

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.

THE KWIK
ALGORITHM FOR
COULOMB
INTERACTIONS AND
ITS APPLICATIONS

by

Jeremy P. Dombroski
B.Sc.(Hons)

A thesis presented in partial fulfilment
of the requirements for the degree
of Doctor of Philosophy

Department of Chemistry
Massey University
1997

ABSTRACT

The KWIK algorithm is introduced, generalised and applied to the problem of determining the Coulomb energy of N localised charge distributions. Coulomb interactions are typical of N -body problems which require the exhaustive pairing of all distributions, which leads to prohibitive computational cost scaling characteristics for large N .

The KWIK algorithm for Coulomb interactions begins by optimally separating the Coulomb operator into rapidly decaying real- and Fourier- space partitions yielding a hybrid technique not dissimilar in concept to other approximations methods. KWIK's superiority lies in that its efficiency increases with distribution size, so that large distributions become computationally advantageous for increasing accuracy.

Model calculations on a distribution consisting of one million particles using KWIK afforded energies, to high accuracy, within minutes compared with days for quadratic methods. The extension of such a feat to even larger distributions is now limited by machine hardware configurations.

Particular emphasis is placed on the application of the algorithm to Molecular Quantum Mechanics where it is illustrated that the algorithm may be applied to linearise single-point self consistent field calculations. In particular, KWIK can be used to form the Exchange matrix in linear computational cost. This has previously only been achieved by crude approximation techniques and cannot be achieved using Coulomb multipole based methods.

ACKNOWLEDGMENTS

There are many people to whom I owe a great deal of thanks.

Firstly, I offer my sincere thanks to my supervisor Dr Peter M. W. Gill. Peter is a scholar and a gentleman, an outstanding scientist and a friend. He has guided me well throughout the last three years, maintaining an open door (or email) offering clear and well thought out assistance for all problems, and a clever knack for simplifying the complex. His enthusiasm for science, particularly quantum chemistry, has been a great inspiration. Thanks also to the second in command, Dr Ken Jolley, who took over the reins after Peter departed for Cambridge.

There have been others who have aided in increasing my understanding of research, quantum chemistry, mathematics and computing which has had a direct benefit on this thesis. Dr Benny Johnson of Q-Chem Inc., Dr Steve Taylor of the Mathematics Department at the University of Auckland and Chris White, presently at University of California at Berkeley, have been particularly helpful in this regard.

I thank my family for their continued encouragement and support which tends to be required more through difficult times, rather than during the more productive and successful ones. I imagine this thesis will end up on the coffee table as an answer to the question "... and what *exactly* is Jeremy doing at Massey?" continually posed by extended family and visitors.

To the members of Peter's research group who have had to put up with me hacking my way through system administration and bringing workstations to a standstill with my large scale calculations, I thank you for your patience. Of the group members, Dr Terry Adams is one who has had the largest positive effect on broadening my knowledge and improving my outlook. Thanks Terry for your friendship. I would also like to thank Mrs 'Bot' Gill for many fine lunches and offer my apologies to Malcolm and Stephanie who weren't allowed seconds at dinner time because the remainder was set aside to become my lunch the next day. Thanks also goes to Gwen Adams for providing me with a far superior poster than I could have possibly even attempted to construct for my trip to Paris and Cambridge.

There are also many other friends who have helped maintain a level of sanity (and often reality) outside of Massey whom I have chosen not to name for fear forgetting anyone, or placing them in the incorrect order. I must however, thank Gaile Peddie for putting up with endless sleepless nights, grumpy moods, hogging the phone-line with the modem and general ungentlemanly behaviour, especially during the write-up phase. Past flatmates have also had to suffer through similar agony.

I have been most fortunate in my bids for funding grants and scholarships which allowed me to attend four conferences, three of these international. I offer my humble thanks to the New Zealand Vice Chancellors Committee for a William Georgetti Scholarship and to Massey University for a Doctoral Scholarship. For assistance towards travel to conferences and for computer software, I wish to thank: - Q-Chem Inc., The New Zealand Institute of Chemistry, DFT-95 Conference organisers, Massey University Research Fund, Department of Chemistry, Massey University Graduate Research Fund and the Science Faculty Dean's Fund for generous contributions.

TABLE OF CONTENTS

ABSTRACT	ii
ACKNOWLEDGMENTS	iii
TABLE OF CONTENTS	v
LIST OF TABLES, SCHEMES, FIGURES AND THEOREMS	ix
LIST OF ABBREVIATIONS	xiv
LIST OF PUBLICATIONS	xvii
NOTE ON CALCULATIONS	xvii
CHAPTER SUMMARIES	xviii
<i>Chapter One</i>	xviii
<i>Chapter Two</i>	xviii
<i>Chapter Three</i>	xviii
<i>Chapter Four</i>	xviii
<i>Chapter Five</i>	xix
<i>Chapter Six</i>	xix
<i>Chapter Seven</i>	xix
<i>Chapter Eight</i>	xix
<i>Chapter Nine</i>	xix
INTRODUCTION	1
CHAPTER ONE	3
THEORETICAL MOTIVATION	3
1.0. <i>Introduction</i>	3
1.1. <i>Classical Beginnings</i>	3
1.2. <i>Schrödinger's Equation</i>	4
1.3. <i>Solving Schrödinger's Equation</i>	5
1.4. <i>The Variation Principle and Variation Method</i>	7
1.5. <i>The Hartree-Fock Approximation</i>	8
1.5.1. <i>Restricted Closed-Shell Hartree-Fock (RHF)</i>	8
1.5.2. <i>Unrestricted Open-Shell Hartree-Fock (UHF)</i>	10
1.6. <i>Matrix Elements</i>	10
1.6.1. <i>Unrestricted Hartree-Fock Matrix Elements</i>	13
1.7. <i>The Correlation Problem</i>	14
1.8. <i>Conventional Methods</i>	15
1.8.1. <i>Configuration Interaction</i>	16
1.8.2. <i>Quadratic Configuration Interaction</i>	16
1.8.3. <i>Coupled Cluster</i>	17
1.8.4. <i>Perturbation Theory</i>	18
1.8.5. <i>Other approaches</i>	20
1.8.6. <i>Additional Comments</i>	20
1.9. <i>Density Functional Theory</i>	22
1.10. <i>Commentary</i>	25
CHAPTER TWO	27
TECHNOLOGICAL MOTIVATION	27
2.0. <i>Introduction</i>	27
2.1. <i>High-Performance Computing</i>	27
2.2. <i>Historical Prospective</i>	28
2.2.1. <i>Computing Facility</i>	28
2.2.2. <i>Programming Languages</i>	28
2.3. <i>Algorithm Development</i>	30
2.4. <i>Modern-day Computer Architectures</i>	31
2.4.1. <i>Vector Machines</i>	32
2.4.2. <i>CISC and RISC</i>	33
2.4.3. <i>Parallel Computers</i>	34
2.5. <i>Summary</i>	35
CHAPTER THREE	36

HISTORY OF THE QUANTUM CHEMICAL COULOMB PROBLEM	36
3.0. <i>Introduction</i>	36
3.1. <i>The Quantum Chemical Coulomb Problem</i>	37
3.1.0. <i>Introduction</i>	37
3.1.1. <i>Gaussian Basis Functions</i>	37
3.1.2. <i>Electron Repulsion Integrals Over GTO's</i>	39
3.1.3. <i>Linear Methods</i>	40
3.2. <i>The Coulomb Problem</i>	43
3.2.1. <i>The N Body Problem</i>	43
3.2.2. <i>The Ewald Summation Technique</i>	44
3.2.3. <i>Particle-Particle-Particle-Mesh</i>	45
3.2.4. <i>The Fast Multipole Method</i>	46
3.2.5. <i>The Fast Wavelet Transform</i>	47
3.3. <i>Summary</i>	47
CHAPTER FOUR	49
PARTITIONING THE COULOMB OPERATOR	49
4.0. <i>Introduction</i>	49
4.1. <i>The KWIK Algorithm</i>	50
4.2. <i>Adapting KWIK for Coulomb Interactions</i>	51
4.3. <i>Separating the operator</i>	52
4.4. <i>The Ultimate Separator</i>	55
4.5. <i>The Error Function Separator</i>	59
4.6. <i>Summary</i>	61
CHAPTER FIVE	63
SHORT-RANGE INTERACTIONS	63
5.0. <i>Introduction</i>	63
5.1. <i>Computational Methods</i>	64
5.1.0. <i>Introduction</i>	64
5.1.1. <i>Computing the Interaction</i>	65
5.1.2. <i>Verlet Neighbour Lists</i>	65
5.1.3. <i>Linked-Cell Method</i>	66
5.2. <i>Improvements to the Linked-Cell Method</i>	68
5.2.0. <i>Introduction</i>	68
5.2.1. <i>Ordering an Ordinate</i>	69
5.2.2. <i>Changing the Cell Geometry</i>	73
5.3. <i>Approximating Spheres</i>	73
5.3.0. <i>Introduction</i>	73
5.3.1. <i>New Basic Cell Structure</i>	73
5.3.2. <i>Cell Size Reduction</i>	75
5.4. <i>Maintaining Vectorisability</i>	81
5.5. <i>Conclusions</i>	84
CHAPTER SIX	85
LONG RANGE KWIK	85
6.0. <i>Introduction</i>	85
6.1. <i>Fourier Transform of the Long-Range Partition</i>	85
6.2. <i>Numerical Quadrature</i>	87
6.2.0. <i>Introduction</i>	87
6.2.1. <i>Simple Quadrature</i>	88
6.2.2. <i>Gauss Type Quadrature Rules</i>	90
6.2.3. <i>Basis Spanning</i>	91
6.3. <i>Integration of FT Long-Range KWIK</i>	91
6.3.0. <i>Introduction</i>	91
6.3.1. <i>The Separator Transform</i>	91
6.3.2. <i>Simple Quadrature</i>	92
6.3.3. <i>Gauss Type Rules</i>	94
6.3.4. <i>COP's and ROP's</i>	97
6.3.5. <i>The Fourier Intensity I(k) and the Random Walk Advantage</i>	98
6.3.6. <i>Discussion</i>	100
6.4. <i>Fourier Series</i>	101

6.4.0. Introduction	101
6.4.1. Fourier Coefficients	102
6.4.2. Recasting KWIK using a Fourier Series	104
6.4.3. Evaluating Fourier Coefficients in KWIK	105
6.4.3.1. Fourier Coefficients in One Dimension	105
6.4.3.2. Fourier Coefficients in Three Dimensions	109
6.4.4. Gibbs Phenomenon	111
6.5. Summary	113
CHAPTER SEVEN	115
APPLICATION OF KWIK TO DISTRIBUTIONS OF PARTICLES	115
7.0. Introduction	115
7.1. Particles in One Dimension	115
7.1.1. Derivation	116
7.1.2. Short-Range	117
7.1.2.0. Introduction	117
7.1.2.1. Interpolation	118
7.1.2.2. Cut-offs	120
7.1.2.3. Memory Requirements	121
7.1.2.4. Distribution Dependence	122
7.1.3. Long-Range	122
7.1.3.0. Introduction	122
7.1.3.1. MOP's vs FLOP's	123
7.1.3.2. Memory Requirements	123
7.1.3.3. Cut-offs	124
7.1.4. Charged Systems	125
7.1.4.0. Introduction	125
7.1.4.1. Illustrative Timings	125
7.1.4.2. The Random Walk and $I(k)$	127
7.1.5. Neutral Systems	134
7.1.5.0. Introduction	134
7.1.5.1. A Random Walk Advantage?	135
7.1.5.2. Illustrative Timings	137
7.2. Particles in Three dimensions	141
7.2.0. Introduction	141
7.2.1. Derivation	141
7.2.2. Short-Range	142
7.2.2.1. Interpolation	142
7.2.2.2. Cut-offs	142
7.2.2.3. Memory Storage	143
7.2.3. Long-Range	143
7.2.3.0. Introduction	143
7.2.3.1. Memory Requirements	143
7.2.3.2. 2-D Fourier Coefficients	144
7.2.4. Charged Systems	146
7.2.4.0. Introduction	146
7.2.4.1. Illustrative Timings	146
7.3. Summary and Discussion	148
CHAPTER EIGHT	150
KWIK APPLIED TO QUANTUM CHEMISTRY	150
8.0. Introduction	150
8.1. KWIK for $O(N)$ Fock Matrix Formation	150
8.1.0. Introduction	150
8.1.1. KWIK in Dirac's Notation	151
8.1.2. Core Hamiltonian Matrix	152
8.1.3. Coulomb and Exchange Matrices	153
8.2. Evaluating Short-Range KWIK	155
8.2.0. Introduction	155
8.2.1. Integrals Over Gaussian S Type Functions	155
8.2.2. Screening Insignificant Integrals in $O(N)$ Work	156
8.3. Evaluating Long-Range KWIK	157
8.3.1. CASE and CAP's	157
8.3.2. Approximating the Coulomb Operator	159

8.4. <i>Self Consistent Field Calculations</i>	160
8.5. <i>Quantum Chemistry Using Slater Functions</i>	160
8.5.1. STO's.....	160
8.5.2. Slater Function Density.....	162
8.5.3. Stewart-Slater Atoms	163
8.6. <i>KWIK in Correlated methods</i>	164
CHAPTER NINE	166
CRITICAL ANALYSIS	166
9.0. <i>Introduction</i>	166
9.1. <i>Scaling</i>	166
9.2. <i>Significant Findings</i>	168
9.3. <i>KWIK vs The Rest</i>	169
9.3.1. KWIK vs Ewald	169
9.3.2. KWIK vs FMM.....	170
9.3.3. KWIK vs CFMM	171
9.4. <i>Further Work Directions</i>	172
9.5. <i>Summary</i>	175
APPENDIX A	177
DATA FOR THE SHORT-RANGE INVESTIGATIONS.....	177
APPENDIX B	179
DERIVATION OF NUMERICAL APPROXIMATION FOR DETERMINING THREE DIMENSIONAL FOURIER COEFFICIENTS.....	179
APPENDIX C	186
QUADRATURE RULES AND POLYNOMIALS FOR THE EXPONENTIAL INTEGRAL WEIGHT FUNCTION	186
APPENDIX D	190
DETAILS OF KWIK TIMINGS PRESENTED IN CHAPTER SEVEN.....	190
REFERENCES	194

LIST OF TABLES, SCHEMES, FIGURES AND THEOREMS

Table 1.1.	Cost scaling behaviour of Conventional Correlated methods with the size (N) of the molecular system.	22
Table 4.1.	Value of the functional $J[g(\mathbf{r})]$ (4.21.3) for various separators with their respective optimal decay parameters, ω . The difference in the value of the functional between that of the optimal separator and error function is minimal.	59
Table 5.1.	Efficiencies of the linked cell method in all three dimensions. Note in 2-D and 3-D the majority of the computational effort is put into calculating insignificant interactions.	69
Table 5.2.	Efficiencies of the linked cell method with x -direction ordinate ordering for all three dimensions. Note in 3-D the majority of the computational effort is still put into calculating insignificant interactions and that 2-D efficiency is still unacceptable.	70
Table 5.3.	Efficiencies of alternative cell geometries in two dimensions.	75
Table 5.4.	The efficiency of cell size reduction. * denotes the total number of cells required relative to the $k=1$ case.	76
Table 5.5.	Effectiveness of ordinate ordering is reduced on increased dimension and cell size reduction. * ordinate ordering has been applied, ‡ some outlying cells become completely insignificant and are neglected as the simulation cell becomes more representative of a sphere/circle.	76
Table 5.6.	Theoretical efficiencies and numbers of significant near neighbour cells for subdivided cubes.	80
Table 7.1.	CPU times (seconds) for evaluating 400,000 interactions in the interval (0,4) involving the erfc function. Key: Function - denotes evaluating the erfc function itself; 1-D - the one-dimensional interaction potential; 3-D - the three-dimensional interaction including a square root; Standard - is the time for evaluating the same potential using the standard mathematical library.	120
Table 7.2.	CPU times (seconds) for a short-range KWIK calculation ($\omega=0.008$) on distributions of 100,000 unit point charges. Ordered implies a binary search has been used and division indicates the level of cell reduction.	122
Table 7.3.	CPU times (seconds) for determining the Coulomb energy using KWIK for collinear, uniformly distributed, positive unit point charges for various predetermined accuracies.	126
Table 7.4.	CPU times (seconds) for determining the Coulomb energy using KWIK for collinear, uniformly distributed, positive unit point charges for various predetermined accuracies. (As reported in reference [148]).	127
Table 7.5.	CPU times (seconds) for determining the Coulomb energy for various system sizes (N) using the classical double sum quadratic method.	128
Table 7.6.	CPU times (seconds) for determining the Coulomb energy using KWIK for collinear, uniformly distributed, net-neutral systems of unit point charges for various predetermined accuracies. Decay parameter and degree of Fourier expansion determined as for charged systems.	138
Table 7.7.	Total KWIK timings for various predetermined accuracies for net-neutral, uniformly distributed particles using a fixed decay parameter of $\omega = 0.01$	139
Table 7.8.	Total KWIK timings for various predetermined accuracies for net-neutral, uniformly distributed particles using a fixed decay parameter of $\omega = 0.005$	139
Table 7.9.	Short-range KWIK energies for net-neutral and charged uniformly distributed unit point charges. A zero decay parameter corresponds to the total coulomb energy. The decay parameter of $\omega = 0.01$ corresponds to a real space cut-off of ≈ 302 for $\epsilon = 10^{-6}$	140

Table 7.10.	Short-range KWIK energies for net-neutral, uniformly distributed unit point charges. The decay parameters of $\omega = 0.01$ and 0.005 have real space cut-offs of ≈ 302 and 603 respectively for $\epsilon = 10^{-6}$	140
Table 7.11.	Two dimensional KWIK Fourier coefficients for $\omega = 1$. k_x and k_y correspond to the x and y components of the k-space sum.....	144
Table 7.12.	Two dimensional KWIK Fourier coefficients for $\omega = 0.1$. k_x and k_y correspond to the x and y components of the k-space sum.....	145
Table 7.13.	Two dimensional KWIK Fourier coefficients for $\omega = 0$. k_x and k_y correspond to the x and y components of the k-space sum.....	145
Table 7.14.	KWIK timings for charged systems of three dimensional, uniformly distributed unit point charges as previously reported [148].....	146
Table 7.15.	KWIK timings for charged systems of three dimensional, uniformly distributed unit point charges with incorporation of interpolation and a crude implementation of long-range cut-offs.....	147
Table A.1.	Theoretical efficiencies and number of near neighbour cells for the use of Hexagons as the parent cell in 2-D short-range calculations.....	177
Table A.2.	Theoretical efficiencies and number of near neighbour cells for the use of Squares as the parent cell in 2-D short-range calculations.....	178
Table A.3.	Theoretical efficiencies and number of near neighbour cells for the use of Triangles as the parent cell in 2-D short-range calculations.....	178
Table B.1.	Table of approximation coefficients for equation (B.22) as obtained using <i>Mathematica</i> [168].....	185
Table C.1.	Roots and weights for the Gauss type quadrature scheme over the weight function $E_1(k^2/4)$	188
Table D.1.	Details for KWIK calculations conducted on collinear charged uniformly distributed systems as presented in Table 7.3. Key: NBOX - the number of cells in the short-range calculation; M - the degree of the Fourier expansion; ω - the decay parameter; R_c - the real space cut-off radius; RLONG - the relative contribution of the long range energy to the total energy; E - the total Coulomb energy; N - the size (and range) of the distribution.....	190
Table D.2.	Details for KWIK calculations conducted on collinear, net-neutral uniformly distributed systems as presented in Table 7.6. Key: NBOX - the number of cells in the short-range calculation; M - the degree of the Fourier expansion; ω - the decay parameter; R_c - the real space cut-off radius; RLONG - the relative contribution of the long range energy to the total energy; E - the total Coulomb energy; N - the size (and range) of the distribution.....	191
Table D.3.	Details for KWIK calculations conducted on collinear, net-neutral uniformly distributed systems for fixed decay parameter $\omega = 0.01$, as presented in Table 7.7. Key: R_c - the real space cut-off radius; M - the degree of the Fourier expansion; RTIME - the relative contribution of the long-range time to the total time; RLONG - the relative contribution of the long range energy to the total energy; N - the size (and range) of the distribution.....	192
Table D.4.	Details for KWIK calculations conducted on collinear, net-neutral uniformly distributed systems for fixed decay parameter $\omega = 0.005$, as presented in Table 7.7. Key: R_c - the real space cut-off radius; M - the degree of the Fourier expansion; RTIME - the relative contribution of the long-range time to the total time; RLONG - the relative contribution of the long range energy to the total energy; N - the size (and range) of the distribution.....	193
Scheme 5.1.	Sorting particles into Verlet neighbour lists requires work that scales with the square of the number of particles which can be seen by the nested loop over particles above. Note also that the memory requirements for storing lists of neighbours for each particle is significant.....	66

Scheme 5.2.	Sorting particles/molecules into cells in linear work using integers. ICOUNT(I) indicates the number of particles in cell I, and IPOINT(I) points to the first particle in cell I in a cell-ordered coordinate array XORD. The first three elements of X are particle coordinates, and the final entry is the charge.	67
Scheme 5.3.	The sorting of an ordinate to improve efficiency is achieved similarly to that without the sort and simply requires an extra loop over all Cells and a call to the sorting routine XSORT. This routine sorts the x ordinate into increasing value maintaining the integrity of the other coordinates and associated charge.....	71
Scheme 5.4.	Algorithm structure for short-range interactions with an ordered ordinate.....	72
Scheme 7.1.	Partial FORTRAN code illustrating the use of interpolating the erfc function in a 3-D simulation. The array T contains the interpolating constants f_k which are a function of the polynomial coefficients.	120
Scheme 7.2.	Illustration of memory requirements for the one dimensional sorting process.	121
Scheme 7.3.	Illustrating the use of recursion to save FLOP's and unrolling the k loop to save on MOP's. Unrolling beyond 5 gave little improvement in speed-up on the IBM.	124
Scheme 7.4.	Illustration of memory requirements for the three dimensional sorting process.	143
Scheme 7.5.	Array storage for long-range KWIK.....	144
Figure 1.1.	Molecular energy level diagram for H ₂	6
Figure 1.2.	Pople diagram illustrating the conventional approach to improved quality of calculation. The quality of a calculation can be improved either by an increase in the size of the basis set or by considering a greater number of configurations.....	21
Figure 3.1.	Radial dependence of a Gaussian vs Slater basis function. Note that the Slater has a cusp at the origin, whereas the Gaussian is flat and that the long-range behaviour of a Gaussian is flatter than a Slater.....	38
Figure 4.1.	Graphical representation of the short and long range characteristics of the Coulomb, short- and long- range operators.	53
Figure 5.1.	The particle denoted P need only interact with particles within the cut-off radius R _C . Using the linked-cell method, particle P interacts with all the particles within the region depicted by the bold square, which is significantly more than necessary.	68
Figure 5.2.	As for Figure 5.1, the particle denoted P need only interact with particles within the cut-off radius R _C . Using the ordinate ordered linked-cell method, particle P interacts only with those particles within the unshaded region. This is somewhat less than the previous requirement.....	72
Figure 5.3.	Dimensions of the squares, triangles and hexagons used in our investigations of alternative two dimensional cell geometries. One arrow length is equal to the cut-off distance R _C	74
Figure 5.4.	Efficiency vs relative total number of cells contrasting the use of squares, triangles and hexagons as the basic geometric structure with which to divide a 2-D simulation system.	78
Figure 5.5.	A closer view of efficiency vs relative total number of cells contrasting the use of squares, triangles and hexagons over a practical range of relative total number of cells.	79
Figure 5.6.	Efficiency vs relative total number of cells for the case of using cubes and the division of cells method in 3-D.	81
Figure 5.7.	Illustration of the organisation of particles into boxes in the mega-array. The square represents the simulation cell divided into squares. Particles are organised into the mega-array (the tall column) by row with each square representing a list of particles.	82
Figure 5.8.	Scheme illustrating the importance of ordering the particles into their respective boxes.	83

Figure 6.1.	Fourier transform of the long-range operator obtained from using the error function separator. The logarithmic singularity and rapid long-range decay typify the behaviour of all the trialed separator functions.....	92
Figure 6.2.	Plots of the 'even' series representation of the erfc long-range operator. The true operator is the smooth line, whereas the series approximations are the oscillating lines. Note the changing frequency and amplitude of the 'tail' with increasing M	96
Figure 6.3.	Plots of the 'odd' series representation of the erfc long-range operator. The true operator is the smooth line, whereas the series approximations are the oscillating lines. Note the changing amplitude and frequency of the 'tail' with increasing M	97
Figure 6.4.	Log of the absolute value of the 1-D Fourier coefficients vs k for $\omega=10$. Note that $A_x(k,10)$ decreases by approximately 5 orders of magnitude between $k=0$ and $k=60$. It is not until $k=10,000$ that a second such reduction is achieved.	108
Figure 6.5.	Log of the absolute value of the 1-D Fourier coefficients vs k for $\omega=100$	109
Figure 6.6.	Illustrating the Gibbs phenomenon for the unit step function. Note that near the discontinuities the Fourier series approximation overshoots the true function significantly more than in the centre of the steps.	112
Figure 6.7.	1-D Fourier series representation of the erfc long-range operator superimposed upon a plot of the true function.....	113
Figure 7.1.	CPU time vs N (the number of particles) for determining the Coulomb energy using the KWIK algorithm for increasing accuracy. Note the circled data point indicating a higher efficiency than expected obtained due to statistical fluctuations. Actual timings are tabulated in Table 7.3.	126
Figure 7.2.	$I(k)$ vs k for discrete values of k for 1,000,000 collinear <u>uniformly</u> distributed, positive, unit point charges.	128
Figure 7.3.	$I(k)$ vs k for discrete values of k for 100,000 collinear <u>uniformly</u> distributed, positive, unit point charges.	129
Figure 7.4.	Histogram of the uniformly distributed particles used in Figure 7.3.	130
Figure 7.5.	$I(k)$ vs k for discrete values of k for 100,000 collinear <u>exponentially</u> distributed, positive, unit point charges.	131
Figure 7.6.	Histogram of the exponentially distributed particles as used for Figure 7.5.....	132
Figure 7.7.	$I(k)$ vs k for discrete values of k for 100,000 collinear <u>normally</u> distributed, positive, unit point charges.	133
Figure 7.8.	Histogram of the normally distributed particles used in Figure 7.7.	134
Figure 7.9.	$I(k)$ vs k for a <u>uniform</u> , net neutral distribution of 1,000,000 unit point charges.	135
Figure 7.10.	$I(k)$ vs k for a <u>uniform</u> , net-neutral distribution of 100,000 unit point charges.....	136
Figure 7.11.	$I(k)$ vs k for 100,000 <u>exponentially</u> distributed, net-neutral system of unit point charges.	137
Figure 7.12.	CPU time (seconds) vs N (number of particles) for determining the total Coulomb energy using KWIK for a net-neutral, uniformly distributed system of unit point charges. The circled data point indicates a higher accuracy than expected was determined due to statistical fluctuations. Times are tabulated in Table 7.6.....	138
Figure 7.13.	CPU Time (seconds) vs N (the number of particles). The filled symbols correspond to the computationally cheaper 1-D systems, while the open symbols the slightly more expensive 3-D systems. Like symbols represent similar accuracies.	148
Figure 9.1.	A plot of linear and $x\log(x)$ functions to illustrate that the scaling behaviour of the two are remarkably similar	167

Theorem 6.1. If a periodic function $f(x)$ with period $2l$ is piece-wise continuous in the interval $-l \leq x \leq l$ and has a left-hand derivative and right-hand derivative at each point of that interval, then the Fourier series representation of $f(x)$ is convergent. Its sum is $f(x)$, except at a point x_0 at which $f(x)$ is discontinuous and the sum of the series is the average of the left- and right-hand limits of $f(x)$ at x_0 102

LIST OF ABBREVIATIONS

1-D	One Dimensional Space
2-D	Two Dimensional Space
3-D	Three Dimensional Space
AO	Atomic Orbital
BLAS	Basis Linear Algebra Subroutines
C++	(object oriented programming language)
CAP	Coulomb Attenuated Potential
CASE	Coulomb Attenuated Schrödinger Equation
CC	Coupled Cluster
CCD	Coupled Cluster with Doubles excitations
CCSD(T)	Coupled Cluster with Singles and Doubles excitations, incorporating Triples perturbatively
CCSDT	Coupled Cluster with Singles, Doubles and Triples excitations
CFMM	Continuous Fast Multipole Method
CI	Configuration Interaction
CISC	Complex Instruction Set Computer
CISD	Configuration Interaction with Singles and Doubles excitations
COP	Coulomb Orthonormal Polynomial
CNDO	Complete Neglect of Differential Overlap
CPU	Central Processing Unit
DE	Differential Equation
DFT	Density Functional Theory
DIIS	Direct Inversion in the Iterative Subspace
ERI	Electron (two) Repulsion Integral
FCI	Full Configuration Interaction
FFT	Fast Fourier Transform
FLOP	Floating Point Operation

FMA	Floating-point Multiply-Add
FMM	Fast Multipole Method
FORTRAN	Formula Translation (programming language)
FT	Fourier Transform
G2	Gaussian-2 theory
GTO	Gaussian type Orbital
GvFMM	Gaussian very Fast Multipole Method
HF	Hartree-Fock
HOMO	Highest Occupied Molecular Orbital
INDO	Intermediate Neglect of Differential Overlap
I/O	Input/Output
KWIK	Name given to the algorithm originally devised by Gill for combinatorial problems which are highly dependent. See Chapter 6 equation (6.9).
LAPACK	Linear Algebra Package
LCAO	Linear Combination of Atomic Orbitals
LHS	Left Hand Side
LUMO	Lowest Unoccupied Molecular Orbital
MA	Multiply-Add
MASPAR	Massively Parallel
MINDO	Modified Intermediate Neglect of Differential Overlap
MO	Molecular Orbital
MOP	Memory Operation
MP	Møller-Plesset
MP _n	n th -order Møller-Plesset
MQM	Molecular Quantum Mechanics
NDDO	Neglect of Diatomic Differential Overlap
PC	Personal Computer
PNDO	Partial Neglect of Differential Overlap
PPPM	Particle-Particle-Particle-Mesh

QCI	Quadratic Configuration Interaction
QCID	Quadratic Configuration Interaction with Doubles excitations
QCISD	Quadratic Configuration Interaction with Singles and Doubles excitations
QCISD(T)	Quadratic Configuration Interaction with Singles and Doubles excitations, incorporating Triples perturbatively
RHF	Restricted Hartree-Fock
RHS	Right Hand Side
RISC	Reduced Instruction Set Computer
ROP	Repulsive Orthonormal Polynomial
SCF	Self-Consistent Field
STO	Slater Type Orbital
UHF	Unrestricted Hartree-Fock
WS	Well-Separated

LIST OF PUBLICATIONS

The following papers have been published as a direct consequence of the work undertaken for this thesis:

1. “KWIK: Coulomb Energies in $O(N)$ Work.” Jeremy P. Dombroski, Stephen W. Taylor and Peter M. W. Gill. *Journal of Physical Chemistry*, (1996) **100**, 6272.
2. “Chemistry without Coulomb Tails.” Ross D. Adamson, Jeremy P. Dombroski and Peter M. W. Gill. *Chemical Physics Letters*, (1996) **254**, 329.
3. “The Optimal Partition of the Coulomb Operator.” Aaron M. Lee, Stephen W. Taylor, Jeremy P. Dombroski and Peter M. W. Gill. *Physical Review A*, (1997), in press.

NOTE ON CALCULATIONS

All calculations and efficiency developments reported herein, were obtained on an IBM RS/6000 model 355 with 64 MB RAM. All programming was written in FORTRAN 77 and compiled with the IBM XL FORTRAN for AIX compiler, versions 3.2.5 and 4.1.0 with O2 optimisation. Code was linked against the BLAS, LAPACK and Q-CHEM libraries when required.

CHAPTER SUMMARIES

Chapter One

Considers the theoretical basis of modern quantum chemistry indicating many of the basic quantities required for computation. Illustrates that the two electron integrals are central to all basic approaches.

