica E **19**, 153 (2003).

171 G. F. Calvo and R. F. Alvarez-Estrada, Nanotechnology **15**, 1870 (2004).

172 D. C. Rapaport, *The Art of Molecular Dynamics Simulations*, 2 ed. (Cambridge University Press, 2004).

173 D. T. Gillespie, J. Comp. Phys. **22**, 403 (1976).

174 D. T. Gillespie, J. Phys. Chem. **81**, 2340 (1977).

175 D. Frenkel and B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications*. (Academic Press, 2002).

176 Y. Chen, D. Li, J. Yang, Y. Wu, J. Lukes, and A. Majumdar, Physica B **349**, 270 (2004).

177 J. Lukes and H. Zhong, J. Heat. Trans. **129**, 705 (2007).

178 H. C. Andersen, J. Chem. Phys **72**, 2384 (1980).

179 S. Nose, Mol. Phys. **52**, 255 (1984).
180 S. Nose, J. Chem. Phys **81**, 511 (1984).
181 W. H. Hoover, Phys. Rev. A **31**, 1695 (1985).
182 B. L. Holian, A. J. De Groot, W. H. Hoover, and C. G. Hoover, Phys. Rev. A **41**, 4552 (1990).
183 C. Bratschi and H. Huber, J. Chem. Phys. **126**, 164104 (2007).
184 S. Berber, Y.-K. Kwon, and D. Tomanek, Phys. Rev. Lett. **84**, 4613 (2000).
185 S. Lepri, R. Livi, and A. Politi, Phys. Rep. **377**, 1 (2003).
186 D. A. McQuarrie, *Statistical Mechanics*. (University Science Books, Sausalito, 2000).
187 S. Lepri, R. Livi, and A. Politi, Phys. Rev. E. **68**, 067102 (2003).
188 S. Lepri, R. Livi, and A. Politi, Phys. Rev. Lett. **78**, 1896 (1997).
189 B. Hu, B. Li, and H. Zhao, Phys. Rev. E. **61**, 3828 (2000).
190 H. Muthurajan, R. Sivabalan, M. B. Talawar, M. Anniyappan, and S. Venugopalan, J. Hazard. Mater. **A133**, 30 (2006).
191 B. Li, L. Wang, and G. Casati, Phys. Rev. Lett. **93**, 1843011 (2004).
192 J. M. Bowman, X. Zhang, and A. Brown, J. Chem. Phys. **119**, 646 (2003).

Appendix A: Spectral Lineshapes, Peak Parameters and Parity Plots

Comparison of Gaussian, Lorentzian and Voigt Profiles

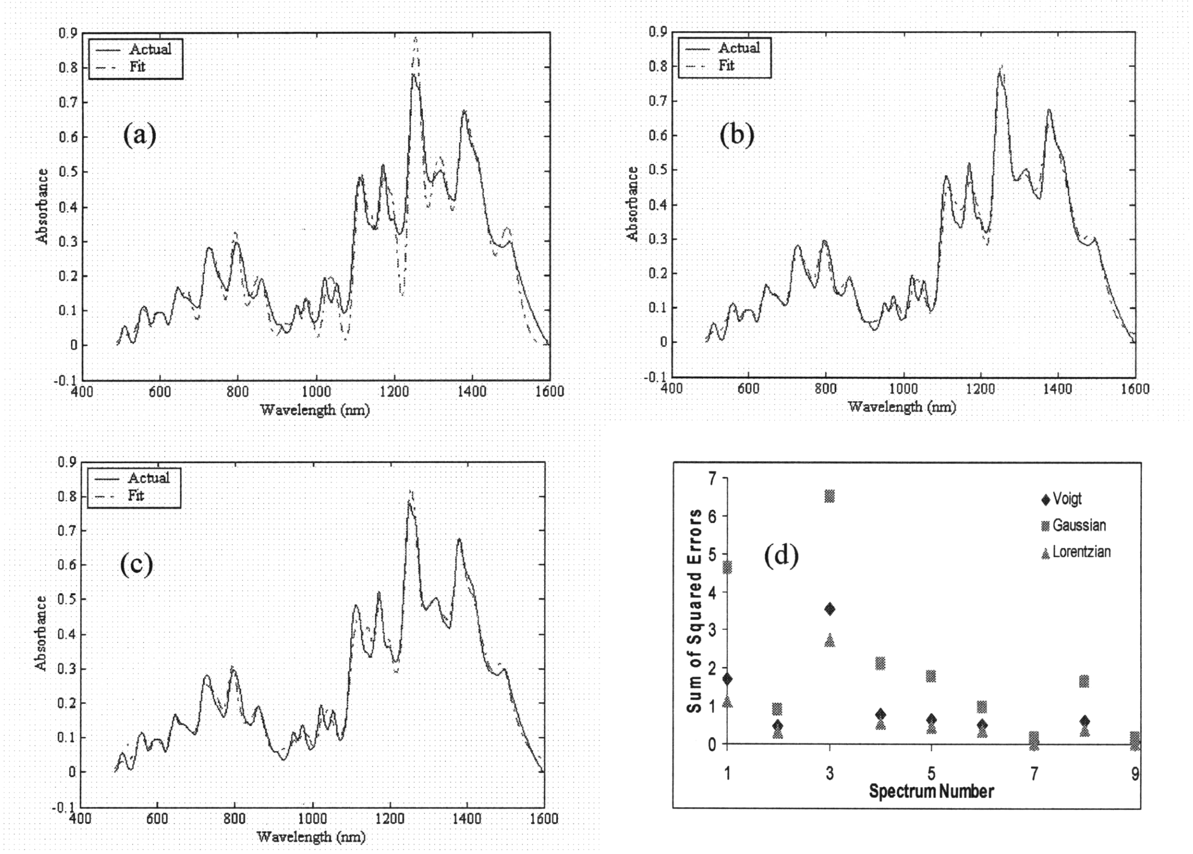


Figure A1. Figures (a)-(c) compare spectral fits using Gaussian, Lorentzian, and Voigt profiles respectively. The sum of squared errors was determined for the fits of 9 absorption spectra and is displayed in Figure (d) for each lineshape. It can be seen that the errors for the Voigt and Lorentzian fits are much lower than the ones for the Gaussian fits.

Fit of a DNA-SWNT Absorption Spectrum

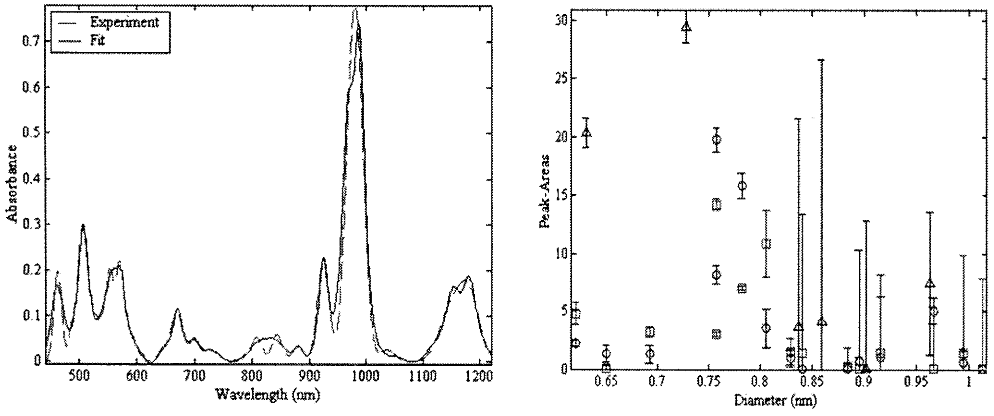


Figure A2. No weighting scheme was applied, since most of the peaks correspond to single nanotubes. In the diameter-distribution plots, the blue circles denote the E_{11} peaks, the red squares denote the E_{22} peaks and the black triangles denote the metallic peaks. The following peak-parameters were estimated: $\Gamma_{V,11}$ =29.86 meV, $\Gamma_{V,22}$ =57.96 meV, $\Gamma_{V,M}$ =93.42 meV

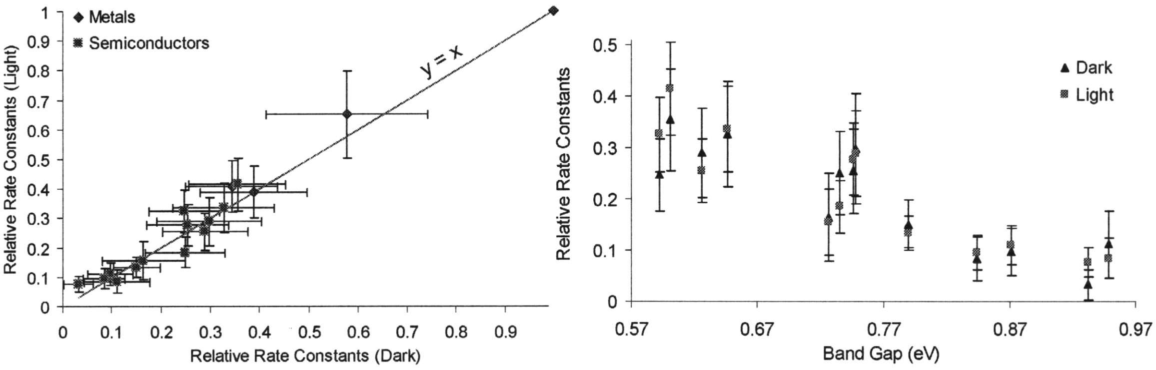
Comparison of metallic relative rate constants with theory

	Relative Rate Constants (Dark)			**Relative Rate Constants (Light)**		
(*n*,*m*)	**Spectral Fits**	**Marcus Theory**	**Gerischer-Marcus Theory**	**Spectral Fits**	**Marcus Theory**	**Gerischer-Marcus Theory**
(7,7)	0.388 ± 0.108	1.000	0.143	0.389 ± 0.086	1.000	0.143
(10,4)	0.577 ± 0.164	1.000	0.500	0.652 ± 0.146	1.000	0.500
(11,5)*	1.000	1.000	1.000	1.000	1.000	1.000
(11,8)	0.344 ± 0.094	1.000	1.000	0.408 ± 0.087	1.000	1.001

*All rate constants were calculated relative to the (11,5) nanotube

Table A1. Relative rate constants for metallic nanotubes as obtained from the spectral fits, Marcus theory and Gerischer-Marcus theory. Figures 4(a)-(b) show that the electron transfer theories can explain the trend in the relative rate constants with band gap for semiconductors. While the theories predict higher relative rate constants for metallic SWNT than semiconducting SWNT, they fail in detecting trends among the metals themselves.

Parity Plot



(a) (b)

Figure A3. (a) Parity plot with the 95% confidence intervals for the light and dark relative rate constants. Shown are the extremes on either side of the unit slope line between which the fit parameters can lie. (b) Light and dark relative rate constants for semiconducting SWNT. The high degree of overlap between the 95% confidence intervals, especially for the large-diameter nanotubes, shows that the values are statistically similar.

Appendix B: Nanotube Transport in a Centrifugal Field

Radial variation of pressure in the centrifuge tube

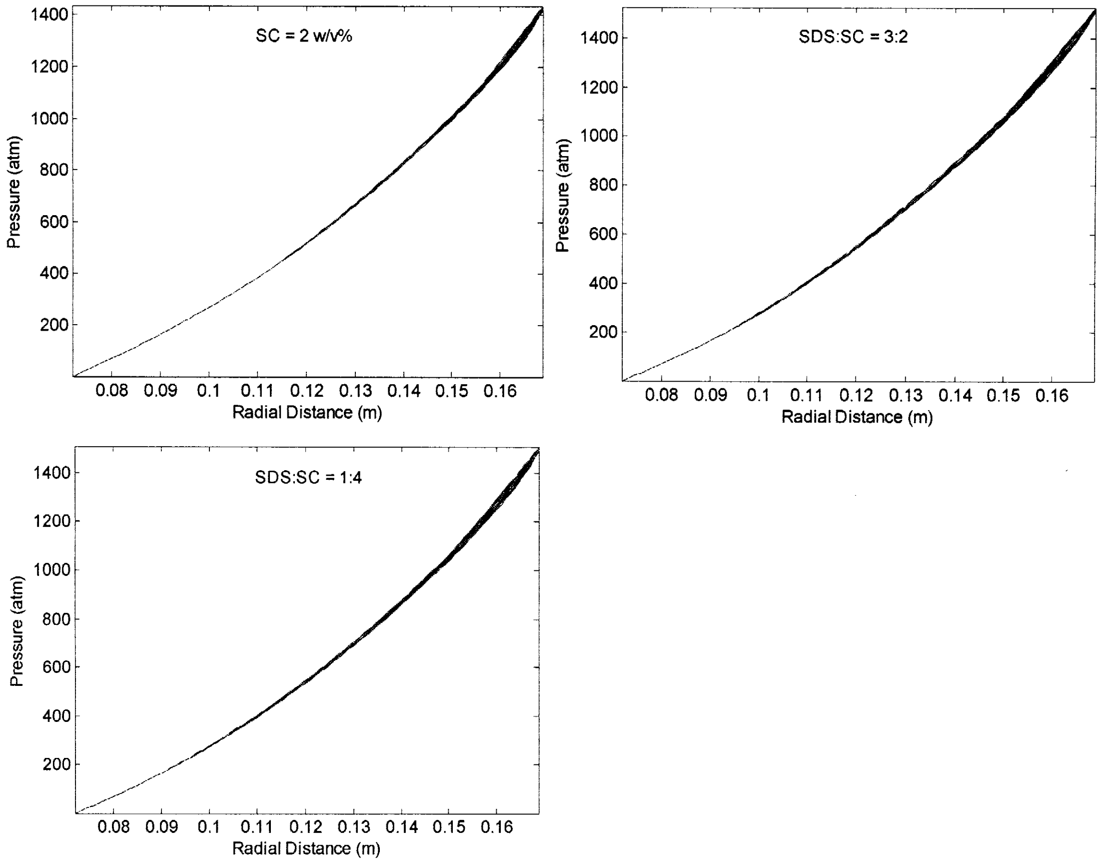


Figure B1. Radial variation of pressure for the (a) 2 w/v% SC, (b) 3:2 and (c) 1:4 cases. The region of interest, within which SWNT motion takes place, lies between 0.1138-0.1538 m for (a) and 0.0979-0.1488 m for (b)-(c). The pressures in these domains are below 1000 atm, which indicates that SDS is not likely to precipitate out of solution. Nothing can be said about SC, since no information regarding the effect of pressure on its properties could be found.