Chapter Two

Outlines some of the important considerations required in algorithm development, especially that the mathematics does not dictate the algorithm and that there are many different ways of arriving at the same answer. Includes a considerable amount of the author's personal opinion regarding choices of programming language and software development platforms. Very important chapter in the context of the work undertaken for this thesis.

Chapter Three

Literature review of the recent development of fast methods for determining self-consistent field energies, electron densities and wave functions including historic perspectives. Illustrates that the Coulomb problem is central to the inhibition of the widespread application of quantum chemistry, and that this problem is of a type universally encountered in many other disciplines. This is followed by a summary of the N -body problem and a number of generalised methods that have been developed to overcome it.

Chapter Four

Outlines the origins of the KWIK algorithm and then discusses the ideas for generalising the algorithm so that it may be applied to determining Coulomb energies which will be developed in this thesis. This is followed by investigations and discussions into determining how best to split the Coulomb operator.

Chapter Five

A chapter dedicated to determining the best methodology for computing short-range interactions. Highlights for the first time extreme inefficiencies with currently employed methods, summarises the investigations undertaken to overcome the problem and concisely concludes how the inefficiency problem is best tackled in the context of high performance computing.

Chapter Six

Highlights investigations into determining how one may afford the KWIK long-range energy partition. Indicates how the algorithm becomes linear and explains the success of earlier described statistics algorithm.

Chapter Seven

KWIK is applied to point charges in one and three dimensions to illustrate the theoretical developments and how the algorithm was developed for generalisation to Quantum Chemistry. In one dimension, KWIK is applied to charged and net neutral systems and the effect of polarised non-uniform distributions are discussed. Three dimensional timings are reported to illustrate that this extension is trivial, requires no major changes in the algorithm, and scales in a similar fashion to one dimensional KWIK.

Chapter Eight

Discusses how KWIK can be used in quantum chemistry for linearising the computational cost of self-consistent field calculations. Brief discussions are presented on applications to correlated methods and the use of KWIK with Slater-type functions.

Chapter Nine

A critical analysis is given of the method, its current implementation, its strengths and weaknesses, and where further work should be directed in terms of the general Coulomb problem, and in particular, for application of KWIK to Quantum Chemical

problems. KWIK is compared and contrasted with other methods, in particular the Fast Multipole Method.

INTRODUCTION

Seventy years ago, quantum theory reached a climax when Schrödinger postulated the equations of quantum chemistry and solved them for the archetypal hydrogen atom. Directly proceeding from Schrödinger's postulates, a theoretical foundation was laid for approximating the equations of quantum chemistry which are too difficult to solve for other than the most trivial case of hydrogenic atoms. The theoretical foundation laid all those years ago remains the cornerstone from which all quantitative modern quantum chemistry is based. (Modern quantum chemistry contrasts with classical quantum mechanics which is purely descriptive and qualitative.)

The beginning of modern quantum mechanics is marked in a paper by Boys [1], the first paper containing chemistry obtained from approximating solutions to Schrödinger's equation *via* a computer. In 1956, programs were written on ticker-tape and had to be recalculated such was the unreliability of the 'automatic machines'. Since this beginning, (thankfully) much has changed.

Firstly, there have been major technological advancements in computer hardware and secondly, major developments of faster and smarter algorithms for approximating solutions to Schrödinger's equation to obtain chemistry. The latter have been expressed in an increasing number of commercial and non-commercial 'black-box' software packages such as Q-CHEM, DISCO, TURBOMOL, GAUSSIAN, POLYATOM, HONDO, SPARTAN, MOLPRO, CADPAC *etc.* The most important of these was GAUSSIAN-70 [2] (released in 1970) because it is considered to be the first of the non-expert programs.

The efficient development of non-expert programs, as well as the aforementioned technological advances, has seen an upsurge in both their use and, the acceptance of molecular quantum mechanics (MQM) by experimental chemical researchers, particularly in the last two decades. In fact, MQM has become so popular that in 1996, computational chemists were second only to weather forecasters in total use of central processing unit (CPU) time worldwide.

With the development of the computer and black-box programs has been a split in those chemists whose research interests lie in MQM. The first division is that of the computational chemist who is interested in exploiting the technology to solve real chemical problems. The second, is theoretical chemists, who (confusingly) could also be referred to as computational scientists. A theoretician is interested in developing algorithms to better approximate Schrödinger's equation so that chemistry can be obtained faster and more accurately. The problem for all MQM chemists is the time requirement for the calculations.

Time or, as it is often termed, computational cost, is mainly measured in relative terms by the nature of a methods cost scaling with molecule size. If, in the limit of large molecules, the cost of a method doubles on first considering a monomer, then its dimer, it is a linear or $O(N)$ method. If the computational cost quadruples from monomer to dimer calculation, it is quadratic or $O(N^2)$. If the cost sextuples it is $O(N^3)$, *etc.* Obviously one must consider absolute costs when making comparisons between methods for small to medium sized systems and also when contrasting methods with the same cost scaling characteristics.

Cost scaling behaviour has been at the fore of quantum chemical research over the last decade. The single biggest hindrance of the application of MQM to very large molecules, perhaps even those of biological interest, is the horrific cost scaling behaviour of many of the more accurate MQM methods. Even if current computing speed was to increase an order of magnitude overnight, the increase in the size of molecule that could be subject to accurate calculation, (that is, one yielding an accurate approximation to Schrödinger's equation) would be small.

Of all the methods, the self-consistent field (SCF) approaches have received the most prominent attention, where the computational bottle-neck is in the evaluation of electron repulsion integrals, scaling at a formal cost of $O(N^4)$. This thesis is concerned with reducing the computational bottle-neck in SCF calculations by careful consideration of the underlying cause of the problem, and developing an alternative algorithm with a view to maximise its impact by taking into account current trends in high performance computing during its development.

CHAPTER ONE

THEORETICAL MOTIVATION

1.0. Introduction

Many a molecular quantum mechanics research seminar, thesis and review has begun by quoting a remark made by Dirac - "The underlying physical laws necessary for the mathematical theory of a large part of physics and the whole of chemistry are thus completely known, and the difficulty is only that the exact application of these laws leads to equations much too complicated to be soluble" [3].

While this remark is as true today as it was then, that is, we are still unable to afford exact solutions to the equations, the development of the digital computer has enabled the solutions to be approximated to high accuracy. Presented in this Chapter are, the equations of molecular quantum mechanics which explain the majority of chemistry referred to by Dirac, the basis from which we approximate their solution, and the ways in which we look to improve such approximations. In the context of this thesis, emphasis is placed on the self-consistent field methods, and post-SCF treatments are introduced for completeness and later discussion.

1.1. Classical Beginnings

In 1911, Rutherford [4] proposed that an atom must contain a small central region called the nucleus. He believed the nucleus contained all the positive charge of the atom and was surrounded by a cloud of electrons with equal and opposite charge. Later, Bohr [5] suggested that electrons moved in orbits around the nucleus and could not spiral inwards out of these orbits emitting continuous radiation as they were only allowed to emit quanta. This Rutherford-Bohr model mixes classical physics ideas of electrons as particles in orbits with concepts of energy quanta from quantum theory, and while this model successfully explains the emission spectrum of the hydrogen atom, it is incorrect. This, however, was the introduction of quantum theory to chemistry.

In 1925 a Frenchman, Louis de Broglie [6] used Einstein's equations for light [7] and applied them to electrons, proposing that electrons behaved as waves. Almost simultaneously, but independently, two American physicists, Davisson and Kunsman had been studying the peculiar behaviour of electrons scattering from crystals. This aided in establishing the existence of the effect [8] predicted by de Broglie.

1.2. Schrödinger's Equation

In 1926 Schrödinger [9] combined the wave nature of the electron with the statistical knowledge of the electron position *viz.* Heisenberg's Uncertainty Principle [10] to formulate the time independent, non-relativistic Schrödinger equation usually written in terms of the Hamiltonian wave equation

$$H(\mathbf{R}, \mathbf{r})\Psi(\mathbf{R}, \mathbf{r}) = E(\mathbf{R})\Psi(\mathbf{R}, \mathbf{r}) \quad (1.1)$$

where the coordinates \mathbf{R} and \mathbf{r} refer to nuclei and electron position vectors respectively and H is the Hamiltonian operator (in atomic units)

$$H = -\frac{1}{2} \sum_{i=1}^N \nabla_i^2 - \frac{1}{2} \sum_{A=1}^M \frac{1}{M_A} \nabla_A^2 - \sum_{i=1}^N \sum_{A=1}^M \frac{Z_A}{r_{iA}} + \sum_{i=1}^N \sum_{j>i}^N \frac{1}{r_{ij}} + \sum_{A=1}^M \sum_{B>A}^M \frac{Z_A Z_B}{R_{AB}} \quad (1.2)$$

∇^2 is the Laplacian operator

$$\nabla^2 \equiv \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \quad (1.3)$$

Z is the nuclear charge, $1/M_A$ is the ratio of the mass of nucleus A to the mass of an electron, $R_{AB} = |\mathbf{R}_A - \mathbf{R}_B|$ is the distance between the A^{th} and B^{th} nucleus, $r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$ is the distance between the i^{th} and j^{th} electrons, $r_{iA} = |\mathbf{r}_i - \mathbf{R}_A|$ is the distance between the i^{th} electron and A^{th} nucleus, M is the number of nuclei and N is the number of electrons. E is an eigenvalue of H , equal to the total energy and the wave function Ψ , is an eigenfunction of H .

Since the rest mass of an electron is very much less than that of nuclei, the nuclei move much more slowly than the electrons and we can separate the rotational and

vibrational motions from electronic motions. This separation was originally due to Born and Oppenheimer [11] and yields the electronic Hamiltonian operator

$$H_{elec} = -\frac{1}{2} \sum_{i=1}^N \nabla_i^2 - \sum_{i=1}^N \sum_{A=1}^M \frac{Z_A}{r_{iA}} + \sum_{i=1}^N \sum_{j>i}^N \frac{1}{r_{ij}} \quad (1.4)$$

The solution of the corresponding electronic Schrödinger equation

$$H_{elec} \Psi_{elec} = E_{elec} \Psi_{elec} \quad (1.5)$$

gives the electronic wave function Ψ_{elec} , which describes the motion of the electrons and E_{elec} , the total electronic energy. The total energy is afforded by simply adding the nuclear-nuclear repulsion energy (fifth term of (1.2)) to the total electronic energy

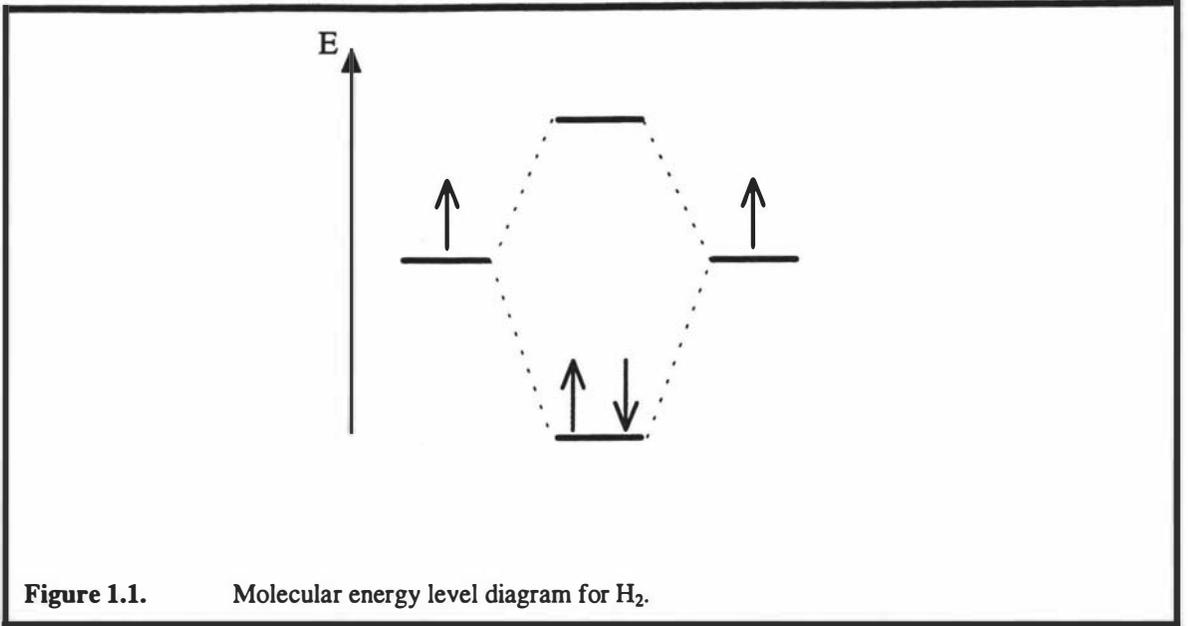
$$E_{Tot} = E_{elec} + E_{nuc} \quad (1.6)$$

For the sake of brevity, the ‘Hamiltonian’ and ‘energy’ referred to in the remainder of this thesis, implies electronic, as in (1.5).

1.3. Solving Schrödinger’s Equation

Partial differential equations in $3N$ unknowns, such as (1.5), are totally intractable to solve exactly for anything other than the case where $N = 1$. Therefore, Schrödinger’s equation is never solved, but rather approximated. Typically, two important approximations are made.

The first approximation is that electrons move independently within molecular orbitals (MO’s) whereby each molecular orbital describes the probability distribution of a single electron. Each MO is determined by considering the electron as moving within an average field of all the other electrons, hence, the term independent particle approximation. Such an approximation is perhaps the chemist’s view of electrons in orbitals (Figure 1.1).



The wave function for non-interacting particles is simply given by the product of individual wave functions

$$\Psi = \psi_1(1)\psi_2(2)\dots\psi_{n-1}(n-1)\psi_n(n) \quad (1.7)$$

where Ψ is the total wave function and ψ_i is wave function of the i^{th} particle. Wave functions such as (1.7) are termed Hartree [12-14] products. A suitable wave function for electrons based on the Hartree product is

$$\Psi = \chi_1(1)\chi_2(2)\dots\chi_{n-1}(n-1)\chi_n(n) \quad (1.8)$$

where each χ_i is termed a spin orbital and is a product of a spatial orbital ψ_i and one of two orthonormal spin functions, $\alpha(\omega)$ or $\beta(\omega)$, *i.e.* spin up (\uparrow) or spin down (\downarrow). Fock [15] pointed out that since electrons are fermions, Hartree product wave functions cannot be used, because the wave function must be anti-symmetric on electron interchange [10]. A wave function based on the Hartree product model, but which conveys the fermion property is a Slater [16,17] determinant

$$\Psi = \frac{1}{\sqrt{n!}} \begin{vmatrix} \chi_1(1) & \chi_2(1) & \cdots & \chi_n(1) \\ \chi_1(2) & \chi_2(2) & \cdots & \chi_n(2) \\ \vdots & \vdots & & \vdots \\ \chi_1(n) & \chi_2(n) & \cdots & \chi_n(n) \end{vmatrix} \quad (1.9)$$

1.4. The Variation Principle and Variation Method

When a system is in a state $\hat{\Psi}$, which may or may not satisfy (1.5), we can write (using the concise Dirac notation [18]) the average of many measurements of the energy as

$$E[\hat{\Psi}] = \langle \hat{\Psi} | H | \hat{\Psi} \rangle \quad (1.10)$$

where the $\hat{\Psi}$ is considered to be normalised. The energy E , is said to be a functional of $\hat{\Psi}$. The approximate wave function $\hat{\Psi}$ can be written as an expansion in terms of the exact eigenstates of H

$$\hat{\Psi} = \sum_j c_j \Psi_j \quad (1.11)$$

therefore, (1.10) can be written as

$$E = \sum_j |c_j|^2 E_j \quad (1.12)$$

which implies that energy obtained from our approximate wave function $\hat{\Psi}$ is always

$$E \geq E_0 \quad (1.13)$$

Full minimisation of the functional $E[\hat{\Psi}]$ (1.10) with respect to all allowed N -electron wave functions will give the true ground state wave function and energy E_0 . In this sense, one could replace Schrödinger's equation (1.5) with the variation principle

$$\delta E[\hat{\Psi}] = 0 \quad (1.14)$$

In practice one chooses a trial wave function with variable parameters and varies these so that (1.14) is true, *i.e.* the energy is minimised. The resulting energy satisfies (1.13) and can be improved, along with the approximate wave function, by adding more arbitrary parameters into the trial wave function.

It is both time-consuming and difficult to minimise the energy with respect to parameters which occur non-linearly in the molecular orbitals, hence it usual to expand the molecular orbitals as a linear combination of basis functions

$$\psi_i = \sum_{\mu} c_{\mu i} \phi_{\mu} \quad (1.15)$$

which is usually referred to as the Linear Combination of Atomic Orbitals (LCAO) approximation. If the set of functions $\{\phi_{\mu}\}$ were complete, (1.15) would be an exact expansion. Unfortunately, for computational reasons, one is restricted to finite and incomplete sets (basis sets). The introduction of a basis set is the second important approximation made to aid the approximate solution of (1.5). The nature of the functions which make up basis sets [19] is an important consideration for computation and much effort has been put into developing, and consideration given when choosing them.

1.5. The Hartree-Fock Approximation

In short, the Hartree-Fock (HF) approximation [18] is the method whereby spatial molecular orbitals, typically of the form (1.15) are found that satisfy (1.14) with a wave function which is of the determinantal form (1.9). That is, the HF approximation is the method whereby electrons move independently within molecular orbitals. Typically these MO's are linear combinations of atomic orbitals (basis functions), which is an additional approximation.

The rigours of the mathematics in the derivation of the Hartree-Fock equations and the elimination of spin have been omitted in the above summary for conciseness, and can be found in suitable texts [18,20,21]. While computation necessitates the elimination of the spin functions from the Hartree-Fock equations we still need to consider the two basic types of spin orbitals used - restricted and unrestricted.

1.5.1. Restricted Closed-Shell Hartree-Fock (RHF)

Restricted spin orbitals are constrained to have the same spatial function for α and β spin functions

$$\left. \begin{matrix} \chi_i(\mathbf{r}) \\ \chi_j(\mathbf{r}) \end{matrix} \right\} = \begin{cases} \psi_i(\mathbf{r})\alpha(\omega) \\ \psi_i(\mathbf{r})\beta(\omega) \end{cases} \quad (1.16)$$

Such wave functions are the chemist's view of electronic structure where electrons tend to associate in pairs of opposite spin (Figure 1.1). The Hartree-Fock wave function for the closed-shell restricted ground state becomes

$$\Psi = \frac{1}{\sqrt{n!}} \begin{vmatrix} \psi_1(1) & \bar{\psi}_1(1) & \psi_2(1) & \bar{\psi}_2(1) & \cdots & \psi_{n/2}(1) & \bar{\psi}_{n/2}(1) \\ \psi_1(2) & \bar{\psi}_1(2) & \psi_2(2) & \bar{\psi}_2(2) & \cdots & \psi_{n/2}(2) & \bar{\psi}_{n/2}(2) \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ \psi_1(n) & \bar{\psi}_1(n) & \psi_2(n) & \bar{\psi}_2(n) & \cdots & \psi_{n/2}(n) & \bar{\psi}_{n/2}(n) \end{vmatrix} \quad (1.17)$$

where the bar denotes a beta spin orbital. Elimination of spin leads to the calculation of molecular orbitals becoming equivalent to the problem of solving the spatial integro-differential equation

$$f(\mathbf{r})\psi_i(\mathbf{r}) = \varepsilon_i\psi_i(\mathbf{r}) \quad (1.18)$$

where f is the Fock operator and ε_i are MO energies. Introducing a basis of the type (1.15), substituting into (1.18), multiplying by ϕ_μ and integrating, results in a matrix equation

$$\sum_{\nu} C_{\nu i} \int \phi_{\mu}(1) f(1) \phi_{\nu}(1) d\mathbf{r}_1 = \varepsilon_i \sum_{\nu} C_{\nu i} \int \phi_{\mu}(1) \phi_{\nu}(1) d\mathbf{r}_1 \quad (1.19)$$

which can be rewritten as

$$\sum_{\nu} F_{\mu\nu} C_{\nu i} = \varepsilon_i \sum_{\nu} S_{\mu\nu} C_{\nu i} \quad (1.20)$$

or

$$\mathbf{FC} = \varepsilon\mathbf{SC} \quad (1.21)$$

(The latter commonly referred to as the Roothaan-Hall equations).

1.5.2. Unrestricted Open-Shell Hartree-Fock (UHF)

The unrestricted Hartree-Fock approach, developed by Pople and Nesbet [22], introduces separate spatial orbitals for α and β spin:

$$\left. \begin{array}{l} \chi_i(\mathbf{r}) \\ \chi_j(\mathbf{r}) \end{array} \right\} = \left\{ \begin{array}{l} \psi_i^\alpha(\mathbf{r})\alpha(\omega) \\ \psi_i^\beta(\mathbf{r})\beta(\omega) \end{array} \right. \quad (1.22)$$

which sometimes leads to lower energy wave functions for many molecular systems, which, by the variational principle, implies we have obtained a better wave function. Adopting such wave functions leads to the generalised coupled matrix equations (Pople-Nesbet equations)

$$\begin{aligned} \mathbf{F}^\alpha \mathbf{C}^\alpha &= \boldsymbol{\varepsilon}^\alpha \mathbf{S} \mathbf{C}^\alpha \\ \mathbf{F}^\beta \mathbf{C}^\beta &= \boldsymbol{\varepsilon}^\beta \mathbf{S} \mathbf{C}^\beta \end{aligned} \quad (1.23)$$

It is important to note that the unrestricted matrix equations above are coupled and must be solved simultaneously.

In general, solutions to the restricted case are also solutions to the unrestricted case when the number of alpha electrons is equal to the number of beta electrons, but there are many exceptions [18] whereby a second unrestricted solutions exists.

1.6. Matrix Elements

\mathbf{C} (1.21) is a square matrix of molecular orbital coefficients where $C_{\mu i}$ corresponds to μ^{th} coefficient of the i^{th} molecular orbital, $\boldsymbol{\varepsilon}$ is a diagonal matrix of the orbital energies ε_i , \mathbf{S} is the overlap matrix with elements

$$S_{\mu\nu} = \int \phi_\mu(\mathbf{r})\phi_\nu(\mathbf{r}) d\mathbf{r} \quad (1.24)$$

and \mathbf{F} is the Fock matrix. Before considering the important components that make up the Fock matrix, consider an electron described by the spatial wave function $\psi_a(\mathbf{r})$. The probability distribution or the charge density for the electron is given by

$$\rho_a(\mathbf{r}) = |\psi_a(\mathbf{r})|^2 \quad (1.25)$$

The total charge density of a molecule using a closed-shell wave function is the summation of individual electron charge densities

$$\rho(\mathbf{r}) = 2 \sum_{a=1}^{N/2} |\psi_a(\mathbf{r})|^2 \quad (1.26)$$

with the multiplication by two reinforcing the restricted form of the wave function having two electrons in each occupied MO. By substituting the MO expansion (1.15) into (1.26) we obtain

$$\rho(\mathbf{r}) = \sum_{\mu\nu} P_{\mu\nu} \phi_\mu(\mathbf{r}) \phi_\nu(\mathbf{r}) \quad (1.27)$$

where we have defined a density matrix

$$P_{\mu\nu} = 2 \sum_{a=1}^{N/2} C_{\mu a} C_{\nu a} \quad (1.28)$$

It is often useful to characterise the charge density by the density matrix.

The Fock matrix \mathbf{F} , is the matrix representation of the Fock operator

$$f(1) = h(1) + \sum_a^{N/2} 2J_a(1) - K_a(1) \quad (1.29)$$

in a LCAO MO (linear combination of atomic orbitals - molecular orbital) basis. The Fock operator, and subsequently, the Fock matrix elements

$$F_{\mu\nu} = \int \phi_\mu(\mathbf{r}_1) f(1) \phi_\nu(\mathbf{r}_1) d\mathbf{r}_1 \quad (1.30)$$

can be separated into several important pieces. The one electron operator

$$h(1) = -\frac{1}{2} \nabla_1^2 - \sum_{A=1}^M \frac{Z_A}{r_{1A}} \quad (1.31)$$

which affords the core-Hamiltonian matrix consisting of kinetic energy integrals

$$T_{\mu\nu} = \int \phi_\mu(\mathbf{r}) \left[-\frac{1}{2} \nabla^2 \right] \phi_\nu(\mathbf{r}) d\mathbf{r} \quad (1.32)$$

and nuclear attraction integrals,

$$V_{\mu\nu} = \int \phi_{\mu}(\mathbf{r}) \left[-\sum_A \frac{Z_A}{|\mathbf{R}_A - \mathbf{r}|} \right] \phi_{\nu}(\mathbf{r}) d\mathbf{r} \quad (1.33)$$

yielding

$$H_{\mu\nu}^{core} = T_{\mu\nu} + V_{\mu\nu} \quad (1.34)$$

or

$$\mathbf{H} = \mathbf{T} + \mathbf{V} \quad (1.35)$$

The Coulomb

$$J_a(1) = \sum_{\lambda\sigma} C_{\lambda a} C_{\sigma a} \int \left[\frac{1}{r_{12}} \right] \phi_{\lambda}(\mathbf{r}_2) \phi_{\sigma}(\mathbf{r}_2) d\mathbf{r}_2 \quad (1.36)$$

and exchange operators

$$K_a(1) = \sum_{\lambda\sigma} C_{\lambda a} C_{\sigma a} \int \phi_{\sigma}(\mathbf{r}_2) \left[\frac{1}{r_{12}} \right] \phi_{\lambda}(\mathbf{r}_2) P_{12} d\mathbf{r}_2 \quad (1.37)$$

(where P_{12} is an operator that interchanges the coordinates of electron one and two) yield the Coulomb and exchange matrix elements

$$J_{\mu\nu} = \sum_{\lambda\sigma} P_{\lambda\sigma}(\mu\nu|\lambda\sigma) \quad (1.38)$$

$$K_{\mu\nu} = \frac{1}{2} \sum_{\lambda\sigma} P_{\lambda\sigma}(\mu\lambda|\nu\sigma) \quad (1.39)$$

respectively. The two electron integrals are simply

$$(\mu\nu|\lambda\sigma) = \int \phi_{\mu}(\mathbf{r}_1) \phi_{\nu}(\mathbf{r}_1) \left[\frac{1}{r_{12}} \right] \phi_{\lambda}(\mathbf{r}_2) \phi_{\sigma}(\mathbf{r}_2) d\mathbf{r}_1 d\mathbf{r}_2 \quad (1.40)$$

Therefore, the Fock matrix elements can be written as

$$F_{\mu\nu} = H_{\mu\nu}^{core} + J_{\mu\nu} - K_{\mu\nu} \quad (1.41)$$

Thus, we may conveniently (by adopting the compact Dirac notation) write the total restricted Hartree-Fock ground state electronic energy as

$$E_0 = \sum_{\mu}^N \sum_{\nu}^N P_{\mu\nu} \langle \mu|h|\nu \rangle + \sum_{\mu}^N \sum_{\nu}^N \sum_{\lambda}^N \sum_{\sigma}^N P_{\mu\nu} P_{\lambda\sigma} \left[(\mu\nu|\lambda\sigma) - \frac{1}{2}(\mu\lambda|\nu\sigma) \right] \quad (1.42)$$

Note that the sum of individual orbital energies is not equal to the total energy. Individual orbital energies are the sums of the kinetic and nuclei attraction energies of an electron in its spin orbital plus the Coulomb and exchange interactions between all other electrons. Summing these energies doubles the Coulomb and exchange contribution, as each orbital adds in interactions with all other orbitals.

1.6.1. Unrestricted Hartree-Fock Matrix Elements

In the unrestricted case we firstly need to define both an alpha and beta density matrix

$$\begin{aligned} P_{\mu\nu}^{\alpha} &= \sum_{a=1}^{n_{\alpha}} C_{\mu a}^{\alpha} C_{\nu a}^{\alpha} \\ P_{\mu\nu}^{\beta} &= \sum_{a=1}^{n_{\beta}} C_{\mu a}^{\beta} C_{\nu a}^{\beta} \end{aligned} \quad (1.43)$$

where $C_{\mu a}^{\alpha}$ is the μ^{th} coefficient of the a^{th} α molecular orbital. Thus, the total electron density is

$$\mathbf{P}^T = \mathbf{P}^{\alpha} + \mathbf{P}^{\beta} \quad (1.44)$$

The Fock matrix for unrestricted calculations varies only in the exchange piece

$$\begin{aligned} F_{\mu\nu}^{\alpha} &= H_{\mu\nu}^{core} + J_{\mu\nu} - K_{\mu\nu}^{\alpha} \\ &= H_{\mu\nu}^{core} + \sum_{\lambda}^N \sum_{\sigma}^N \left[P_{\mu\nu}^T (\mu\nu|\lambda\sigma) - P_{\lambda\sigma}^{\alpha} (\mu\lambda|\nu\sigma) \right] \end{aligned} \quad (1.45)$$

so that the total ground state energy can now be written as (again adopting the compact Dirac notation)

$$E_0 = \sum_{\mu}^N \sum_{\nu}^N P_{\mu\nu} \langle \mu | h | \nu \rangle + \sum_{\mu}^N \sum_{\nu}^N \sum_{\lambda}^N \sum_{\sigma}^N \left\{ P_{\mu\nu}^T P_{\lambda\sigma}^T (\mu\nu | \lambda\sigma) - \frac{1}{2} [P_{\mu\nu}^{\alpha} P_{\lambda\sigma}^{\alpha} + P_{\mu\nu}^{\beta} P_{\lambda\sigma}^{\beta}] (\mu\lambda | \nu\sigma) \right\} \quad (1.46)$$

In practice, to obtain a HF wave function (restricted or unrestricted) one uses a guess for the initial density matrix and applies an iterative process until self consistency is achieved. In such, the Hartree-Fock approximation is also a self-consistent field method.

1.7. The Correlation Problem

While the HF approximation is surprisingly successful in its ability to approximate some real world chemistry, HF calculations tend to give qualitative, rather than quantitative chemistry and it is considered that in order to yield more quantitatively satisfying results it is usual to incorporate correlation energy [23-25].

The correlation energy, is by definition, the difference between the exact HF energy (that obtained when using the exact HF molecular orbitals, or in the limit, commonly referred to as the Hartree-Fock-limit) and the corresponding eigenvalue of the Electronic Schrödinger Equation (1.5)

$$E_{corr} = E_0 - E_{HF} \quad (1.47)$$

This definition, by the nature of (1.5), excludes the zero-point vibrational energy, Born-Oppenheimer and relativistic corrections. Note that the independent particle approximation allows electrons to interact more closely than they would wish, which leads to a negative value for E_{corr} .

It should also be noted that while the HF approximation is described as the independent particle approximation, (which would imply that the probability of finding electron j at a point in space and that of finding electron k at another point is independent) confusingly the Hartree-Fock approach incorporates what is known as exchange correlation. Exchange correlation between electrons of parallel spin means that the probability of finding two electrons of the same spin at the same point in space,

is zero. If the electrons of parallel spin were not correlated, the probability of finding two at the same point in space would be the same as for opposite spin electrons, which is, the probability of finding electron j multiplied by the probability of finding electron k at that point. This technicality can be overlooked for purposes of clarity, and it is re-emphasised that the correlation energy is defined by equation (1.47).

There are two general approaches for incorporating correlation into a MQM calculation. Conventional methods, which base themselves on the so-called single-configuration HF theory (multi-reference configurations have been excluded in this discussion), and those which use the Density Functional Theory (DFT).

1.8. Conventional Methods

Conventional methods for determining the electronic correlation energy use the general approach, where the correlation effects are introduced by choosing the wave function to be a linear combination of many electron configurations. Configurations are generated by replacing the occupied orbitals in the Hartree-Fock reference wave function by virtual orbitals to yield single, double, triple ... (S,D,T ...) excitations. This general approach can be subsequently split into several classes which are outlined briefly in the following subsections.

Pople [26] has outlined a number of desirable features for evaluating a correlated technique, each of which will be addressed for all methods considered below. The first desirable is that the method must be well-defined, which simply means that the technique is suitably unambiguous to ensure that the same energy will be obtained for identical nuclear configurations. We will only consider well-defined methods here. Secondly, size consistency must prevail, that is, the energy of two non-interacting systems must be equal to the sum of the individual energies. It is also desirable for a correlated method to be exact for two-electron systems and accurate *c.f.* Full Configuration Interaction (FCI) (see later), for systems with more than two electrons. In addition, Pople suggested that it would be desirable if the method was also variational and computationally inexpensive.

1.8.1. Configuration Interaction

Configuration interaction (CI) [18] is perhaps the most simple and general of the conventional techniques. CI uses a linear combination of configurations where the HF wave function is mixed with single, double, triple ... “excited” configurations

$$\Psi = c_0 \Psi_0 + \sum_{ra} c_a^r \Psi_a^r + \sum_{\substack{a<b \\ r<s}} c_{ab}^{rs} \Psi_{ab}^{rs} + \sum_{\substack{a<b<c \\ r<s<t}} c_{abc}^{rst} \Psi_{abc}^{rst} + \dots \quad (1.48)$$

where the symbol $\Psi_{ab\dots}^{rs\dots}$ represents the configuration where the electrons in occupied orbitals a, b, ... are placed into virtual orbitals r, s, ..., and the summations are over all distinct determinants. The coefficients are obtained in a variational approach to afford a better many-electron wave function.

Full configuration interaction refers to when the expansion (1.48) is extended to include all possible determinants. Such calculations can only be considered for finite basis and given that the number of FCI determinants grows exponentially with the system size, the technique is only applicable to small molecules.

Truncated CI expansions, for example CISD (configuration interaction with singles and doubles excitations), whereby only the singly and doubly excited configurations are considered in the CI wave function, yields a technique which lacks size-consistency (as opposed to FCI which is size consistent). For example, consider a CISD calculation on two non-interacting H₂ molecules. The composite system does not allow us to consider the case where both molecules are doubly excited, as this requires a quadruple excitation. Thus, the energy of twice the monomer calculation cannot be equal to that of the composite system.

1.8.2. Quadratic Configuration Interaction

Quadratic configuration interaction (QCI) [26] was developed for the purpose of introducing size consistency exactly for CISD, yielding an acronym QCISD. QCI is best considered as a special case of the more general coupled cluster technique - in fact, QCID and CCD are identical methodologies.

It has been shown [27] that the inclusion of up to and including triple excitations are important for obtaining accurate molecular energies, thus, the QCI approach was extended to incorporate triples excitations using a perturbative approach to yield QCISD(T). The application of such a method with a suitably large basis has a very expensive computational cost.

Pople constructed a series of methodologies [28-31] with the aim of evaluating bond energies, heats of formation, ionisation energies and electron affinities of atoms and molecules to within 1-2 kcal/mol. The resulting 'G2' theory is a sequence of well-defined calculations which uses considerably less expensive methods to approximate a high accuracy/large basis QCISD(T) calculation. While the G2 approach is less expensive than the large calculation it is approximating, it still uses highly expensive correlated techniques and thus can still only be applied to relatively small molecules. Note also that when G2 theory was constructed there was surprisingly little experimental data available to gauge its success, and while it appears to be generally very accurate, it has been found to fail when applied to some quite simple problems [32].

1.8.3. Coupled Cluster

Coupled cluster (CC) theory [33] is a general size consistent technique and has become the more prominent and generally accepted method for the effective and accurate treatment of electron correlation. In particular, the approximation denoted CCSD(T) has been shown to consistently yield predictive chemistry [33].

The approach used in Coupled Cluster theory is based on an exponential wave function

$$\Psi_{CC} = \exp(T)\Psi_0 \quad (1.49)$$

where T

$$T = \sum_i T_i = T_1 + T_2 + T_3 + \dots \quad (1.50)$$

is an operator that creates excitations from the independent particle HF reference. The operator, for example T_2 ,

$$T_2\Psi_0 = \sum_{\substack{a>b \\ r>s}} t_{ab}^{rs} \Psi_{ab}^{rs} \quad (1.51)$$

has coefficients, t_{ab}^{rs} , which are different from the CI coefficients c_{ab}^{rs} as the CCD wave function exponential expansion

$$\Psi_{CCD} = \exp(T_2)\Psi_0 = (1 + T_2 + T_2^2/2 + T_2^3/3! + \dots)\Psi_0 \quad (1.52)$$

yields

$$\Psi_{CCD} = \Psi_0 + \sum_{\substack{a>b \\ r>s}} t_{ab}^{rs} \Psi_{ab}^{rs} + \frac{1}{2} \sum_{\substack{a>b \\ r>s}} \sum_{\substack{c>d \\ t>u}} t_{ab}^{rs} t_{cd}^{tu} \Psi_{abcd}^{rstu} + \dots \quad (1.53)$$

so that we have introduced quadruple, sextuple, *etc.* excitations into the wave function until $T_2^n/n!$ vanishes for $n/2$ electrons. The higher excitation terms are not introduced generally, as the coefficients are simply products of double excitation coefficients.

Unfortunately, cost scaling characteristics of highly accurate approaches, such as CCSD(T), renders them unsuitable for anything other than small molecules.

1.8.4. Perturbation Theory

Many-body perturbation theory, or better known as Møller-Plesset perturbation theory from the authors of the original theoretical development [34], considers and assumes electron correlation as a small perturbation from the Hartree-Fock approximation. In obtaining the HF wave function and energy, Ψ_0 and E_0 we have obtained an approximate wave function and energy for the exact Hamiltonian eigenvalue problem (1.5). The HF wave function and energy are exact eigenfunctions and eigenvalues for the Hartree-Fock Hamiltonian H_0 . The exact Hamiltonian operator can thus be written as

$$H = H_0 + \lambda V \quad (1.54)$$

In doing so we consider the perturbation, V , to be small, and have introduced the parameter λ for mathematical convenience which we will set to unity later. We expand the exact wave function and energy (1.5) in terms of the HF wave function and energy

$$E = E^{(0)} + \lambda E^{(1)} + \lambda^2 E^{(2)} + \lambda^3 E^{(3)} + \dots \quad (1.55)$$

$$\Psi = \Psi_0 + \lambda \Psi^{(1)} + \lambda^2 \Psi^{(2)} + \lambda^3 \Psi^{(3)} + \dots \quad (1.56)$$

Substituting the expansions (1.54-6) into (1.5), gathering terms in λ and then multiplying by Ψ_0 and integrating over all space yields equations for the n^{th} order energies $E^{(n)}$. For example, the zeroth and first order energies are given by

$$E_0^{(0)} = \langle \Psi_0 | H_0 | \Psi_0 \rangle \quad (1.57)$$

$$E_0^{(1)} = \langle \Psi_0 | V | \Psi_0 \rangle \quad (1.58)$$

such that the HF energy is

$$\begin{aligned} E_0 &= \langle \Psi_0 | H | \Psi_0 \rangle \\ &= \langle \Psi_0 | H_0 + V | \Psi_0 \rangle \\ &= E_0^{(0)} + E_0^{(1)} \end{aligned} \quad (1.59)$$

which is the sum of the zeroth and first order energies. The correlation energy is thus

$$E_{corr} = E_0^{(2)} + E_0^{(3)} + E_0^{(4)} + \dots \quad (1.60)$$

of which the first term (called the MP2 energy) is given by

$$E_0^{(2)} = \frac{1}{4} \sum_{abrs} \frac{|\langle ab || rs \rangle|^2}{\epsilon_a + \epsilon_b - \epsilon_r - \epsilon_s} \quad (1.61)$$

where

$$\langle ab || rs \rangle = \langle ab | rs \rangle - \langle ab | sr \rangle \quad (1.62)$$

and

$$\langle ab|cd\rangle = \int \psi_a(\mathbf{r}_1)\psi_c(\mathbf{r}_1) \left[\frac{1}{r_{12}} \right] \psi_b(\mathbf{r}_2)\psi_d(\mathbf{r}_2) d\mathbf{r}_1 d\mathbf{r}_2 \quad (1.63)$$

which can be rewritten in terms of the two electron integrals

$$\langle ab|cd\rangle = \sum_{\mu} \sum_{\nu} \sum_{\lambda} \sum_{\sigma} C_{\mu a} C_{\nu c} C_{\lambda b} C_{\sigma d} (\mu\nu|\lambda\sigma) \quad (1.64)$$

This perturbative approach yields non-variational but size consistent energies with the energy expansion (1.55) tending to converge rapidly. The rate of convergence [35] can be adversely affected when the HOMO-LUMO (highest occupied molecular orbital - lowest unoccupied molecular orbital) gap is small or in systems with stretched bonds.

1.8.5. Other approaches

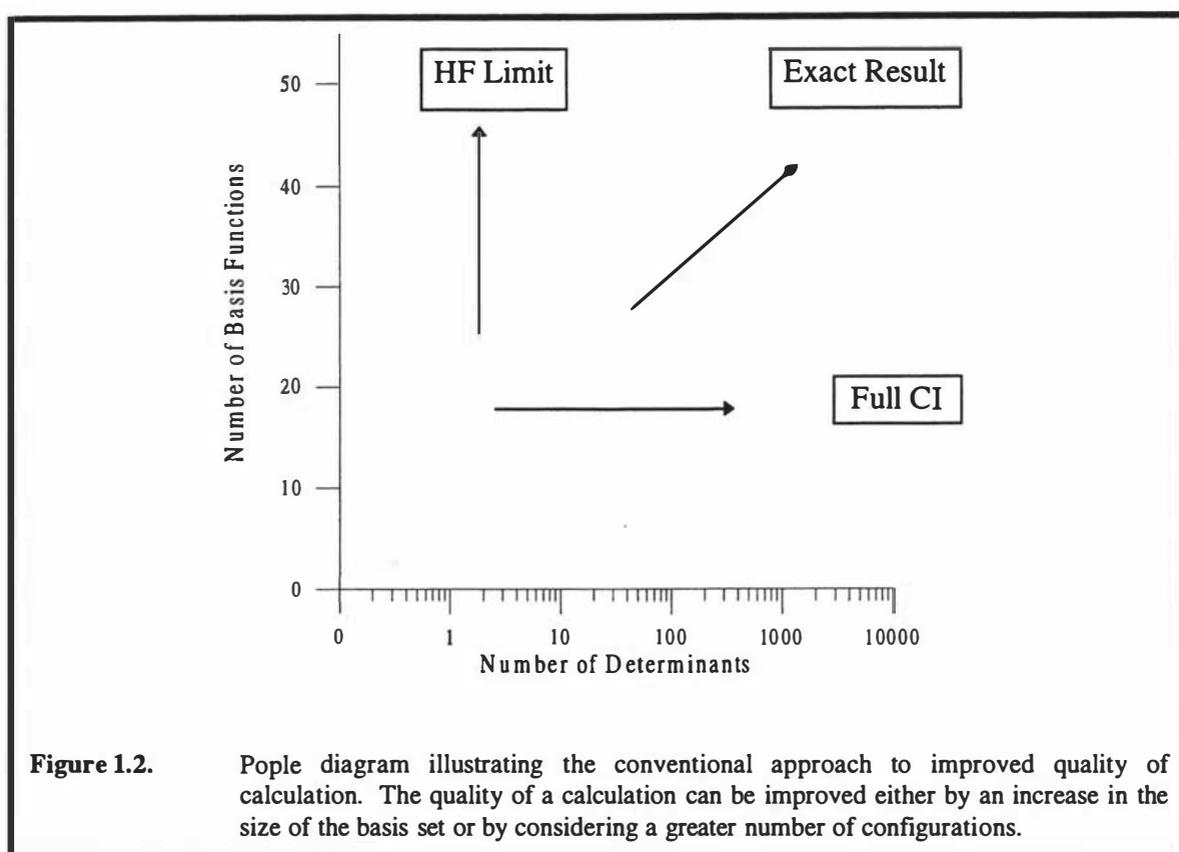
There have been many other approaches trialed and considered for obtaining corrections to the HF energy, with none any more promising, or simple in concept, than that used by Hylleraas [36-38]. The difficulty in solving the equation (1.5) is the inter-electronic repulsion term which couples the electron motions - the HF approach is to use an averaging of all other $n-1$ electrons. Hylleraas' approach, unlike those mentioned above, does not begin with the HF wave function and energy, but begins by incorporating an r_{12} term into the wave function. Using this approach he was able to obtain very accurate numerical energies for He, and others [39] have obtained similar accuracies for H_2 dissociation. While the approach is simple in concept, like the use of Slater-type orbitals (STO's) in MQM, the resulting integrations are difficult and state-of-the-art implementations remain numerical [40].

1.8.6 Additional Comments

The big drawback with all the above mentioned conventional correlated methods is their computational expense (Table 1.1). The underlying reason that the application of MQM has not grown as rapidly as perhaps anticipated one to two decades ago, is the very unfavourable cost scaling characteristics of the conventional correlated techniques. For example, if a computer manufacturer made a huge breakthrough in central processing unit (CPU) design which saw an increase in the average speed of a

workstation by an order of magnitude, the increase in the size of molecules which could be subject to calculation using these correlated techniques would be minuscule.

The positive aspect of the conventional methods is that the quality of one's calculation can be systematically improved (Figure 1.2). *e.g.* The size of the basis can be increased to obtain more flexibility in the modelling of MO's and/or the CI (1.48) or perturbation series can be expanded (1.56), which in the limit, tends towards the exact energy and wavefunction.



Within Pople's list of desirables the consideration of accuracy is typical of the approach of the correlated techniques. Aiming to obtain as much of the correlation energy as possible appears to have become an over-riding factor and this has been their major downfall (with the present algorithms and computing facility). Chemists are concerned mainly with differences in energy. The underlying success of quantum chemistry is a reliance on cancellation of errors which makes systematic errors in total energies tolerable. In fact, the level of accuracy afforded in energy differences is often an order of magnitude or more than that of the total energies [41]. Similarly, the

variational requirement for total energies does not aid the accuracy of resulting chemistry, as a difference of two variational energies is a non-variational energy.

Method	Formal Cost
HF	N^4
MPn	N^{3+n}
QCISD	N^6
QCISD(T)	N^7
CCSD	N^6
CCSD(T)	N^7
CCSDT	N^8
FCI	$N!$

Table 1.1. Cost scaling behaviour of Conventional Correlated methods with the size (N) of the molecular system.

1.9. Density Functional Theory

The correlated methods described above all have horrific cost scaling characteristics (Table 1.1). They begin scaling at N^5 , become as large as N^7 for highly accurate methods, and exponential for full configuration interaction. Given that the HF approximation captures the majority of the total energy, the down side to the conventional approach is that the majority of the computational effort is spent on calculating a small fraction of the total energy. This is highly undesirable.

The Density Functional Theory by Hohenberg, Kohn and Sham [42-44] provides an alternative to the Hartree-Fock approximation. On the other hand, density functionalists view the HF approximation as just another functional, rationalising DFT as the superset of SCF methodologies.

The Density Functional Theory is based on work by Dirac [45], who found that the exchange energy of a uniform electron gas may be calculated exactly knowing only the charge density. This idea has been extended to correlation energies, and corrections to Dirac's exchange functional introduced to account for the fact that molecular charge densities are poorly approximated by a uniform electron gas. This has culminated in an increasingly growing number of density functionals (see [46] for examples). Gill [47] has recently challenged the uniform electron gas origin and has developed a functional based on an electron in a ball. But whatever the basis of the derivation of above

mentioned approximate functionals, the important fact is an existence theorem [43] for a functional which yields the exact ground state energy.

Within the Kohn-Sham formulation [42-44,48], the electronic energy can be written as

$$E = E_T + E_V + E_J + E_{XC} \quad (1.65)$$

where E_T is the kinetic energy, E_V is the electron-nuclear interaction energy, E_J is the Coulomb self interaction of the electron density $\rho(\mathbf{r})$ and E_{XC} is the exchange-correlation energy. Using spin-unrestricted molecular orbitals we can write the alpha and beta electron densities as

$$\begin{aligned} \rho_\alpha(\mathbf{r}) &= \sum_{i=1}^{n_\alpha} |\psi_i^\alpha|^2 \\ \rho_\beta(\mathbf{r}) &= \sum_{i=1}^{n_\beta} |\psi_i^\beta|^2 \end{aligned} \quad (1.66)$$

where n_α and n_β are the number of alpha and beta electrons respectively, and the MO's are as given by (1.15). Thus the total electron density is

$$\begin{aligned} \rho(\mathbf{r}) &= \rho_\alpha(\mathbf{r}) + \rho_\beta(\mathbf{r}) \\ &= \sum_{\mu\nu} P_{\mu\nu}^T \phi_\mu(\mathbf{r}) \phi_\nu(\mathbf{r}) \end{aligned} \quad (1.67)$$

The components of (1.65) can now be written as

$$\begin{aligned} E_T &= \sum_{i=1}^{n_\alpha} \langle \psi_i^\alpha | -\frac{1}{2} \nabla^2 | \psi_i^\alpha \rangle + \sum_{i=1}^{n_\beta} \langle \psi_i^\beta | -\frac{1}{2} \nabla^2 | \psi_i^\beta \rangle \\ &= \sum_{\mu\nu} P_{\mu\nu}^T \langle \phi_\mu(\mathbf{r}) | -\frac{1}{2} \nabla^2 | \phi_\nu(\mathbf{r}) \rangle \end{aligned} \quad (1.68)$$

$$\begin{aligned} E_V &= -\sum_{A=1}^M Z_A \int \frac{\rho(\mathbf{r})}{|\mathbf{r} - \mathbf{R}_A|} d\mathbf{r} \\ &= -\sum_{\mu\nu} P_{\mu\nu}^T \sum_A \langle \phi_\mu(\mathbf{r}) | \frac{Z_A}{|\mathbf{r} - \mathbf{R}_A|} | \phi_\nu(\mathbf{r}) \rangle \end{aligned} \quad (1.69)$$

$$\begin{aligned}
 E_J &= \frac{1}{2} \langle \rho(\mathbf{r}_1) | \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} | \rho(\mathbf{r}_2) \rangle \\
 &= \sum_{\mu\nu} \sum_{\lambda\sigma} P_{\mu\nu}^T P_{\lambda\sigma} (\mu\nu|\lambda\sigma)
 \end{aligned}
 \tag{1.70}$$

$$E_{xc} = \int f(\rho(\mathbf{r}), \nabla\rho(\mathbf{r})) d\mathbf{r} \tag{1.71}$$

Substituting these expressions into (1.65) and minimising with respect to the unknown MO coefficients yields a set of matrix equations exactly analogous to the UHF case

$$\begin{aligned}
 \mathbf{F}^\alpha \mathbf{C}^\alpha &= \varepsilon^\alpha \mathbf{S} \mathbf{C}^\alpha \\
 \mathbf{F}^\beta \mathbf{C}^\beta &= \varepsilon^\beta \mathbf{S} \mathbf{C}^\beta
 \end{aligned}
 \tag{1.23}$$

where the Fock matrix elements are now generalised to

$$\begin{aligned}
 F_{\mu\nu}^\alpha &= H_{\mu\nu}^{core} + J_{\mu\nu} - F_{\mu\nu}^{XC\alpha} \\
 F_{\mu\nu}^\beta &= H_{\mu\nu}^{core} + J_{\mu\nu} - F_{\mu\nu}^{XC\beta}
 \end{aligned}
 \tag{1.72}$$

where $F_{\mu\nu}^{XC\alpha}$ and $F_{\mu\nu}^{XC\beta}$ are the exchange-correlation parts of the Fock matrices dependent on the exchange-correlation functional used. The Pople-Nesbet equations are obtained simply by allowing

$$F_{\mu\nu}^{XC\alpha} = K_{\mu\nu}^\alpha \tag{1.73}$$

and similarly for the beta electron piece. Thus, the density and energy is obtained as for the Hartree-Fock approximation, where initial guesses are applied for the MO coefficients and an iterative process applied until self consistency obtained.

The exciting feature of applying DFT methods to solve chemical problems is that one can cheaply incorporate correlation into a calculation. The rate limiting step in determining the total electronic energy (1.65) is determining the Coulomb energy - all other energy contributions can be obtained in work that scales linearly with the size of the system [46]. DFT does however, have a drawback. While DFT has been shown to yield good chemistry (for example, see [49]), it can often, unexpectedly, yield quite disastrous chemistry (for example see [50,51]) without the capacity to systematically improve the quality of result as for the conventional correlated methods.

1.10. Commentary

The main focus of this thesis, introduced in the preceding chapter, is to enhance the means by which we may quickly obtain chemistry from approximations to Schrödinger's equation. Presented in this chapter are the common approximations to the solution of this equation. How one may implement these approximations is completely undefined, allows the possibility for alternative routes for the computing the mathematics, and has been the focus of vigorous research for many hundreds of man-years. During this time, a great deal of insight has been obtained, the most important features are noted thus.

Firstly, that the potential exists to obtain exact answers to all chemical problems through the solution of Schrödinger's equation which we can approximate to any given precision given unlimited resources. Since resources are very finite, a number of important theoretical approximations have been developed of which the most exciting is the existence theorem for an exact density functional which, if known, will yield the exact energy and electron density. The second important observation, is that given the exact functional (and an implementation similar to present state-of-the-art functionals) the rate limiting step in obtaining the exact energy and density is the formation of the two electron integrals used in constructing the **J** matrix.

While DFT accounts for approximately 90% of all quantum chemical calculations being performed, the sometimes unpredictable nature of results and the inability to systematically improve the quality of calculation may mean that a place for the conventional correlated techniques remains in the quantum chemist's tool kit. The basis of the next tier of approximations (which will lead to a solution of any given precision granted unlimited resources) is the HF wave function and energy. The rate limiting step in obtaining the HF wave function and energy is exactly analogous to the DFT case - the **J** matrix/Coulomb energy. The third and final important observation is that the post-Hartree-Fock approximations presented above deal mainly with manipulations of combinations of intermediates formed in a HF calculation, particularly the two electron integrals. It seems that the key to obtaining approximate solutions is in the formation of

two-electron integrals, hence the key to rapidly obtaining approximate solutions is in swiftly forming the two-electron integrals.

CHAPTER TWO

TECHNOLOGICAL MOTIVATION

2.0. Introduction

Computer technology has had a major role in the development of MQM. Presented in this Chapter are a number of features of computing technology and a brief history of the development of the computer, computer architectures, operating systems and programming languages. A discussion on the development of algorithms and the effect of modern architectures/platforms, operating systems and programming languages is made which ties in the two main factors which have contributed to the increasing realisation of MQM, that is, the computer and the development of more efficient algorithms.

2.1. High-Performance Computing

High-performance scientific computing, or computational science is the link between computers and algorithms. High-performance scientific computing is solving problems whose complexity renders an analytical solution impossible through the exploitation of the technology produced by computer scientists. Problems requiring numerical simulation, symbolic manipulation, visualisation or plain old number crunching, once considered impossible or at best impractical can now be routinely accomplished.

Computational science is the transformation and implementation of scientific theory into efficient algorithms which requires both theoretical and experimental skill. The transformation of a new theory into an efficient algorithm requires understanding of programming concepts, mathematical and physical intuition and theoretical insight, whereas the production of the computer code is much like experimentation, requiring debugging, testing and organisation to yield a highly efficient product.

2.2. Historical Prospective

2.2.1. Computing Facility

In the 1960's computers were housed in buildings, used only by experts and were somewhat error prone. Data input was provided through ticker-tape which led on to punch cards, and user input mistakes found on the return of the printed output days after submission, required correction of the mis-punched character and re-submission of the job to the batch queue.

The 1970's saw the advent of remote job entry stations where it became possible to edit and submit jobs to the mainframe electronically. This kind of direct user access resulted in overloading of computational resources and led to the development of the mini-computer. Mini's made computing resources affordable for groups or departments and led to the development of personal or micro computers. These latter resources, however, offered little for scientific computation due to their slowness and small size, and as the supercomputer with its large scale storage, memory and speed began to appear, Mini's became redundant for scientists, even though access to the supercomputer was limited.

The 1980's were revolutionary as workstations, Unix and graphical interfaces allowed individuals access to affordable computing resources on the desktop. Networks then enabled sharing of printers, disks and CPU's. Communication between computers became communication amongst users and computers started to become a necessity rather than a luxury.

The popularity of the workstation has continued into the 1990's, with speed, memory and storage configurations increasing dramatically in magnitude year by year. Computer clusters and multi-processor servers have reduced the necessity of supercomputing expense for serious computation.

2.2.2. Programming Languages

As computers, operating systems and the number and expertise of users have developed, so too have programming languages. Programming is the art of transforming the science and mathematics into a concise set of instructions so that the computer can

perform the required task. Computers only understand machine code, and given that our aim in writing a program is not only to obtain an answer but also to document the algorithm and enable ease of modification, programs tend not to be written in machine code. Instead, programming languages are used to construct a portable generic code which is then usually compiled (translated into machine code) for the specific machine.

The FORTRAN programming language was one of the first compiled languages developed and has stood the test of time, at least for science and engineering applications. It has been well tested and optimised for most computer systems and it still yields the fastest binaries for basic number crunching. Its big advantages are its simplicity, portability and advanced optimisation within compilers.

BASIC contrasts to FORTRAN in that it is an interpreted language rather than compiled. It is quite similar to FORTRAN in that a beginner can learn a few basics and begin to write code with limited tuition, but being an interpreted language it is far too slow for other than trivial applications.

The rise of the Unix operating system which is written in the C programming language and the programming freedom the C language offers, has stunted the continuation of FORTRAN as a mainstream language. However, the C language has many pitfalls and it is especially easy to misuse. Visually, C is much less appealing and difficult to follow for the beginner. This may seem pedantic, but if one considers that the aim of developing code is to both express and document the algorithm for others to modify at a later stage as well as obtaining information after compilation, this visual/readable aspect is quite important.

More recently the object-oriented programming languages, such as C++, have appeared and offer significant advantages over its procedural predecessors. C++ can be considered as a superset of the C programming language and suffers similar pitfalls to C in that it requires a more advanced understanding of the language to begin programming effectively. Whilst for science and engineering applications the concept of merging the rapid number crunching advantages of FORTRAN with the object-oriented C++ appears to enable a large gain in programming freedom and ease of development, the successful marriage [52] proves more difficult than it perhaps should be.

FORTRAN, throughout all the programming language developments, has taken a very conservative approach in its own development. Trying to maintain a simplicity with semantics, and incorporating positive aspects from new languages while avoiding their mistakes, has allowed compiler developers to concentrate on preserving the level of optimisation of resulting binaries. FORTRAN 90, for example, has incorporated a number of useful structures for parallel programming and it is likely that the next generation FORTRAN language will also incorporate positive aspects of the object-oriented developments.

2.3. Algorithm Development

An algorithm is a set of rules for obtaining the required mathematics. The mathematics does not however, dictate how an algorithm is to be constructed as there are many different ways of achieving the end. Within the various possibilities, one or a few algorithms will be more efficient than the rest. Consider, for example, an arithmetic or geometric series. A bad algorithm would involve summing all the terms, or until convergence to within some predetermined accuracy was obtained. A good algorithm would most likely use the well known summation formulae [53]. Both approaches achieve the required mathematics, one approach is far superior to the other.

Developing an algorithm can be likened to an experiment. At each stage of development one considers generalisations, makes observations and looks for ways to improve performance. In developing an algorithm, one's aim is to produce a program for others to use, usually not on the particular machine or type of machine on which it was developed. Such is the necessity for portability.

Portability requires that the algorithm has been written in a suitable language with widespread availability. Portability also means that while the code may not be fully optimised for all platforms/architectures, it has been constructed in such a way that it will compile on most platforms and be expected to perform adequately. This has been aided by the development of machine specific mathematical libraries [54]. Computer vendors have developed highly optimised libraries of common subroutines, specific to their machine. Thus, a program with calls to such a subroutine on platform A could be expected to perform similarly on platform B, relative to peak performance on each

individual machine. The archetypal example are the Basic Linear Algebra Subroutines [55-57] (BLAS). Most modern platforms come with a BLAS library so that any program with calls to such can consider that part of the code fully optimised. The Linear Algebra Package [58] (LAPACK) library is an example of a suite of subroutines that uses calls to the BLAS to afford almost total portability - amongst scalar and vector platforms at least.

Developing a computer algorithm is not only about getting the right answer. It is also about documentation, efficiency, portability and adaptability. While a piece of code cannot be written to afford complete optimisation on all platforms, one can take suitable steps in the choice of language and the use of libraries such as BLAS. Algorithm development is mainly about translating new science and new theories into code. Approximately 5% of the time is spent developing the new theory, and 95% of the time writing, debugging and rewriting code to afford a well documented, efficient, portable and adaptable algorithm.

For large scale computation, the desktop workstation has become the developer's workbench. Code is written and debugged on the workstation and then ported to the supercomputer where performance is evaluated. The developer can then return to the workstation, making changes to the original code, debugging and then returning to the supercomputer for further evaluation. This reduces cost by developing code on a much cheaper platform and frees up the supercomputer for the more serious tasks of large scale computation.

A major consideration in the development process is the platform for which the code is being developed. Scalar, vector and parallel architectures require specialist knowledge and understanding of the machine and how the compiler will optimise the code, although it is easier to afford optimal code on the scalar machines relative to their vector and parallel contemporaries.

2.4. Modern-day Computer Architectures

Modern-day computers can be grouped into three main levels. The personal computer, the workstation and the supercomputer. Presently, the speed (memory and

storage size) of the computer dollar doubles every eighteen months such that yesterday's workstation becomes tomorrow's PC and the prospect of outlaying many thousands of dollars on computing facility is erred by the notion that in a short period of time it will be rendered out of date and incapable of efficiently running the latest operating system and software.

Within the three levels of computing there are several different architectures from which to choose. This is more evident at the high end of the spectrum than at the PC level, but none the less, as the high end of the market develops, the trickle down effect creates a wider market for the high end PC user. Silicon Graphics, for example, concentrates all its research and development budget at the high end. Low end users receive the benefits in either a down sized version or as newer technology becomes available at the high end, the older technology becomes more affordable.

2.4.1. Vector Machines

In the past, the term supercomputer has always referred to vector computing, but nowadays refers to either vector or parallel computers. Vector computing or vector processing tackles the most demanding task in scientific computation of matrix operations more efficiently than its scalar counterpart. For example, to add two vectors of length N to form a third vector requires N sequential additions. In a scalar machine each addition requires the i^{th} array element of the first vector to be obtained from its location in memory followed by obtaining the i^{th} array element of the second vector from its memory location, the addition of the two values in a CPU register and then the storage of the resultant in the i^{th} location of the third array. Much time is wasted in this scalar process telling the computer repetitiously where to fetch, add and store. A vector machine is capable of performing operations on entire sections of an array. (To a computer an n -dimensional matrix is simply another vector, so one shouldn't confuse the mathematical vector with the computational vector.)

Vector performance is restricted both by the nature of the algorithm (such is an algorithm's vectorisability) and by the limitations of the programmer. While an algorithm may be in theory vectorisable, if the code has been written poorly the resulting performance may be poor. The design of code for vector machines requires an

understanding of what the compiler is going to do with the code and how the compiler may or may not be tricked into increased optimisation by unrolling outer loops, for example.

The rate of computer operations per unit time r , can be written as [59]

$$r = (t_0/n + \tau)^{-1} \quad (2.1)$$

where n is the vector length, τ is the clock period and t_0 is the start-up time. The clock speed is simply the time taken to execute the simplest instruction, and the start-up time accounts for the time it takes to load up the vector pipe-line. Thus, for long vectors the computational rate tends towards the clock speed. The size of vector required to approach half peak performance can be shown to be

$$n_{1/2} = \frac{t_0}{\tau} \quad (2.2)$$

and is dependent on the hardware. Variations in hardware characteristics also result in widely varying performance, so while it is often possible to write code that will run at or near peak performance for scalar machines, this is often not the case for vector machines.

Vector processing has been and is still associated with supercomputers because the obvious success of such processing is reliant also on fast processors, large and fast memory, and fast communication between the various units. Such expense is not utilisable nor affordable for desktop machines at present.

2.4.2. CISC and RISC

Complex Instruction Set Computers (CISC) utilise the early CPU chip design where hundreds of thousands of devices were placed on a single chip. As revolutionary as this was in the early 1970's, much of each chip was dedicated to a microcode of over 1000 machine language instructions. This code mimicked the higher level languages and, as a result, was exceedingly slow - up to 10 computer cycles per instruction. As compilers translating high level languages into machine code improved, many of the 1000 available instructions became redundant. As few as 10 low-level machine

instructions accounted for 80% of executed instructions [60], and 30 accounted for 99% of the use.