Fits of 12- and 44.53-hour centrifuge runs

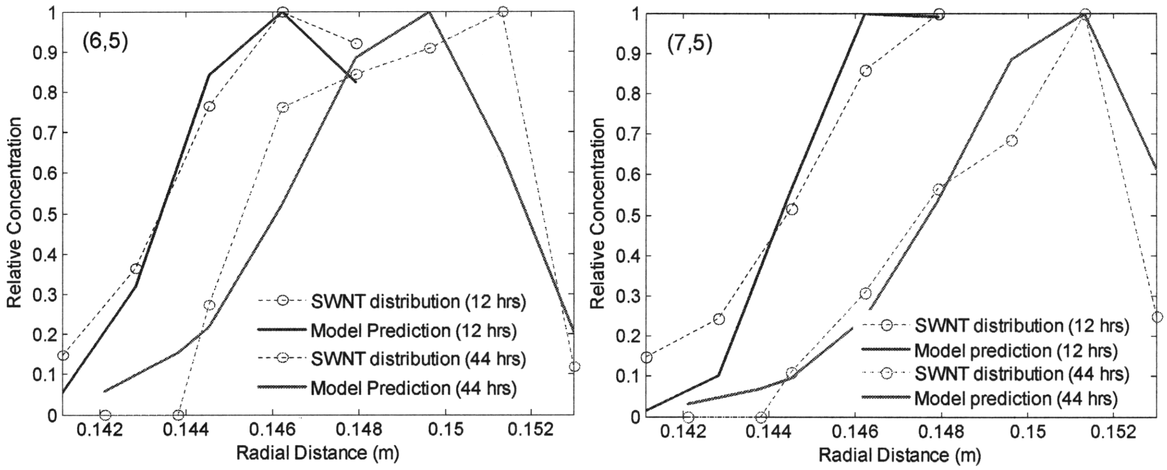


Figure B2. Fits for the (6,5) and (7,5) nanotubes after 12-hr and 44.53-hr runs. The parameter estimate might be skewed (~1.84 molec/nm for the first two cases vs. ~2.04 molec/nm for the third) by the fact that the fractions are located close to the point of injection, where the bulkier SWNT lie.

Appendix C: Nanostructure-Guided Chain Reactions

Thermal Conductivity of Oscillator Chains (FPU-β and φ^4)

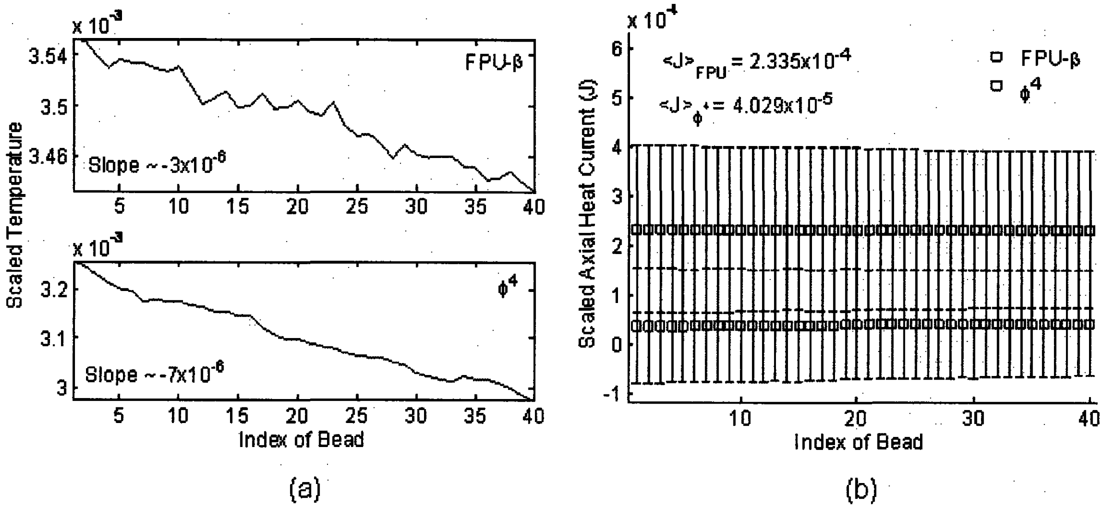


Figure C1. (a) Comparison of temperature variation in the FPU (top panel) and φ^4 (bottom panel) lattices when one end of the chain was set at T^*=0.004 and the other at T^*=0.00003. Here, T^* is the scaled temperature that is comparable to those generated in the lattice during successive energetic reactions. The scaling factor is $m_{sc}(u_{sc}*u_{sc})/k_B$, where m_{sc} is the mass of the unreacted bead, u_{sc} is the speed of sound in the lattice, and k_B is the Boltzmann factor. The calculated temperature gradient (scaled units) is shown in blue. (b) Time-averaged heat current in the axial direction (x) at each node. Stationarity requires that the averaged value be approximately constant across the chain. The mean heat current for the entire chain, $<J>$, is shown for the 2 lattices. The ratio of $<J>$ to the negative of the temperature gradient yields the thermal conductivity of the respective lattice in scaled units: $\kappa_{FPU}\sim$ 77.83, $\kappa_{\Phi}\sim$ 5.76, i.e., κ_{FPU} $\sim$ 13.5κ_{φ}.

Effect of EM load on contribution of higher modes

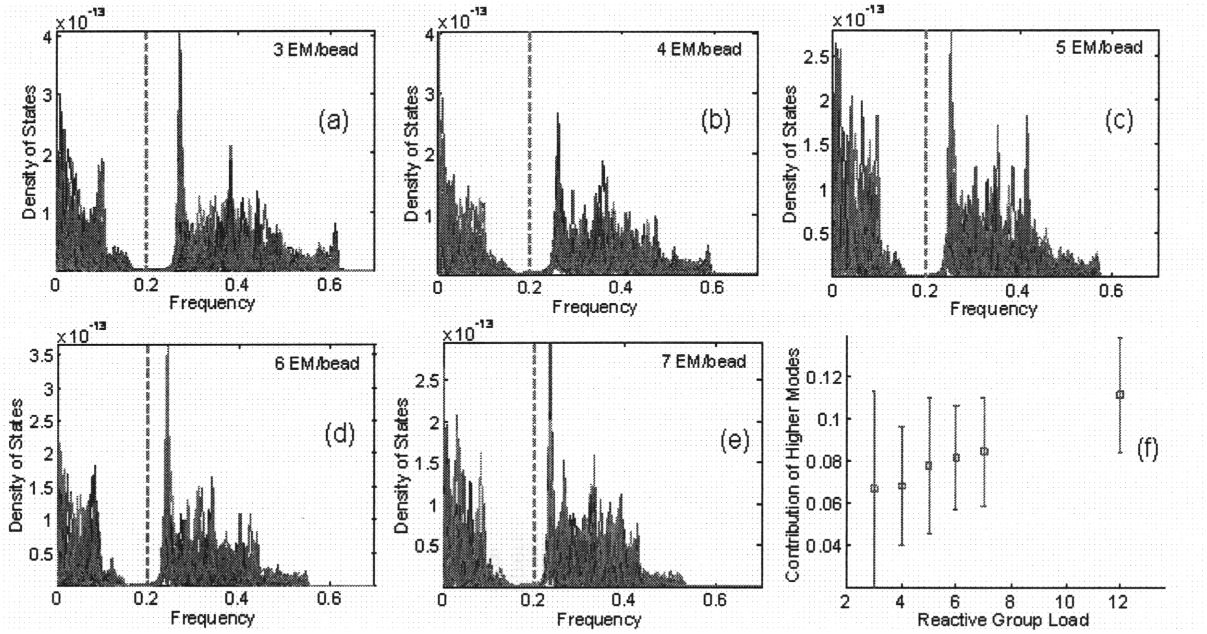


Figure C2. (a)-(e) Density of states of individual beads in a φ^4 system at the specified reactive group loadings. The orange line indicates the position of ω=0.2, which was used as the boundary for the calculation of the areas under the low- and high-frequency regions of the spectrum. (f) Ratio of the area under the high-frequency portion to the total spectral area plotted against reactive group loading shows that the contribution of the higher modes increases with the energy input into the lattice.

Optimal driving forces for Driven Molecular Dynamics

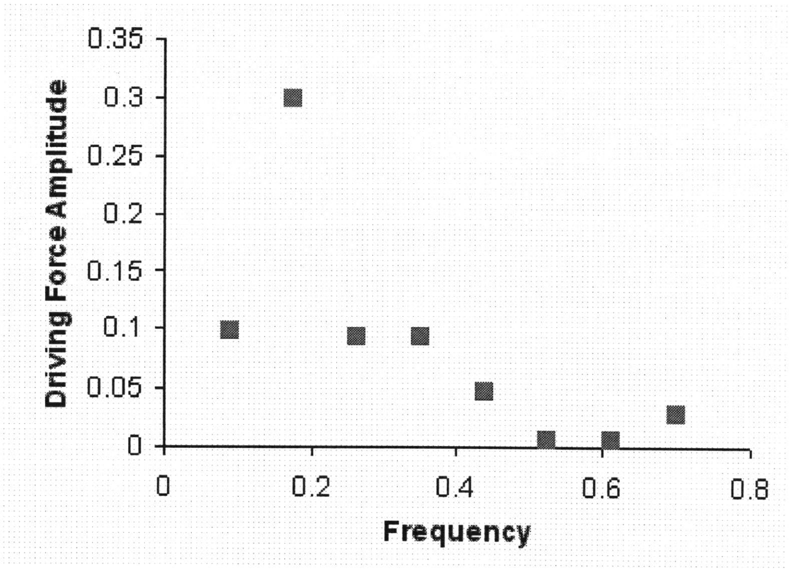


Figure C3. Estimation of the driving force amplitudes that gave an optimal signal at each of the applied frequencies

Effect of EM load on φ^4 reaction velocity

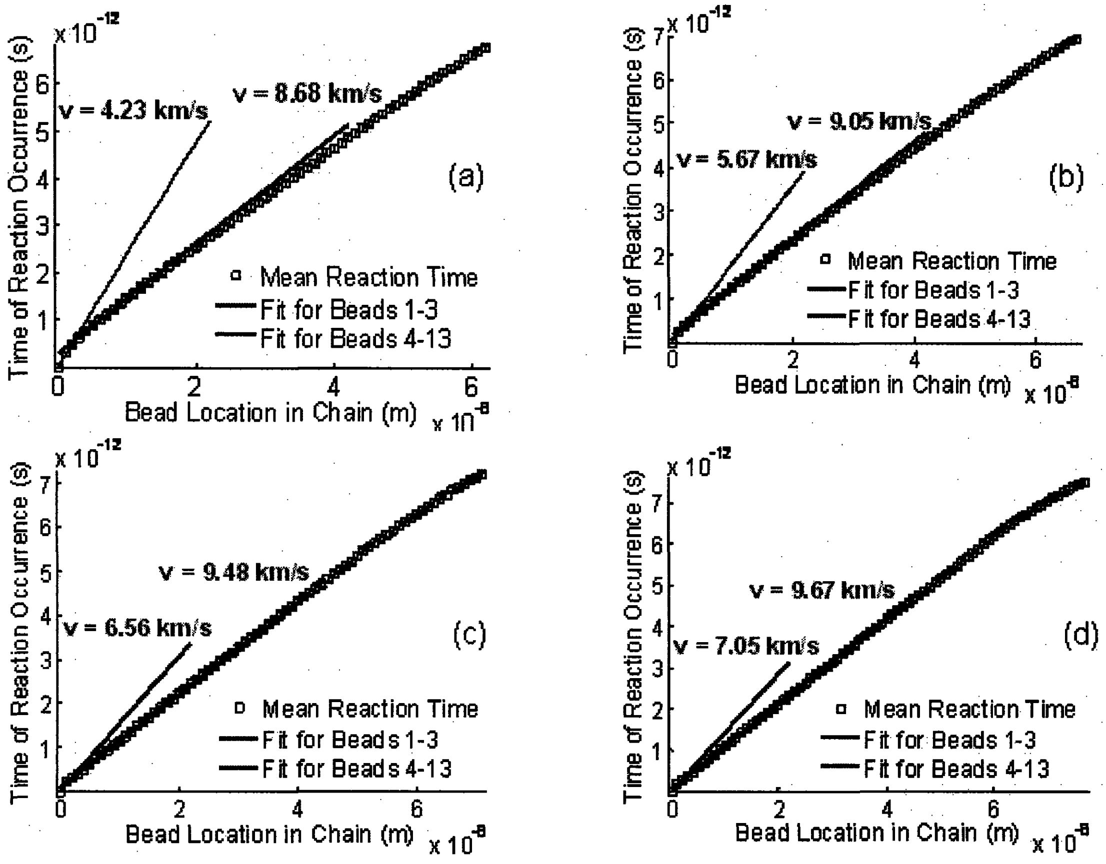


Figure C4. Comparison of reaction velocities for a φ^4 lattice at (a) 4, (b) 5, (c) 6 and (d) 7 EM/bead, respectively (E_a=35 kcal/mol, α=0.7). The probability cutoffs were set at 0.90. With increased loads, the difference between the slower start-up and faster bulk phases gradually disappears.

Higher energies at free boundaries of a harmonic chain

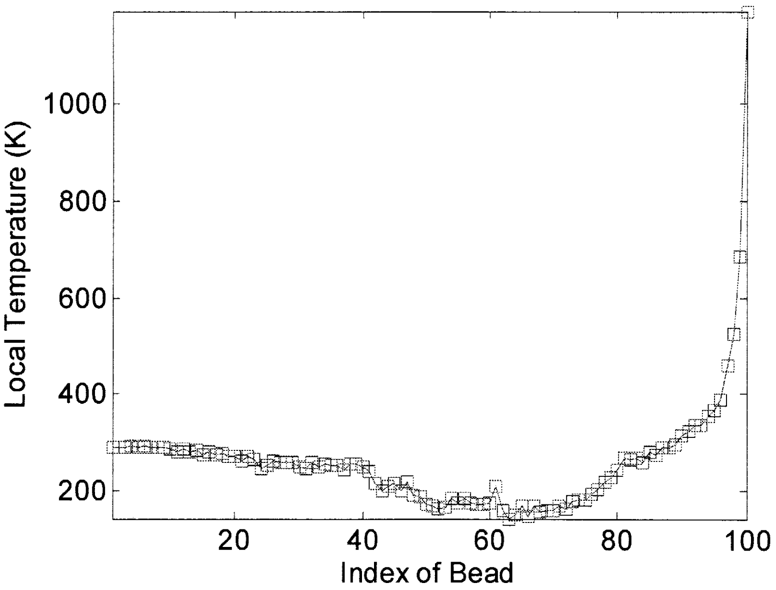


Figure C5. Variation of local temperature in a harmonic chain with 2 EM/bead loading. The system was initially set up at 300 K and simulated in an NVE ensemble to obtain the time-averaged temperatures at each location. Free boundary conditions at both ends result at higher energies at the extreme beads in comparison to the central portion of the lattice.