The philosophy behind the Reduced Instruction Set Computer (RISC) system was to enable the production of a cheaper, more efficient and faster chip requiring compilers to utilise fewer instructions within the chip. The aim was to reduce the number of cycles required per instruction. So while on the one hand the CISC system was trying to reduce the CPU time requirement by reducing the required number of instructions to perform a given task, the RISC system on the other hand was aiming to reduce the number of machine cycles (to one) for a given instruction, so that while the total number of instructions required to perform a task may be larger than that of the CISC system, the total number of machine cycles would be fewer.

An additional advantage of the RISC architecture is the freeing up of space as using a reduced instruction set allows an increase in the number of internal registers. This has resulted in RISC processors gaining a further edge in floating point arithmetic. The classic example is the IBM RS/6000 [61] where a floating point multiply-add (FMA) can be obtained in one clock cycle.

2.4.3. Parallel Computers

Given that a single processor's speed is physically limited by the nature of the material from which it is made, the next logical step in the move to obtain higher performance is the use of multiple processors. In terms of the supercomputer this is in the form of massively parallel machines which contain many hundreds, even thousands of processors, and on the desktop, either a small number of processors in the one machine or a cluster of single processor workstations linked together over a network.

To the casual observer, the prospect of splitting a job into two, three and even six pieces to spread over such a number of processors to afford a similar factor of speed increase would seem easily obtainable with a moderate level of due thought. However, to afford such theoretical speed-ups on the Massively Parallel (MASPAR) machines requires significant effort and often a major rethink into the layout of the algorithm.

While the basic design of vector machines follows similar principles from vendor to vendor, the same cannot be said for parallel machines, and in fact the variation is much worse. Shared memory vs processors with individual memory not to mention the differences in philosophy in parallel programming language libraries [62], yields an extreme lack of uniformity in the parallel domain.

2.5. Summary

Computational science is the adaptation of new scientific theory into computer code exploiting the advances in compiler, programming language and hardware technology. The aim is to afford an algorithm to enable efficient computation, portability of code, ease of adaptability and to document the science. To afford such an algorithm requires an intuitive understanding of the science to be implemented, much experimentation with optimisation and debugging of the developing code, a suitable choice of programming language, as well as a basic overview of the nature of the platforms for which the code is intended.

CHAPTER THREE

HISTORY OF THE QUANTUM CHEMICAL COULOMB PROBLEM

3.0. Introduction

A common misconception held by many chemists would suggest that the task of modelling real chemistry is extremely difficult, perhaps unachievable, because of the many different types of interactions and forces that must be considered. The misconception is that they consider hydrogen bonds, dispersion forces, dipole-dipole interactions *etc.*, (many of which are used to explain protein folding and conformations, entropies of vaporisation and boiling points of non-polar liquids, and commonplace in many texts) to be quite different forces. There is in fact only one force determining chemical phenomena, the Coulomb force.

The Coulomb force manifests itself in molecular quantum mechanics in the form of two electron repulsion integrals (ERI's), nuclear-nuclear repulsion and electron-nuclear attraction integrals. Given that the Coulomb force determines all of chemistry and that molecular quantum mechanics uses Born-Oppenheimer separation of nuclei from electrons, it is not surprising that the majority of the computational effort in a MQM calculation goes into determining or manipulating the ERI's.

In Chapter One, the theoretical foundation and motivation for MQM was expressed and in Chapter Two the technological means by which Chapter One's potential may be realised was discussed. In this Chapter we seek to illustrate that new algorithms have had a significant impact on the realisation of some of MQM potential, in particular, within the SCF methods. Furthermore, significant hindrance to the more widespread application of MQM incorporating very large molecules (for example, in biological fields) is due to what has become known as, the Quantum Chemical Coulomb Problem. It should also be noted that this problem is more general, is common to many other areas of chemistry, physics and mathematics and that it has, and will continue to receive much attention and research focus.

This Chapter is divided into two main sections. The first section will present, chronologically, the rapid realisation that the bottle-neck within SCF calculations is the formation of the two electron repulsion integrals, a brief summary of the huge research effort which has been put into optimising the formation of the integrals, and various approximations imposed in an attempt to offset the extreme computational cost. The apparent general nature of the Coulomb problem will then be highlighted, along with the need for $O(N)$ methods. The more recently implemented CFMM, the prospect of using Fast Wavelet transforms and the concession of the Divide and Conquer method will be summarised. The second section will discuss the N body Coulomb problem in general and some of the sub-quadratic approximation techniques previously employed. A critique of these methods will be left for Chapter Nine where the KWIK algorithm will be contrasted with present state-of-the-art algorithms.

3.1. The Quantum Chemical Coulomb Problem

3.1.0. Introduction

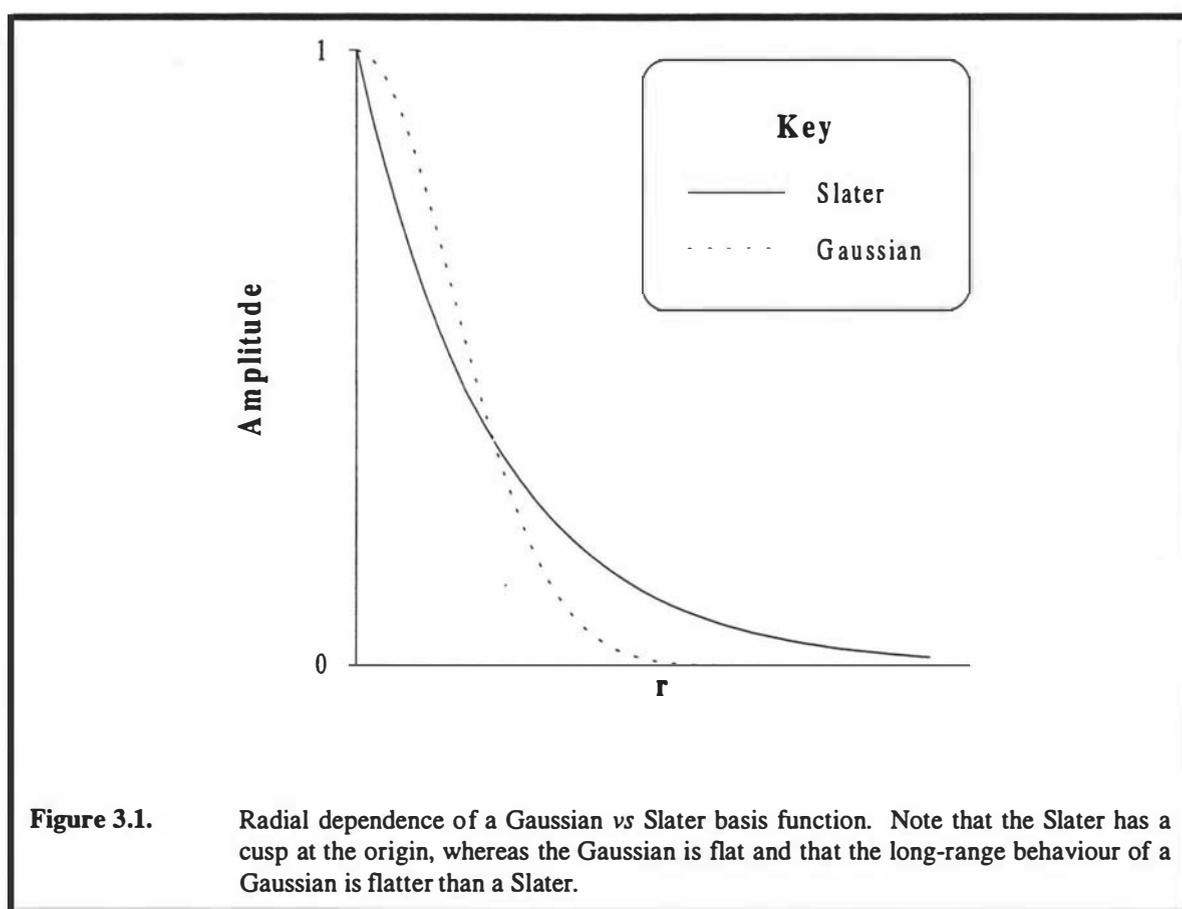
Schrödinger solved his equation for the hydrogen atom approximately thirty years before a computer was used to solve quantum chemical calculations, so it was possibly thirty years before any detailed consideration had been given to the practical difficulties of obtaining approximate solutions to Schrödinger's equation using computers, MQM being purely a descriptive science until the development of the digital computer. During the descriptive era, molecular orbitals were expanded as a linear combination of Slater-type orbitals (STO) - they being 'natural' basis functions and having the 'correct' behaviour at the origin and asymptotically. However, evaluation of the ERI's using STO's were at best expensive and, at worst, intractable for molecules. Such were the beginnings of the Coulomb problem - the task of evaluating the ERI's in the first instance, difficult by virtue of their intractability.

3.1.1. Gaussian Basis Functions

Boys [63] suggested that Gaussian-type orbitals (GTO's) could be used on the basis that all the integrals could be evaluated easily (analytically) and efficiently. Thus overcoming laborious numerical integrations (for example, see [64]). The success of a

Gaussian function is due to the product of two Gaussians, concentric or non-concentric, being a third Gaussian, and the Fourier transform of a Gaussian again being a Gaussian function.

As a Gaussian function has the ‘wrong’ behaviour both at the origin and asymptotically (Figure 3.1), then clearly more GTO’s would be required to describe an atomic orbital than if STO’s were used. However, as standard basis sets of GTO’s were constructed [65] and the concept of theoretical model chemistries developed [66], this ‘incorrect’ behaviour became somewhat overshadowed by the success of the GTO’s in obtaining good chemistry [23]. Furthermore, it later became apparent that STO’s do not offer any significant advantage over GTO’s for obtaining correlation [41].



It is important to note that while the Gaussian basis function was a major step forward in enabling routine calculations on non-linear molecules and large basis sets, we need not limit ourselves to such functions when developing radically new algorithms for the evaluation of ERI’s as other authors have [67].

3.1.2. Electron Repulsion Integrals Over GTO's

The evaluation of the electron repulsion integral using GTO's is of fundamental importance and is a major hindrance to the widespread application of quantum chemistry. As such it has been the focus of hundreds of man-years of research. The many algorithms developed for evaluating ERI's have recently been reviewed [68] (the reader is referred to this review for a more in depth account of ERI evaluation) and further efficiencies continue to be achieved [69,70]. Our interest here is simply to highlight a number of developments and the generality of the problem.

Historically, the evaluation of the ERI's appeared to scale as the fourth power of the number of basis functions. The ERI is based on two pairs of basis functions (1.40). The first pair describing electron one and the second pair describing electron two. Given there are N functions in the basis set means there are [68]

$$N_{Total} = \frac{1}{8} N(N+1)(N^2 + N + 2) \quad (3.1)$$

total ERI's. However, many of these integrals are negligible and it is possible to screen most of these in two steps.

The first screening occurs during the formation of shell-pair data. The basic ERI is an interaction between two charge distributions, each of which are the products of two basis functions. By considering all pairs of shells in the basis set, and discarding those which are negligible on the basis of their overlap, it has been found that the number of significant shell-pairs grows only linearly with the size of molecule for a given basis set. In fact, in large molecules, most of the shell-pairs are insignificant. This can be easily rationalised by considering the rapid decay of the Gaussian function and in the case of Slater basis sets, exponential decay of shells. Thus, the total number of ERI's scales only quadratically with the size of the basis.

Discarding insignificant shell-pairs can be further refined by a least-squares fit of the contracted shell-pair data by a fewer Gaussian functions. This economisation of the shell-pairs [71] is carefully designed to ensure that the error imposed on ERI evaluation remains below the required threshold. Others have taken this economisation further by employing the Fast Gauss Transform [72,73] which uses an auxiliary set of basis

functions to model the charge density. Computationally, the advantage arises due to the reduction in the number auxiliary functions required *c.f.* significant shell-pairs.

The second step in screening negligible ERI's is simply a refinement of the shell-pair screening, achieved by employing bounds on the ERI's [74]. What remains, however, is still $O(N^2)$ ERI's and is rate limiting for SCF calculations. What is required is a sub-quadratic, preferably $O(N)$ method for determining the ERI's.

3.1.3. Linear Methods

In our discussions to date, we have indicated that the rate limiting step in determining the total energy is determining the Coulomb energy. This is a completely true statement given the MO coefficients. However, in order to determine the MO coefficients, one must diagonalise the Fock matrix [18]. The cost of diagonalising the Fock matrix scales as $O(N^3)$, but does not become significant until N becomes very large [75]. However, for large N it may be possible to reduce the cost of the diagonalisation step by exploiting the sparsity of the Fock matrix, although arguments to the contrary suggest that the Fock matrix remains dense for large systems [76]. Whatever the case, N must be very large before dense matrix diagonalisation becomes rate limiting using linear Coulomb methods, and such limits have yet to be reached with current computing technology.

Before turning our attention purely to the Coulomb problem, we must also consider the non-classical exchange terms. The computation of the HF exchange energy is similar to the Coulomb energy in that it also formally scales as $O(N^4)$, and the employment of cut-offs again reduces this to $O(N^2)$. Schwegler and Challacombe [76] have recently implemented theoretical conjectures (see [76] for references) that the continuous density matrix decays exponentially for insulators, thus affording construction of the exchange matrix in $O(N)$ work. Kohn [77] has recently extended the possibility of $O(N)$ determination of exchange to metals and is currently implementing these ideas. Note that within the framework of DFT, exchange has been shown [46] to be calculable in work that scales linearly with the system size. Thus, determining the Coulomb energy is the rate limiting step.

The first linear method available for MQM calculations was the Continuous Fast Multipole Method by White *et al* [78] which is a generalisation of the FMM for classical point charges to arbitrary continuous charges. Their intention was to derive a general method for continuous distributions of charge, then to apply the general method to the more specific quantum chemical Coulomb problem. Prior to their $O(N)$ implementation [79] White *et al* had made significant improvements to the original FMM [80] and upon implementation of the CFMM made further efficiencies [81,82]. Others [67,83-85] have followed White and implemented the CFMM, although their implementations have not yet been optimised, and some [67] have resorted to uncontracting shell-pairs, which severely impairs performance of ERI code required in parts of the CFMM calculation and thus reduces the performance of their implementation.

As mentioned, White *et al* have made numerous improvements to the original FMM and subsequently the CFMM. It became clear however, that improving the performance of the CFMM could also be achieved by improving the performance of ERI code as this was still required to obtain the short-range contributions (see the section below on the FMM for a more detailed account of FMM's). This observation culminated in a so-called J-Matrix engine [86] approach, whereby formal evaluation of the ERI's was avoided. Instead the desired **J** matrix elements are calculated directly, which significantly reduces memory requirements and improves performance.

The Fast Wavelet transform [87,88] has recently been applied to electronic structure theory [89-92], but to date has only been used as an alternative numerical method to the use of more standard basis functions. That is, it has not been used in an attempt to reduce the scaling behaviour of the Coulomb problem.

The 'divide and conquer' approach [93] has been claimed to be an $O(N)$ DFT method. The divide and conquer approach is a re-derivation of the standard Kohn-Sham [42-44] equations eliminating the global representation of localised density by the Kohn-Sham orbitals, with a set of localised partition functions. Thus, the whole system is divided into small subsystems and the total energy and electron density is calculated as a sum of the contributions from individual subsystems. Others [94] have modified the method, but the overall approach remains the same.

Divide and conquer's appeal is that its overall philosophy of partitioning a molecule into small subsystems is a philosophy not dissimilar to that of an organic chemist. Organic chemists tend to consider molecules as a 'sum' of interacting functional groups, talking of the effects that neighbouring functional groups have on the reactivity of other local functional groups. However, the problem with such a philosophy is that it tends not to be quantitative and extremely difficult to generalise. So while a number of workers have illustrated the success of divide and conquer in a number of molecular energy calculations [95-97] and have further developed the methodology for geometry optimisations [98], it has yet to be shown to be a general method - a major problem being how to *a priori* partition the molecular system. Furthermore, in implementations of the methodology to date, the Coulomb interactions have all been evaluated using $O(N^2)$ algorithms. Thus, divide and conquer *per se*, is not an $O(N)$ technique.

We cannot dismiss the usefulness of divide and conquer, as it does away with the $O(N^3)$ diagonalisation step. Such a bottle-neck exists within the framework of the semi-empirical methods [99]. The semi-empirical methods differ from the so-called *ab initio* methods in that they use experimentally determined parameters. Many of the semi-empirical methods are based on the Pople Approximation [100] which neglects differential overlap. That is, it is assumed that the Atomic Orbitals (AO's) are orthonormal under certain conditions. This has led to approximations such as CNDO, NDDO, PNDO, INDO and MINDO, which are, respectively, complete neglect of, neglect of diatomic, partial neglect of, intermediate neglect of, modified intermediate neglect of differential overlap. Such approximations are inherently crude and perform much better than one would anticipate as much of the error tends to be absorbed by the empirical parameters. Interestingly, the 'improvements' made to the basic Pople approximation tend to increase the number of Coulombic terms. As a final aside, Hückel theory, developed prior to the Pople approximation based methods, uses similar neglect of differential overlap, but only considers π electrons.

There has also been significant work put into reducing the level of $O(N^2)$ work required, either by cleverly modelling contracted shell pairs by another shell pair of reduced contraction [71] such that the error introduced in the ERI's remains below a

given threshold, or by modelling the charge density using auxiliary basis sets [101,102]. However, the fitting procedures have yet to be optimised as either the cost of the fit outweighs the savings or the error introduced by the fit is unacceptable.

As a final note, no $O(N)$ Coulomb method has yet been devised to determine the exchange energy in linear work. The linear method for non-classical exchange outlined above, works on the basis that Exchange terms can be approximated so that it can be calculated in $O(N)$ work.

3.2. The Coulomb Problem

3.2.1. The N Body Problem

Many important problem in physics and chemistry can be treated by methods based on the total interaction energy E between N localised matter distributions. In many cases, the interaction potential follows an inverse-square force law, *e.g.* Coulomb or gravitational potentials. Because such potentials are long-range, even widely separated distributions interact significantly, thus the employment of straightforward cut-off techniques introduces substantial errors and all pair-wise interactions must be considered. The computational cost of an algorithm requiring the consideration of all pair-wise interactions as a result of such slowly decaying potentials, scales quadratically with N .

The interaction of well-localised matter distributions *via* slowly decaying inverse-square law type potentials, requiring consideration of all pair-wise interactions, is known as the N body problem. However, given a large system of well-localised matter distributions the N body problem should now not be considered a problem that scales quadratically with the system size - even though such problems are most compactly expressed as a double sum which indicates such scaling. A number of algorithms have been developed that treat all pair-wise interactions in an N -‘particle’ system whose computational cost scales less than quadratically with N . A number of these will be discussed in the following sections.

Emphasis must be placed on the well-localised nature of the matter distributions - any distribution which contains totally delocalised matter cannot be treated in time that

scales linearly with the distribution size by any of the so-called $O(N)$ methods discussed here. However, this localisation consideration need not concern us greatly as $O(N)$ algorithms are designed primarily for considering large systems, the majority of large systems tend to be very well-localised, and smaller systems, in which one is more likely to come across total delocalisation of matter, can still be easily treated by quadratic methods.

3.2.2. The Ewald Summation Technique

In studying bulk properties of condensed matter systems, problems arise with small sample ($10 < N < 10,000$) surface effects. Molecules at the surface experience quite different forces from those in the bulk material. So, for example, a system comprising of 1000 molecules in a $10 \times 10 \times 10$ cube which has 488 molecules appearing on the cube face, is a very poor representation of bulk material.

Calculating the electrostatics of condensed matter systems, avoiding small sample surface effects, can be accomplished *via* the employment of periodic boundary conditions [103], and was first solved by Ewald [104]. Ewald splits the Coulomb potential into two rapidly converging real and Fourier space lattice sums, where the relative contributions of each sum is controlled by a parameter β .

The Ewald summation technique splits the Coulomb potential of a periodic system into a sum of two potentials

$$\phi(\mathbf{r}) = \phi_\gamma(\mathbf{r}) + \phi_c(\mathbf{r}) \quad (3.2)$$

where $\phi_\gamma(\mathbf{r})$ is written as a Fourier series typically of the form

$$\phi_\gamma(\mathbf{r}) = \frac{1}{\epsilon_0 V} \sum_{\mathbf{l} \neq 0} \frac{\exp(-k^2/(4\beta^2))}{k^2} \sum_{B=1}^N q_B \exp(i\mathbf{k} \cdot (\mathbf{r} - \mathbf{R}_B)) \quad (3.3)$$

where V is the volume of the simulation cell, q_B is the charge on the B^{th} particle, ϵ_0 is the permittivity of free space and $\phi_c(\mathbf{r})$ is the 'real space' part of the Ewald potential which can be shown to be [105]

$$\phi_c(\mathbf{r}) = \frac{1}{4\pi\epsilon_0} \sum_{B=1}^N q_B \frac{\text{erfc}(\beta|\mathbf{r} - \mathbf{R}_B|)}{|\mathbf{r} - \mathbf{R}_B|} \quad (3.4)$$

The ‘k-space’ potential is usually rewritten in terms of the total electrostatic potential energy of the original point charges q_A in the more compact double sum form

$$E_\gamma = \frac{1}{2\epsilon_0 V} \sum_{\mathbf{k} \neq 0} \frac{\exp(-k^2/(4\beta^2))}{k^2} \left| \sum_A^N q_A \exp(i\mathbf{k} \cdot \mathbf{R}_A) \right|^2 \quad (3.5)$$

The real space part can also be written similarly as an energy term.

In a simulation, if the length of the cubic system was doubled, then the number of particles and the volume of the system will increase eight fold. In order to maintain the same density of k-vectors, the k-space summation (3.5) would also need to increase giving rise to $O(N^2)$ scaling. By varying the β parameter, the scaling can be reduced to $N^{3/2}$ [105,106]. Typically however, in the study of condensed matter systems, short-range dispersion/repulsion interactions are incorporated (such as Lennard-Jones). Thus, it is usual to fix the β parameter such that only near neighbour interactions within the parent simulation cell are significant, allowing the use of efficient short-range techniques, but forces the k-space summation to scale as $O(N^2)$.

The Ewald technique is very general in that it may be applied to long-range potentials [107] other than the Coulomb potential. The Ewald technique is also very mature in that it has been highly optimised [106,108,109] for a variety of applications including a recent intriguing DFT application [110] which will be discussed in more detail in Chapter Nine.

3.2.3. Particle-Particle-Particle-Mesh

The particle-particle-particle-mesh (PPPM) method, developed from the particle-in-cell methods [111], is based on a separation of the total interaction potential between particles into a sum of short-range interactions and long-range interactions. The former are calculated directly and the latter by solving Poisson’s equation using periodic, and more recently [112], non-periodic boundary conditions. The overlap between the Ewald summation technique and the PPPM method is significant and the difference lies in the

evaluation of the long-range function. Ewald evaluates this analytically, whilst in the PPPM the grid based Fast Fourier Transform (FFT) is used. A comparison of the two methods with respect to relative computational efficiency [105] is best considered for larger systems, where the PPPM method's employment of the FFT, resulting in cost scaling of $N \log_2(N)$, gains considerable advantage over the Ewald method.

3.2.4. The Fast Multipole Method

The FMM was a culmination of the seminal work in tree codes of Barnes and Hut [113], and Rokhlin [114]. Tree codes are based on the observation that although gravitational or Coulombic fields have a complex local structure, the far field is smooth. In tree codes a cluster of charge is replaced by some simpler representation which is used to compute the influence of that cluster at large distances [115]. The simulation box is subdivided $\log_8(N)$ times and interactions are calculated downward recursively for all well-separated boxes, (where well-separatedness refers to non-near-neighbour boxes) until the lowest level, where it remains only to compute interactions between near neighbours. The FMM can also be considered as a tree code, but in the FMM multipole expansions are calculated only at the lowest level and are passed upwards to parent boxes *via* ingenious transformations and the localisation of the potential field using a Taylor's expansion.

The Fast Multipole Method was the first $O(N)$ method developed for evaluating Coulomb fields of particle distributions and has been applied to problems in astrophysics, plasma physics, molecular dynamics, fluid dynamics, partial differential equations and numerical complex analysis [116]. The method reduces the computational expense and improves the accuracy of simulations where crude approximation techniques were previously required to overcome prohibitive computational cost. Note also that both the tree codes and the FMM have strict error bounds which allows *a priori* fixing of error tolerance.

The maturity of the FMM, having been implemented in many different applications and correspondingly optimised [117], re-derived [80] and generalised [78,118] for those specific applications, would suggest that the scope for significant increases in efficiency and applicability is minimal, implying the algorithm is reaching a

peak in its level of performance. The maturity of the FMM should be taken into account when comparing and contrasting a new algorithm (such as KWIK) with the FMM.

In summary, the FMM splits the total interaction into short- and long-range contributions which are defined by the 'well-separatedness' of low level boxes. At the lowest level, particles within boxes which are near neighbours, are interacted directly, particles in well-separated boxes are interacted through a combination of multipole expansions and localised representations with well defined error bounds. Both the near and far field contributions can be afforded in linear work yielding an $O(N)$ algorithm for Coulomb interactions.

3.2.5. The Fast Wavelet Transform

The Fast Wavelet transform is a recent and exciting mathematical advancement that can be considered as a generalisation of the Fourier transform. Wavelets have received most attention in the fields of image analysis [119] and data reduction, where the algorithm's linearity and efficiency are far superior to those techniques previously employed, such as the $O(N \log(N))$ scaling FFT.

Its advantage is that the 'mother' wavelet can be shifted (translated) and scaled (dilated) so that, for example in the case of image analysis, parts of the image which are more detailed can be represented by a greater number of finer wavelets and those parts less detailed, by fewer and coarser wavelets. There is also a greater flexibility, in that Fourier based techniques which perform well for repetitive signals, require adjustment for analysing transient signals - wavelets on the other hand can handle both [120].

While wavelet theory is maturing and applications becoming more widespread and efficient [87-92,119-125], the theory remains general and coupled with the flexibility of the mother wavelet [121] it is not yet fully understood how the technique may be applied to the N -body problem.

3.3. Summary

Self consistent field molecular quantum mechanical calculations are rate limited by the evaluation of the total Coulomb energy, in particular, the electron-electron

repulsion energy. The interactions considered in the calculations are typical of the so-called N body problem type interactions. Thus, the Coulomb problem in SCF calculations need only be considered as a part of the more general problem. Such thinking has led to the recent development and implementation of the $O(N)$ CFMM. This has led to further rethinks in the way intermediate quantities are evaluated, especially the **J** matrix.

The implementation and development of the CFMM, and the success of DFT is enabling much larger molecules to be subject to accurate calculation than previously possible. However, the linear cost of the CFMM is still high and the extent of size applicability is not as great as initially anticipated. While an efficient implementation of the Fast Wavelet transform appears that it may lead to a still faster $O(N)$ method, such an implementation has yet to be reported.

Regardless of the relative speeds of these linear methods, and indeed those currently under development, $O(N)$ methods are dramatically reducing the total time requirement for calculations on large molecules. It is likely that while new $O(N)$ algorithms may or may not be faster than current ones, no single algorithm is likely to be ideal for all cases and each may have deficiencies and fail where others don't. Thus, we must look to find still faster, more general, and preferably simpler linear methods.

CHAPTER FOUR

PARTITIONING THE COULOMB OPERATOR

4.0. Introduction

The common theme amongst Coulomb field approximation techniques is that of partitioning of the Coulomb operator. Separating the singularity from the smooth long-range behaviour and then treating each piece in a specialised fashion is commonly practiced in many research areas where Coulomb force calculations are undertaken. For example, the Ewald summation technique partitions the potential to allow for exploitation of periodic boundary conditions. The FMM and its variants [78,80,116] separate the Coulomb operator using a grid technique. Near-field interactions are calculated directly, far-field using multipole expansions. In each of these examples the near-field partition is calculated directly and the far-field is calculated by exploiting the smoothness of the Coulomb operator's long-range behaviour and, in the case of Ewald, further taking advantage of periodic boundary conditions.

The KWIK algorithm for Coulomb energies is no exception in that it is an approximation technique which requires a separation of the Coulomb operator. However, the way separation is obtained is quite different to other approximation techniques and the resulting methodology is much simpler and does not require special boundary conditions. Furthermore, it appears that the size of system considered becomes an asset in obtaining higher accuracy, rather than a hindrance.

Surprisingly, KWIK's origin stems from finite-population theory in statistics and it is firstly considered how KWIK solved the statistics problem, why it was successful and then contemplate why it may be applicable to the Coulomb problem. Before application of KWIK to the Coulomb problem a few necessary generalisations will be considered, the most important being the partitioning of the Coulomb operator.

4.1. The KWIK Algorithm

Recently, Gill [126] developed a technique for solving a class of combinatorial problems extremely efficiently. The problem addressed was, if we are given a set X of known real numbers $\{x_1, x_2, \dots, x_n\}$, and we let A be the sum of a random sample of size $m \leq n$ numbers drawn without replacement from X , what is the probability $p = \Pr(A > 0) + \Pr(A = 0)/2$? This could be solved by exhaustive sampling. However, when $1 \ll m \ll n$ the number of possible samples ${}^n C_m$ is extremely large and prohibits the use of this method.

Gill's approach is to recast the problem by writing the probability as

$$p = \frac{1}{N} \sum_{i=1}^N H(A_i) \quad (4.1)$$

where N is the total number of possible samples and H is the Heaviside function (defined by $H(x) = 0, 1/2$ or 1 if $x < 0, x = 0$ or $x > 0$, respectively). By replacing H by its Fourier representation [53]

$$H(x) = \frac{1}{2} + \frac{2}{\pi} \sum_{k=1}^{\infty} \frac{\sin((2k-1)x)}{(2k-1)} \quad (4.2)$$

and inverting the summations, a convergent infinite series is obtained for p

$$p = \frac{1}{2} + \frac{2}{\pi N} \sum_{k=1}^{\infty} \frac{1}{(2k-1)} \sum_{i=1}^N \sin((2k-1)A_i) \quad (4.3)$$

valid for $(-\pi < A_j < \pi)$. Gill then defines two functions which enables the use of elementary trigonometric identities to recursively determine the inner sum. This is more compactly presented (and for reasons which will become apparent) by adopting the complex form. Thus, by defining

$$E_l(m, n) = \sum_{j=1}^n \exp(ilA_j) \quad (4.4)$$

which enables the use of the recursion formula

$$E_k(u, v) = E_k(u, v-1) + E_k(u-1, v-1) \exp(ikx_v) \quad (4.5)$$

the inner sum can be obtained in work that scales linearly with n , the size of the set X , for fixed m . *i.e.*

$$\sum_{i=1}^N \sin((2k-1)A_i) = \text{Im}(E_{2k-1}(m, n)) \quad (4.6)$$

The success of the method relies not upon the rapid convergence of the Fourier series, as the series converges slowly, but on the rapid decay of the inner sums (4.6) with k . Gill did not rationalise this behaviour and indicated that a systematic study was required to obtain a better understanding. However, it was observed that the new algorithm was most successful when the total number of possible samples was large and, whereas a larger total number of possibilities was unfavourable for the exhaustive approach, the larger number aided the convergence of the series in the KWIK approach.

The underlying success of the method can be attributed to the fact that although the total number of possible A_i 's grows as $O(n^m)$, they are highly dependent as they have been generated from only n independent values. The Coulomb interaction can be likened to this. One could consider the Coulomb interaction as an exhaustive sampling of all pairs of particles. Although there are $O(n^2)$ interactions, they arise from only n independent charge distributions. Thus, it was anticipated that the methodology applied successfully in the combinatorial problem could also be successfully applied to the Coulomb problem.

4.2. Adapting KWIK for Coulomb Interactions

In one of the key steps of Gill's combinatorial KWIK algorithm the Heaviside function is replaced with its Fourier series representation. This series approximation is slowly convergent, which has little effect on the success of the algorithm due to the rapid convergence of the inner sum, even for small populations. The total number of interactions appeared to be the key to the rapid convergence of the inner sum and for small populations a moderate sample size (m) still leads to a very large number of interactions. The series does, however, fall victim to the Gibbs phenomenon due to the discontinuities of the Heaviside function requiring scaling of the A_i 's to within the

boundaries of the periodic function in order to avoid serious error, but this appeared not to hinder convergence to any significant degree.