Appendix D: MATLAB and C Programs

Selected MATLAB functions for deconvolution of reaction spectra

This program is used to analyse the reactions of single-walled carbon nanotubes. "reaction" is the main function from where all the remaining functions are called to perform calculations, store data and plot the trends in reactivity for certain nanotubes. “Semiconductors” have been abbreviated as “sc”, and metals are denoted as “met”.

```
function reaction
clear sub lsub conc
global sub lsub conc

file=input('Enter Excel file name (without ".xls"): ','s');
sheet=input('Enter sheet name: ','s');
spec=xlsread([file '.xls'],sheet);                    %Reads absorption data from specified Excel file
op=input('Has background subtraction already been performed (y/n)? ','s');
if op=='y'
   sub=spec;
   lsub=length(sub);
   smooth;
else
   colm=input('Which data set would you like to use to calculate the background? ');
   [sub lsub]=bkgnd(spec,colm);                       %Background subtraction based on selected column
end
[val peak sub lsub]=pkval(sub,lsub);                  %Locates peaks and valleys in the spectrum
[lamda_11 lamda_22 lamda_m lamda nmdp]=getinfo;       %Stores SWNT-related information
gamma=weight(lamda,val,peak);                         %Calculates weights for each SWNT
X=detX(gamma,lamda_m,lamda_22,lamda_11);              %Calculates coefficient matrix
[SC,M,abso]=rxn(X,lamda,nmdp,gamma);                  %Fits multiple reaction spectra

function smooth                                       %Smooths the spectrum
global sub lsub

s=sub(1:2,:);
k=2;
for i=3:lsub-2
   k=k+1;
   s(k,1)=sub(i,1);
   s(k,2:size(sub,2))=(sub(i-2,2:size(sub,2))+2*sub(i-
1,2:size(sub,2))+3*sub(i,2:size(sub,2))+2*sub(i+1,2:size(sub,2))+sub(i+2,2:size(sub,2)))/9;
end
s=[s;sub(lsub-1:lsub,:)];
sub=s;

function [sub,lsub]=bkgnd(spec,colm)                  %Front-end for background subtraction
s1=spec(1,1);
while 1>0
   w=input('Enter the starting and ending wavelengths: ','s');
   w=sscanf(w,'%i');
   [temp bg]=bkg(spec(w(1)-s1+1:w(2)-s1+1,:),w(2)-w(1)+1,colm);
   plot(spec(w(1)-s1+1:w(2)-s1+1,1),spec(w(1)-s1+1:w(2)-s1+1,2:size(spec,2)))
   hold on
   plot(spec(w(1)-s1+1:w(2)-s1+1,1),bg,'r')
```

```
   opt=input('Are you satisfied with the background (y/n)? ','s');
   if opt=='y'
      close
      break
   end
   close
end
t1=temp(1,1);
sub=temp;                                          %Subtracted spectrum stored in "sub"
lsub=length(sub);                                  %Number of data points in "sub"

function [sub,bg]=bkg(spec,lspec,colm)             %Back-end routine for subtraction
global data ldata col

data=spec;
ldata=lspec;
col=colm;
kb=[200 0.9];                                      %Initial guess for the fit parameters "k" and "b"
log_kb=[log(kb(1)) kb(2)];
A=[ones(ldata,1) -log(data(:,1))];
B=log(data(:,col));
log_kopt=fmincon(@fit3,log_kb,A,B);                %Calculation of optimum "k" and "b"
kopt=[exp(log_kopt(1)) log_kopt(2)];
bg=kopt(1)./(data(:,1).^kopt(2));
sub=data(:,1);
for i=2:size(data,2)
   sub(:,i)=data(:,i)-bg;
end
s=sub(1:2,:);
k=2;
for i=3:length(sub)-2
   k=k+1;
   s(k,1)=sub(i,1);
   s(k,2:size(data,2))=(sub(i-2,2:size(data,2))+2*sub(i-
1,2:size(data,2))+3*sub(i,2:size(data,2))+2*sub(i+1,2)+sub(i+2,2:size(data,2)))/9;
end
s=[s;sub(length(sub)-1:length(sub),:)];
sub=s;

function err=fit3(log_kb)                          %Function minimized by the "fmincon"
global data ldata col
log_data=log(data);
log_bkg=log_kb(1)*ones(ldata,1)-log_kb(2)*log_data(:,1);
F=log_data(:,col)-log_bkg;
err=F'*F;

function [val,peak,sub,lsub]=pkval(sub,lsub)       %Locates spectral peaks and valleys
s1=sub(1,1);
s2=sub(lsub,1);
pwav=[];
vwav=[];

gr=gradient(sub(:,2));
q=0;
for i=2:(lsub-1)
   if (gr(i-1)<0 & gr(i+1)>0) & sub(i,1)<1500      %Identifies points where slope changes sign
```

```
      q=q+1;
      s(q,1)=sub(i,1);
      s(q,2)=sub(i,2);
   end
end
len_s=q;

val(1,1)=sub(1,1);
val(1,2)=sub(1,2);
val(2,1)=s(1,1);
val(2,2)=s(1,2);
i=1;
k=2;
while i<(len_s-2)
   k=k+1;
   j=i+1;
   j_in=j;
   while (s(j,1)-s(i,1))<9
      j=j+1;
   end
   j_fin=j;
   val(k,1)=s(j,1);
   val(k,2)=s(j,2);
   i=i+(j_fin-j_in+1);
end
val(length(val)+1,:)=sub(lsub,1:2);

q=0;
for i=1:length(val)-1
   q=q+1;
   L1=val(i,1);
   L2=val(i+1,1);
   m=max(sub(L1-s1+1:L2-s1+1,2));
   for L=L1:L2
      if sub(L-s1+1,2)==m
         wav=sub(L-s1+1,1);
      end
   end
   peak(q,1)=wav;
   peak(q,2)=m;
end

plot(sub(:,1),sub(:,2))
hold on
plot(peak(:,1),peak(:,2),'or')
plot(val(:,1),val(:,2),'sk')
legend('Spectrum','Peaks','Valleys')
op=input('Do you wish to add more peaks and valleys (y/n)? ','s');
if op=='y'
   pwav=input('Enter wavelengths of additional peaks: ','s');
   vwav=input('Enter wavelengths of additional valleys: ','s');
   pwav=sscanf(pwav,'%i');
   pnum=pwav-sub(1,1)+1;
   pwav=[pwav sub(pnum,2)];
   vwav=sscanf(vwav,'%i');
   vnum=vwav-sub(1,1)+1;
```

```
    vwav=[vwav sub(vnum,2)];
  end
  peak=sortrows([peak;pwav],1);
  val=sortrows([val;vwav],1);
  op=input('Are there extraneous peaks or valleys (y/n)? ','s');
  if op=='y'
    wstart=input('Enter new starting wavelength: ');
    for i=1:length(val)
      if peak(i,1)<wstart
        peak(i,:)=[];
      end
      if val(i,1)<wstart
        val(i,:)=[];
      end
      if val(i,1)>wstart
        break
      end
    end
    wstart=wstart-sub(1,1)+1;
    sub=sub(wstart:length(sub),:);
    lsub=length(sub);
  else
    sub=sub;
    lsub=lsub;
  end
  close

  function gamma=weight(lamda,val,peak)          %Calculate weights based on transition energies
  temp=repmat(Inf,1,length(lamda));
  gamma=[];
  for i=1:length(val)-1
    for j=1:length(lamda)
      if lamda(j)>=val(i,1) & lamda(j)<val(i+1,1)
        temp(j)=abs(1/lamda(j)-1/peak(i,1))/(1/val(i,1)-1/val(i+1,1));
      end
    end
    gamma=[gamma;exp(-temp)./sum(exp(-temp))];
    temp=repmat(Inf,1,length(lamda));
  end
  gamma=gamma';
  gamma=[lamda gamma];
  gamma=sortrows(gamma,1);
  gamma=gamma(:,2:size(gamma,2));

  function X=detX(gamma,lamda_m,lamda_22,lamda_11)
  global sub lsub

  Lm=length(lamda_m);Lsc=length(lamda_11);
  hc=1239842;
  fv_11=25; fv_22=57.96; fv_m=93.42;            %Voigt FWHMs
  fg_11=fv_11/1.6376; fg_22=fv_22/1.6376; fg_m=fv_m/1.6376;
  fl_11=fg_11; fl_22=fg_22; fl_m=fg_m;          %Lorentzian component FWHMs
  l_start=sub(1,1);
  l_end=sub(lsub,1);
  Ej=-100:0.1:100;
  X=[];
```

```
for L=l_start:l_end
   met=calc(Ej,L,lamda_m,fg_m,fl_m);                %Coefficients for metallic peak areas
   one=calc(Ej,L,lamda_11,fg_11,fl_11);             %Coefficients for E-11 peak areas
   two=calc(Ej,L,lamda_22,fg_22,fl_22);             %Coefficients for E-22 peak areas
   temp(1:Lm)=met;
   temp(Lm+1:Lm+Lsc)=two;
   temp(Lm+Lsc+1:Lm+2*Lsc)=one;
   X=[X;temp];
end
lamda=[lamda_m;lamda_22;lamda_11];
X=[lamda X'];
X=sortrows(X,1);
X=(X(:,2:size(X,2)))';
X=X*gamma;

function val=calc(Ej,L,wav,fg,fl)
val=[];
delE=0.1;
E=1239842/L;
for i=1:length(wav)
   Eo=1239842/wav(i);
   func=exp(-2.7726*(Ej/fg).^2)./(fl^2/4+(E-Ej-Eo).^2)*delE;
   val=[val 0.1495*sum(func)];
end

function [SC,M,abso]=rxn(X,lamda,nmdp,gamma)
global sub lsub

SC=[];
M=[];
abso=[];
C=[];
for num=2:size(sub,2)
   [Abs C beta res err]=voigtfit(X,gamma,num,C);    %Calls the linear solver that solves for the peak areas
   LU=confint(C,res,X,gamma,beta);                  %Calculates confidence intervals for each SWNT
   [sc,met]=select(nmdp,lamda,beta,LU);             %Separates parameter vector into sc & met portions
   SC=[SC sc];
   M=[M met];
   abso=[abso Abs];
end

function [Abs,C,beta,R,err]=voigtfit(X,gamma,num,C)
global sub lsub

Y_data=sub(:,num);
H=X'*X;
f=-X'*Y_data;
lb=zeros(size(gamma,2),1);
if num==2
   ub=repmat(Inf,size(gamma,2),1);
else
   ub=C;
end
options=optimset('TolFun',1e-20,'MaxIter',100000000);
C=quadprog(H,f,[],[],[],[],lb,ub,[],options);        %Solved "parent Voigt" peak areas
beta=gamma*C;                                       %Peak areas for nanotube Voigts
```

```
Abs=X*C;                                  %Calculated Absorbance values
R=Y_data-Abs;                             %Residuals at each wavelength
err=R'*R;


function LU=confint(C,res,jac,gamma,beta)     lres=length(res);
n=100;
[l m]=size(jac);
v=l-m-2;

temp=find(max(abs(jac))==0);
if ~isempty(temp)
  jac(temp,:)=jac(temp,:)+sqrt(eps);
end
for i=1:lres
   if i<=n
      sig(i)=sqrt(sum(res(1:i+n).*res(1:i+n)/(i+n)));
   elseif i>n & i<=(lres-n)
      sig(i)=sqrt(sum(res(i-n:i+n).*res(i-n:i+n)/(2*n+1)));
   elseif i>lres-n
      sig(i)=sqrt(sum(res(i-n:lres).*res(i-n:lres)/(lres-i+n+1)));
   end
end
Ve=diag(sig.^2,0);
Ve_inv=Ve\eye(size(Ve));
t1=jac'*Ve_inv;
t2=t1*jac;
Vb=t2\eye(size(t2));

t1=gamma*Vb;
t2=t1*gamma';
delta=tinv(0.975,v)*sqrt(diag(t2,0));
ci=[beta-delta beta+delta];
for i=1:length(ci)
   if ci(i,1)<0
      ci(i,1)=0;
   end
end
LU=[beta-ci(:,1) ci(:,2)-beta];

function [sc,met]=select(nmdp,lamda,beta,LU)          %Separates the results into sc and met portions
sc=[];
met=[];
lamda=sort(lamda);
for i=1:length(beta)
   if rem(nmdp(i,1)-nmdp(i,2),3)~=0
      for j=1:length(beta)
         if j~=i & nmdp(i,1)==nmdp(j,1) & nmdp(i,2)==nmdp(j,2) & lamda(i)<lamda(j)
            sc=[sc;[lamda(j) nmdp(i,1:4) beta(i) LU(i,1:2) beta(j) LU(j,1:2)]];
         end
      end
   else
     met=[met;[nmdp(i,1:3) beta(i) LU(i,1:2)]];
   end
end
sc=sortrows(sc,1);
```

```
sc=sc(:,2:size(sc,2));

function [sc,met]=degree(SC,M)                %Calculates degree of functionalisation
q=0;
for i=1:size(SC,1)
   for j=1:size(SC,2)
      if j==8+q
         sc(i,j)=(SC(i,8)-SC(i,j))/SC(i,8);
         q=q+10;
      elseif j==9+q
         Y=SC(i,8)-SC(i,j-1);
         X=Y/SC(i,8);
         dY=sqrt(SC(i,9)^2+SC(i,j)^2);
         dX=X*sqrt((dY/Y)^2+(SC(i,9)/SC(i,8))^2);
         sc(i,j)=dX;
         sc(i,j+1)=sc(i,j);
      elseif j~=10+q
         sc(i,j)=SC(i,j);
      end
   end
   q=0;
end
q=0;
for i=1:size(M,1)
   for j=1:size(M,2)
      if j==4+q
         met(i,j)=(M(i,4)-M(i,j))/M(i,4);
         q=q+6;
      elseif j==5+q
         Y=M(i,4)-M(i,j-1);
         X=Y/M(i,4);
         dY=sqrt(M(i,5)^2+M(i,j)^2);
         dX=X*sqrt((dY/Y)^2+(M(i,4)/M(i,5))^2);
         met(i,j)=dX;
         met(i,j+1)=met(i,j);
      elseif j~=6+q
         met(i,j)=M(i,j);
      end
   end
   q=0;
end
```

Selected MATLAB functions for rate calculations

This program was used to fit steady state surface coverage data from the semibatch reactor and obtain relative rate constants for representative nanotubes.

```
function ode_sites_nd
global Ntot_range A0 fit_data tspan ic nm vals wavs index gamma

[ic A0 fit_data guess tspan Ntot_range nm vals wavs]=getinfo;
gamma=matrix;
opt=optimset('TolFun',1e-10,'TolX',1e-10,'MaxFunEvals',1e3,'MaxIter',2e3);
trial=0;
choice='y'; index=[]; sq_err=[];
lb= zeros(length(guess),1);                   %Lower bound for parameters
while choice=='y'                             %Loop continues running until convergence
  [sol_vec,d,res,d,d,d,jac]=lsqnonlin(@solver,guess,lb,[],opt);
  trial=trial+1;
  index=[index;trial];
  [err cf cd]=solver(sol_vec);
  sq_err=[sq_err;err'*err];
  plot_res(sq_err,cf,cd)
  choice=input('Continue? ','s');
  if choice=='y'
    guess=sol_vec+guess*0.001;                %Perturbation of current solution for next run
    close all
  else
    break
  end
end

function [err,cf,cd]=solver(guess)
global Nnt K_nt Ntot_range Ntot fit_data tspan ic vals wavs nm gamma
Nnt=guess(1); k=guess(2:length(guess));
for i=1:length(k)                             %Locates SWNT with maximum rate constant
  if k(i)==max(k)
    k(i)=1;
    break
  end
end
K_nt=gamma*k*1e3;
cf=[];
option=odeset('Jacobian',@jacobian);
indicator=0;
alpha_m=find_alpha; max_prod=max(K_nt)*alpha_m;
V0=5; tau=V0/max_prod;
Tspan=tspan/tau;                              %Normalisation w.r.t. chosen time scale
for Ntot=Ntot_range                           %Solves ODEs for different amounts of reagent
  [t cm]=ode15s(@rhs,Tspan,ic,option);
  cf=[cf;cm(size(cm,1),2:size(cm,2))];
end
data=[]; model=[];
for col=1:size(fit_data,2)                    %Stacks data into a column vector
  data=[data;fit_data(:,col)];
  model=[model;cf(:,col)];
end
```

```
err=(data-model);                              %Computes error at each data point
function vec=rhs(t,c)                          %ODEs defined
global Ntot K_nt Nnt A0
alpha_m=find_alpha; max_prod=max(K_nt)*alpha_m;
V0=5; v0=0.02083;
tau=V0/max_prod; Vr=V0+v0*tau*t;
Fd=Ntot/24;
vec(1)=Fd*tau/Nnt-c(1)*sum(K_nt.*A0.*(1-c(2:length(c))))*V0/(Vr*max_prod);
vec(2:length(c))=K_nt*c(1).*(1-c(2:length(c)))*V0/(Vr*max_prod);
vec=vec';

function jac=jacobian(t,c)                     %Jacobian for ODE solver
global A0 Nnt K_nt Ntot
alpha_m=find_alpha; max_prod=max(K_nt)*alpha_m;
V0=5.25; v0=0.02083;
tau=V0/max_prod; Vr=5+v0*tau*t;
N=length(c);
diag_vec(1)=-sum(K_nt.*A0.*(1-c(2:length(c))))*V0/(Vr*max_prod);
diag_vec(2:N)=-c(1)*K_nt*V0/(Vr*max_prod);
jac=diag(diag_vec);
jac(1,2:N)=c(1)*(K_nt.*A0)'*V0/(Vr*max_prod);
jac(2:N,1)=K_nt.*(1-c(2:length(c)))*V0/(Vr*max_prod);

function alpha_m=find_alpha                    %Locate absorbance for most reactive SWNT
global K_nt A0
for i=1:length(K_nt)
   if K_nt(i)==max(K_nt)
      alpha_m=A0(i);
      break
   end
end

function gamma=matrix
global vals wavs
gamma=zeros(length(wavs),size(vals,1));
for i=1:size(vals,1)
   for j=1:size(wavs,1)
      if (wavs(j)>vals(i,1)) & (wavs(j)<vals(i,2))
         gamma(j,i)=1;
      end
   end
end

function plot_res(err,cf,cd)
global fit_data Ntot_range index
plot(Ntot_range,cf)                            %Compares model to data
hold on
plot(Ntot_range,fit_data,'o')

figure(2)                                      %Plots error after each run
plot(index,err,'--rs')
xlabel('Trial')
ylabel('Squared Error')
```

MATLAB program for evolution of density gradient in centrifuge

This program computes the sedimentation of the density gradient medium (Iodixanol) in a centrifuge tube while under the influence of a centrifugal field. The effect of diffusion also has been taken into account. A finite volume method with the first order upwind scheme has been used to solve the partial differential equations. Flow of the solute is always oriented along the positive radial direction. The resulting spatiotemporal coordinates of the Iodixanol are stored in a look-up table for use in the subsequent program to calculate the concentration profiles of the nanotubes.

```
function centrif_fvm_iodix
global conc del_r t_range tau del_t D_t
clc

conc=30;                                  %Iodix conc (wt/vol)
[del_r t_range tau del_t D_t]=grid_stuff;
time_march;                               %output saved to disk
clear all

function time_march
global ic N_cells r_d t_range tau press
eta=ic;                                   %vector
d_prof=[];                                %initial density profile
press=[];
for t=t_range
   density=1005.3+8.215*eta;              %soln density (kg/m^3)
   d_prof=[d_prof density];               %row=dist,col=time
   update_vel(eta);                       %update vels for new gradient
   M_rhs=grid_mat;                        %recalc matrix
   eta=M_rhs*eta;                         %conc (mol/L)
end
T_grad=t_range*tau;                       %abs times
save C:\MATLAB7\work\Centrif\WJData\grad_data d_prof T_grad

ic=1005.3+8.215*ic;
eta=1005.3+8.215*eta;
plot(r_d,ic,r_d,eta,'r')
xlabel('Radial Distance (m)'),ylabel('Density (kg/m^3)')
axis tight
clear M_rhs eta ic count density
% xlswrite([r_d d_prof(:,1) d_prof(:,end)],[],[],'test.xls')

function update_vel(eta)
global u_t r_d del

rho=update_density(eta);                  %incl transport/pressure fx
u_t=r_d.*(1-rho/2161)/del;                %updated velocity

function rho=update_density(eta)
global w r_d del_r ic rho_ic N_cells press
rho=1005.3+8.215*eta;                     %wt/vol -> kg/m^3 (transport)

kappa=0.46e-9;                            %1/Pa (water comp)
del_P=w^2*rho(1:N_cells-1).*(r_d(2:N_cells).^2-r_d(1:N_cells-1).^2)/2;
```

```
P=101300+cumsum(del_P);
press=[press [101300;cumsum(del_P)]];
rho(2:N_cells)=rho(1:N_cells-1).*exp(kappa*del_P);

clear del_rho

function M_rhs=grid_mat
global r_d del_r del_t D_t u_t N_cells

ratio=del_t/del_r;                              %ubiquitous factor

r_f=r_d(3:N_cells);                             %'r' of front cell
r_c=r_d(2:N_cells-1);                           %'r' of curr cell
r_b=r_d(1:N_cells-2);                           %'r' of back cell
r_R=(r_f+r_c)/2;                                %posn of right edge
r_L=(r_c+r_b)/2;                                %posn of left edge

u_f=u_t(3:N_cells,:);                           %'u' of front cell
u_c=u_t(2:N_cells-1,:);                         %'u' of curr cell
u_b=u_t(1:N_cells-2,:);                         %'u' of back cell

d=[-1 0 1];                                     %diagonal vector
[A B C]=direxn(ratio,r_R,r_L,r_c,u_f,u_c,u_b);  %i+1,i,i-1 cells
[A B C]=bcells(ratio,A,B,C);                    %add boundary cell data
M_rhs=spdiags([[C;0] B [0;A]],d,N_cells,N_cells);

clear d r_f r_c r_b r_R r_L
clear ratio u_f u_c u_b

function [A,B,C]=direxn(ratio,r_R,r_L,r_c,u_f,u_c,u_b)
global del_r D_t

A=-ratio*(-r_R*D_t/del_r+r_R.*min(u_f,0))./r_c;     %i+1 (2->N_cells-1)
B=1-ratio*(D_t*(r_R+r_L)/del_r+r_R.*max(u_c,0)-r_L.*min(u_c,0))./r_c; %i (2->N_cells-1)
C=ratio*(r_L*D_t/del_r+r_L.*max(u_b,0))./r_c;       %i-1 (2->N_cells-1)

function [A,B,C]=bcells(ratio,A,B,C)
global r_d del_r D_t u_t N_cells

r_R=(r_d(2)+r_d(1))/2;
r_L=(r_d(N_cells)+r_d(N_cells-1))/2;

%1st cell BC
A_1=-ratio*(-r_R*D_t/del_r+r_R*min(u_t(2),0))/r_d(1);
B_1=1-ratio*(r_R*D_t/del_r+r_R*max(u_t(1),0))/r_d(1);

%Nth cell BC
C_N=ratio*(r_L*D_t/del_r+r_L*max(u_t(N_cells-1),0))/r_d(N_cells);
B_N=1+ratio*(-r_L*D_t/del_r+r_L*min(u_t(N_cells),0))/r_d(N_cells);

A=[A_1;A];                                      %i+1 (1->N_cells-1)
B=[B_1;B;B_N];                                  %i (1->N_cells)
C=[C;C_N];                                      %i-1 (2->N_cells)

clear A_* B_* C_*
```

```
function [del_r,t_range,tau,del_t,D_t]=grid_stuff
global conc w r_d ic rho_ic N_cells del

w=pi*32000/30;                                  %rad/s

r_a=0.0718-0.00566; r_b=0.1688; del=r_b-r_a;    %computation bdys (m)

kT=4.0721e-21;                                  %J/atom
D=2.5e-10;                                      %Iodixanol diffusion coeff (m^2/s)
T=22*3600; tau=kT/(2.5833e-24*D*w^2);           %run time (sec)
t_lim=T/tau;

p_bd=[0.0979 0.1488];                           %original grad bdys (m)
slope=79.6/diff(p_bd);                          %initial grad slope
p_inj=p_bd(1)+5*diff(p_bd)/6;                   %SWNT injexn pt (m)
sp=0.00566;                                     %spread of SWNT soln (m)
p_sp=p_inj+sp*[-0.35 0.65];                     %bdys of spread (m)
p_bd=p_bd-[sp 0];                               %new grad bdys (m)

N_cells=101; del_r_d=del/(N_cells-1);           %cell bdys,dim spacing
del_r=del_r_d/del; del_t=0.1*del_r;             %non-dim spacing,time step
t_range=0:del_t:t_lim;
r_d=(r_a:del_r_d:r_b)';                         %spatial grid (m)
r_d=chop(r_d,4);                                %chop to 4 decimal places
i_bd=find(r_d==p_bd(1) | r_d==p_bd(2));         %bdy indices
i_bd=modify(i_bd,p_bd,del_r_d);                 %insert non-existent pts
i_sp=find(r_d==p_sp(1) | r_d==p_sp(2));         %spread indices
i_sp=modify(i_sp,p_sp,del_r_d);                 %insert non-existent pts

%overlayer
rho(1:i_bd(1)-1)=1005.3;
%1st sloped region
density=1005.3+5.3*conc;                        %starting density (kg/m^3)
rho(i_bd(1):i_sp(1))=density+slope*(r_d(i_bd(1):i_sp(1))-r_d(i_bd(1)));
%SWNT injexn layer
rho(i_sp(1)+1:i_sp(2)-1)=rho(i_sp(1));
%2nd sloped region
rho(i_sp(2):i_bd(2))=rho(i_sp(1))+slope*(r_d(i_sp(2):i_bd(2))-r_d(i_sp(2)));
%stop layer
rho(i_bd(2)+1:length(r_d))=1323.3;
rho_ic=rho';
ic=(rho_ic-1005.3)/8.215;                       %conc (mol/L)
plot(r_d,rho)
fac=1e-23;                                      %kg
D_t=kT./(w^2*0.2573*fac*del^2);                 %dim-less number

clear temp d_* kT del fac rho

function chopped=chop(toRound,decimalPlaces)
powerOf=10^decimalPlaces;
chopped=toRound.*powerOf;
remainder=mod(chopped,1);
whole=chopped-remainder;
chopped=whole/powerOf;

function ind=modify(ind,p,del_r_d)
```

```
global r_d

if length(ind)==2                                %no change reqd
   ind=ind;
elseif length(ind)==1                            %add 1 elem to 'r_d'
   i=find(p~=r_d(ind));                          %elem not in 'r_d'
   gap=abs(r_d-p(i));                            %absolute distances
   min_gap=min(gap);                             %closest points
   ind=[ind find(gap==min_gap)];                 %indices of closest pts
elseif length(ind)==0                            %add 2 elem to 'r_d'
   gaps=[abs(r_d-p(1)) abs(r_d-p(2))];           %absolute distances
   min_gap=[min(gaps(:,1)) min(gaps(:,2))];      %closest points
   ind=find(gaps==min_gap(1) | gaps==min_gap(2))';%indices of closest pts
   ind=ind-size(r_d,1)*[0 1];                    %row numbers
end
ind=sort(ind);
```