The first consideration when adapting the KWIK algorithm is that the total number of ‘possibilities’ in a Coulomb KWIK algorithm of distribution size n , is $O(n^2)$, which is significantly less than those encountered in Gill’s combinatorial problem. Hence, there is a greater likelihood of a more slowly convergent inner sum than those seen in the statistics problem. Furthermore, the Fourier Transform of the Coulomb operator (*c.f.* the Heaviside function in the statistics problem) decays only as k^{-2} . Thus it yields both a slowly convergent inner sum and a slowly convergent Fourier series and a larger number of Fourier terms would need to be considered before truncation in order to obtain the accuracy necessary for quantum chemical calculations. *i.e.* 10^{-6} , 10^{-10} .

To alleviate this problem, it was considered that introducing a ‘separator function’ to separate the singularity from the long-range behaviour may be essential.

4.3. Separating the operator

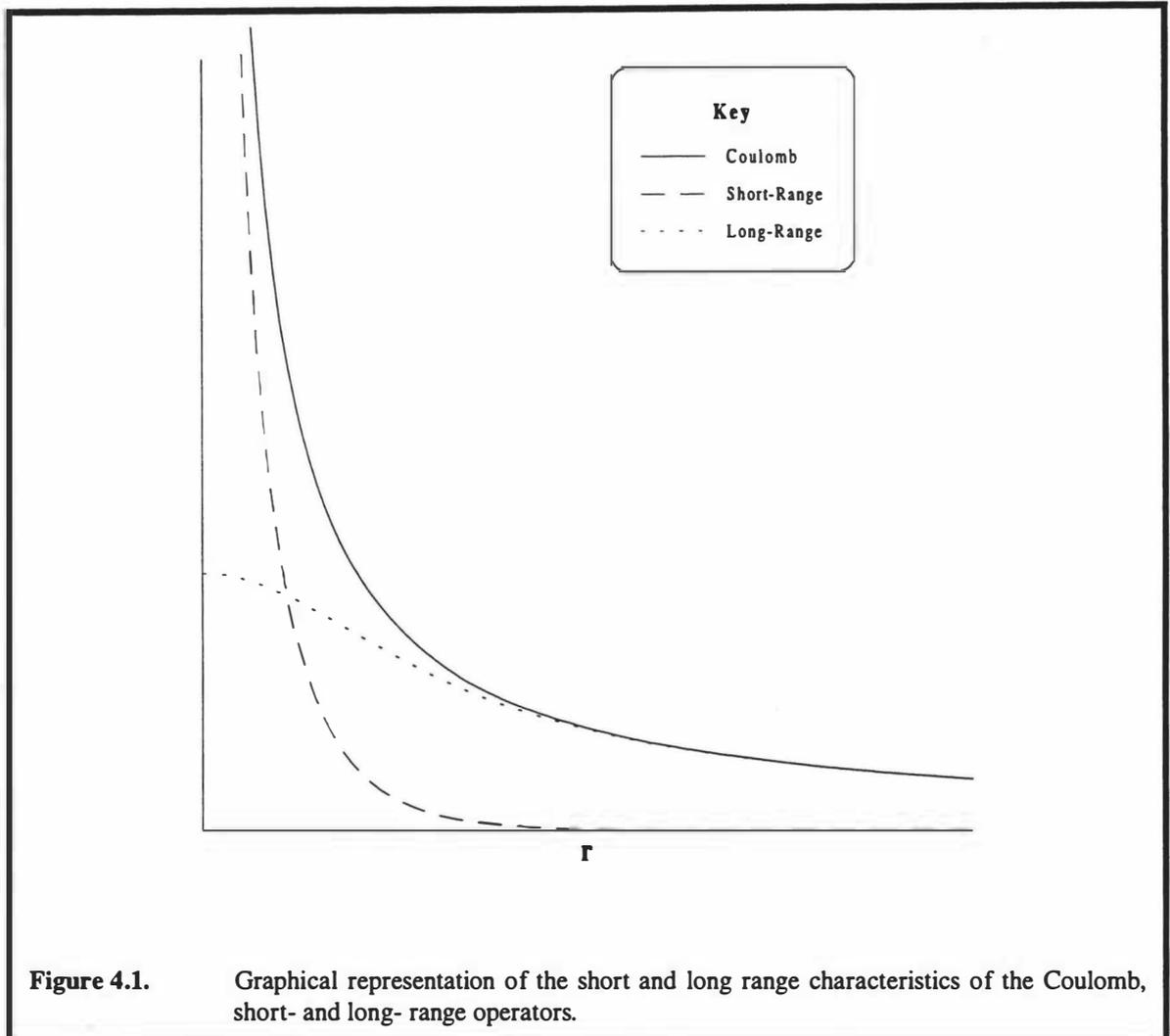
The Coulomb operator, r^{-1} , where r is the inter-particle/distribution separation, is difficult to approximate because it has a singularity at the origin and exhibits very slowly decaying long-range behaviour. By introducing a smooth and rapidly decaying function f , henceforth the ‘separator’, with $f(0) = 1$ the identity

$$\frac{1}{r} \equiv \frac{f(r)}{r} + \frac{1-f(r)}{r} \quad (4.7)$$

then separates the Coulomb operator into a singular, rapidly decaying piece and a non-singular, slowly decaying piece (Figure 4.1).

The rapidly decaying short-range operator will reduce the number of significant interactions in a system of charged particles to $O(n)$, based purely on a proximity argument. The FT of the long-range operator will be a rapidly decaying function whose interactions can be evaluated by a factorisation, leading to summations over Fourier coefficients and a single sum over particles/distributions - achievable in $O(n)$ work assuming that the number of Fourier coefficients required is $O(1)$. To re-emphasise the

importance of separation, recall that the long-range behaviour of the FT of a function is governed mainly by that function's short-range behaviour. The Coulomb operator has a singularity at the origin giving unfavourable long-range behaviour. On separation (4.7) however, the long-range partition has more favourable short-range behaviour leading to a rapidly decaying FT.



What then is the most effective separator? Given the arbitrary requirements of our function f above, there are no less than an infinite number of candidates! However, the aim is to afford a separation such that a long-range operator is obtained whose FT decays rapidly and so that the short range operator also decays rapidly.

The first approach in obtaining a separator was trial-and-error where 'common' functions exhibiting the required separator behaviour mentioned above were considered, and at each stage evaluating a list of necessary features. Initial investigations centred on

the functions $\exp(-\omega r)$, $\exp(-\omega r^2)$, $\tanh(\omega r)$ and $\operatorname{erfc}(\omega r)$. (A decay parameter, ω , was inserted as it seemed that it may be necessary to be able to tune the separator - larger distributions may require a different partitioning of long- and short-range operators compared to smaller or polarised distributions.) All these trial-and-error functions exhibit similar qualitative features. That is, when substituted into (4.7) they yield a rapidly decaying short-range operator, and a long-range operator whose FT also decays rapidly. However, quantitatively and practically they are somewhat different.

To illustrate the findings of the trial-and-error approach, consider the problem in parallel in both one- and three- dimensions. The respective Fourier transforms of an even/spherically symmetric function $f(r)$ is defined [127]:

$$f(r) = \int \hat{f}(k) \exp(ikr) dk \Leftrightarrow \hat{f}(k) = \frac{1}{\pi} \int_0^{\infty} f(r) \cos(kr) dr \quad (4.8.1)$$

$$f(r) = \int \hat{f}(k) \exp(i\mathbf{k} \cdot \mathbf{r}) d\mathbf{k} \Leftrightarrow \hat{f}(k) = \frac{1}{2\pi^2 k} \int_0^{\infty} r f(r) \sin(kr) dr \quad (4.8.3)$$

The exponential function has a short-range operator that decays exponentially whilst the Fourier transform of the long-range operator

$$\begin{aligned} \mathfrak{F}\left\{\frac{1 - \exp(-\omega r)}{r}\right\} &= \frac{1}{2\pi} \ln\left(1 + \frac{\omega^2}{k^2}\right) \\ &= \left(\frac{1}{2\pi}\right) \frac{\omega^2}{k^2} - \frac{\omega^4}{2k^4} + \frac{\omega^6}{3k^6} - \dots \end{aligned} \quad (4.9.1)$$

$$\mathfrak{F}\left\{\frac{1 - \exp(-\omega r)}{r}\right\} = \frac{\omega^2}{2\pi^2 k^2 (k^2 + \omega^2)} \quad (4.9.3)$$

decays only as k^{-2} , for large k . On the other hand the Gaussian function has a more rapid Gaussianly decaying short-range operator but the FT of the long-range operator leads to a combination of a hypergeometric and logarithmic functions in 1-D and Laguerre and Gaussian functions for 3-D, both of which have slow decay characteristics. The FT of the long-range operator obtained from a Gaussian separator is represented only as a power series. While in a final implementation of the algorithm in quantum

chemical code, the separator would, for the sake of maximising speed, most likely be evaluated using an interpolation scheme [128], implying that the form of the separator matters little other than being able to evaluate it once, accurately, for forming the interpolation tables, prior to commencing calculations. However, for development purposes, it was hoped a more easily handled function could be obtained. Thus, the Gaussian functions' improved short-range characteristics were noted, but noted too was that the separator required a few important practical features.

Separators involving $\tanh(r)$ and $\text{erf}(r)$ functions were next considered. Both led to short-range operators that decayed Gaussianly however, the long-range operators lead to FT's which decay exponentially and Gaussianly, respectively.

$$\mathfrak{S}\left\{\frac{\tanh(\omega r)}{r}\right\} = \frac{2}{\pi} \ln\left(\coth\left(\frac{\pi k}{4\omega}\right)\right) \quad (4.10.1)$$

$$\mathfrak{S}\left\{\frac{\tanh(\omega r)}{r}\right\} = \frac{1}{8\pi k} \text{cosech}\left(\frac{\pi k}{2\omega}\right) \quad (4.10.3)$$

$$\mathfrak{S}\left\{\frac{\text{erf}(\omega r)}{r}\right\} = -\frac{1}{\pi} \text{Ei}\left(-\frac{k^2}{4\omega^2}\right) \quad (4.11.1)$$

$$\mathfrak{S}\left\{\frac{\text{erf}(\omega r)}{r}\right\} = \frac{1}{(2\pi k)^2} \exp\left(-\frac{k^2}{4\omega^2}\right) \quad (4.11.3)$$

The latter considered erfc separator function exhibits a Gaussianly decaying short-range operator and a Gaussianly decaying FT of the long-range operator and it seemed that our trial-and-error approach to determine the 'ultimate' separator was converging.

4.4. The Ultimate Separator

It is well known that the Gaussian function, whose FT is another Gaussian, is the function which has the most rapidly decaying real and k space functions. While it is possible to devise a more rapidly decaying real space function, the consequence will be a more slowly decaying k space function and vice versa. The calculus of variations [129] provides a convenient means to prove this. Consider the sum of the second

moments [53] of the squares of our unknown function and its FT, again in one- and three- dimensions:

$$\int r^2 |f(r)|^2 dr + \int k^2 |\hat{f}(k)|^2 dk \quad (4.12.1)$$

$$\int |\mathbf{r}|^2 |f(\mathbf{r})|^2 d\mathbf{r} + \int |\mathbf{k}|^2 |\hat{f}(\mathbf{k})|^2 d\mathbf{k} \quad (4.12.3)$$

where the integration is over all space and we use the symmetric definition [130] of the FT rather than that described above. The aim is to minimise this sum to obtain a function with maximum real and k space decay. Using the definition of the FT of a derivative

$$\mathfrak{F}\left\{\frac{df(r)}{dr}\right\} = ik \hat{f}(k) \quad (4.13.1)$$

$$\mathfrak{F}\left\{\frac{df(\mathbf{r})}{dr_j}\right\} = ik_j \hat{f}(\mathbf{k}) \quad (4.13.3)$$

and Parseval's Theorem[127]

$$\int f(x) g^*(x) dx = \int \hat{f}(k) \hat{g}^*(k) dk \quad (4.14.1)$$

$$\int f(\mathbf{r}) g^*(\mathbf{r}) d\mathbf{r} = \int \hat{f}(\mathbf{k}) \hat{g}^*(\mathbf{k}) d\mathbf{k} \quad (4.14.3)$$

(where * denotes complex conjugate) lets us rewrite (4.12) as

$$\int r^2 (f(r))^2 + (f'(r))^2 dr \quad (4.15.1)$$

$$\int |\mathbf{r}|^2 |f(\mathbf{r})|^2 + |\nabla f(\mathbf{r})|^2 d\mathbf{r} \quad (4.15.3)$$

The extremals of this functional subject to the constraint

$$\int (f(r))^2 dr = 1 \quad (4.16.1)$$

$$\int |\mathbf{f}(\mathbf{r})|^2 d\mathbf{r} = 1 \quad (4.16.3)$$

are obtained on solving the Euler-Lagrange equation

$$r^2 f(r) - f'(r) = \lambda f(r) \quad (4.17.1)$$

$$r^2 f(r) - f'(r) - \frac{2}{r} f(r) = \lambda f(r) \quad (4.17.3)$$

A solution to this Differential Equation (DE) which results in a function that minimises (4.15) is a normalised Gaussian function:

$$f(r) = \frac{1}{\sqrt{\pi}} \exp\left(-\frac{r^2}{2}\right); \lambda = 1 \quad (4.18.1)$$

$$f(r) = \frac{1}{\sqrt{\pi^3}} \exp\left(-\frac{r^2}{2}\right); \lambda = 3 \quad (4.18.3)$$

Applying this same logic to our separator problem yields the sum to minimise:

$$\int r^2 \left(\frac{1}{r} - g(r)\right)^2 dr + \int k^2 |\hat{g}(k)|^2 dk \quad (4.19.1)$$

$$\int |\mathbf{r}|^2 \left|\frac{1}{|\mathbf{r}|} - g(\mathbf{r})\right|^2 d\mathbf{r} + \int |\mathbf{k}|^2 |\hat{g}(\mathbf{k})|^2 d\mathbf{k} \quad (4.19.3)$$

where for simplicity, we have modified the definition of the separator given in (4.7) so that

$$g(r) = \frac{1 - f(r)}{r} \quad (4.20)$$

Applying the definition of the FT of the derivative and applying Parseval's Theorem to this problem yields

$$J[g(r)] = \int r^2 \left(\frac{1}{r} - g(r)\right)^2 + (g'(r))^2 dr \quad (4.21.1)$$

$$J[g(\mathbf{r})] = \int |\mathbf{r}|^2 \left| \frac{1}{|\mathbf{r}|} - g(\mathbf{r}) \right|^2 + |\nabla g(\mathbf{r})|^2 d\mathbf{r} \quad (4.21.3)$$

The associated Euler-Lagrange equation leads to the DE

$$r^2 g(r) - g''(r) = r \quad (4.22.1)$$

$$r^2 g(r) - g''(r) - \frac{2}{r} g'(r) = r \quad (4.22.3)$$

which, while appearing simple, initially leads only to a series solution [130]

$$g(r) = C_0 + C_1 r - \frac{r^3}{3.2} + C_0 \frac{r^4}{4.3} + C_1 \frac{r^5}{5.4} - \frac{r^7}{7.6.3.2} + C_0 \frac{r^8}{8.7.4.3} \dots \quad (4.23.1)$$

$$g(r) = C_0 - \frac{r^3}{4.3} + C_0 \frac{r^4}{5.4} - \frac{r^7}{8.7.4.3} + C_0 \frac{r^8}{9.8.5.4} \dots \quad (4.23.3)$$

During these investigations, it was suspected and subsequently concluded that, if not solely for development purposes, the ultimate separator must not only lead to a short-range operator and FT of a long-range operator that decayed at least Gaussianly, but must also satisfy practical criteria. Firstly, to aid the rapid development of the algorithm the separator must lead to operators and FT's that were easily calculable. This first criterion is in line with the general goal of developing not only a fast linear scaling algorithm, but also a conceptually simple one.

Secondly, the ERI's that are to be encountered in a quantum chemical calculation involving the short-range operator must be calculable analytically - preferably so that present integral technology need only be perturbed, rather than rewritten. While these criteria may appear to be convenient and more indicative of our failure to determine the best separator (4.22), the convenience of the second criterion cannot be understated. Boys [63] introduced the Gaussian basis function in the early 1950's to replace the generally more accepted and natural Slater functions, purely on the basis that the ERI's could be evaluated analytically. It would be a significant backward step in MQM if alternative numerical integration techniques had to be reverted to, especially given the highly optimised nature of present ERI technology.

Recently, an analytic solution has been found to (4.22) [131] resulting in an optimal separator (the ultimate separator) which can be expressed in terms of modified Bessel, Hermite or parabolic cylindrical functions [53]. In terms of the latter, this is written as

$$1 - f(r) = U(0, r\sqrt{2}) / U(0, 0) \quad (4.24)$$

which has an asymptotic decay which is slightly faster than a Gaussian. The resulting long-range partition has a somewhat flatter origin than that obtained with the error function, the overall improvement gained by the analytic solution can be gauged by evaluating equation (4.21.3) (Table 4.1). It is apparent that the difference between the analytic solution and the erfc separator is limited. However, an important benchmark will be the comparison between the implemented separator and the analytic solution.

Separator, $f(r)$	Optimal ω	$J[g(r)]$
ref [131]	-	8.4946
$\text{erf}(\omega r)$	0.6366	8.5102
$\tanh(\omega r)$	0.7723	8.6272
$1 - \exp(-\omega r)$	1.1067	9.2713
$1 - \exp(-\omega r^2)$	0.9400	9.7711

Table 4.1. Value of the functional $J[g(r)]$ (4.21.3) for various separators with their respective optimal decay parameters, ω . The difference in the value of the functional between that of the optimal separator and error function is minimal.

4.5. The Error Function Separator

The error function [53], $\text{erf}(x)$ and the complementary error function $\text{erfc}(x) = 1 - \text{erf}(x)$, are two of many functions which are defined by an integral

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-t^2) dt \quad (4.25)$$

The error function can be expanded in both a power series

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{n!(2n+1)} \quad (4.26)$$

and asymptotically,

$$\sqrt{\pi}x\exp(x^2)\operatorname{erfc}(x) \sim 1 + \sum_{m=1}^{\infty} (-1)^m \frac{(2m-1)!!}{(2x^2)^m} \quad (4.27)$$

and is easily differentiated

$$\frac{d^{n+1}}{dx^{n+1}} \operatorname{erf}(x) = (-1)^n \frac{2}{\sqrt{\pi}} H_n(x) \exp(-x^2) \quad (4.28)$$

where $H_n(x)$ is the n^{th} Hermite polynomial. Most importantly, substitution of $\operatorname{erf}(r)$ into (4.7) with the addition of the decay parameter ω , leads to both a long-range operator

$$\frac{\operatorname{erf}(\omega r)}{r} \quad (4.29)$$

whose FT can be found analytically in one (4.30.1), two (4.30.2) and three (4.30.3) dimensions,

$$\mathfrak{S}\left\{\frac{\operatorname{erf}(\omega r)}{r}\right\} = -\frac{1}{\pi} \operatorname{Ei}\left(-\frac{k^2}{4\omega^2}\right) \quad (4.30.1)$$

$$\mathfrak{S}\left\{\frac{\operatorname{erf}(\omega r)}{r}\right\} = \frac{1}{2\pi k} \operatorname{erfc}\left(\frac{k}{2\omega}\right) \quad (4.30.2)$$

$$\mathfrak{S}\left\{\frac{\operatorname{erf}(\omega r)}{r}\right\} = \frac{1}{2\pi^2 k^2} \exp\left(-\frac{k^2}{4\omega^2}\right) \quad (4.30.3)$$

and a short-range operator

$$\frac{\operatorname{erfc}(\omega r)}{r} \quad (4.31)$$

that will enable ease of implementation into existing MQM programmes. On a practical note the error function has widespread availability in mathematical libraries on many computer platforms. Furthermore, on substitution of the trial-and-error separators into equation (4.21.3) it is clear that the error function gives the lowest value for this sum, and the observed ‘convergence’ of the considered separators discussed in the previous section (4.3) is also confirmed (Table 4.1).

Based on these considerations, the immediate unavailability of an analytic solution to the DE (4.22) and the short-range decay similarities of the series solution (4.23) the error function, $\text{erf}(\alpha r)$, was chosen as the separator for development of KWIK for Coulomb interactions.

4.6. Summary

Partitioning or separating the short- and long- range characteristics of the Coulomb operator, that is, the singularity and the slow, long-range decay, and subsequently treating each entity in a specialised fashion is commonplace in Coulomb field approximation techniques. The KWIK methodology for Coulomb energies is no exception in that the short-range partition is calculated directly and the long-range partition is calculated using an approximation technique.

It has been postulated that the optimal function to obtain the best separation can be determined using the calculus of variations to yield a functional that minimises the sums of the second moments of the short-range operator and the FT of the long-range operator, thus affording a least-squares minimisation of the extents of the short-range operator and FT of the long-range operator.

On determining the so-called ultimate separator it was found that it would very likely lead to a short-range operator and a FT of the long-range operator that decay Gaussianly. Considered were a number of important features of the separator necessary to aid ease of development, implementation and generalisation. Those of the decay parameter - to enable fine tuning of our separation, a separator that leads to an analytic function for the FT of the long-range operator - for ease of algorithm development - and a separator that leads to a short-range operator such that the ERI's in MQM calculations can be evaluated analytically. The error function satisfies all these criteria.

The recent solution of the ultimate separator problem indicates that the choice of the error function as a separator is close to optimal. It is likely that practical considerations and the ease of adjusting ERI evaluation for MQM implementation will far outweigh the minor theoretical improvement in the efficiency of KWIK likely using

the ultimate separator, although the ultimate separator would provide a suitable benchmark to gauge these practicalities.

CHAPTER FIVE

SHORT-RANGE INTERACTIONS

5.0. Introduction

The aim of maximising the overall efficiency of the KWIK algorithm requires careful consideration of the individual components of the algorithm as the total efficiency will be no less than the sum of the efficiencies of the components. The first of the two major components of the KWIK algorithm is that of the short-range partition. While the short-range potential in the KWIK algorithm may (or may not) be unique, the problem encountered in trying to evaluate these specific short-range interactions is part of a more general problem which has received considerable attention.

Short-range potentials have widespread application in many areas of physics and chemistry, such as astronomy, molecular dynamics, plasma modeling and quantum chemistry [103,111,132]. Short-range interactions are typified by a rapidly decaying function of inter-particle separation such that beyond some radial distance R_C , the interaction has a negligible contribution to the total energy/force. This means that cut-offs can be introduced to reduce the computational effort to afford the energy/forces within some predetermined accuracy. The most significant aspect of a short-range potential is that the number of significant (relative to some predetermined accuracy, ϵ) interactions, M , per particle, is $O(1)$ where $M \ll N$, and N is the size of the system (number of particles/atoms/distributions). Thus, the total number of significant short-range interactions scales linearly with N .

We emphasise here the proximity basis of the short-range interaction, that beyond the critical cut-off parameter R_C , the interactions are insignificant and contribute negligibly to the total energy. Contrasting with the proximity based short-ranged interactions are those methods that use a discontinuous separation of a long-range potential, such as the FMM and the more general CFMM. In methods such as these, the total interaction potential is separated into a near-field partition and a far-field partition. The near-field is evaluated directly and the cut-off is defined by the extremities of the

near-field cell geometry - in general a cube. That is to say, the short-range interaction is defined to the extremities of the nearest neighbour cells (or boxes).

In this Chapter we outline and discuss the methods currently employed to rapidly evaluate the proximity based short-range potentials, identify weaknesses in these methodologies and hypothesise and investigate how one may improve on them.

5.1. Computational Methods

5.1.0. Introduction

As is the case for many problems in computational science, there are a number of methods available to afford a solution - each having realms of application where a given method delivers the answer faster and with greater precision than the alternatives. The objective of the KWIK algorithm is to enable application to very large systems - thus efficiency, memory requirements, cost scaling behaviour and speed are all of paramount importance.

Primarily, as the short-range potential requires only $O(N)$ work to be carried out, the foremost requirement of a short-range algorithm is that it scale linearly. With the rate at which computer technology is developing, yesterday's simulation size limit becomes routine today, and even methods with a very small $O(N^2)$ coefficient will eventually reach the cross-over point, and this may arrive sooner than expected given this rapidly advancing technology.

Further important considerations come from the theme of Chapter Two, which are the platforms on which we are to run our short-range code. As program developers tend to develop mainly on the more affordable scalar machines, the awareness of a new method's parallelisability and vectorisability are most important.

The two methods usually discussed when considering efficient methods of evaluating short-range interaction are the Verlet neighbour list method and the link-cell method. In this section we firstly consider the zeroth order code optimisation, that of efficiently computing the potential, and then briefly outline the two methods for determining significant particle pairs.

5.1.1. Computing the Interaction

Often the form of the potential, that is the function that defines the level of interaction for a given inter-particle separation, is computationally expensive to calculate. Even simply determining the inter-particle distance is computationally expensive (relative to multiplies or additions) as a square root is required

$$r_{AB} = \sqrt{(A_x - B_x)^2 + (A_y - B_y)^2 + (A_z - B_z)^2} \quad (5.1)$$

The computational expense of evaluating the potential can be significantly reduced *via* interpolation tables (for example see [59,128]) and polynomial expansions [103]. However, the performance of intrinsic functions varies from machine to machine and while significant speed may be obtained on one platform, no improvement may be observed on an other. Computer vendors are well aware of the limitations imposed by slow intrinsic functions and significant effort [133] has, and will, continue to be made by manufacturers to improve such limitations. This means that most of the more common functions are relatively highly optimised.

5.1.2. Verlet Neighbour Lists

The Verlet neighbour list method [134] keeps a list of the identities of near-neighbours, that is those within the cut-off radius R_C , for each individual particle/molecule. The neighbours are stored in a large array, say called LIST of length MN , and a second array POINT of length N , indexes the neighbours. That is, POINT(I), points to the first neighbour of particle/molecule I in LIST and POINT(I+1) - 1 points to the last neighbour of molecule I, thus we can easily identify the part of the large array LIST containing the neighbours of I.

The Verlet neighbour list method is particularly useful in areas where the computer modelling will result in the displacement of molecules/particles and a recalculation of the energy/forces is then required. The neighbour list can then simply be updated for the next cycle. However, the method does have many disadvantages. Firstly, recent descriptions and implementations [103] of the Verlet method use nested loops over all particles to determine the neighbour list for each particle/molecule (Scheme 5.1). This implies a cost that scales as $O(N^2)$. Secondly, the size of the array

LIST may be significantly large, and coupled with the $O(N^2)$ scaling characteristic prohibits the use of the method for large systems.

```

INTEGER POINT(N) , LIST(MAXLIST)
REAL*8 X(4,N)

RCUTSQ = RCUT**2
NLIST = 0

DO i=1,N-1
  POINT(i) = NLIST + 1
  DO j=i+1,N
    RXIJ = X(1,i) - X(1,j)
    RYIJ = X(2,i) - X(2,j)
    RZIJ = X(3,i) - X(3,j)
    RIJ = RXIJ**2 + RYIJ**2 + RZIJ**2
    IF (RIJ.LT.RCUTSQ) THEN
      NLIST = NLIST + 1
      LIST(NLIST) = J
    ENDIF
  ENDDO
ENDDO

POINT(N) = NLIST + 1

END

```

Scheme 5.1. Sorting particles into Verlet neighbour lists requires work that scales with the square of the number of particles which can be seen by the nested loop over particles above. Note also that the memory requirements for storing lists of neighbours for each particle is significant.

5.1.3. Linked-Cell Method

The linked-cell method [111] contrasts with the Verlet neighbour list in two respects. Firstly, its cost scales linearly with the size of the system, and secondly, while simple updates can be applied in applications where particles/molecules are displaced, the ease and speed of the sorting process means that the complete sort can just as easily be reapplied.

The method begins by dividing the simulation system into a set of cells (cubes/squares/lines depending on the dimension of the system) of length R_C and then determining which cell each particle resides in. For cache considerations, it is best if the coordinate array is then re-ordered so that particles within the same cell are listed consecutively. This division and reordering is easily afforded in linear work (Scheme 5.2). Note also that the total memory requirements are significantly less than that of the Verlet neighbour list approach and the method is similarly simple.

```

INTEGER Nx, Ny, Nz, INBOX(N)
INTEGER ICOUNT(Nx*Ny*Nz), IPOINT(Nx*Ny*Nz)
REAL X(4,N), XORD(N,4), Rc

DO i=1,N
  Kx = INT(X(1,i)/Rc) + 1
  Ky = INT(X(2,i)/Rc) + 1
  Kz = INT(X(3,i)/Rc) + 1
  NBOX = Kx + (Ky - 1) * Nx + (Kz - 1) * Nx * Ny
  INBOX(i) = NBOX
  ICOUNT(NBOX) = ICOUNT(NBOX) + 1
ENDDO

IPOINT(1) = 1
DO j=2, Nx*Ny*Nz
  IPOINT(j) = IPOINT(j-1) + ICOUNT(j-1)
ENDDO

DO j=1,N
  NBOX = INBOX(j)
  IPT = IPOINT(NBOX)
  XORD(IPT,1) = X(1,j)
  XORD(IPT,2) = X(2,j)
  XORD(IPT,3) = X(3,j)
  XORD(IPT,4) = X(4,j)
  IPOINT(NBOX) = IPOINT(NBOX) + 1
ENDDO

DO i=1, ICOUNT(1)
  INBOX(i) = 1
ENDDO
ICT = ICOUNT(1)
IPOINT(1) = 1
DO 20 i=2, Nx*Ny*Nz
  IPOINT(i) = IPOINT(i-1) + ICOUNT(i-1)
  DO 10 j=ICT+1, ICT + ICOUNT(i)
    INBOX(j) = i
  10 CONTINUE
  ICT = ICT + ICOUNT(i)
20 CONTINUE

END

```

Scheme 5.2. Sorting particles/molecules into cells in linear work using integers. ICOUNT(I) indicates the number of particles in cell I, and IPOINT(I) points to the first particle in cell I in a cell-ordered coordinate array XORD. The first three elements of X are particle coordinates, and the final entry is the charge.

This linked-cell method is the approach adopted by the discontinuous separation techniques, *i.e.* FMM, CFMM. Near-field interactions are determined by interacting all particles within the target cell with both themselves and all the particles in near-neighbour cells. While one is unlikely to see the FMM mentioned in the context of efficient short-range algorithms, in the context of this thesis it is important to re-emphasise the distinctions.

5.2. Improvements to the Linked-Cell Method

5.2.0. Introduction

Short-range potentials such as the Lennard-Jones potential, and more recently those which are found in the KWIK and CASE approximations, are being applied to larger and larger systems and it is becoming increasingly evident that present methodologies (notably the most widely adopted linked-cell method), which are used to screen insignificant short-range interactions, are very inefficient. Inefficient to the extent that most of the computational effort goes into evaluating interactions which need not be considered.

When applying the linked cell method to cases such as particle-particle-particle-mesh, KWIK and CASE, we begin by dividing the simulation box into cubes or squares, (depending on the dimension of the problem) where the width of the cells is equal to the distance where the short-range interaction becomes insignificant, R_C . Target cells then need only interact with nearest neighbours to ensure that all significant interactions are calculated. Thus, the average number of interactions per particle is $9M$ for planar and $27M$ for non-planar systems as the number of near neighbour cells is 9 and 27 for planar and non-planar systems respectively, and M being the average number of particles per cell. While this method ensures that each particle interacts with all significant particles, each particle also interacts with many particles it need not. (Figure 5.1).

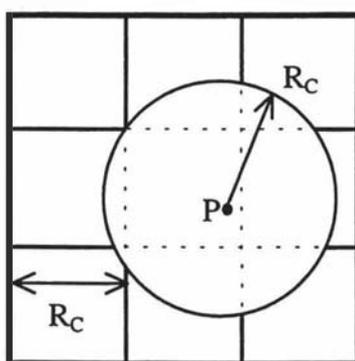


Figure 5.1. The particle denoted P need only interact with particles within the cut-off radius R_C . Using the linked-cell method, particle P interacts with all the particles within the region depicted by the bold square, which is significantly more than necessary.