MATLAB program for diameter-based separation of SWNT in centrifuge

The following code is used to fit experimentally obtained concentration profiles of carbon nanotubes that have been separated according to diameter in a density gradient via ultracentrifugation. The density of the sedimenting medium at all times and positions are obtained by interpolating from the look-up table created by the preceding program. The sole fit parameter is the number of surfactant molecules adsorbed per unit length of the nanotube sidewall.

```
function centrif_fvm_mod
global n_ad N_n L_range N_L w d_prof T_grad dia
global del del_r del_t N_c r_data eta_data run_time

clc
warning off all
datafile='Centrif_32';                        %Excel file with data
tube='87';                                    %SWNT of interest
load C:\MATLAB7\work\Centrif\WJData\grad_data %calculated gradient profile
run_time=22*3600;                             %run time (sec)

[r_data eta_data w n_ad N_n L_range N_L dia]=get_data([datafile '.xls'],tube);
[del N_c del_r del_t]=grid_stuff;
[eta_model err]=time_march;
% [low up]=conf_int(err)
plot_res(eta_model,err);
clear all

function [r_data,eta_data,w,n_ad,N_n,L_range,N_L,dia]=get_data(file,tube)
global N_s num_C n_D

data=xlsread(file,tube);
r_data=data(:,1);                             %radial positions (m)
eta_data=data(:,2);                           %relative conc

w=pi*32000/30;                                %angular speed of rotor (rad/s)

n_1=4.05; n_2=0;
n_ad=[n_1 n_2];
[N_n N_s]=size(n_ad);                         %# of choices,# of surfs

start=10; stop=1010; num=20;
step=(stop-start)/num;
L_range=(start:step:stop)';
N_L=length(L_range);

n=str2num(tube(1)); m=str2num(tube(2));
data=xlsread('CoMoData.xls','Data');
row=find(data(:,1)==n & data(:,2)==m);        %find row corresp to "n","m"
dia=data(row,3);                              %diameters (nm)
num_C=data(row,11);                           %#C/nm
n_D=0;
% n_D=num_C/10;
clear start step stop data

function [del,N_c,del_r,del_t]=grid_stuff
global r_d N_L N_n w ic exp_int exp_all
```

```
r_a=0.0661; r_b=0.1688; del=r_b-r_a;             %computation bdys (m)

p_bd=[0.0979 0.1488];                            %original grad bdys (m)
sp=0.00566;                                      %spread of SWNT soln (m)
p_inj=p_bd(1)+5*diff(p_bd)/6;                    %SWNT injexn pt (m)
p_sp=p_inj+sp*[-0.35 0.65];                      %bdys of spread (m)
p_bd=p_bd-[sp 0];                                %new grad bdys (m)

N_c=101; del_r_d=del/(N_c-1);                    %cell bdys,dim spacing
del_r=del_r_d/del; del_t=1*del_r;                %non-dim spacing,time step
r_d=(r_a:del_r_d:r_b)';                          %spatial grid (m)
r_d=chop(r_d,4);                                 %chop to 4 decimal places
i_sp=find(r_d==p_sp(1) | r_d==p_sp(2));          %spread indices
i_sp=modify(i_sp,p_sp,del_r_d);                  %insert non-existent pts

ic=zeros(length(r_d),1); ic(i_sp(1):i_sp(2),:)=1;    %unif conc over this range
ic=repmat(ic,N_L,1);                             %"ic" for all lengths
ic=repmat(ic,N_n,1);                             %"ic" for all "n_ad"

exp_int=zeros(N_L,N_L*(N_c-2));                  %expand "D" for interior
exp_all=zeros(N_L,N_L*N_c);                      %expand "D" for all cells
for i=1:N_L
   int_range=(i-1)*(N_c-2)+1:i*(N_c-2);          %range of cols to be filled
   all_range=(i-1)*N_c+1:i*N_c;
   exp_int(i,int_range)=ones(1,length(int_range));
   exp_all(i,all_range)=ones(1,length(all_range));
end

clear temp int_* all_*

function ind=modify(ind,p,del_r_d)
global r_d

if length(ind)==2                                %no change reqd
   ind=ind;
elseif length(ind)==1                            %add 1 elem to 'r_d'
   i=find(p~=r_d(ind));                          %elem not in 'r_d'
   gap=abs(r_d-p(i));                            %absolute distances
   min_gap=min(gap);                             %closest points
   ind=[ind find(gap==min_gap)];                 %indices of closest pts
elseif length(ind)==0                            %add 2 elem to 'r_d'
   gaps=[abs(r_d-p(1)) abs(r_d-p(2))];           %absolute distances
   min_gap=[min(gaps(:,1)) min(gaps(:,2))];      %closest points
   ind=find(gaps==min_gap(1) | gaps==min_gap(2))';%indices of closest pts
   ind=ind-size(r_d,1)*[0 1];                    %row numbers
end
ind=sort(ind);

function [eta_model,err]=time_march
global N_c N_n N_L ic r_data r_d eta_data del_t run_time

eta=ic;                                          %initialise for each "n_ad"
[m_tot density tau]=calc_qty;                    %output calc from "n_ad"
tau_min=min(tau);                                %time constant for fastest NT
t_L=run_time./tau;                               %scaled times for each L
```

```
t_L=chop(t_L,2);
t_L_start=t_L;                              %"t_L" is reduced later
t_max=max(t_L);                             %max scaled time (loop limit)
prof=zeros(N_n*N_L*N_c,1);
for t=0:del_t:t_max                         %march fwd wrt fastest NT
   ind=find(t_L<=t);                        %length indices to be removed
   if ~isempty(ind)
      range=[];
      for i=ind'
         j=find(t_L_start==t_L(i));         %actual posns of removed pts
         range=[range;(j-1)*N_c+1:j*N_c];   %collect ranges to be extracted
      end
      prof(range)=eta(range);               %collect completed profiles
      t_L(ind)=[];
   end
   [U R]=update_vel(t*tau_min,density);
   M_rhs=grid_mat(U,R);
   eta=M_rhs*eta;
end
[eta_model err]=profiles(prof);

clear M_rhs eta ic profs

function [m_tot,density,tau]=calc_qty
global del L_range N_L n_ad N_n N_s N_c num_C n_D dia w exp_int exp_all D_int D_all

N_av=6.023e23;                              %Avogadro's Num (molec/mol)
NT_mass=num_C*0.012/N_av;                   %SWNT mass per length (kg/nm)
diaz_mass=n_D*0.093/N_av;
n_2=1;

term_1=[];
for i=1:N_n
   term_1=[term_1;repmat(n_ad(i,:),N_L,1)];  %replicate "n_ad" for each "L"
end
term_2=repmat(L_range,N_n,1);
if N_s==1                                   %one surfactant used
   mol_mass=0.43055/N_av;                   %surf molec mass (kg/molecule)
   mol_vol=6.13e-4/N_av;                    %surf molec vol (m^3/molecule)
else
   mol_mass=[0.43055 0.28838]/N_av;
   mol_vol=[6.13e-4 4.03e-4]/N_av;
end
surf_mass=sum(term_1.*repmat(mol_mass,size(term_1,1),1),2);
surf_mass=surf_mass+n_2*mol_mass(1);
m_tot=(NT_mass+diaz_mass+surf_mass).*term_2;  %total mass for each length

r_C=0.172;                                  %radius (nm)
NT_vol=0.25*pi*(dia+2*r_C)^2*1e-27;         %CNT vol (m^3/molec)

diaz_vol=105.304e-30;
surf_vol=sum(term_1.*repmat(mol_vol,size(term_1,1),1),2);
tot_vol=NT_vol+n_D*diaz_vol+surf_vol;       %m^3/length replicated by "L"
density=(NT_mass+diaz_mass+surf_mass)./tot_vol;  %kg/m^3 replicated by "L"
density(1)
```

```
kT=4.07e-21;                                          %J/atom
D=kT./(m_tot*w^2*del^2);                              %n-d diff coeff for each "L"
D_int=zeros(N_n*N_L*(N_c-2),1);
D_all=zeros(N_n*N_L*N_c,1);
temp=zeros(N_n*N_L*N_c,1);
for i=1:N_n
  range=(i-1)*N_L+1:i*N_L;
  D_int((i-1)*N_L*(N_c-2)+1:i*N_L*(N_c-2))=D(range)'*exp_int;
  D_all((i-1)*N_L*N_c+1:i*N_L*N_c)=D(range)'*exp_all;
  temp((i-1)*N_L*N_c+1:i*N_L*N_c)=density(range)'*exp_all;
end
density=temp;

f=3*pi*8.94e-4*term_2*1e-9./(log(term_2/dia)+0.32);
tau=f./(m_tot*w^2);                                   %time consts for each L (s)

clear term_* temp range f

function chopped=chop(toRound,decimalPlaces)
powerOf=10^decimalPlaces;
chopped=toRound.*powerOf;
remainder=mod(chopped,1);
whole=chopped-remainder;
chopped=whole/powerOf;

function [U,R]=update_vel(t,density)
global r_d del d_prof T_grad w L_range N_c N_n N_L
ind=max(find(T_grad<=t));                             %interp posn
alpha=(T_grad(ind+1)-t)/(T_grad(ind+1)-T_grad(ind));
rho=alpha*d_prof(:,ind)+(1-alpha)*d_prof(:,ind+1);  %density profile at "t"
u=repmat(r_d,N_n*N_L,1).*(1-repmat(rho,N_n*N_L,1)./density)/del;
r_exp=repmat(r_d,N_n*N_L,1);

temp=zeros(N_c,1);
temp(1:N_c-2)=1;                                      %back cell
ind=find(repmat(temp,N_n*N_L,1)==1);                  %back cell positions
u_b=u(ind);                                           %vel in back cell
r_b=r_exp(ind);                                       %'r' of back cell

temp=zeros(N_c,1);
temp(2:N_c-1)=1;                                      %current cell
ind=find(repmat(temp,N_n*N_L,1)==1);                  %current cell positions
u_c=u(ind);                                           %vel in current cell
r_c=r_exp(ind);                                       %'r' of front cell

temp=zeros(N_c,1);
temp(3:N_c)=1;                                        %forward cell
ind=find(repmat(temp,N_n*N_L,1)==1);                  %forward cell positions
u_f=u(ind);                                           %vel in forward cell
r_f=r_exp(ind);

u_R=(u_f+u_c)/2;                                      %vel at right edge
u_L=(u_c+u_b)/2;                                      %vel at left edge
r_R=(r_f+r_c)/2;                                      %posn of right edge
r_L=(r_c+r_b)/2;                                      %posn of left edge
clear *_b *_f
```

```
den_uniq=unique(density);                         %pluck out unique densities
down=[]; up=[];
template_d=zeros(N_n*N_L*(N_c-2),1);              %template for down motion
template_u=zeros(N_n*N_L*(N_c-2),1);              %template for up motion
for i=1:N_n
  v_inv=max(find(rho(2:N_c-1)<den_uniq(i)));

  temp=zeros((N_c-2),1);                          %replicated in length
  start=(i-1)*N_L*(N_c-2)+1; stop=i*N_L*(N_c-2);
  gap=v_inv-1;                                    %added to starting point
  temp(1:(1+gap))=1;                              %down range=non-zero
  template_d(start:stop)=repmat(temp,N_L,1);      %same for all "L" per "n_ad"

  temp=zeros((N_c-2),1);                          %replicated in length
  start=(i-1)*N_L*(N_c-2)+1; stop=i*N_L*(N_c-2);
  gap=N_c-v_inv-3;                                %added to starting point
  temp((v_inv+1):(v_inv+1+gap))=1;                %up range=non-zero
  template_u(start:stop)=repmat(temp,N_L,1);      %same for all "L" per "n_ad"
end

u_R_d=template_d; u_L_d=template_d;
r_R_d=template_d; r_L_d=template_d; r_c_d=template_d;
ind=find(u_R_d==1);                               %indices of "down" range
u_R_d(ind)=u_R(ind); u_L_d(ind)=u_L(ind);
r_R_d(ind)=r_R(ind); r_L_d(ind)=r_L(ind); r_c_d(ind)=r_c(ind);

u_R_u=template_u; u_L_u=template_u;
r_R_u=template_u; r_L_u=template_u; r_c_u=template_u;
ind=find(u_R_u==1);                               %indices of "up" range
u_R_u(ind)=u_R(ind); u_L_u(ind)=u_L(ind);
r_R_u(ind)=r_R(ind); r_L_u(ind)=r_L(ind); r_c_u(ind)=r_c(ind);

U=[u_R_d u_L_d u_R_u u_L_u];
R=[r_R_d r_L_d r_c_d r_R_u r_L_u r_c_u];

clear den_* ind alpha v_inv temp st* template_*
clear *_R_* *_L_*

function M_rhs=grid_mat(U,R)
global del_r del_t N_c N_L N_n
ratio=del_t/del_r;                                %ubiquitous factor
d=[-1 0 1];                                       %diagonal vector
[A_i B_i C_i]=direxn(ratio,U,R);                  %interior cells
[A_f B_f C_f]=bcells(ratio,U,A_i,B_i,C_i);        %A/C=N_c-1,B=N_c (size)
n_row=N_n*N_L*N_c;
M_rhs=spdiags([[C_f;0] B_f [0;A_f]],d,n_row,n_row);

clear ratio *_f U R

function [A,B,C]=direxn(ratio,U,R)
global r_d del_r N_c L_range N_L N_n D_int

u_R_d=U(:,1); u_L_d=U(:,2);
u_R_u=U(:,3); u_L_u=U(:,4);
```

```
r_R_d=R(:,1); r_L_d=R(:,2); r_c_d=R(:,3);
r_R_u=R(:,4); r_L_u=R(:,5); r_c_u=R(:,6);

%used when particle moves down (r<r0)
A_d=-ratio*D_int.*r_R_d./(del_r*r_c_d);
B_d=ratio*(D_int.*(r_R_d+r_L_d)/del_r+u_R_d.*r_R_d)./r_c_d;
C_d=-ratio*(D_int.*r_L_d/del_r+u_L_d.*r_L_d)./r_c_d;

%used when particle moves up (r>r0)
A_u=ratio*(-D_int.*r_R_u/del_r+u_R_u.*r_R_u)./r_c_u;
B_u=ratio*(D_int.*(r_R_u+r_L_u)/del_r-u_L_u.*r_L_u)./r_c_u;
C_u=-ratio*D_int.*r_L_u./(del_r*r_c_u);

%replace "NaN" in "down" vector with finite values in "up" vector
A_d(find(isnan(A_d)==1))=A_u(find(isnan(A_u)==0));
B_d(find(isnan(B_d)==1))=B_u(find(isnan(B_u)==0));
C_d(find(isnan(C_d)==1))=C_u(find(isnan(C_u)==0));

A=-A_d;                                       %i+1 (2->N_c-1)
B=1-B_d;                                      %i (2->N_c-1)
C=-C_d;                                       %i-1 (2->N_c-1)

clear A_* B_* C_* *_R_* *_L_* *_c_*

function [A_f,B_f,C_f]=bcells(ratio,U,A_i,B_i,C_i);
global r_d del_r N_c L_range N_L N_n D_all

r_R=repmat([(r_d(2)+r_d(1))/2;zeros(N_c-1,1)],N_n*N_L,1);
r_L=repmat([zeros(N_c-1,1);(r_d(N_c)+r_d(N_c-1))/2],N_n*N_L,1);

u_L_d=U(:,2);                                 %for 1st BC (1st cell)
u_R=zeros(N_n*N_L*N_c,1);                     %consider all cells
u_R(1:N_c:N_n*N_L*N_c)=u_L_d(1:(N_c-2):N_n*N_L*(N_c-2));
u_R_u=U(:,3);                                 %for 2nd BC (N_cell-th cell)
u_L=zeros(N_n*N_L*N_c,1);                     %consider all cells
u_L(N_c:N_c:N_n*N_L*N_c)=u_R_u((N_c-2):(N_c-2):N_n*N_L*(N_c-2));

A_f=repmat(zeros(N_L*(N_c+1),1),N_n,1);       %pad with ghost cell at the end
C_f=repmat(zeros(N_L*(N_c+1),1),N_n,1);       %pad with ghost cell in front

%1st cell BC
range=1:N_n*N_L*(N_c+1);                      %1->N_c+1 for each length
range((N_c+1):(N_c+1):N_n*N_L*(N_c+1))=[];    %N_cell+1-th cells removed
A_f(range)=ratio*D_all.*r_R/(del_r*r_d(1));   %fill 1->N_c for each "L"
rem_range=N_c:(N_c+1):N_n*N_L*(N_c+1);        %remove N_cell-th cell
A_f(rem_range)=[];                            %1->N_c-1 for each length
B_1=1-ratio*(D_all.*r_R/del_r+u_R.*r_R)/r_d(1);

%Nth cell BC
range=1:N_n*N_L*(N_c+1);                      %1->N_c+1 for each length
range(1:(N_c+1):N_n*N_L*(N_c+1))=[];          %1st cells removed
C_f(range)=ratio*D_all.*r_L/(del_r*r_d(N_c)); %fill 1->N_c for each "L"
rem_range=2:(N_c+1):N_n*N_L*(N_c+1);          %remove 1st cell (below ghost)
C_f(rem_range)=[];                            %2->N_c for each length
B_N=1-ratio*(D_all.*r_L/del_r-u_L.*r_L)/r_d(N_c);
```

```
range=1:N_n*N_L*N_c;                          %"N_c" cells for all L
cell_1=1:N_c:N_n*N_L*N_c;                     %posns of 1st cell
cell_N=N_c:N_c:N_n*N_L*N_c;                   %posns of Nth cell
range([cell_1 cell_N])=[];                    %remove 1st and last cells
A_f(range)=A_i;                               %fill in interior cells
A_f(end)=[];                                  %remove end cell for "spdiags"

B_f(cell_1)=B_1(cell_1);                      %fill in 1st cells
B_f(cell_N)=B_N(cell_N);                      %fill in last cells
B_f(range)=B_i;                               %fill in interior cells
B_f=B_f';

C_f(range)=C_i;                               %fill in interior cells
C_f(1)=[];                                    %remove 1st cell for "spdiags"

clear A_i B_i C_i *range cell_*

function [eta_model,err]=profiles(prof)
global r_d r_data eta_data L_range N_L N_c N_n

mu=311; sig=290;
P_L=1./(sqrt(2*pi)*sig).*exp(-(L_range-mu).^2/(2*sig^2));
P_L=repmat(P_L,1,N_n);
P_L=ones(size(P_L));
temp=[]; n_prof=[];
for i=1:N_n*N_L
   range=(i-1)*N_c+1:i*N_c;
   if rem(i,N_L)~=0                           %collect profs for all L
      temp=[temp prof(range)*P_L(i)];
   else
      n_prof=[n_prof sum(temp,2)];            %collect profs for all "n_ad"
      temp=[];
   end
end
plot(r_d,n_prof)
eta_model=interp1(r_d,n_prof,r_data);         %prof corresp to data
eta_max=repmat(max(eta_model,[],1),length(r_data),1);  %max for each "n_ad"
eta_model=eta_model./eta_max;
err=repmat(eta_data,1,N_n)-eta_model;
err=sum(err.^2,1);                            %sq err for each "n_ad"

clear range temp P_L

function [low,up]=conf_int(err)
global r_data n_ad ic N_n N_s n_0

ic_temp=ic;
N_beta=1;                                     %num of parameters for N_s=1
% temp=[1 0;1 0];
temp=[0 1;0 1];
beta=temp*n_ad';                              %nominal value vector (2x1)
gamma=repmat(n_ad(1,2),size(beta,1),1);       %only one surf
delta=repmat(0.0001,size(beta,1),1);          %starting perturbn (2x1)
coeff=[-1;1];                                 %'delta' coeffs (2x1)
n_ad=repmat(n_ad,size(beta,1),1);             %2x1
n_d=size(r_data,1);                           %number of data points
```

```
F_val=finv(0.95,N_beta,n_d-N_beta);
err_bnd=err*(1+F_val*N_beta/(n_d-N_beta))          %err bnd for 95% conf level

while 1>0
  beta_old=beta;                                    %'beta' frm prev step
  beta=beta+delta.*coeff;                           %current 'beta'
  n_ad(find(temp==1))=beta;
  n_ad(find(temp==0))=n_0-n_ad(:,2);
  N_n=size(n_ad,1);
  ic=repmat(ic_temp,N_n,1);                         %"ic" for all "n_ad"
  [eta_model,err]=time_march;
  ind=find(err<=err_bnd)';
  [ind n_ad(ind,:) err(ind)']
  ind=find(err>err_bnd)';
  beta(ind)=beta_old(ind);
  delta(ind)=0.0001;                                %modfify corresp 'delta'

  if length(ind)==size(beta,1)                      %all exceed 'err_bnd'
    break
  end
end

low=beta(1)-0.0001;                                 %lower bound
up=beta(2)-0.0001;                                  %upper bound

function plot_res(eta_model,err)
global r_data r_d eta_data n_ad

err
plot(r_data,eta_data,'o',r_data,eta_model,'-r')
xlabel('Radial Distance (m)'),ylabel('Relative Concentration'),hold on
```

C program for nanostructure-guided chain reactions

The suite of programs (main + headers) outlined below is used to calculate the properties of the reaction wave in a nanostructure coated with energetic molecules. As described in the main text, a carbon nanotube has been approximated by a 1D chain of oscillators interacting with both harmonic and anharmonic force fields. The output is a table of reaction times relative to the origin at $t = 0$ at each node in the lattice and for each Monte Carlo run. A separate MATLAB program is then invoked to generate statistics on these results and calculate the velocity of the reaction wave and plot the various figures in Chapter 5.

Main program:

```
#include <math.h>
#include <stdlib.h>
#include <gsl/gsl_rng.h>
#include "mex.h"
#include "pars.h"                              /*parameter file*/
#include "funcns.h"                            /*function definitions*/
#define pi 3.141593
#define small 1e-10
#define N 100
#define nmol 100

gsl_rng *rng;                                  /*random number generator (global defn)*/
int rb[nmol],n_r[nmol],Ng;
double F_s[3*N],F_b[3*N],F_nb[3*N],mass[N],t_rxn[nmol],u_cm[3];
double r[3*N],u[3*N],E_s[3],E_b[3],E_nb[3],E_k[3],m_sc;
double IS[N],uE[N],v_0[N];

void mexFunction(int nlhs,mxArray *plhs[],int nrhs,const mxArray *prhs[])
{
   exec_loops();
   gsl_rng_free(rng);
}

void exec_loops(void)
{
   int m;
   double Ea,alpha,beta_x,beta_y,vals[2],T;
   FILE *f;

   f=fopen("res.txt","w+");
   rng=gsl_rng_alloc(gsl_rng_mt19937);         /*MT algorithm*/
   gsl_rng_set(rng,time(NULL));                /*seed RNG*/

   for(Ea=5;Ea<41;Ea+=5)                       /*activn energy loop*/
      for(alpha=0.0;alpha<1.01;alpha+=0.1)     /*KE frxn loop*/
         for(m=1;m<=nruns;m++)                 /*MC trials loop*/
         {
            initialise();                      /*init 'r','u',masses*/
            equilib();                         /*set at ~300 K*/
            vals[0]=Ea; vals[1]=alpha;
            calc_force(&T,vals,f);
```

```
        }
    fclose(f);
}

void equilib(void)
{
    int i=0,j=0;
    double fac,tol,T,T_des,flag[1]={0};
    FILE *d;

    T_des=300; tol=T_des/20.0;
    for(i=0;i<100;i++)
    {
        calc_force(&T,flag,d);                    /*force computation-rxn*/

        if(T<0.67*T_des) fac=sqrt(0.67*T_des/T);
        else fac=sqrt(T_des/T);
        for(j=0;j<N;j++)
            u[j]*=fac;                            /*scale vels*/
        if(fabs(T-T_des)<tol) break;              /*exit loop*/
    }
}

void initialise(void)
{
    int i;
    double sum1,sum2,sum3,step;

    Ng=(int)(N-nmol)/2;                           /*half the # of ghost beads*/
    b1=1;                                         /*locn of 1st reac bead*/
    assign(0);                                    /*init masses*/
    assign(1);                                    /*init rxn stuff*/
    sum1=sum2=sum3=0.0;

    for(i=0;i<N;i++)
    {
        r[i]=r[N+i]=r[2*N+i]=0.0;

        u[i]=gsl_rng_uniform(rng)/100;
        u[N+i]=gsl_rng_uniform(rng)/100;
        u[2*N+i]=gsl_rng_uniform(rng)/100;

        sum1+=u[i];
        sum2+=u[N+i];
        sum3+=u[2*N+i];
    }

    for(i=0;i<N;i++)                              /*subtract CM vel*/
    {
        u[i]-=sum1/N;
        u[N+i]-=sum2/N;
        u[2*N+i]-=sum3/N;

        r[i]=i-u[i]*dt;
        r[N+i]-=u[N+i]*dt;
        r[2*N+i]-=u[2*N+i]*dt;
```

```
    }
}

void calc_force(double *T,double *vals,FILE *f)
{
    int count,i,start,stop;

    m_sc=m_b+load*m_e;                              /*mass scale (kg)*/
    t_sc=sqrt(1.4395*m_sc/k_0);                     /*time scale (s)*/
    u_sc=1e-10*r_0/t_sc;                            /*vel scale (m/s)*/

    if(vals[0]>0.0) acceptor(0.01,b1-1,vals,0);     /*cleave 1st bond*/

    start=100; stop=2099;
    for(count=1;count<=steps;count++)               /*time march loop*/
    {
        if(vals[0]>0.0) sub_vel_cm();               /*calc CM vel & subtract*/

        assign(2);                                  /*zero elems in F,E_k arrays*/
        stretch();
        bond();
        LJ();
        update(T,0);                                /*update posns*/

        assign(2);                                  /*zero elems in F,E_k arrays*/
        stretch();
        bond();
        LJ();
        update(T,1);                                /*update posns*/

        if(vals[0]>0.0)
        {
            prob(vals,count);                       /*call rxn code*/
            /*if(count>=start && count <=stop)
                calc_acf(start,count,f);            /*VACF*/

            /*calc_modes(f);                        /*energy in modes*/
            /*calc_dist(f);                         /*inter-bead dists*/
            /*calc_image(f);                        /*bead energies at each instant*/
            /*calc_wav_locn(f);                     /*wav locn,std devn*/
            /*calc_E(f);                            /*total E as func of time*/
            /*calc_curr(f);                         /*heat curr*/
        }
    }
    *T*=15.524e30/(3*N-3);                          /*temp (K)*/
    if(vals[0]>0) write_array(f);                   /*write 't_rxn' into file*/
    /*if(vals[0]>0) for(i=0;i<N;i++) fprintf(f,"%f %f %f\n",r[i],r[N+i],r[2*N+i]);*/
}

void stretch(void)
{
    int i,j;
    double k_T[2],r_1,d[2],f1,f2,dot_prod,diff,E,prod;
    double s,k,r_ij,x,y,z,sum0,sum1,sum2,dx,dy,dz,dr;

    k_T[0]=1000/k_0; k_T[1]=700/k_0;                /*spring consts (nd)*/
```

```
   r_1=10.5/r_0;                                    /*cutoff (nd)*/
   d[0]=1; d[1]=r_1-k_T[0]/k_T[1]*(r_1-1);          /*sub qtys (nd)*/
   sum0=sum1=sum2=0.0;

   for(i=0;i<N-1;i++)
   {
      choose_par(i,k_T,r_1,d,&s,&k,&r_ij);          /*par for RHS*/
      x=(r[i]-r[i+1])/r_ij;                         /*x-comp factor;denom=centre-centre dist*/
      y=(r[N+i]-r[N+i+1])/r_ij;                     /*y-comp factor*/
      z=(r[2*N+i]-r[2*N+i+1])/r_ij;                 /*z-comp factor*/
      dx=r[i]-i;
      dy=r[N+i];
      dz=r[2*N+i];
      dr=sqrt(dx*dx+dy*dy+dz*dz);

      diff=r_ij-1;                                  /*RHS force magnitude*/
      f1=-diff*(1+diff*diff*r_0*r_0);               /*FPU-β case*/
      f1=-diff;                                     /*harmonic case*/
      f2=0;
      f2=-dr*(1+dr*dr);                             /*Φ^4 case*/

      F_s[i]+=f1*x+f2*dx/dr;                        /*x-comp for curr bead*/
      F_s[N+i]+=f1*y+f2*dy/dr;                      /*y-comp for curr bead*/
      F_s[2*N+i]+=f1*z+f2*dz/dr;                    /*z-comp for curr bead*/
      dot_prod=F_s[i]*u[i]+F_s[N+i]*u[N+i]+F_s[2*N+i]*u[2*N+i];
      IS[i]=(r[i+1]-r[i])*dot_prod;                 /*stretch comp for 'i'*/

      F_s[i+1]-=f1*x;                               /*x-comp for 2nd bead*/
      F_s[N+i+1]-=f1*y;                             /*y-comp for 2nd bead*/
      F_s[2*N+i+1]-=f1*z;                           /*z-comp for 2nd bead*/
      dot_prod=F_s[i+1]*u[i+1]+F_s[N+i+1]*u[N+i+1]+F_s[2*N+i+1]*u[2*N+i+1];
      /*IS[i+1]+=(r[i]-r[i+1])*dot_prod;            /*stretch comp for 'i+1'*/

      E=r_0*r_0*diff*diff*diff*diff/4.0;
      j=fabs(i-b1+1);
      sum0+=E;                                      /*0-th moment*/
      sum1+=j*E;                                    /*1st moment*/
      sum2+=j*j*E;                                  /*2nd moment*/
   }
   E_s[0]=sum0*r_0*r_0;
   E_s[1]=sum1*r_0*r_0;
   E_s[2]=sum2*r_0*r_0;
}

void choose_par(int i,double *k_T,double r_1,double *d,double *s,double *k,double *r_ij)
{
   double a,dx,dy,dz;

   dx=r[i]-r[i+1];                                  /*x-comp*/
   dy=r[N+i]-r[N+i+1];                              /*y-comp*/
   dz=r[2*N+i]-r[2*N+i+1];                          /*z-comp*/
   *r_ij=sqrt(dx*dx+dy*dy+dz*dz);                   /*edge-edge dist*/

   if(*r_ij<r_1)
   {
      *s=d[0];                                      /*sub qty*/
```

```
        *k=k_T[0];                                    /*spring const*/
    }
    else
    {
        *s=d[1];                                      /*sub qty*/
        *k=k_T[1];                                    /*spring const*/
    }
}

void bond(void)
{
    int i,j;
    double d_1[3],d_2[3],mod_1,mod_2,prod,cos_phi,sin_phi,E,product;
    double term_1,term_2,phi=0,w_1[3],w_2[3],sum0,sum1,sum2,k_B,dot_prod;

    k_B=14300/(k_0*r_0*r_0);                          /*bending const (nd)*/
    sum0=sum1=sum2=0.0;

    for(i=0;i<N-2;i++)                                /*bead index (denom)*/
    {
        d_1[0]=r[i]-r[i+1];                           /*LHS bond direxn (x)*/
        d_1[1]=r[N+i]-r[N+i+1];                       /*LHS bond direxn (y)*/
        d_1[2]=r[2*N+i]-r[2*N+i+1];                   /*LHS bond direxn (z)*/
        mod_1=sqrt(d_1[0]*d_1[0]+d_1[1]*d_1[1]+d_1[2]*d_1[2]);

        d_2[0]=r[i+2]-r[i+1];                         /*RHS bond direxn (x)*/
        d_2[1]=r[N+i+2]-r[N+i+1];                     /*RHS bond direxn (y)*/
        d_2[2]=r[2*N+i+2]-r[2*N+i+1];                 /*RHS bond direxn (z)*/
        mod_2=sqrt(d_2[0]*d_2[0]+d_2[1]*d_2[1]+d_2[2]*d_2[2]);

        prod=d_1[0]*d_2[0]+d_1[1]*d_2[1]+d_1[2]*d_2[2];
        cos_phi=prod/(mod_1*mod_2);
        if(cos_phi>1.0) cos_phi=1.0;
        if(cos_phi<-1.0) cos_phi=-1.0;

        phi=acos(cos_phi);                            /*obtuse bond angle*/
        term_1=-k_B*(phi-pi);

        sin_phi=sqrt(1-cos_phi*cos_phi);
        if(sin_phi<1.0e-3)                            /*avoid denom=0*/
            term_2=1/(1.0e-3*mod_1*mod_2);
        else
            term_2=1/(sin_phi*mod_1*mod_2);

        w_1[0]=prod/(mod_1*mod_1)*d_1[0]-d_2[0];
        w_1[1]=prod/(mod_1*mod_1)*d_1[1]-d_2[1];
        w_1[2]=prod/(mod_1*mod_1)*d_1[2]-d_2[2];
        w_2[0]=prod/(mod_2*mod_2)*d_2[0]-d_1[0];
        w_2[1]=prod/(mod_2*mod_2)*d_2[1]-d_1[1];
        w_2[2]=prod/(mod_2*mod_2)*d_2[2]-d_1[2];

        F_b[i]+=term_1*term_2*w_1[0];                 /*force on left bead*/
        F_b[N+i]+=term_1*term_2*w_1[1];
        F_b[2*N+i]+=term_1*term_2*w_1[2];
        dot_prod=F_b[i]*u[i]+F_b[N+i]*u[N+i]+F_b[2*N+i]*u[2*N+i];
```

```
    F_b[i+1]-=term_1*term_2*(w_1[0]+w_2[0]);    /*force on centre bead*/
    F_b[N+i+1]-=term_1*term_2*(w_1[1]+w_2[1]);
    F_b[2*N+i+1]-=term_1*term_2*(w_1[2]+w_2[2]);
    dot_prod=F_b[i+1]*u[i+1]+F_b[N+i+1]*u[N+i+1]+F_b[2*N+i+1]*u[2*N+i+1];

    F_b[i+2]+=term_1*term_2*w_2[0];             /*force on right bead*/
    F_b[N+i+2]+=term_1*term_2*w_2[1];
    F_b[2*N+i+2]+=term_1*term_2*w_2[2];
    dot_prod=F_b[i+2]*u[i+2]+F_b[N+i+2]*u[N+i+2]+F_b[2*N+i+2]*u[2*N+i+2];

    E=k_B*(phi-pi)*(phi-pi)/2;
    j=fabs(i-b1+1);