While it appears that the linked cell method is screening the majority of insignificant interactions, (and by comparison to considering all possible $O(N^2)$ interactions, it is) if we consider the efficiency of a method to be:

$$\epsilon = \frac{\text{No. of significant interactions}}{\text{No. of interactions computed}} \quad (5.2)$$

it can be easily shown (Table 5.1) that the majority of the computational effort is still spent calculating insignificant interactions. Furthermore, while significant effort has been made to optimise the implementation of the linked-cell method for vector [59] and parallel [135] processing, little mention has been made of the inherent inefficiency in the primary algorithm, such as that illustrated (Figure 5.1) and tabulated (Table 5.1).

Dimension	One	Two	Three
Efficiency	0.6667	0.3491	0.1551

Table 5.1. Efficiencies of the linked cell method in all three dimensions. Note in 2-D and 3-D the majority of the computational effort is put into calculating insignificant interactions.

The efficiencies detailed in Table 5.1 are obtained by dividing the minimum ‘amount’ of interaction required by the ‘amount’ (distance, area or volume) of interaction a test particle (*e.g.* P in Figure 5.1) is subjected to using the given scheme. Consider Figure 5.1. The test particle P will be interacted with 9 near neighbour squares using the linked cell method. The total ‘amount’ of interaction will be 9 times the area of a single square, which is $9R_C^2$. However, the test particle needs only interact with those within a radius R_C which is an area of πR_C^2 . This is also the case for the 1-D and 3-D systems, where distance along the line and volume of interactions are the ‘amounts’ considered.

5.2.1. Ordering an Ordinate

The first way a significant improvement can be made to the linked cell method is if particles within a cell are ordered in, for example, increasing x coordinate [111]. Once particle-particle inter-cell Δx becomes greater than R_C , the calculation can be terminated for that target cell particle. Alternatively, a binary search may be performed to predetermine the extremity particle in the neighbour box. This removes the

requirement to compare Δx with R_C for each interaction and will aid in maintaining vectorisability.

The extra work required to order the particles within cells is minimal, given that we already rearrange the coordinates of the particles/molecules into that of cells. This work is illustrated in Scheme 5.3 and it should be made clear that this sorting does not alter the linear scaling property of the algorithm. All cells are looped over ($O(N)$ work) and then all particles within a cell are sorted and ordered. While it is well known that sorting routines scale as much as $O(N^2)$ [136], when the number of particles within each cell is M , which is $O(1)$ we can, for collinear particles, obtain a completed ordered array in $O(N)$ work and, for higher dimensions, obtain an array with an ordered ordinate in the same amount of work.

The effect of ordering an ordinate on the basic algorithm for calculating short-range particle-particle interactions remains reasonably simple, and this is depicted in a compact form in Scheme 5.4. To illustrate the effect of ordering an ordinate on theoretical efficiencies, Figure 5.1 has been reconstructed to indicate the new computation regions (Figure 5.2).

If we incorrectly (see later) consider binary searches as not computing interactions then the theoretical efficiencies can be easily calculated following the methodology used for Table 5.1 by subtracting off the total interaction ‘amount’, that ‘amount’ ignored by the binary search (Table 5.2).

Dimension	One	Two	Three
Efficiency	1.0000	0.5236	0.2327

Table 5.2. Efficiencies of the linked cell method with x -direction ordinate ordering for all three dimensions. Note in 3-D the majority of the computational effort is still put into calculating insignificant interactions and that 2-D efficiency is still unacceptable.

Considering binary searches as not computing interactions is incorrect because a binary search does require us to consider a small proportion of the outlying particles. The significance of the binary search and its effect on efficiencies is small, when the total number of interactions avoided by using this approach is large.

```

INTEGER Nx, Ny, Nz, INBOX(N)
INTEGER ICOUNT(Nx*Ny*Nz), IPOINT(Nx*Ny*Nz)
REAL X(4, N), XORD(N, 4), Rc

DO i=1, N
  Kx = INT(X(1, i)/Rc) + 1
  Ky = INT(X(2, i)/Rc) + 1
  Kz = INT(X(3, i)/Rc) + 1
  NBOX = Kx + (Ky - 1) * Nx + (Kz - 1) * Nx * Ny
  INBOX(i) = NBOX
  ICOUNT(NBOX) = ICOUNT(NBOX) + 1
ENDDO

IPOINT(1) = 1
DO j=2, Nx*Ny*Nz
  IPOINT(j) = IPOINT(j-1) + ICOUNT(j-1)
ENDDO

DO j=1, N
  NBOX          = INBOX(j)
  IPT           = IPOINT(NBOX)
  XORD(IPT, 1)  = X(1, j)
  XORD(IPT, 2)  = X(2, j)
  XORD(IPT, 3)  = X(3, j)
  XORD(IPT, 4)  = X(4, j)
  IPOINT(NBOX)  = IPOINT(NBOX) + 1
ENDDO

DO i=1, ICOUNT(1)
  INBOX(i) = 1
ENDDO
ICT = ICOUNT(1)
IPOINT(1) = 1
DO 20 i=2, Nx*Ny*Nz
  IPOINT(i) = IPOINT(i-1) + ICOUNT(i-1)
  DO 10 j=ICT+1, ICT + ICOUNT(i)
    INBOX(j) = i
  10 CONTINUE
  ICT = ICT + ICOUNT(i)
  20 CONTINUE

DO i=1, Nx*Ny*Nz
  IF(ICOUNT(i).GT.1) THEN
    CALL XSORT(XORD(IPOINT(i), 2), XORD(IPOINT(i), 1),
$           XORD(IPOINT(i), 3), XORD(IPOINT(i), 4), ICOUNT(i), 4)
    ENDF
ENDDO

END

```

Scheme 5.3. The sorting of an ordinate to improve efficiency is achieved similarly to that without the sort and simply requires an extra loop over all Cells and a call to the sorting routine XSORT. This routine sorts the *x* ordinate into increasing value maintaining the integrity of the other coordinates and associated charge.

```

{loop over all boxes}
  {loop over particles within box}
    {complete intra-box interactions}
    {loop over neighbouring boxes}
      {determine farthest significant particle in
      neighbour box via binary search}
      {loop over significant neighbour box particles}
        {calculate inter-particle interactions}
      {end loop}
    {end loop}
  {end loop}
{end loop}

```

Scheme 5.4. Algorithm structure for short-range interactions with an ordered ordinate.

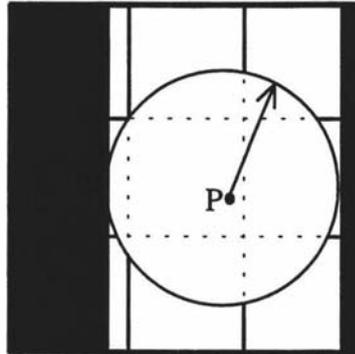


Figure 5.2. As for Figure 5.1, the particle denoted P need only interact with particles within the cut-off radius R_C . Using the ordinate ordered linked-cell method, particle P interacts only with those particles within the unshaded region. This is somewhat less than the previous requirement.

5.2.2. Changing the Cell Geometry

While the extra work in ordering an ordinate and performing binary searches is minimal, the 2-D and 3-D efficiencies remain unacceptable, and so further improvements to the linked-cell method have been investigated by reconsidering the basic geometry of our parent cell. This has been done in two ways. Firstly, by an integer reduction in the cell size and secondly by reconsidering the basic geometry of the cell. Both ideas are attempting to produce an interaction region, that is the target cell and its neighbours, that are better approximations to a circle or sphere of radius R_C . As outlined above, a square (oversized) is a poor approximation of a circle.

While it would appear that a reduction in cell size, with subsequent reduction in cell occupancy will result in reduced vector lengths for algorithms of the type depicted in Scheme 5.4, therefore decreased performance on vector machines, by carefully reconstructing the algorithm [59] the increased number of cells can be used as an advantage to maintain (increase) vectorisability. While emphasising the importance of maintaining vectorisability for implementation on large machines, it should also be noted that scalar architectures also perform poorly with very short vector lengths.

5.3. Approximating Spheres

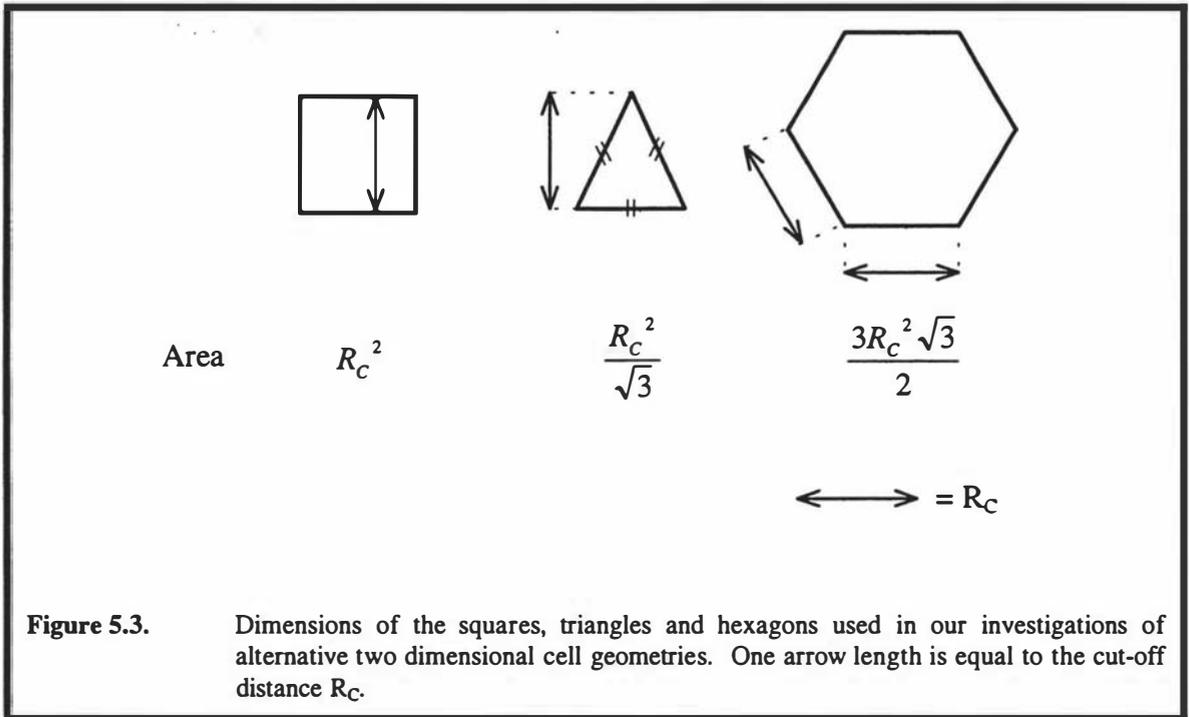
5.3.0. Introduction

The underlying reason for the inefficiency of the linked-cell method is that a square/cube is a poor approximation of a circle/sphere, especially when the square/cube is an oversized one, as is the case here (Figure 5.1). It is hypothesised that to improve the efficiency of linked-cell method and the key to greater efficiency, is the basic structure of our cell. While it has been suggested that alternate geometries may be used [103], implementations have not been illustrated and it was not suggested that this may lead to an improved algorithm.

5.3.1. New Basic Cell Structure

In 1-D there is no choice as to the nature of the basic cell structure, but as is outlined in the previous section, assuming the cost of binary searches and ordinate

ordering is minimal and linear, maximum efficiency can be afforded. For higher dimensions the question is: “What is the optimal polyhedron with which to divide space to obtain the most efficient evaluation of short range interactions?”



A basic requirement of a new basic cell structure for two and three dimensional systems is that it tessellate all n -space. Socrates had assumed regular tetrahedra tessellated three-space but it is now well known that this is fallacious [137]. Two dimensions is much easier to visualise and investigate than three dimensions. Squares, triangles and hexagons (Figure 5.3) have been considered in some detail for two dimensions, and the use of multiple cell structures has been dismissed for simplicity. *e.g.* a combination of regular tetrahedra and octahedra tessellate three space at a ratio of 2:1. The efficiencies (5.2) of using the alternative 2-D geometries are presented in Table 5.3. These efficiencies were again obtained by considering the area (‘amount’) of interaction a test particle would be subjected to using the linked-cell method for each of the alternative geometries. This is simply the number of significant near-neighbour cells (relative to the cell in which the test particle resides) multiplied by the area of a single cell. This area is then compared to the minimum area of interaction (a circle) to obtain the efficiency.

Geometry	Square	Hexagon	Triangle
Efficiency	0.3491	0.1727	0.4186
Rel. No. Cells	1	$2/3\sqrt{3}$	$\sqrt{3}$

Table 5.3. Efficiencies of alternative cell geometries in two dimensions.

As a result of the findings in two dimensions in the next section, three dimensional alternatives are not considered in any detail. Suffice it to say, that alternative structures have previously been suggested for three space [103], for example the rhombic dodecahedron and truncated octahedron, but the cube has always maintained favour due to its geometric simplicity.

It has recently been suggested [118] that the FMM, as applied to periodic assemblies, may be improved in the 2-D case by adjusting the geometry of the base cell to that of a hexagon. The reasoning given is that the hexagon is a better approximation to a circle than a square. This may or may not lead to improved efficiency of the FMM in 2-D, but again, it must be emphasised that the FMM uses a discontinuous separation and that KWIK uses a proximity based short-range interaction.

5.3.2. Cell Size Reduction

In 1-D systems where the assumption of cheap ordinate ordering and binary searches does not hold, the total time to evaluate the short-range interactions will become more expensive than the case where they are not ordered and such situations should be avoided. Furthermore, the cell geometries introduced above, increase efficiencies, but not to a satisfactory level. Thus, we need to consider an alternative approach. (All discussions herein apply equally to all three dimensions considered).

The basic idea for the new approach is to reduce the intrinsic cell dimension R_C by a factor of k , where k is an integer. A target cell will then interact with k shells of cells, rather than simply nearest neighbours. In terms of approximating spheres, we are simply using a smaller sub-cell which in the limit of infinitely small sub-cell approximates the sphere exactly. This is illustrated in Table 5.4 for the case of 1-D systems.

k	Near Neighbours	Efficiency	Rel*
1	3	0.6667	1
2	5	0.8000	2
3	7	0.8571	3
k	2k+1	2k/(2k+1)	k
∞	∞	1.0000	∞

Table 5.4. The efficiency of cell size reduction. * denotes the total number of cells required relative to the k=1 case.

The efficiencies in Table 5.4 are evaluated by considering the total distance ('amount') a test particle will be subjected to using the k-divided linked-cell approach. This will be $(2k+1)*R_C$. The efficiency is afforded by dividing the distance of significant interaction which is $2k*R_C$, (a length R_C either side of the test particle) by this number.

Determining efficiencies to construct a table such as Table 5.4 for arbitrary cell geometry and dimension, is slightly more difficult as complete cells become insignificant and are not necessary to consider for large k and $d>1$, where d is the system dimension. Note also, that for higher dimensions and cell size reduction, the overall effect of ordering an ordinate is also reduced (Table 5.5).

d	Two			Three		
	Efficiency	Efficiency*	% Imp.	Efficiency	Efficiency*	% Imp.
1	0.3491	0.5236	50	0.1551	0.2327	50
2	0.5027	0.5585	11	0.2681	0.2979	11
3	0.5770	0.6059	5.0	0.3637 [‡]	0.3787 [‡]	4.1
4 [‡]	0.6528	0.6680	2.3	0.4373	0.4455	1.9

Table 5.5. Effectiveness of ordinate ordering is reduced on increased dimension and cell size reduction. * ordinate ordering has been applied, [‡] some outlying cells become completely insignificant and are neglected as the simulation cell becomes more representative of a sphere/circle.

There are two important observations to be made at this point. The first is that on reducing the cell size, using the integer reduction outlined above, the total number of cells increases resulting in reduced occupancy number and therefore reduced inner loop lengths (Scheme 5.4). The second observation is that changing the cell geometry also alters occupancy (see Table 5.3 for relative total numbers of cells). Both approaches can improve efficiencies and simultaneously reduce occupancy. Notwithstanding the general problem of reduced occupancy (addressed in section 5.4), an important question

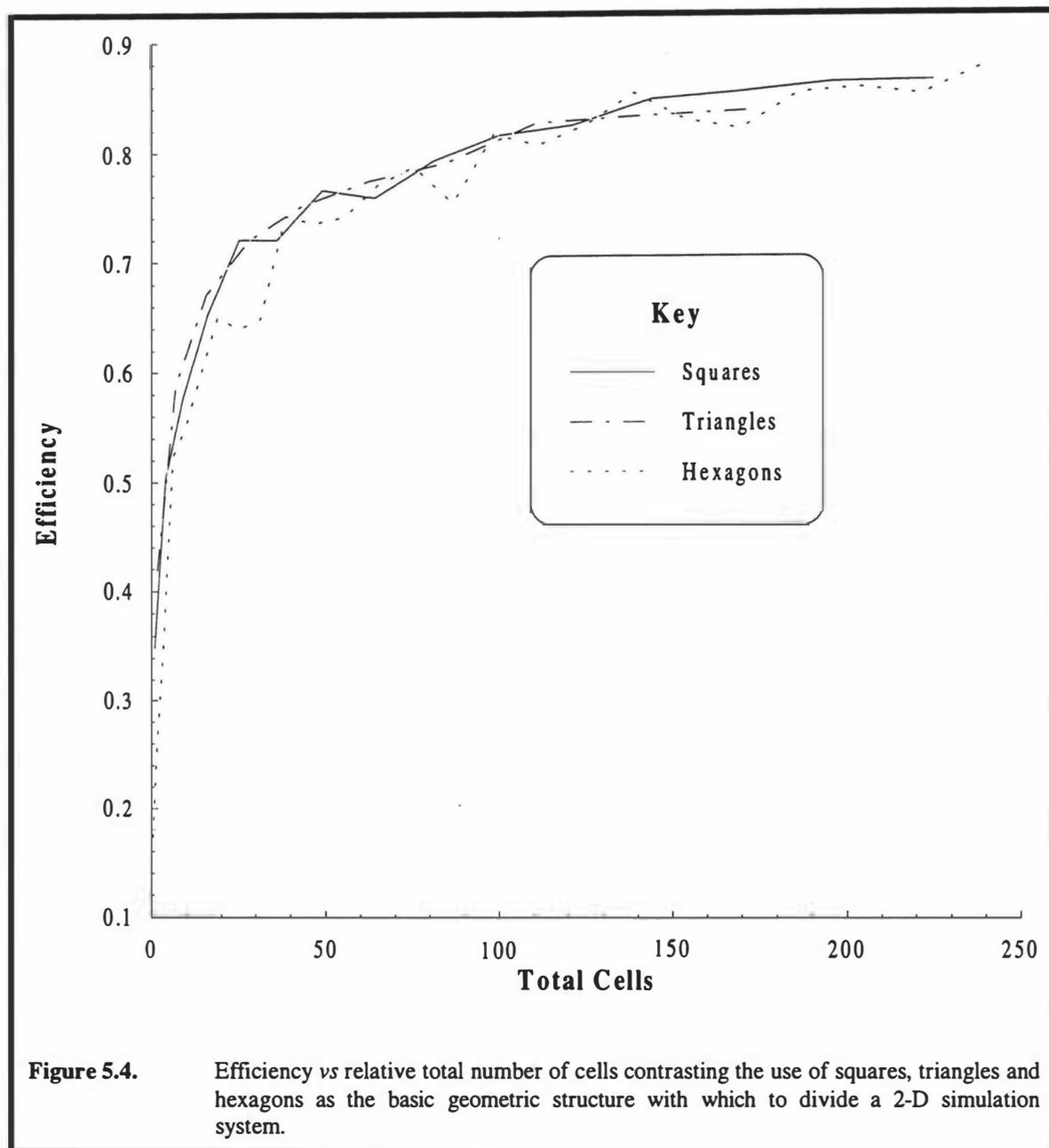
is, what combination of geometry and cell reduction affords maximum efficiency with minimal occupancy reduction? To answer this we need to consider the relationship between theoretical efficiency and the relative total number of cells. The greater the efficiency for fewer total cells means greater occupancy and a reduced loop length over all cells.

The total numbers of significant near-neighbours, efficiencies and total numbers of cells relative to the $k=1$ square case for 2-D systems using squares, triangles and hexagons as the basic cell geometry are tabulated in Appendix A Tables A.1, A.2 and A.3. These data are plotted in Figures 5.4 and 5.5 to illustrate the relative effectiveness of each parent cell geometry.

Figure 5.4 illustrates that in the limit of large subdivisions, the efficiency is not dependent on the shape of the parent cell. It also indicates that above about 0.75 the rate of gains in efficiency diminish with increasing cell division.

In determining the efficiencies in Figure 5.4 and 5.5, the total number of near neighbours for larger k values are reduced significantly from simply considering k -shells of interaction cells. This is because some distant cells contain particles which are all outside the radius of significance and thus the whole cell can be disregarded. *e.g.* consider the case of square cells for $k=4$. The cells at the $(\pm 3, \pm 3)$ positions (relative to the home cell of our test particle) are all greater than R_C from both the test particle and the test particle home cell, thus can be ignored.

To better compare the merits of hexagons, squares and triangles, Figure 5.4 has been re-plotted over a more sensible range of cell-reduction (Figure 5.5).



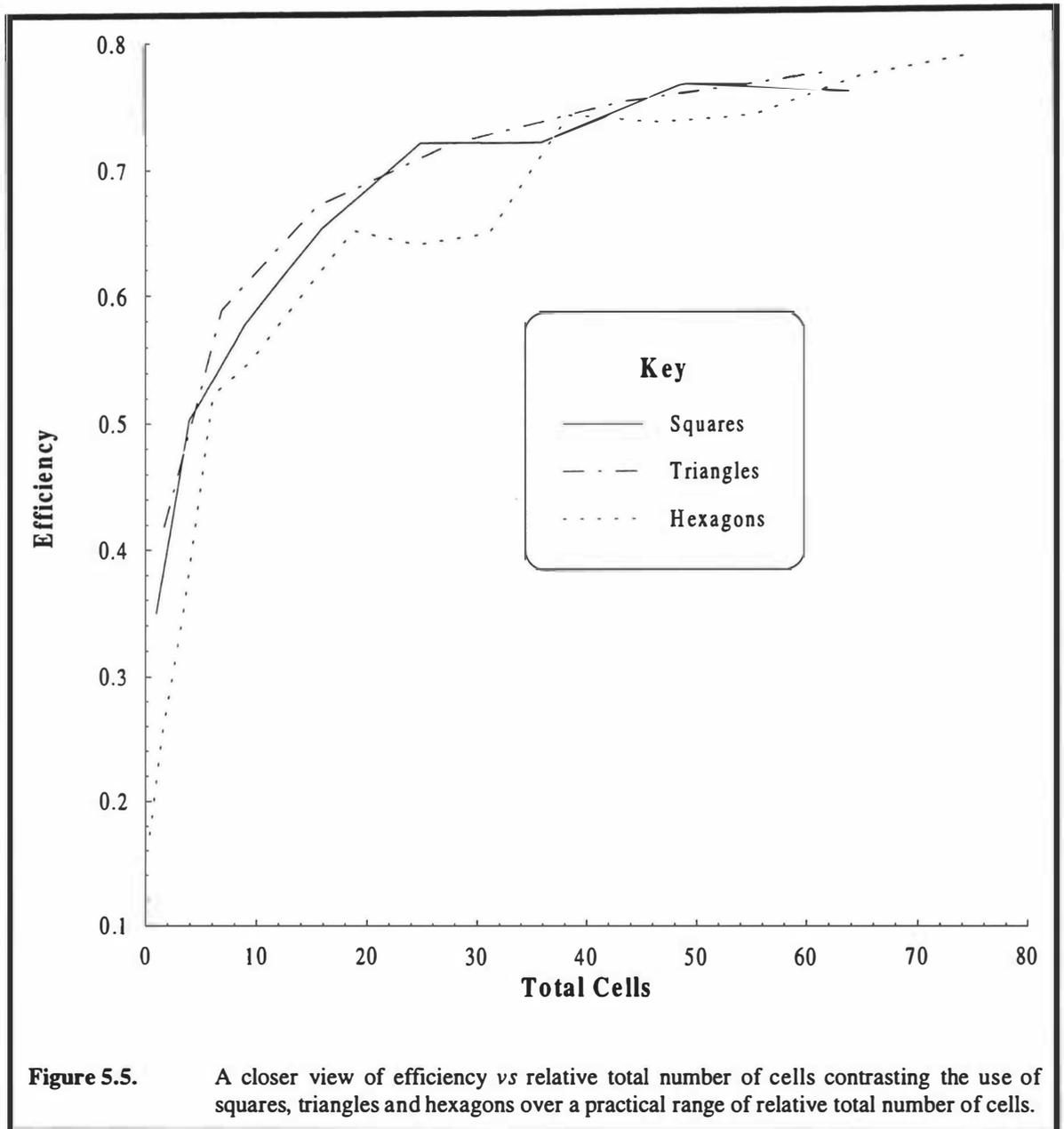


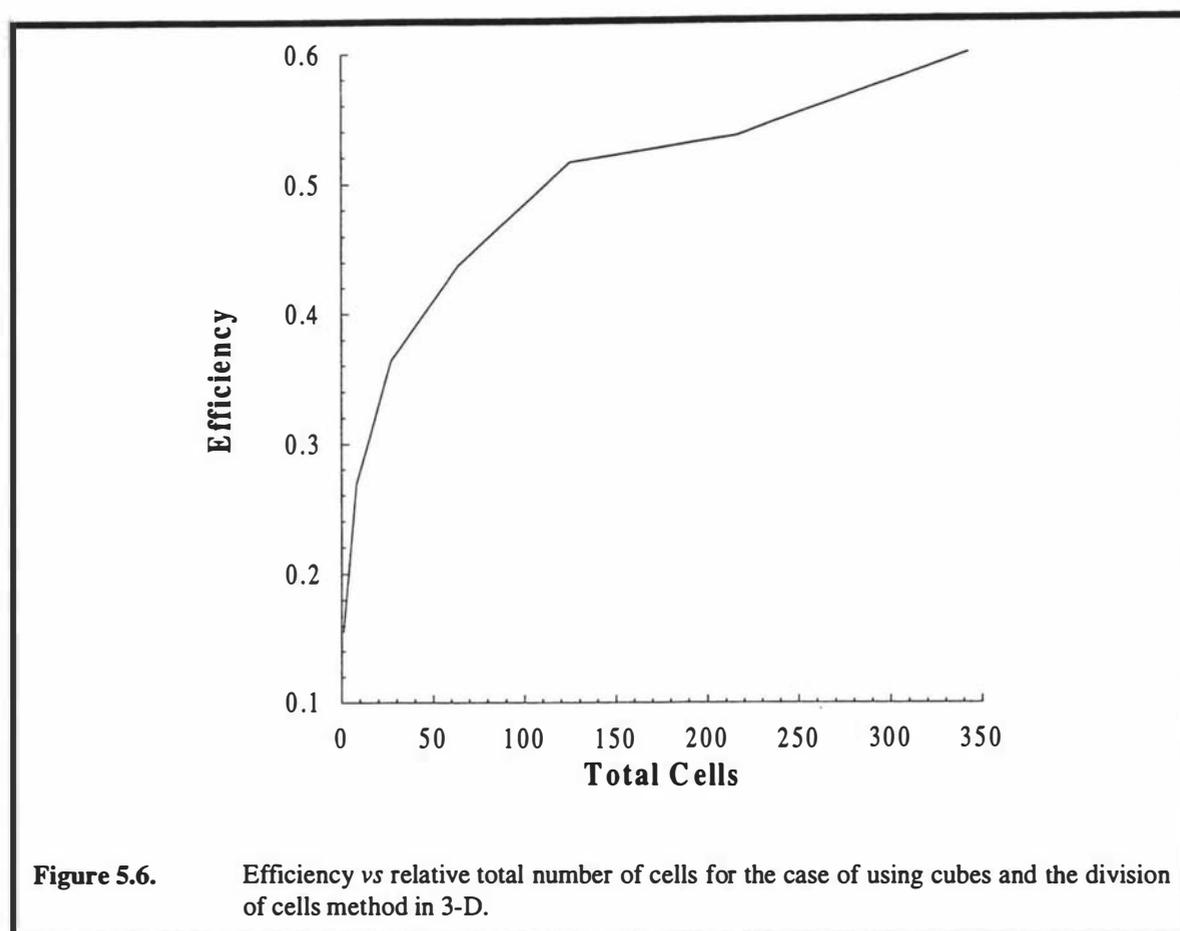
Figure 5.5 indicates that hexagons are inferior to triangles and squares and that triangles hold a very slight advantage over squares, but the difference is much less than Table 5.3 may have indicated. On this basis of theoretical efficiency, we could conclude that triangles are a better shape of parent cell, albeit only slightly better. However, while for smaller k triangles have a slight edge, in the limit, the advantage disappears - all shapes are equal. More significantly, when using a square as a parent cell it is easy to determine the significant near-neighbours by the use of integers. The conclusion is that the combined net effect of considering both theoretical efficiency and near-neighbour identification leads to no gain in the use of triangles over squares, and thus we suggest squares should be used because of their simpler shape.

Given this finding for 2-D, the more difficult case of 3-D was not investigated in any detail - as while it may be likely that a small theoretical advantage could be obtained by considering an alternative, determining near neighbours will be even more complicated. Presented in Table 5.6 are theoretical efficiencies and numbers of near-neighbours for case of cubes in 3-D, which on the basis of 2-D findings is likely to be optimal, or at least very close to it. This data is plotted in an analogous manner to the 2-D data, in Figure 5.6.

k	Near Cells	Efficiency	Rel.
1	27	0.1551	1
2	125	0.2681	8
3	311	0.3637	27
4	613	0.4373	64
5	1015	0.5159	125
6	1689	0.5357	216
7	2399	0.5989	343

Table 5.6. Theoretical efficiencies and numbers of significant near neighbour cells for subdivided cubes.

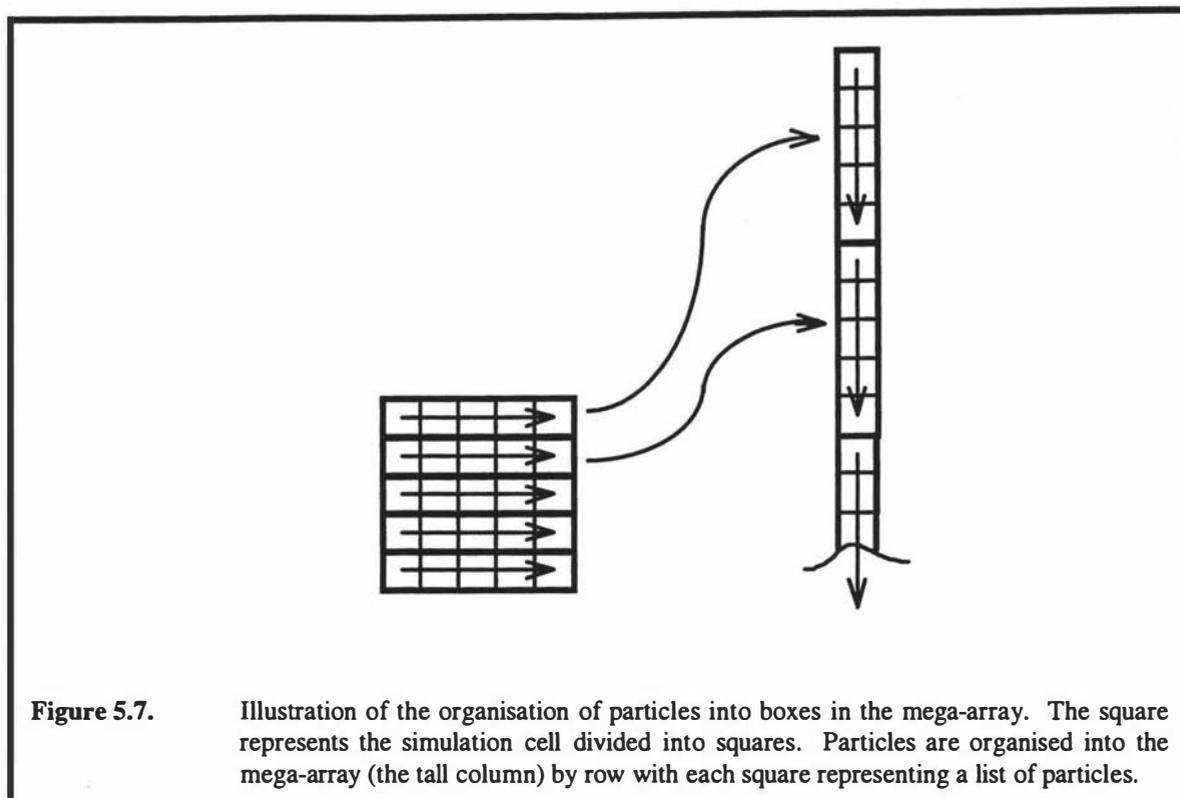
Figures 5.5 and 5.6 best illustrate the effect of changes in geometry and cell division. It appears that changing the cell geometry is not advantageous in the endeavour of improving overall algorithm efficiencies and a boxing schemes efficiency is best improved by dividing the cell into as many boxes as possible.



5.4. Maintaining Vectorisability

While it has been suggested here that a reduction in the basic cell size reduces vectorisability and to maximise the effectiveness of code based on Scheme 5.4 this should be avoided, the aim in constructing a short-range algorithm should be to minimise the amount this small number M is subdivided, *i.e.* minimise the reduction of inner loop lengths. However, the sheer nature of short-range interactions, whereby a target particle needs only to interact with a small number of neighbouring particles, does not lend itself well to vectorisation. The largest average particle loop length will only be M , which is extremely unfavourable for very short-range interactions. This inherently limited vectorisability potential of short-range interactions is not affected as greatly by cell reduction as one would initially anticipate, especially if the particles are reordered in

the mega-array by box, as suggested earlier. This organisation is illustrated in Figure 5.7.



In Figure 5.8 it can be seen how the use of re-ordering of particles increases loop lengths, if adopting a strategy of the type depicted in Scheme 5.4. Figure 5.8 illustrates the case where particles within cells up to $k=2$ are significant, *i.e.* cell reduction is $R_C/2$. The shaded box represents the target cell and we wish to loop over all neighbouring boxes and particles within them. The darkened rectangle within the simulation square holds a large subset of near neighbour cells for the shaded box. Since the mega-array holds the particles ordered by box, we do not have to loop over each individual box within the darkened square, but need only determine the pointer in the mega-array to the left most particle of left-most cell in the darkened square (A) and the pointer to the right-most particle of the right-most cell in the darkened square (B). Effectively this has increased loop lengths by a factor of five.

Unfortunately, as k increases so do the number of rows, and while the box ordering assists in maintaining vectorisability, loop lengths are quickly reduced - even more rapidly in 3-D. An alternative is to adopt the layer method proposed by Rapaport [59]. Rapaport discusses a number of important considerations for the computation of

short-range interactions and is primarily concerned with efficient implementations on to vector and parallel architectures. He also suggested the layer algorithm as a solution to the problem where interactions are very short-ranged and where much of the computational time could be spent screening insignificant interactions as calculating the interactions themselves.

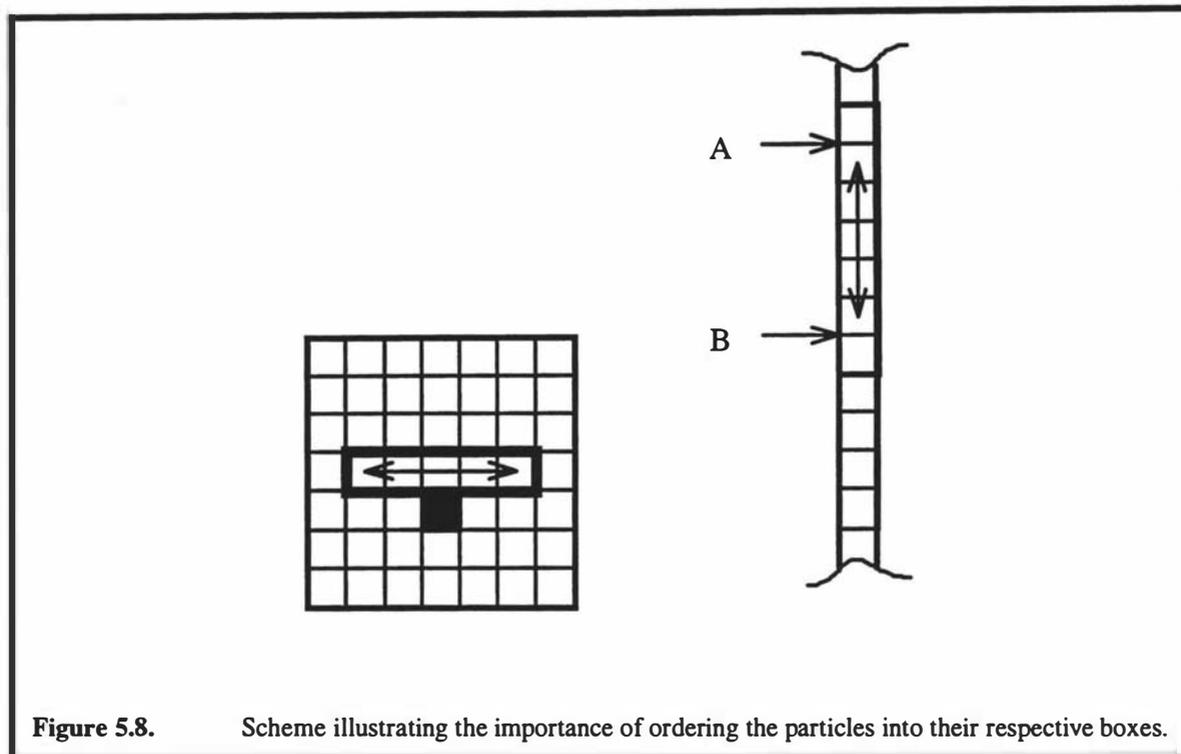


Figure 5.8. Scheme illustrating the importance of ordering the particles into their respective boxes.

The layer method begins by considering the occupancy of the cells. The identities of all first-listed particles within each cell are stored in an array which has one element assigned to each cell. This is referred to as the first layer. The second particles of the multiply occupied cells are assigned to a second layer and so on, until all particles have been recorded in a layer. The total number of layers will be equal to highest cell occupancy number, and it is best if this number is small. Low maximum occupancy numbers are encountered for very short-range interactions and in instances where large cell-division has been applied.

In Scheme 5.4 the outer loop is over cell pairs and the inner loop over particle pairs belonging to the cells. This inner loop is very short for weak interactions or when large cell-division has been applied and in these cases is not well suited to vectorisation. The layer approach, on the other hand, has an outer loop over pairs of layers and an

inner loop over cells. A large number of cells is thus required for optimal vectorisation which is also a consequence of increased cell-division.

Rapaport presents a detailed discussion for optimised vector and parallel processing implementations of the layer method. It is envisaged that cubes/squares be reduced to afford an average occupancy of one to achieve maximum efficiency of the type (5.2) and increase the effectiveness of an implemented vectorised layer method.

5.5. Conclusions

The primary conclusion from the investigations presented here is that efficiencies must be compared for alternative boxing schemes with similar total numbers of cells. The second conclusion is that when theoretical efficiencies are similar other factors need to be reconsidered, such as the ease by which only the significant near-neighbour cells may be determined. From this, it has been concluded that the square, and subsequently the cube, are the most effective cells with which to divide a simulation box. This is based on the fact that no other parent cell geometry offers any significant advantage over squares and most likely cubes. Increased efficiencies can be obtained by reduction of the parent cell size using cell reduction and while occupancies of cells are reduced vectorisation can be maintained *via* the use of the 'layer' approach to short-range interactions.

CHAPTER SIX

LONG RANGE KWIK

6.0. Introduction

In the statistical origins of KWIK and in the discussions on determining the optimal separator in Chapter Four, the emphasis was on the replacement of the operator (in the statistics problem the Heaviside function, in the Coulomb problem the long-range operator) by its Fourier series representation. However, while the Fourier series representation of the Heaviside function is well known [53], it became clear that Fourier series representations of the types of operators to be encountered in the Coulomb case were less so. Thus, the decay characteristics of the long-range Fourier series were discussed in terms of the Fourier transform and, given the connection of the Fourier series with the Fourier transform, it is postulated that recasting the long-range KWIK Coulomb partition in terms of its Fourier transform rather than Fourier series, would have little effect on the underlying success of the algorithm, perhaps even enabling the development of a more efficient method of determining the long-range partition.

In discussions presented thus far, generality has attempted to be maintained in the derivations and while the separator of choice has been stipulated and implemented in calculations presented in Chapter Seven, generality will be maintained in this chapter as the techniques used apply to any suitable long-range operator. In the development of the algorithm the simplest possible cases have been considered, optimised and then the complexity of the problem increased until a general procedure is obtained. This is achieved by first considering collinear unit point charges, that is, points on a line. This is then expanded to three dimensions, where randomly placed particles and continuous distributions of charge are considered.

6.1. Fourier Transform of the Long-Range Partition

The total long-range KWIK energy for a system of N charge distributions $Q_j(\mathbf{r}, \mathbf{R}_j)$, where \mathbf{R}_j is the centre of the distribution, is given by

$$E_{long} = \frac{1}{2} \sum_{l=1}^N \sum_{j=1}^N \iint Q_l(\mathbf{r}_l, \mathbf{R}_l) \left(\frac{1 - f(r_{l2})}{r_{l2}} \right) Q_j(\mathbf{r}_2, \mathbf{R}_j) d\mathbf{r}_1 d\mathbf{r}_2 \quad (6.1)$$

Substituting the long-range operator by its Fourier transform yields

$$E_{long} = \frac{1}{2} \sum_{l=1}^N \sum_{j=1}^N \iint Q_l(\mathbf{r}_l, \mathbf{R}_l) \int \hat{f}(\mathbf{k}) \exp(i\mathbf{k} \cdot (\mathbf{r}_1 - \mathbf{r}_2)) d\mathbf{k} Q_j(\mathbf{r}_2, \mathbf{R}_j) d\mathbf{r}_1 d\mathbf{r}_2 \quad (6.2)$$

where $\hat{f}(\mathbf{k})$ is the FT of the long-range operator. Inverting the integration and factorising the exponential yields

$$E_{long} = \frac{1}{2} \int \hat{f}(\mathbf{k}) \sum_{l=1}^N \int Q_l(\mathbf{r}_l, \mathbf{R}_l) \exp(i\mathbf{k} \cdot \mathbf{r}_l) d\mathbf{r}_l \sum_{j=1}^N \int Q_j(\mathbf{r}_2, \mathbf{R}_j) \exp(-i\mathbf{k} \cdot \mathbf{r}_2) d\mathbf{r}_2 d\mathbf{k} \quad (6.3)$$

This can be rewritten more concisely as

$$E_{long} = \frac{1}{2} \int I(\mathbf{k}) \hat{f}(\mathbf{k}) d\mathbf{k} \quad (6.4)$$

where the auxiliary function is defined

$$I(\mathbf{k}) = \left| \sum_{j=1}^N \int Q_j(\mathbf{r}, \mathbf{R}_j) \exp(i\mathbf{k} \cdot \mathbf{r}) d\mathbf{r} \right|^2 \quad (6.5)$$

which is the Fourier intensity of the charge distribution.

In order to determine (6.4) in linear work and therefore the long-range energy partition in linear work, it must be evaluated numerically, as if (6.5) was to be expanded, $O(N^2)$ terms would be obtained.

To illustrate the development of the algorithm, consider the special case when the charges are positive unit point charges

$$Q_j(\mathbf{r}, \mathbf{R}_j) = \delta(\mathbf{r} - \mathbf{R}_j) \quad (6.6)$$

such that the Fourier intensity after integration becomes

$$I(\mathbf{k}) = \left| \sum_{j=1}^N \exp(i\mathbf{k} \cdot \mathbf{R}_j) \right|^2 \quad (6.7)$$

Consider now the case where the point charges are collinear which yields the long-range energy in the form

$$E_{long} = \frac{1}{2} \int I(k) \hat{f}(k) dk \quad (6.8)$$

and with

$$I(k) = \left| \sum_{j=1}^N \exp(ikR_j) \right|^2 \quad (6.9)$$

This simplification enables the close study of the important characteristics of integrals of the type (6.4).

6.2. Numerical Quadrature

6.2.0. Introduction

In order to evaluate (6.8) and later (6.4) as efficiently as possible, in a general fashion and in $O(N)$ work, we must first re-emphasise the assumption that our charge distribution is finite and that we must evaluate this numerically.

Consider the following integral

$$I = \int_a^b f(x) w(x) dx \quad (6.10)$$

and let us assume that both f and w are smooth functions and the integral converges over the region (a,b) . To evaluate this integral to some predetermined accuracy, any one of four basic approaches can be adopted.

1. Use a general quadrature technique, assuming nothing about the nature of the two functions f and w (other than those outlined above regarding convergence, *etc.*). Such an assumption yields a scheme

$$I = \int_a^b f(x) w(x) dx \approx \sum_{k=1}^M a_k f(x_k) w(x_k) \quad (6.11a)$$

2. Use approximations of both f and w to afford a quadrature scheme which is exact for a n^{th} order approximation of f and m^{th} order approximation of w . This will often lead to a more efficient quadrature than (6.11 a) (but looks essentially the same)

$$I = \int_a^b f(x) w(x) dx \approx \sum_{k=1}^M a_k f(x_k) w(x_k) \quad (6.11b)$$

3. Develop a quadrature scheme that approximates w to some degree m , and is exact for f , affording a scheme

$$I = \int_a^b f(x) w(x) dx \approx \sum_{k=1}^M f_k w(x_k) \quad (6.11c)$$

4. Develop a quadrature scheme that approximates f to some degree n , and is exact for w , affording a scheme exactly analogous to (6.11 c)

$$I = \int_a^b f(x) w(x) dx \approx \sum_{k=1}^M w_k f(x_k) \quad (6.11d)$$

(Assuming that the length of the expansion M , and the roots $\{x_k\}$ and weights $\{a_k, f_k, w_k\}$ are specific to the individual case.)

Clearly, the more information about f and w that can be incorporated into a quadrature scheme the more likely a more efficient method of evaluating (6.10) will be obtained.

6.2.1. Simple Quadrature

The simplest and most general approach to quadrature of the type depicted in the scheme illustrated in (6.11 a), assuming little about the integral I , are those using equally spaced roots. Two such examples are the Trapezoidal rule and Simpson's rule [53,138,139].

The Trapezoidal rule approximates the integral by an interpolatory scheme up to first order (thus affording an error term involving the second derivative of the function), or more physically, by approximating the area by a sum of trapezoids

$$I = \int_a^b g(x) dx = h \left[\frac{g_0 + g_M}{2} + \sum_{k=1}^{M-1} g_k \right] - \frac{Mh^3}{12} g''(\xi) \quad (6.12)$$

where

$$g_j = g(x_j) \quad (6.13)$$

and

$$h = \frac{b-a}{M} \quad (6.14)$$

and

$$\xi \in [a, b] \quad (6.15)$$

Thus, such a formula is exact for polynomials of degree one and can be easily applied to (6.10) by substituting the product of f and w for g . Improved accuracy is obtained by increasing the number of trapezoids (reducing h), which has the effect of reducing the error by the inverse square of that number.

Simpson's rule

$$I = \int_a^b g(x) dx = \frac{h}{3} [g_0 + 4(g_1 + g_3 + \dots + g_{2M-1}) + 2(g_2 + g_4 + \dots + g_{2M-2})] - \frac{Mh^5}{90} g^{(4)}(\xi) \quad (6.16)$$

can be considered an extension of the Trapezoidal rule, by using an interpolatory scheme up to second order. While this would then seem to afford a rule exact only up to degree two, it can be shown [138] that Simpson's rule is exact up to degree three thus, the error term incorporating the fourth derivative of g .

6.2.2. Gauss Type Quadrature Rules

Gauss type quadrature [139] work on the principle that either the integrand, or part of, can be approximated to high accuracy by a polynomial of low order.

In general, the Gauss type rules are of the form

$$I = \int_a^b f(x) w(x) dx \approx \sum_{k=1}^M w_k f(x_k) \quad (6.17)$$

where w is a fixed weight function defined on the interval $[a,b]$ which may be finite or infinite. Thus, the Gauss type quadrature can be applied as (6.11b-d).

Gauss type quadrature rules are derived from the possibility of constructing sets of polynomials which are orthogonal with respect to the weight function w . Normalisation lends to polynomials

$$\int_a^b p_m^*(x) p_n^*(x) w(x) dx = \delta_{mn} \quad (6.18)$$

that satisfy a three term recurrence relationship

$$p_n^*(x) = (a_n x + b_n) p_{n-1}^*(x) - c_n p_{n-2}^*(x), \quad n = 1, 2, 3, \dots \quad (6.19)$$

where $a_n, c_n \neq 0$ and

$$p_{-1}^*(x) = 0 \quad (6.20)$$

and

$$p_0^*(x) = \left(\int_a^b w(x) dx \right)^{1/2} \quad (6.21)$$

The roots of the polynomials x_k , lie within the region (a,b) and by careful selection of the corresponding weights w_k , yield a quadrature scheme (6.17) exact for polynomials up to degree $2M-1$. That is, an M point Gauss type quadrature scheme (6.17) using the weight function w , will give exact answers for functions f of polynomials up to degree $2M-1$.

6.2.3. Basis Spanning

The success of the use of Gauss type quadrature rules is dependent on how well the monomials

$$1, x, x^2, x^3, x^4, \dots, x^{2M-1} \quad (6.22)$$

represent the function f . Often such a basis is a poor representation of the function f in that the basis functions (6.22) do not span f , or simply, such an expansion of f converges very slowly. It is possible to construct alternative quadrature schemes using an incomplete set of basis functions $\{\phi_i\}$ that converge to f more quickly than those of type (6.22).

6.3. Integration of FT Long-Range KWIK

6.3.0. Introduction

The most efficient quadrature scheme is likely to be the one that incorporates as much information about the integrand as possible. In this section the individual components of the special case long-range KWIK integrand (6.8) are considered. That is, the Fourier transform of the long-range operator and the Fourier intensity of the charge distribution, $I(k)$ (6.9). This is followed by discussions as to how this information may be used to afford an accurate quadrature scheme by considering each of the approaches outlined in the previous section.

6.3.1. The Separator Transform

The Fourier transform of all the separators considered in Chapter Four have a logarithmic singularity at the origin and decay rapidly away from the origin (For example, see Figure 6.1). This behaviour arises from the short-range behaviour of the operator dictating long-range behaviour of the FT, and the long-range behaviour of the operator dictating the short-range characteristics of the FT. The operator was constructed to be as flat as possible - flat operator origin gives rapid long-range decay, and a slow operator long-range decay gives a FT singularity.

A discussion as to the nature of the Fourier intensity will be presented later, but it is suffice to say at this point that $I(k)$ is both highly oscillatory and aperiodic.

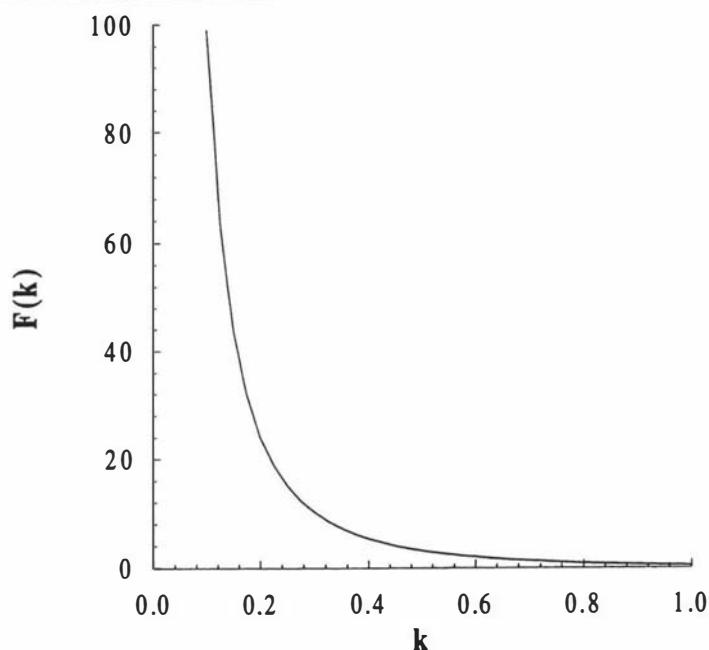


Figure 6.1. Fourier transform of the long-range operator obtained from using the error function separator. The logarithmic singularity and rapid long-range decay typify the behaviour of all the trialed separator functions.

6.3.2. Simple Quadrature

The rapid decay feature aids our task in both simple quadrature schemes and Gauss type quadrature schemes. For simple quadrature schemes we can truncate the right hand bound of the integration region on the basis of magnitude of the FT at that bound. A more rapidly decaying function allows us to truncate much earlier meaning the width of the integration 'strips' h , is smaller for similar numbers of points, giving a reduced error term. In the case of Gauss type quadrature schemes where the FT is treated exactly, then if a more rapidly decaying FT is used the polynomial approximation of the Fourier intensity need not be so accurate for larger k .

The singularity is particularly significant when using simple quadrature rules, and significant problems were initially encountered trying to deal with this. Determining the right hand bound in order to minimise the number of roots was reasonably simple, given some predetermined accuracy ϵ , one simply determined the position at which the value of the FT was less than ϵ . However, the left hand bound was a little more difficult given

that the first and last roots are at each bound, and the origin contained a singularity, which is formally the left hand bound.

Initially, the left hand bound was approximated as

$$a \approx a + \delta \tag{6.23}$$

where δ is a small positive number. In the case of the integral (6.8) the right hand bound simply became δ . Unfortunately, the optimal δ was dependent on both the accuracy and the size of the distribution, implying that a greater number of quadrature points would be necessary for larger systems, which would appear to increase the cost scaling characteristics above $O(N)$. The singularity problem was alleviated by the addition, and subsequent subtraction, of a Gaussian function from the Fourier intensity. If chosen carefully, the integrand containing the term from which the Gaussian was subtracted contains no singularities, and problems determining the left hand bound are revoked. The new term, the addition of the Gaussian to counter the subtraction, can be evaluated analytically.

Such an exercise brought into question the amount of work required to afford higher accuracy. The analytic term captured most of the long-range energy, and the numeric term most of the computation. Could additional Gaussian functions be added and subtracted to improve the amount of total energy afforded in the analytic piece to the extent that the remainder requiring quadrature could be disregarded? In short the answer is no. Effectively the Fourier intensity is approximated by a sum of Gaussian functions, and while a single Gaussian function captures a significant part of the total long-range energy, a sum of Gaussians does not span the Fourier intensity. This observation effectively ended the investigations of simple quadrature rules - that is, that it is possible to approximate $I(k)$ by a single function and to obtain a significant part of the long-range energy, remembering that the remainder term required most of the computational effort and simple quadrature schemes are notable for their inefficient placement of roots.

Two new ideas became the focus of investigations. What function or set of functions would best approximate $I(k)$? Can a new quadrature scheme be developed as

a basis for the latest ideas for evaluating the integrals (6.8)? Let us consider the latter first.

6.3.3. Gauss Type Rules

The most common and widely used of the Gauss type quadrature rules are those resulting from the Legendre polynomials [53,139,140]. The Legendre polynomials are orthogonal with respect to a unit weight function with the resulting quadrature rule being the most efficient for approximating an integrand by a simple polynomial. They are tabulated widely [53,139,140] and can also be readily calculated (for example, GRULE in [139]) to extremely high order. The use of Legendre roots and weights for integrating the long range term gives rise to an approximation technique of the type depicted in (6.11b).

As described in section 6.2.2., it is possible to construct a set of orthogonal polynomials over an appropriate weight function, and thence determine a quadrature scheme which will be exact for polynomials up degree $2M-1$ for an M term quadrature scheme. Taking our long-range energy integral (6.8) and substituting for the erfc separator yields the specific integral for the 1-D long range energy

$$E_{long} = \frac{1}{2\pi} \int_0^{\infty} I(k) E_1\left(\frac{k^2}{4\omega^2}\right) dk \quad (6.24)$$

where E_1 is the exponential integral [53]. By making a substitution, it is possible to determine Gauss type quadrature rules for the exponential integral function which are independent of the decay parameter, hence need only be determined once for all systems. Such an approach is analogous to the following

$$\frac{\text{erf}(\omega r_{12})}{r_{12}} = \sum_{j=0}^{\infty} \alpha_j \cos(\omega \beta_j r_{12}) \quad (6.25)$$

where the α_j 's and the β_j 's for a truncated summation are determined *via* matching derivatives at the origin, *i.e.*

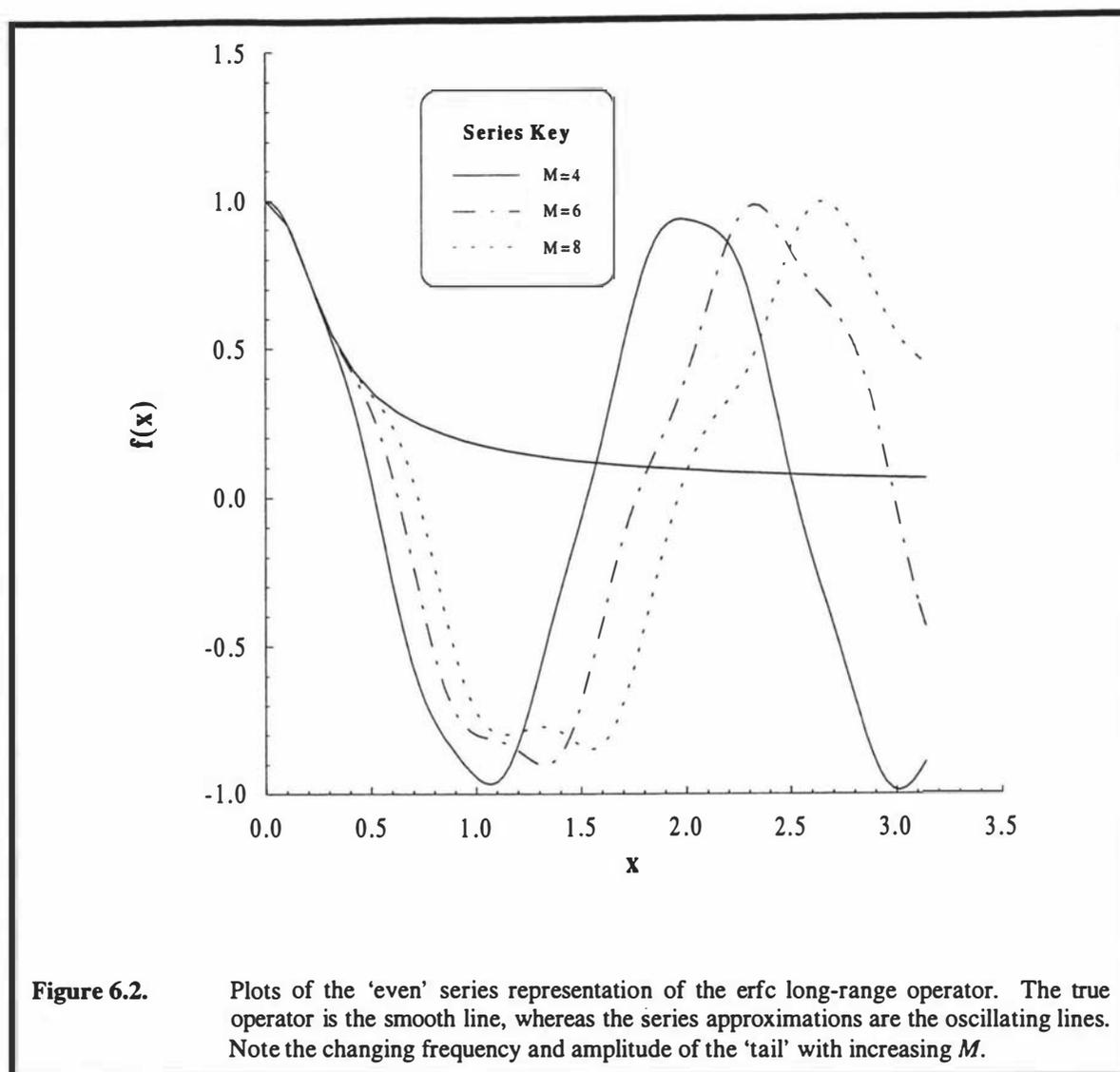
$$\left. \frac{d^n}{d r_{12}^n} \left(\frac{\text{erf}(\omega r_{12})}{r_{12}} \right) \right|_{r_{12}=0} = \left. \frac{d^n}{d r_{12}^n} \left(\sum_{j=0}^{\infty} \alpha_j \cos(\omega \beta_j r_{12}) \right) \right|_{r_{12}=0} \quad (6.26)$$

and both the roots and weights are real. Tables of roots and weights for such a scheme are tabulated in the Appendix.

It is well known that the equations whose solutions yield the roots of Gauss type quadrature schemes are numerically ill-conditioned, and even in quadruple precision it is difficult to obtain quadrature rules above about the 15-point region. Higher order rules are obtained by using the Newton iteration scheme [139] and such an approach is utilised in the GRULE subroutine.

The approach of modelling the long-range operator in the form (6.25-6) was discarded for three reasons. Firstly, it was not possible to construct a three term recurrence relation for the polynomials orthogonal with respect to the exponential integral weight function, as encountered in the long-range energy integrals. Secondly, it is difficult to produce sufficiently high order quadrature schemes - the basic approach for determining quadrature schemes is numerically unstable, and without a three term recurrence relation it is not possible to exploit the methodology used in the GRULE subroutine to construct an analogous quadrature rule engine. Thirdly, the ultimate goal is to consider three dimensional systems. The underlying theory for orthogonal polynomials and relationships with quadrature schemes is very poorly developed for higher dimensions *c.f.* one dimensional orthogonal polynomials.

The 'tail' of the modelled operator (6.25) was also of concern. The true long-range operator is smooth, whereas the modelled operator has an oscillating tail. The nature of this oscillation is dependent on whether the quadrature scheme stems from an odd or an even orthogonal polynomial. If the polynomial is even, the tail oscillates with unit amplitude about the horizontal axis and decreasing frequency with increasing order (Figure 6.2). On the other hand if the quadrature scheme stems from an odd polynomial, the tail oscillates with increasing amplitude (at least initially) above the horizontal axis and decreasing frequency with increasing order (Figure 6.3). Being unable to obtain high order quadrature schemes, it could not be determined if this behaviour would be problematic.



From findings in later CASE (Coulomb Attenuated Schrödinger Equation) and CAP (Coulomb Attenuated Potential's) studies, where the long-range partition was completely neglected, or modelled only crudely, it was found that for neutral systems, the long-range contribution can be successfully treated in a very approximate manner. Thus, concerns regarding the oscillating tail may be without substance.

As a final note, the problem with Gauss type quadrature is that if one wishes to increase the accuracy of the integration, the information obtained in the previous quadrature can be extended with the addition of another set of roots (see [139]) to produce higher, but not optimally higher, accuracy. Further increases in accuracy tend to require that the calculation be re-computed with a complete new set of roots and weights.