    sum0+=E;                                    /*0-th moment*/
    sum1+=j*E;                                  /*1st moment*/
    sum2+=j*j*E;                                /*2nd moment*/
  }
  E_b[0]=sum0;
  E_b[1]=sum1;
  E_b[2]=sum2;
}

void LJ(void)
{
  int i,j,k,start,stop;
  double eps,sig,cutoff,dot_prod,E,prod;
  double dx,dy,dz,r_ij,term,force,sum0,sum1,sum2,ecut;

  eps=15.1/(k_0*r_0*r_0);                       /*L-J (nd)*/
  sig=9.35/r_0;                                 /*L-J (nd)*/
  cutoff=60/r_0;                                /*cutoff dist (nd)*/

  term=pow(sig/cutoff,6);
  ecut=4*eps*term*(term-1);
  sum0=sum1=sum2=0.0;

  for(i=0;i<N-1;i++)
  {
    if(i+10>=N)
      stop=N;
    else
      stop=i+10;

    for(j=i+3;j<stop;j++)
    {
      dx=r[i]-r[j];                             /*x-comp*/
      dy=r[N+i]-r[N+j];                         /*y-comp*/
      dz=r[2*N+i]-r[2*N+j];                     /*z-comp*/
      r_ij=sqrt(dx*dx+dy*dy+dz*dz);             /*edge-edge dist*/

      if(r_ij>cutoff)
        continue;                               /*skip if dist>60 A*/

      term=pow(sig/r_ij,6);
      force=24*eps*term*(2*term-1)/r_ij;        /*total force*/

      F_nb[i]+=force*dx/r_ij;                   /*x-comp for curr bead*/
```

```
        F_nb[N+i]+=force*dy/r_ij;                /*y-comp for curr bead*/
        F_nb[2*N+i]+=force*dz/r_ij;              /*z-comp for curr bead*/
        dot_prod=F_nb[i]*u[i]+F_nb[N+i]*u[N+i]+F_nb[2*N+i]*u[2*N+i];

        F_nb[j]-=force*dx/r_ij;                  /*x-comp for 2nd bead*/
        F_nb[N+j]-=force*dy/r_ij;                /*y-comp for 2nd bead*/
        F_nb[2*N+j]-=force*dz/r_ij;              /*z-comp for 2nd bead*/
        dot_prod=F_nb[j]*u[j]+F_nb[N+j]*u[N+j]+F_nb[2*N+j]*u[2*N+j];

        E=4*eps*term*(term-1)-ecut;
         k=fabs(i-b1+1);
        sum0+=E;                                 /*0-th moment*/
        sum1+=k*E;                               /*1st moment*/
        sum2+=k*k*E;                             /*2nd moment*/
      }
    }
    E_nb[0]=sum0;
    E_nb[1]=sum1;
    E_nb[2]=sum2;
    E_nb[0]=0;
    E_nb[1]=0;
    E_nb[2]=0;
}

void update(double *T,int flag)
{
    int i,j;
    double ratio,temp,gamma;
    struct { double x,y,z; } a;                  /*components of acceln*/

    for(i=0;i<N;i++)
    {
      ratio=mass[i]/m_sc;                        /*mass ratio*/
      gamma=0.0;                                 /*damping for interior*/
      if(i<Ng) gamma=d_max*(1-i/((float)(Ng-1)));  /*for 1st Ng beads*/
      if(i>=N-Ng) gamma=d_max*(i-N+Ng)/(float)(Ng-1);

      a.x=(F_s[i]+F_b[i]+F_nb[i]-gamma*u[i])/ratio;
      a.y=(F_s[N+i]+F_b[N+i]+F_nb[N+i]-gamma*u[N+i])/ratio;
      a.z=(F_s[2*N+i]+F_b[2*N+i]+F_nb[2*N+i]-gamma*u[2*N+i])/ratio;

      switch(flag)
      {
        case 0:                                  /*1st force calcn: r & u*/
          r[i]+=u[i]*dt+a.x*dt*dt/2;
          r[N+i]+=u[N+i]*dt+a.y*dt*dt/2;
          r[2*N+i]+=u[2*N+i]*dt+a.z*dt*dt/2;

          u[i]+=a.x*dt/2;
          u[N+i]+=a.y*dt/2;
          u[2*N+i]+=a.z*dt/2;

          *T=0.0;
          break;

        case 1:                                  /*2nd force calcn: u*/
```

```
        u[i]+=a.x*dt/2;
        u[N+i]+=a.y*dt/2;
        u[2*N+i]+=a.z*dt/2;

        j=fabs(i-b1+1);
        temp=u[i]*u[i]+u[N+i]*u[N+i]+u[2*N+i]*u[2*N+i];
        E_k[0]+=mass[i]*temp/m_sc;              /*0-th moment*/
        E_k[1]+=j*mass[i]*temp/m_sc;            /*1st moment*/
        E_k[2]+=j*j*mass[i]*temp/m_sc;          /*2nd moment*/
        uE[i]+=u[i]*mass[i]*temp/(2*m_sc);
        *T+=mass[i]*temp;
        break;
    }
  }
}

void prob(double *vals,int count)
{
  int i,j,bead=0;
  double sum,tau=10.0,k,Ea;

  for(i=Ng;i<N-Ng;i++)                          /*loop thru reactive interior*/
  {
    /*if(i!=2) rb[i-Ng]=0;*/
    j=rb[i-Ng]-1;                               /*array index of reac bead*/
    if(j<0) continue;                           /*skip reacted bead*/

    sum=mass[j]*(u[j]*u[j]+u[N+j]*u[N+j]+u[2*N+j]*u[2*N+j]);
    Ea=2*vals[0]*6.973e-21/u_sc/u_sc;           /*scaled activn energy*/
    k=1e13*exp(-Ea/sum);                        /*rate const*/
    if(k<small) k=small;                        /*avoid denom=0*/
    locator(k,i,&tau,&bead);                    /*rxn time,posn locator*/
  }
  acceptor(tau,bead,vals,count);                /*rxn acceptor*/
}

void locator(double k,int i,double *tau,int *bead)
{
  double zi,temp;

  zi=gsl_rng_uniform(rng);
  temp=log(1/zi)/(k*3*t_sc);                    /*rxn time @ reac bead 'i' (nd)*/

  if(temp<*tau)                                 /*find earliest rxn*/
  {
    *tau=temp;                                  /*store lower time value*/
    *bead=i;                                    /*possible rxn locn (array index)*/
  }
}

void acceptor(double tau,int bead,double *vals,int count)
{
  int i,n,index;
  double fac,alpha;

  index=bead-Ng;                                /*to match 'rb','n_r' indices*/
```

```
  if(tau<1.0)                                    /*rxn possible*/
  {
    locn=rb[index]-1;                            /*array index of reac bead*/

    for(i=1;i<=load-n_r[index];i++)              /*cycle thru unreac molecs*/
      if(i*tau<=1) n=i;                          /*# molecs reacted in dissipn time*/
      else break;                                /*skip invalid loop indices*/

    if(count==0) n=load;
    mass[locn]-=n*m_e;                           /*sub reac molec mass*/
    n_r[index]+=n;                               /*tot # molecs reacted*/

    if(n_r[index]==load)                         /*all molecs on bead reacted*/
    {
      t_rxn[index]=count+3*tau/dt;               /*time of rxn at full conversion*/
      /*printf("%i %.3f\n",rb[index],t_rxn[index]); /*print rxn locn,time*/
      rb[index]=0;                               /*locn no longer reactive*/
    }

    if(count==0) alpha=1.0;                      /*unconditionally cleave 1st bond*/
    else alpha=vals[1];

    calc_vel(alpha,n);                           /*chem energy->KE*/
  }
}

void calc_vel(double alpha,int n)
{
  int i,sign=1;
  double dir[3],u_sq,E,g_1,g_2,G,frxn;

  g_1=m_b/(n*m_e);
  g_2=load/n;
  G=g_1+g_2;
  u_sq=sqrt(u[locn]*u[locn]+u[N+locn]*u[N+locn]+u[2*N+locn]*u[2*N+locn]);
  dir[0]=u[locn]/u_sq; dir[1]=u[N+locn]/u_sq; dir[2]=u[2*N+locn]/u_sq;

  if(dir[2]<0)                                   /*bead moves downwards*/
  {
    dir[0]=-dir[0];                              /*x-direxnl cosine*/
    dir[1]=-dir[1];                              /*y-direxnl cosine*/
    dir[2]=-dir[2];                              /*z-direxnl cosine (always +ve)*/
  }

  for(i=0;i<3;i++)                               /*do for x,y,z dims*/
  {
    frxn=dir[i]*dir[i];                          /*energy fraction in dim i*/
    E=frxn*(2*alpha*delH/(m_e*u_sc*u_sc));       /*energy deposited in dim 'i'*/
    if(dir[i]>0) sign=-1;                        /*choose correct sign for soln*/
    u[i*N+locn]+=sign*sqrt(E/(G*(G-1)));         /*solve for bead vel*/
  }
}

void sub_vel_cm(void)
{
  int i;
```

```
  double sum1,sum2,sum3,sum4;

  sum1=sum2=sum3=sum4=0.0;
  for(i=0;i<N;i++)
  {
    sum1+=mass[i]*u[i];
    sum2+=mass[i]*u[N+i];
    sum3+=mass[i]*u[2*N+i];
    sum4+=mass[i];
  }
  u_cm[0]=sum1/sum4;
  u_cm[1]=sum2/sum4;
  u_cm[2]=sum3/sum4;

  for(i=0;i<N;i++)                              /*subtract CM vel*/
  {
    u[i]-=u_cm[0];
    u[N+i]-=u_cm[1];
    u[2*N+i]-=u_cm[2];
  }
}

void assign(int flag)
{
  int i,index,del_b;

  switch(flag)
  {
    case 0:                                     /*init masses for all beads*/
      mass[0]=m_b+load*m_e;
      for(i=1;i<N;i++)
        mass[i]=m_b+load*m_e;                   /*intrinsic mass + reac molec load (kg)*/
      break;

    case 1:                                     /*init rxn-related stuff*/
      for(i=Ng;i<N-Ng;i++)                      /*runs over only reactive interior*/
      {
        index=i-Ng;
        t_rxn[index]=0.0;                       /*rxn time storage array*/
        rb[index]=i+1;                          /*actual posns of reac beads*/
        n_r[index]=0;                           /*# of reacted species on bead*/
      }
      break;

    case 2:                                     /*init forces to zero*/
      for(i=0;i<N;i++)                          /*cycles thru beads*/
      {
        F_s[i]=F_s[N+i]=F_s[2*N+i]=0.0;
        F_b[i]=F_b[N+i]=F_b[2*N+i]=0.0;
        F_nb[i]=F_nb[N+i]=F_nb[2*N+i]=0.0;

        IS[i]=0.0;                              /*inner sum in heat current*/
        uE[i]=0.0;
      }

      E_k[0]=E_k[1]=E_k[2]=0.0;                 /*init 'E_k' to 0 for sums*/
```

```
      break;
   }
}

void calc_acf(int start,int count,FILE *f)
{
   int i;
   double acf,v_t;

   acf=0.0;
   for(i=0;i<N;i++)
   {
      if(count==start)                                   /*set origin vels*/
         v_0[i]=sqrt(u[i]*u[i]+u[N+i]*u[N+i]+u[2*N+i]*u[2*N+i]);

      v_t=sqrt(u[i]*u[i]+u[N+i]*u[N+i]+u[2*N+i]*u[2*N+i]);
      acf+=v_0[i]*v_t;
   }
   acf/=(double) N;                                      /*avg sum over all beads*/
   fprintf(f,"%e\n",acf);
}

void calc_modes(FILE *f)
{
   int k,i;
   double a,a_dot,omega,E;

   for(k=1;k<=N;k++)
   {
      a=0; a_dot=0;
      for(i=0;i<N;i++)
      {
         a+=r[i]*sin(i*k*pi/N);
         a_dot+=u[i]*sin(i*k*pi/N);
      }
      omega=2*sin(k*pi/(2*N));
      E=(a_dot*a_dot+a*a*omega*omega)/2;
      fprintf(f,"%f ",E);
   }
   fprintf(f,"\n");
}

void calc_dist(FILE *f)
{
   int i;
   double d,dx,dy,dz;

   for(i=0;i<N-1;i++)
   {
      dx=r[i]-r[i+1];
      dy=r[N+i]-r[N+i+1];
      dz=r[2*N+i]-r[2*N+i+1];
      d=sqrt(dx*dx+dy*dy+dz*dz);
      fprintf(f,"%f ",d);
   }
   fprintf(f,"\n");
```

```
}

void calc_image(FILE *f)
{
  int i;
  double E,k_B=1.38e-23;

  for(i=0;i<N;i++)
  {
    E=mass[i]*(u[i]*u[i]+u[N+i]*u[N+i]+u[2*N+i]*u[2*N+i])/m_sc;
    fprintf(f,"%e ",E*m_sc*u_sc*u_sc/(3*k_B));
  }
  fprintf(f,"\n");
}

void calc_wav_locn(FILE *f)
{
  double E0,E1,E2;

  E0=E_s[0]+E_b[0]+E_nb[0]+E_k[0];              /*sum 0-th moments*/
  E1=E_s[1]+E_b[1]+E_nb[1]+E_k[1];              /*sum 1st moments*/
  E2=E_s[2]+E_b[2]+E_nb[2]+E_k[2];              /*sum 2nd moments*/
  fprintf(f,"%f %f\n",E1/E0,E2/E0-E1*E1/E0/E0);
}

void calc_E(FILE *f)
{
  fprintf(f,"%e\n",E_k[0]);
}

void calc_curr(FILE *f)
{
  int i;
  double j,sum;

  sum=0.0;
  for(i=0;i<N;i++)
  {
    j=uE[i]+IS[i];                              /*local heat flux*/
    sum+=j;
    printf("%i %e %e\n",i,uE[i],IS[i]);
    /*fprintf(f,"%e ",j);*/
  }
  fprintf(f,"%f\n",sum);
}

void write_array(FILE *f)
{
  int i;

  for(i=0;i<nmol;i++)
    fprintf(f,"%f ",t_rxn[i]);
  fprintf(f,"\n");
}
```

Parameter header file:

```
#ifndef pars_h
#define pars_h

/*# of time steps,# of MC runs*/
int steps=1000,nruns=500;

/*spring const (kcal/mol/A^2),inter-bead dist (A),max damping,time step (nd)*/
double k_0=1000.0,r_0=10.0,d_max=0,dt=0.1;

/*1st reac bead,reac bead locn,# reac molec on bead*/
int b1,locn,load=3;
/*mass of bead (kg),mass of reac molec (kg)*/
double m_b=32.426e-25,m_e=3.5198e-25;

/*scaling vel (m/s),time (s),rxn enthalpy (J)*/
double u_sc,t_sc,delH=10.3e-19;

#endif
```

Function definition header file:

```
#ifndef funcns_h
#define funcns_h

void exec_loops(void);
void equilib(void);
void initialise(void);
void calc_force(double *,double *,FILE *);
void stretch(void);
void choose_par(int,double *,double,double *,double *,double *,double *);
void bond(void);
void LJ(void);
void update(double *,int);
void prob(double *,int);
void locator(double,int,double *,int *);
void acceptor(double,int,double *,int);
void calc_vel(double,int);
void sub_vel_cm(void);
void assign(int);
void calc_acf(int,int,FILE *);
void calc_modes(FILE *);
void calc_dist(FILE *);
void calc_image(FILE *);
void calc_wav_locn(FILE *);
void calc_E(FILE *);
void calc_curr(FILE *);
void write_array(FILE *);

#endif
```

MATLAB program to analyse results:

```
function data_plot
clc
mex covariance.c
warning off all

Ea=5:5:40; alpha=0:0.1:1;
n_Ea=length(Ea); n_alpha=length(alpha);
n_runs=500; n_beads=100;
res=funcn(n_Ea,n_alpha,n_runs,n_beads);          %analyse 'time' matrix
fig1(res,Ea,alpha);
clear all

function res=funcn(n_Ea,n_alpha,n_runs,n_beads)
global n_steps t_range del_t
n_rows=n_Ea*n_alpha*n_runs;
file=[pwd '/Results 2/res3.txt'];                %results file to be analysed
data=dlmread(file,' ');
data(:,end)=[];

stds=zeros(1,n_beads);                           %std devns of event times
res=zeros(n_Ea,n_alpha);                         %row=E_a,col=alpha
X=(0:n_beads-1)'*1e-9;                           %dimensional posns of beads (m)
del_t=75.373e-16;                                %time step (2 EM,s)
load=3; m=1+load*0.1085;
del_t=del_t*sqrt(m/1.217);

for i=1:n_Ea                                     %row index
  for j=1:n_alpha                                %col index
    start=(i-1)*n_alpha*n_runs+(j-1)*n_runs+1;
    stop=(i-1)*n_alpha*n_runs+j*n_runs;
    range=start:stop;                            %all runs for (E_a,alpha) pair
    time=data(range,:);                          %rxn times for beads in range
    avg_conv=length(find(time>0))/(n_runs*n_beads);
    res(i,j)=avg_conv;
    dt=sdev(time,n_beads);

    [T v_eps P]=covariance(time,t_range);        %calc mean rxn times,covariance matrix
    v_eps(1,:)=[]; v_eps(:,1)=[];                %del 1st bead
    row=min(find(diag(v_eps)>1e12))              %beads satisfying prob cutoff
    if isempty(row)
      row=size(v_eps,1)+1;
    else
      v_eps(row:end,:)=[];                       %delete low-prob rows
      v_eps(:,row:end)=[];                       %delete low-prob cols
    end
      X=X(1:row);                                %truncate chain length
    T=T(1:row)*del_t; v_eps=v_eps*del_t^2;       %dimensionalise (s,s^2)

    fig2(n_beads,v_eps);
    beta=fit_line(P,X,T,v_eps,n_runs);           %fit line thru N pts
  end
end

function beta=fit_line(P,X,T,v_eps,n_runs)
```

```
global data del_t
start=1; stop=13;                                 %pts to fit to line
var=diag(v_eps);
dT=1.96*sqrt(var/n_runs);                         %error bars for rxn times
data=[X(start:stop)*1e9 T(start:stop)/del_t];     %dataset to be fitted (non-dim)
v=data(end,1)/data(end,2);                        %initial guess for slope (vel)
t_0=T(1)/del_t;                                   %initial guess for 1st rxn time
beta=lsqnonlin(@solver,[v;t_0]);                  %solve for parameters
for i=1:length(dT)
  if dT(i)>T(i)
    dT(i)=T(i);
  end
end
dT=[0;dT];
beta(1)=beta(1)*1e-9/del_t;                       %dimensional vel (km/s)
beta(2)=beta(2)*del_t;                            %dimensional time (s)
confint(X(2:end),T(2:end),v_eps,beta);            %calc confidence intervals
fig3(P,X,T,dT,beta,stop);

function res=solver(beta)
global data
Y=data(:,1)/beta(1)-beta(2);                      %predixn
res=data(:,2)-Y;                                  %residuals

function confint(X,T,v_eps,beta)
v_inv=v_eps\eye(size(v_eps));
S=[-X/beta(1)^2 -ones(size(v_eps,1),1)];          %sensitivity vector
temp=v_inv*S;
temp=S'*temp;
v_beta=temp\eye(size(temp));
[m n]=size(S);
d=m-n;                                            %# degrees of freedom
delta=tinv(0.95,d)*sqrt(diag(v_beta));
b1=[beta-delta beta beta+delta]./repmat([1000;1],1,3);
fprintf(1,'%e %e %e\n',b1');

function dt=sdev(time,n_beads)
global n_steps t_range
for i=1:n_beads
  ind=find(time(:,i)>0);                          %rows with non-zero times
  if ~isempty(ind)
    stds(i)=std(time(ind,i));                     %std devns of non-zero times
  else
    stds(i)=0;                                    %no rxn event at all
  end
end
dt=floor(min(stds(stds>1e-4)));                   %time step = min std devn
t_min=min(min(time(time>0))); t_max=max(max(time));
t_range=t_min:dt:t_max+2*dt;                      %time range
n_steps=length(t_range);

function fig1(res,Ea,alpha)
contourf(alpha,Ea,res);
xlabel('Enthalpy Fraction Converted to KE','FontSize',22);
ylabel('Activation Energy (kcal/mol)','FontSize',22);
colorbar;
```

```
set(gca,'xtick',[0 0.2 0.4 0.6 0.8 1],'ytick',[10 15 20 25 30 35 40])
set(gca,'FontSize',22);

function fig2(n_beads,v_eps)
figure
imagesc(1:size(v_eps,1),1:size(v_eps,1),v_eps*1e25);
colorbar
xlabel('Index of Bead in Chain','FontSize',18);
ylabel('Index of Bead in Chain','FontSize',18);
axis tight
set(gca,'FontSize',18);
plot(1:size(v_eps),diag(v_eps),'m'),hold on
xlabel('Index of Bead in Chain','FontSize',18);
ylabel('Variance (s^2)','FontSize',18);
axis tight
set(gca,'xtick',[5 20 35 50 65 80 95])
set(gca,'FontSize',18);
legend('E_a=25 kcal/mol','E_a=30 kcal/mol','E_a=35 kcal/mol','E_a=40 kcal/mol'); legend('boxoff');

function fig3(P,X,T,dT,beta,stop)
figure
errorbar(X,T,dT,dT,'sb')
xlabel('Bead Location in Chain (m)','FontSize',18);
ylabel('Time of Reaction Occurrence (s)','FontSize',18);
hold on
Y=X(1:stop+30)/beta(1)-beta(2);
plot(X(1:stop+30),Y,'-r','LineWidth',3)
v=num2str(beta(1)/1000); v=v(1:4);
str=['v = ' v ' km/s'];
text(0,0,str,'FontSize',18,'FontWeight','bold','Color','r')
legend('Mean Reaction Time','Fit for Beads 1-3','Fit for Beads 4-13');
legend('boxoff');
axis tight
set(gca,'FontSize',18);

figure
plot(1:length(P),P,'b','LineWidth',6)
xlabel('Bead Index','FontSize',38);
ylabel('P(Reaction)','FontSize',38);
axis tight
set(gca,'xtick',[20 60 100])
set(gca,'FontSize',38);
```

<u>Calculating the covariance of reaction times</u>:

```
#include <math.h>
#include "mex.h"
#define cutoff 0.90
#define Inf 1e8

void calc_qty(double *,double *,double *,double *,double *);
void calc_len(double *,int,int *);
void calc_avg(double *,int,double *,double *,double *,double *);
void calc_cov(double *,int,int,double *,double,double *,double,double *,double *);

int nbeads,nruns,nsteps;

void mexFunction(int nlhs,mxArray *plhs[],int nrhs,const mxArray *prhs[])
{
  int n_b,n_r,n_s;
  double *time,*tr,*T,*v_eps,*P;

  n_r=mxGetM(prhs[0]);                          /*no: of runs (rows)*/
  n_b=mxGetN(prhs[0]);                          /*no: of beads (cols)*/
  time=mxGetPr(prhs[0]);                        /*rxn times*/
  n_s=mxGetN(prhs[1]);                          /*# of time steps*/
  tr=mxGetPr(prhs[1]);                          /*time range*/

  plhs[0]=mxCreateDoubleMatrix(n_b,1,mxREAL);
  T=mxGetPr(plhs[0]);                           /*mean rxn times*/
  plhs[1]=mxCreateDoubleMatrix(n_b,n_b,mxREAL);
  v_eps=mxGetPr(plhs[1]);                       /*covariance matrix*/
  plhs[2]=mxCreateDoubleMatrix(n_b,1,mxREAL);
  P=mxGetPr(plhs[2]);                           /*probability list*/

  nbeads=n_b;
  nruns=n_r;
  nsteps=n_s;
  calc_qty(time,tr,T,v_eps,P);
}

void calc_qty(double *time,double *tr,double *T,double *v_eps,double *P)
{
  int i,j,L=0;
  double ti_b,ti[500],tj_b,tj[500],cov,prob;

  for(i=0;i<nbeads;i++)                         /*col/bead index in 'time'*/
  {
    calc_len(time,i,&L);                        /*# of 0 rxn times*/
    if(L==nruns) continue;

    calc_avg(time,i,ti,&ti_b,&prob,tr);         /*mean rxn time & prob @ 'i'*/
    T[i]=ti_b;                                  /*dimensnl mean rxn time (s)*/
    P[i]=prob;                                  /*prob of rxn @ 'i'*/
    for(j=i;j<nbeads;j++)
    {
      calc_len(time,i,&L);                      /*# of 0 rxn times*/
      if(L==nruns) continue;
```

```
         calc_avg(time,j,tj,&tj_b,&prob,tr);          /*mean rxn time @ 'j'*/
         calc_cov(time,i,j,ti,ti_b,tj,tj_b,tr,&cov);  /*calc cov(i,j)*/
         v_eps[j*nbeads+i]=cov;
         v_eps[i*nbeads+j]=v_eps[j*nbeads+i];         /*symmetric matrix*/
      }
   }
}

void calc_len(double *time,int col,int *L)
{
   int i,start,stop,count=0;

   start=col*nruns;                                   /*1st elem in col*/
   stop=(col+1)*nruns-1;                              /*last elem in col*/
   for(i=start;i<=stop;i++)                           /*traverse col elems*/
      if(time[i]==0) count++;                         /*count # of zeros*/

   *L=count;                                          /*tot # of 0s in col*/
}

void calc_avg(double *time,int col,double *t,double *t_b,double *prob,double *tr)
{
   int i,j,start,stop,count;
   double sum_t,sum_p=0,sum_tp=0,p;

   start=col*nruns;                                   /*1st elem in col*/
   stop=(col+1)*nruns-1;                              /*last elem in col*/
   for(i=0;i<nsteps-1;i++)                            /*'tr' index*/
   {
      sum_t=0.0; count=0;
      for(j=start;j<=stop;j++)                        /*traverse col elems*/
         if(time[j]>=tr[i] && time[j]<tr[i+1])
         {
            sum_t+=time[j];                           /*for mean calculation*/
            count++;                                  /*# of times in interval*/
         }
      p=(double)count/(double)nruns;                  /*P(rxn) in [i i+1)*/
      sum_p+=p;                                       /*tot prob of rxn @ 'i'*/
      if(count>0)
         t[i]=sum_t/(double)count;                    /*mean of runs*/
      else t[i]=(tr[i]+tr[i+1])*0.5;                  /*centre of interval*/
      sum_tp+=t[i]*p;
   }
   *prob=sum_p;                                       /*prob of rxn @ 'i'*/
   if(sum_p>=cutoff) *t_b=sum_tp/sum_p;               /*mean rxn time @ 'j'*/
   else *t_b=Inf;                                     /*no rxn possible*/
}

void calc_cov(double *time,int i,int j,double *ti,double ti_b,double *tj,double tj_b,double *tr,double *cov)
{
   int m,n,p,q,i_0,j_0,i_N,j_N,count;
   double tm1,tm2,tn1,tn2,p_ij,sum_tp=0,sum_p=0;

   i_0=i*nruns;                                       /*1st elem in col 'i'*/
   i_N=(i+1)*nruns-1;                                 /*last elem in col 'i'*/
```

```
  j_0=j*nruns;                                /*1st elem in col 'j'*/
  j_N=(j+1)*nruns-1;                          /*last elem in col 'j'*/

  for(m=0;m<nsteps;m++)
  {
    tm1=tr[m]; tm2=tr[m+1];                   /*interval limits for 'i'*/
    for(n=0;n<nsteps;n++)
    {
      tn1=tr[n]; tn2=tr[n+1];                 /*interval limits for 'j'*/
      count=0;
      for(p=i_0,q=j_0;p<=i_N,q<=j_N;p++,q++)  /*scan cols 'i' & 'j'*/
        if((time[p]>=tm1 && time[p]<tm2) && (time[q]>=tn1 && time[q]<tn2))
          count++;

      p_ij=(double)count/(double)nruns;       /*P(rxn) @ (i,j)*/
      sum_tp+=(ti[m]-ti_b)*p_ij*(tj[n]-tj_b); /*prod of vecs & matrix*/
      sum_p+=p_ij;                            /*total probability*/
    }
  }
  *cov=sum_tp/sum_p;                          /*covariance*/
}
```