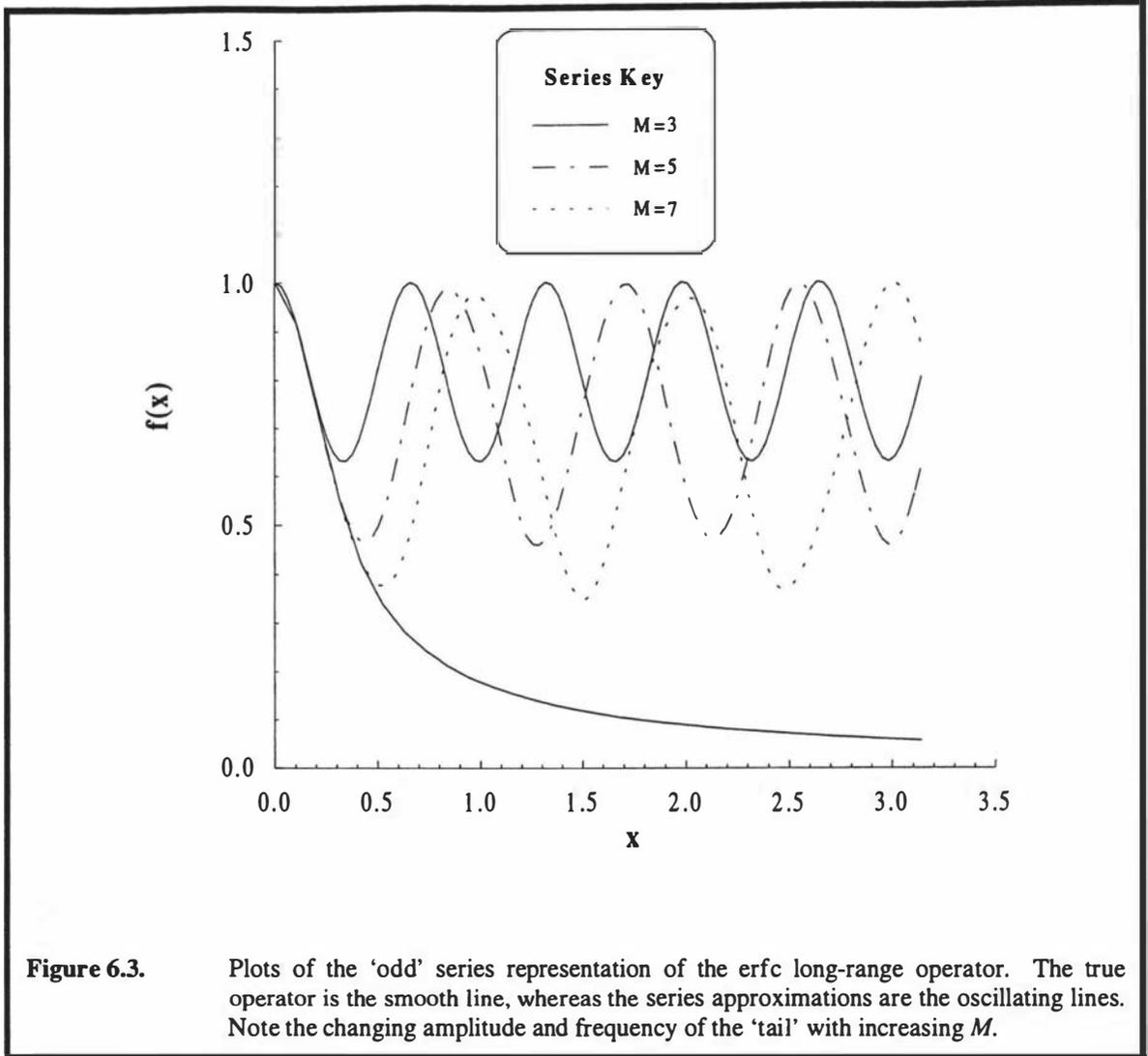


Figure 6.3. Plots of the 'odd' series representation of the erfc long-range operator. The true operator is the smooth line, whereas the series approximations are the oscillating lines. Note the changing amplitude and frequency of the 'tail' with increasing M .

6.3.4. COP's and ROP's

Coulomb Orthonormal Polynomials and Repulsive Orthonormal Polynomials [141] are a set of functions $\{\phi_i\}$ which are used to model the charge density, $\rho(\mathbf{r})$. The functions are constructed such that they are orthonormal with respect to a binary operator (the long-range operator)

$$\int_{\Omega} \phi_i(\mathbf{r}_1) L(r_{12}) \phi_j(\mathbf{r}_2) d\mathbf{r}_1 d\mathbf{r}_2 = \delta_{ij} \quad (6.27)$$

Thus, the charge density is represented as

$$\rho(\mathbf{r}) = \sum_i c_i \phi_i(\mathbf{r}) \quad (6.28)$$

and the coefficients obtained by using the orthonormality condition

$$c_j = \int_{\Omega} \rho(\mathbf{r}_1) L(r_{12}) \phi_j(\mathbf{r}_2) d\mathbf{r}_1 d\mathbf{r}_2 \quad (6.29)$$

The total long-range energy can then be written as (adopting the Einstein summation convention and Dirac notation)

$$E_{long} = \frac{1}{2} \langle \rho | v_j \rangle \langle v_j | \rho \rangle \quad (6.30)$$

where

$$v_j(\mathbf{r}_1) = \int_{\Omega} L(r_{12}) \phi_j(\mathbf{r}_2) d\mathbf{r}_2 \quad (6.31)$$

Thus, COP's and ROP's are a scheme as depicted in (6.11c) and (6.11d). While the COP/ROP approach suffers severe numerical instability problems they offer a more systematic and perhaps efficient approach to determining the long-range energy.

6.3.5. The Fourier Intensity $I(\mathbf{k})$ and the Random Walk Advantage

For large k , the Fourier intensity $I(k)$ (6.9) can be considered as the square of the progress of a Pearson [142] random walk in the complex plane. That is, if we assume that each kR_j is a random phase, then $[I(k)]^{1/2}$ can be interpreted as the net progress of a drunkard who takes N unit steps, each in a random direction, on the complex plane.

Consider a series of steps starting from the origin with length L , but with random direction. *i.e.* $Z_1 = a + ib$, $Z_2 = c + id$. The sum after N steps is then

$$S_N = Z_1 + Z_2 + \dots + Z_N \quad (6.32)$$

What will be the nature of the sum after a large number of walks? The average position of the end points must be at the origin. Given that a large number of steps are performed where Z_1 is selected randomly implies that the position on the unit circle is random, therefore the expectation position of each step is the origin. *i.e.*

$$\langle Z_1 \rangle = \langle Z_2 \rangle = \dots = 0 \quad (6.33)$$

hence the average net displacement vanishes

$$\langle S_N \rangle = \langle Z_1 + Z_2 + \dots + Z_N \rangle = 0 \quad (6.34)$$

The mean square of the deviations from the average net displacement after a large number of walks of N steps is

$$\langle D^2 \rangle = \langle S_N^2 \rangle = \langle (Z_1 + Z_2 + \dots + Z_N)(Z_1 + Z_2 + \dots + Z_N) \rangle \quad (6.35)$$

and only the square terms remain on the RHS, as all of the others, such as $\langle Z_1 Z_2 \rangle$ vanish because Z_1 and Z_2 are not correlated, *i.e.* for any selected value of Z_1 there are as many positive as negative values of Z_2 . Thus,

$$\langle D^2 \rangle = \langle Z_1^2 + Z_2^2 + \dots + Z_N^2 \rangle \quad (6.36)$$

which is N terms of the step length squared. Therefore the mean square of the net displacement is simply

$$\langle D^2 \rangle = NL^2 \quad (6.37)$$

Thus, the expectation value of the Fourier intensity

$$\langle I(k) \rangle = N \quad (6.38)$$

It can be shown [143] that for large N , the probability density function for the mean square of the net displacement of the random walk above $\langle I(k) \rangle$, is an exponential variable and that in the limit of large N , of mean and standard deviation N .

Thus, for the special case of collinear, positive unit point charges, the Fourier intensity $I(k)$, begins at the origin with a value of N^2 , or more correctly

$$I(0) = \left(\sum_{j=1}^N q_j \right)^2 \quad (6.39)$$

($N^2 = (6.39)$ in the special case of unit point charges.) Initially the random phases are limited to the half plane but tend towards a random deviate on the unit circle with increasing k , thus $I(k)$ rapidly decreases from N^2 to a value of N with standard deviation N .

If, on the other hand, we have a neutral system, the sum of the charges (6.39) is zero and while the phases remain limited to the half plane, tending towards a random deviate on the unit circle, negative charges amount to ‘backward stepping’. The effect is that the random walk is applicable for all values of k .

We now revisit the statistics problem investigated by Gill [126]. In this case we have a similar situation to the positive unit point charge case, but in the statistics problem the number of steps in the random walk is

$$N = {}^n C_m \quad (6.40)$$

where m was the size of the sample drawn from the set of n numbers. Again the assumption of random phases does not apply at $k=0$ and the first term of the Fourier sum is N^2 , which by (6.40) is a very large number. As the assumption of random phase becomes valid for large (relatively, quite small) k , the Fourier terms reduce to N . This is how such rapid convergence is obtained for the probability.

6.3.6. Discussion

The major problem with devising appropriate quadrature schemes to evaluate integrals of the type (6.10), particularly integrals that vary moderately from system to system, is that it is always possible to construct a system where the chosen scheme performs extremely badly, and the possibility exists that another more efficient scheme may be available but was not considered. There are several very important questions that must be considered when devising a quadrature scheme. What exactly defines a better, more efficient quadrature scheme? Is it one that uses fewer quadrature points? For example, while the equally spaced quadrature schemes suffer from inefficient placing of quadrature roots, massive efficiencies can be applied when implementing such schemes in a computer algorithm. Also, if one wished to increase the accuracy of one’s numerical integration, can the correct answer be improved simply by the addition of more terms? Similarly, can that answer be improved with similar ease, or must one start from scratch with a higher order scheme to obtain improved accuracy?

Density Functional Theory has been plagued by the requirement that the exchange-correlation energy must be evaluated using quadrature [144]. Rotational

invariance and differences in opinion regarding the efficiency vs flexibility argument are typical of the problems encountered, and have only recently been silenced with the development of the first gridless functional [145].

While it is possible to obtain a gridless long-range KWIK energy by expansion of (6.5), an $O(N^2)$ algorithm results. Furthermore, while the Gauss quadrature techniques are likely to afford an efficient scheme, multiple integrals like those encountered when KWIK is generalised to higher dimensions, pose a greater problem. Gauss quadrature schemes for multiple integrals are either inefficient for product type rules, or not well-defined in terms of the orthogonal polynomials in higher dimensions. Also, unless significant understanding is available, the equations formulated which must be solved to obtain the roots and weights of 1-D quadrature rules of Gauss type (whether it be the more traditional or those using more arbitrary basis functions) become extremely ill-conditioned [136] very quickly. It was found that even using quadruple precision, numerical stability was lost by about $M=15$.

None of the quadrature schemes investigated appeared to show significant promise to warrant further investigation. Difficulties were encountered in both the rate of convergence of the trialed schemes, and more particularly, it was difficult to obtain sufficiently large expansions so that direct comparisons with the lesser considered schemes could be made, for sufficient levels of accuracy.

6.4. Fourier Series

6.4.0. Introduction

In section 6.3 a discussion was presented on the numerical integration of the Fourier transform of the long-range partition of KWIK, which appears to have been less than fruitful. Given the origins of KWIK in the combinatorial problem [126], it may seem that investigating an approach alternative to the use of a Fourier series was premature. However, the use of the Fourier series can be likened to using Simpson's rule for quadrature. While it is well known that it is less than efficient in its choice of quadrature roots, that is, a much more accurate rule can be obtained using fewer points (for example, Gauss type quadrature), a great number of efficiencies can be incorporated

due to the equidistant nature of the roots. Furthermore, unlike the Gauss type quadrature, a Fourier series in higher dimensions is well defined. In hindsight, the investigations in section 6.3 helped highlight a number of important features of the long-range partition - none more important than the majority of the long-range energy partition being easily captured and the remaining detail requiring significant effort.

In this section the Fourier series approach to the long-range partition is considered. A Fourier series representation of a periodic function $f(x)$ is a unique representation of a linear combination of all cosine and sine functions which have the same period, say $2l$. In order to represent a function by a Fourier series the theorem (6.1) [130] must hold.

Theorem 6.1. If a periodic function $f(x)$ with period $2l$ is piece-wise continuous in the interval $-l \leq x \leq l$ and has a left-hand derivative and right-hand derivative at each point of that interval, then the Fourier series representation of $f(x)$ is convergent. Its sum is $f(x)$, except at a point x_0 at which $f(x)$ is discontinuous and the sum of the series is the average of the left- and right-hand limits of $f(x)$ at x_0 .

6.4.1. Fourier Coefficients

Given that the function wished to be represented as a Fourier series satisfies Theorem 6.1, it is possible to determine Fourier series coefficients by using the orthogonality of the trigonometric function over the interval of their periods or multiples of their periods. Two functions are said to be orthogonal if and only if

$$\int_{-l}^l f(x)g(x)dx = \begin{cases} 0, f(x) \neq g(x) \\ c, f(x) = g(x) \end{cases} \quad (6.41)$$

and thus given

$$\int_{-l}^l \left\{ \cos\left(\frac{m\pi x}{l}\right) \sin\left(\frac{n\pi x}{l}\right) \right\} dx = 0, \quad \forall m, n \in I \quad (6.42)$$

and

$$\left. \begin{aligned} \frac{1}{l} \int_{-l}^l \cos\left(\frac{m\pi x}{l}\right) \cos\left(\frac{n\pi x}{l}\right) dx \\ \frac{1}{l} \int_{-l}^l \sin\left(\frac{m\pi x}{l}\right) \sin\left(\frac{n\pi x}{l}\right) dx \end{aligned} \right\} = \begin{cases} 0, m \neq n \\ 1, m = n \end{cases} \quad (6.43)$$

Fourier coefficients for the Fourier series representation of a function

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{n\pi x}{l}\right) + b_n \sin\left(\frac{n\pi x}{l}\right) \right] \quad (6.44)$$

can be obtained by multiplication of (6.44) by the appropriate trigonometric function and integrating over the period $2l$ obtaining

$$a_0 = \frac{1}{2l} \int_{-l}^l f(x) dx \quad (6.45)$$

$$a_n = \frac{1}{l} \int_{-l}^l f(x) \cos\left(\frac{n\pi x}{l}\right) dx \quad (6.46)$$

$$b_n = \frac{1}{l} \int_{-l}^l f(x) \sin\left(\frac{n\pi x}{l}\right) dx \quad (6.47)$$

It is more convenient to recast the Fourier series in its complex form

$$f(x) = \sum_{n=-\infty}^{\infty} c_n \exp\left(\frac{inx}{l}\right) \quad (6.48)$$

where the coefficients are obtained using similar orthogonality conditions

$$c_n = \frac{1}{2l} \int_{-l}^l f(x) \exp\left(\frac{-in\pi x}{l}\right) dx \quad (6.49)$$

where i is the complex number $\sqrt{-1}$.

Similar arguments can be applied for complex Fourier series in higher dimensions, assuming a periodic n -dimensional cube of length l , to yield

$$f(\mathbf{x}) = \sum_{\mathbf{k}} c_{\mathbf{k}} \exp\left(\frac{i\pi}{l} \mathbf{k} \cdot \mathbf{x}\right) \quad (6.50)$$

where the summation is over all \mathbf{k} vectors. Thus, the corresponding Fourier coefficients are given by

$$c_{\mathbf{k}} = \frac{1}{(2l)^n} \int_{\Omega} f(\mathbf{x}) \exp\left(\frac{-i\pi}{l} \mathbf{k} \cdot \mathbf{x}\right) d\mathbf{x} \quad (6.51)$$

6.4.2. Recasting KWIK using a Fourier Series

Reconsider the total long-range KWIK energy of N charge distributions $Q_j(\mathbf{r}, \mathbf{R}_j)$

$$E_{long} = \frac{1}{2} \sum_{l=1}^N \sum_{j=1}^N \iint Q_l(\mathbf{r}_1, \mathbf{R}_l) \left(\frac{1 - f(r_{12})}{r_{12}} \right) Q_j(\mathbf{r}_2, \mathbf{R}_j) d\mathbf{r}_1 d\mathbf{r}_2 \quad (6.1)$$

Substituting the complex Fourier series representation of the long-range operator for a period π yields

$$\left(\frac{1 - f(r_{12})}{r_{12}} \right) = \sum_{\mathbf{k}} A(\mathbf{k}) \exp(i\mathbf{k} \cdot (\mathbf{r}_1 - \mathbf{r}_2)) \quad (6.52)$$

where $A(\mathbf{k})$ are the Fourier coefficients. Inverting the summations and again factorising the exponential yields

$$E_{long} = \frac{1}{2} \sum_{\mathbf{k}} A(\mathbf{k}) \sum_{l=1}^N \int Q_l(\mathbf{r}_1, \mathbf{R}_l) \exp(i\mathbf{k} \cdot \mathbf{r}_1) d\mathbf{r}_1 \sum_{j=1}^N \int Q_j(\mathbf{r}_2, \mathbf{R}_j) \exp(-i\mathbf{k} \cdot \mathbf{r}_2) d\mathbf{r}_2 \quad (6.53)$$

which again can be rewritten more concisely

$$E_{long} = \frac{1}{2} \sum_{\mathbf{k}} A(\mathbf{k}) I(\mathbf{k}) \quad (6.54)$$

and the Fourier intensity remains as was defined in (6.5).

Note that the region over which the series is valid must be defined which will only allow us to use the equation (6.52) if

$$|\max(R_{12})| \leq \pi \quad (6.55)$$

6.4.3. Evaluating Fourier Coefficients in KWIK

In the next Chapter, the application of KWIK will be illustrated by considering the special cases of charged particles randomly scattered on a line and in three space. These special cases are examples of the more general KWIK algorithm in one and three dimensions which will be further specialised by using our chosen separator, the error function. Thus, the associated Fourier coefficients will be required. Once found, they can be applied to any problem in one or three dimensions, irrespective of the nature of charges.

6.4.3.1. Fourier Coefficients in One Dimension

The one dimensional Fourier series representation of the long-range operator using the error function separator is given by

$$\frac{\text{erf}(\omega r)}{r} = \sum_{k=-\infty}^{\infty} A(k, \omega) \exp\left(\frac{i\pi k r}{l}\right) \quad (6.56)$$

where the Fourier coefficients are given by

$$A_l(k, \omega) = \frac{1}{2l} \int_{-l}^l \frac{\text{erf}(\omega r)}{r} \exp\left(\frac{-i\pi k r}{l}\right) d r \quad (6.57)$$

For simplicity, consider the special case where the period $2l$ is 2π whose Fourier coefficients are given by

$$A_\pi(k, \omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{\text{erf}(\omega r)}{r} \exp(-ikr) d r \quad (6.58)$$

To obtain Fourier coefficients for the general case (6.57), make the substitution

$$r = \frac{l}{\pi} x \quad (6.59)$$

such that

$$A_l(k, \omega) = \frac{1}{2l} \int_{-\pi}^{\pi} \frac{\text{erf}\left(\frac{\omega x l}{\pi}\right)}{x} \exp(-ikx) dx \quad (6.60)$$

and the general Fourier coefficients can be obtained from the special case *via*

$$A_l(k, \omega) = \frac{\pi}{l} A_{\pi}\left(k, \frac{\omega l}{\pi}\right) \quad (6.61)$$

Thus, in order to obtain Fourier coefficients valid over a region l , we need only be able to determine those for the special case when $l = \pi$.

Consider the Fourier coefficients given by the integral

$$A_{\pi}(k, \omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{\text{erf}(\omega r)}{r} \exp(-ikr) dr \quad (6.58)$$

Because the long-range operator is an even function

$$A_{\pi}(k, \omega) + A_{\pi}(-k, \omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{\text{erf}(\omega r)}{r} [\exp(ikr) + \exp(-ikr)] dr \quad (6.62)$$

which becomes

$$A_{\pi}(\pm k, \omega) = \frac{2}{\pi} \int_0^{\pi} \frac{\text{erf}(\omega r)}{r} \cos(kr) dr \quad (6.63)$$

and which can now be split into two parts

$$A_{\pi}(\pm k, \omega) = \frac{2}{\pi} \int_0^{\pi} \left(\frac{\cos(kr)}{r} - \frac{\text{erfc}(\omega r) \cos(kr)}{r} \right) dr \quad (6.64)$$

Assuming that $\text{erfc}(\omega\pi)$ is negligible, (6.64) can be approximated by

$$A_{\pi}(\pm k, \omega) = \frac{2}{\pi} \int_0^{\pi} \frac{\cos(kr)}{r} dr - \frac{2}{\pi} \int_0^{\infty} \frac{\text{erfc}(\omega r) \cos(kr)}{r} dr \quad (6.65)$$

which can be found in an analytical form

$$A_{\pi}(\pm k, \omega) = \frac{1}{\pi} \left[E_1 \left(\frac{k^2}{4\omega^2} \right) + 2 \text{Ci}(\pi k) \right] \quad (6.66)$$

where E_1 and Ci are the exponential and cosine integrals [53] respectively. The special case of $k=0$ has a Fourier coefficient

$$A_{\pi}(0, \omega) = \frac{1}{\pi} \left[\ln(2\pi\omega) + \frac{\gamma}{2} \right] \quad (6.67)$$

where γ is Euler's constant. Thus, to afford a Fourier coefficient given k and ω , the exponential and cosine integrals must be accurately determined. The latter integral was obtained using the function `CISI` from numerical recipes [136], and the former by using the well known power series [53] representation and then assuming insignificance for values greater than the radius of the convergence.

For small k , $A(k, \omega)$ are positive and like the exponential error function decay approximately exponentially. However, using asymptotic expansions for E_1 and Ci it can be shown that beyond

$$k_{crit} = 2\omega \sqrt{\ln(2\pi^2 \omega^2)} \quad (6.68)$$

the cosine integral becomes the dominant term and the Fourier coefficients oscillate and decay only as k^{-2} . Figure 6.4 illustrates this for $\omega=10$. The rate of decay of the Fourier coefficients follows a similar trend for larger ω illustrated by considering $\omega=100$ (Figure 6.5).

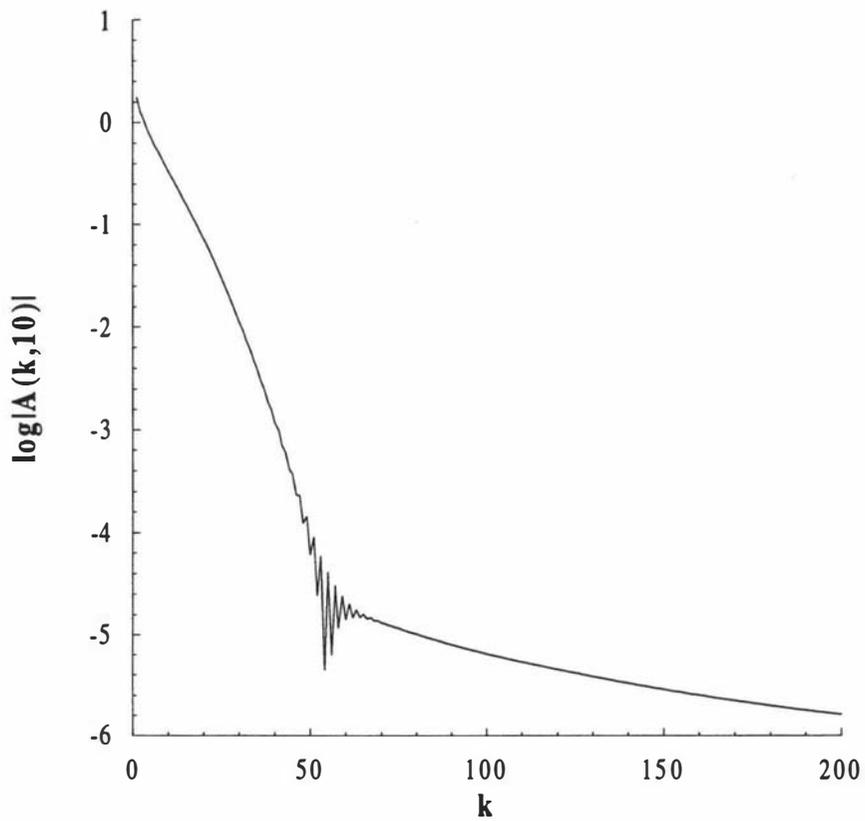


Figure 6.4. Log of the absolute value of the 1-D Fourier coefficients vs k for $\omega=10$. Note that $A_x(k, 10)$ decreases by approximately 5 orders of magnitude between $k=0$ and $k=60$. It is not until $k=10,000$ that a second such reduction is achieved.

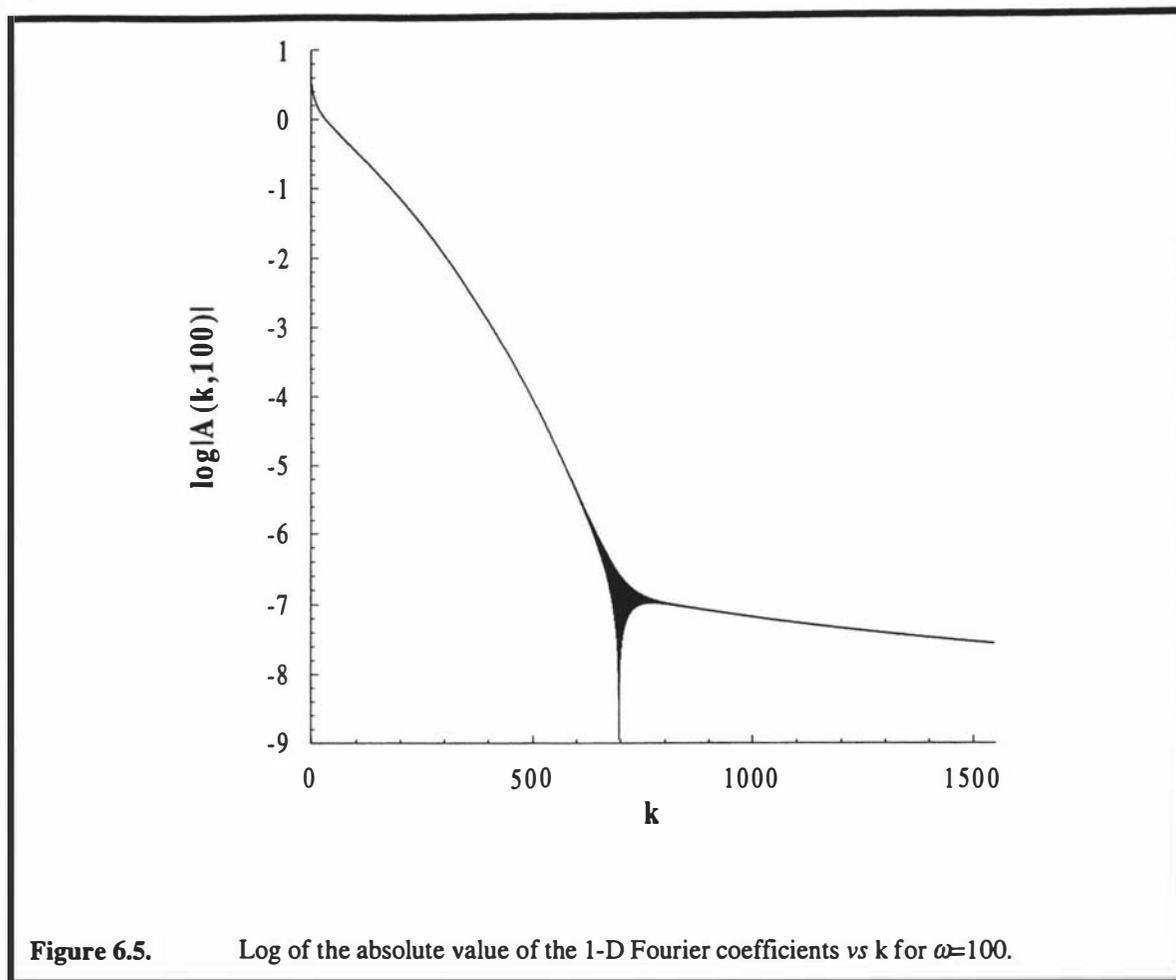


Figure 6.5. Log of the absolute value of the 1-D Fourier coefficients vs k for $\omega=100$.

6.4.3.2. Fourier Coefficients in Three Dimensions

Fourier coefficients in three dimensions require the same orthogonality conditions as for one dimension. Thus a region is required over which the Fourier series representation will be an exact expansion and over which the trigonometric components are also orthogonal. The most obvious region is that of a cube with sides of length $2l$, centred at the origin. Thus, the Fourier representation of the long-range operator in three dimensions is

$$\frac{\text{erf}(\omega|\mathbf{r}|)}{|\mathbf{r}|} = \sum_{k_x=-\infty}^{\infty} \sum_{k_y=-\infty}^{\infty} \sum_{k_z=-\infty}^{\infty} A_l(\mathbf{k}, \omega) \exp\left(\frac{i\pi}{l} \mathbf{k} \cdot \mathbf{r}\right) \quad (6.69)$$

where the Fourier coefficients are given by

$$A_l(\mathbf{k}, \omega) = \frac{1}{(2l)^3} \int_{-l}^l \int_{-l}^l \int_{-l}^l \frac{\text{erf}(\omega|\mathbf{r}|)}{|\mathbf{r}|} \exp\left(\frac{-i\pi}{l} \mathbf{k} \cdot \mathbf{r}\right) dx dy dz \quad (6.70)$$

Fourier coefficients for any cube of sides with length $2l$ may be obtained from those of the cube 2π . Thus, it is only necessary to consider the coefficients for the cube of sides with length 2π

$$A_{\pi}(\mathbf{k}, \omega) = \frac{1}{(2\pi)^3} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \frac{\text{erf}(\omega|\mathbf{r}|)}{|\mathbf{r}|} \exp(-i\mathbf{k} \cdot \mathbf{r}) \, dx \, dy \, dz \quad (6.71)$$

To obtain these, we begin in a similar fashion to the 1-D case and rewrite (6.71) as

$$A_{\pi}(\mathbf{k}, \omega) = \frac{1}{(2\pi)^3} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \left\{ \frac{1}{|\mathbf{r}|} - \frac{\text{erfc}(\omega|\mathbf{r}|)}{|\mathbf{r}|} \right\} \exp(-i\mathbf{k} \cdot \mathbf{r}) \, dx \, dy \, dz \quad (6.72)$$

where the approximation

$$\int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \frac{\text{erfc}(\omega|\mathbf{r}|)}{|\mathbf{r}|} \exp(-i\mathbf{k} \cdot \mathbf{r}) \, dx \, dy \, dz \approx \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{\text{erfc}(\omega|\mathbf{r}|)}{|\mathbf{r}|} \exp(-i\mathbf{k} \cdot \mathbf{r}) \, dx \, dy \, dz \quad (6.73)$$

can be used when ω is sufficiently large, yielding the analytical approximation

$$\int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \frac{\text{erfc}(\omega|\mathbf{r}|)}{|\mathbf{r}|} \exp(-i\mathbf{k} \cdot \mathbf{r}) \, dx \, dy \, dz \approx \frac{4\pi}{|\mathbf{k}|^2} \exp(-|\mathbf{k}|^2/4\omega^2) \quad (6.74)$$

and the focus now turns to evaluating the following integral

$$\int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \frac{\exp(-i\mathbf{k} \cdot \mathbf{r})}{|\mathbf{r}|} \, dx \, dy \, dz \quad (6.75)$$

Unlike 1-D, this integral is not representative of a well known function and initially proved to be something of an obstacle. It is important to note that this integral is completely independent of the decay parameter ω , that only positive components of \mathbf{k} need to be considered and it is independent of the Fourier series region of validity due to the arguments previously outlined.

Determining (6.75) must be done numerically. Initially, this was done *via* the use of Gauss type quadrature product rules using the Legendre roots and weights. The

integrals were evaluated once for k components up to order 120 to give a total of 302,620 distinct integrals. These were stored in a file which was read into an array at the beginning of a calculation. Memory requirements were not stretched as a result of such an approach, but the extra overhead of storing such integrals in a formatted ascii file was significant, and although this could have been reduced by creating an analogous binary file, the requirement of a large data file for KWIK calculations is not desirable and thus a method to obtain such integrals quickly, and as required, was necessary.

A derivation of how to evaluate these difficult integrals (6.75) numerically [146] is presented in the Appendix.

6.4.4. Gibbs Phenomenon

The KWIK algorithm used in the statistics problem (Chapter Four) required the scaling of the sums (A_i) so that they were well within the region over which the Fourier series was defined. The scaling was required because the Fourier series gives a poor representation of the function close to discontinuities, which is known as the Gibbs phenomenon. This is illustrated in Figure 6.6 for the unit step function.

Given the rapid convergence of the series for the probability, it was important to make sure that the scaling was sufficient as the extent of the Gibbs phenomenon is greater for smaller expansions. The extent of the Gibbs phenomenon decreases with increasing Fourier expansion, its amplitude however, does not decrease with increasing the degree of expansion.

The Fourier representation of the KWIK long-range operator does not contain discontinuities and therefore does not exhibit the Gibbs phenomenon and thus scaling is not required. There is, however, a discontinuity in the first derivative of the Fourier series representation of the long-range operator which may cause problems in applications other than determining total energies. These features are illustrated in the Fourier Series representation of the erfc long-range operator in Figure 6.7.

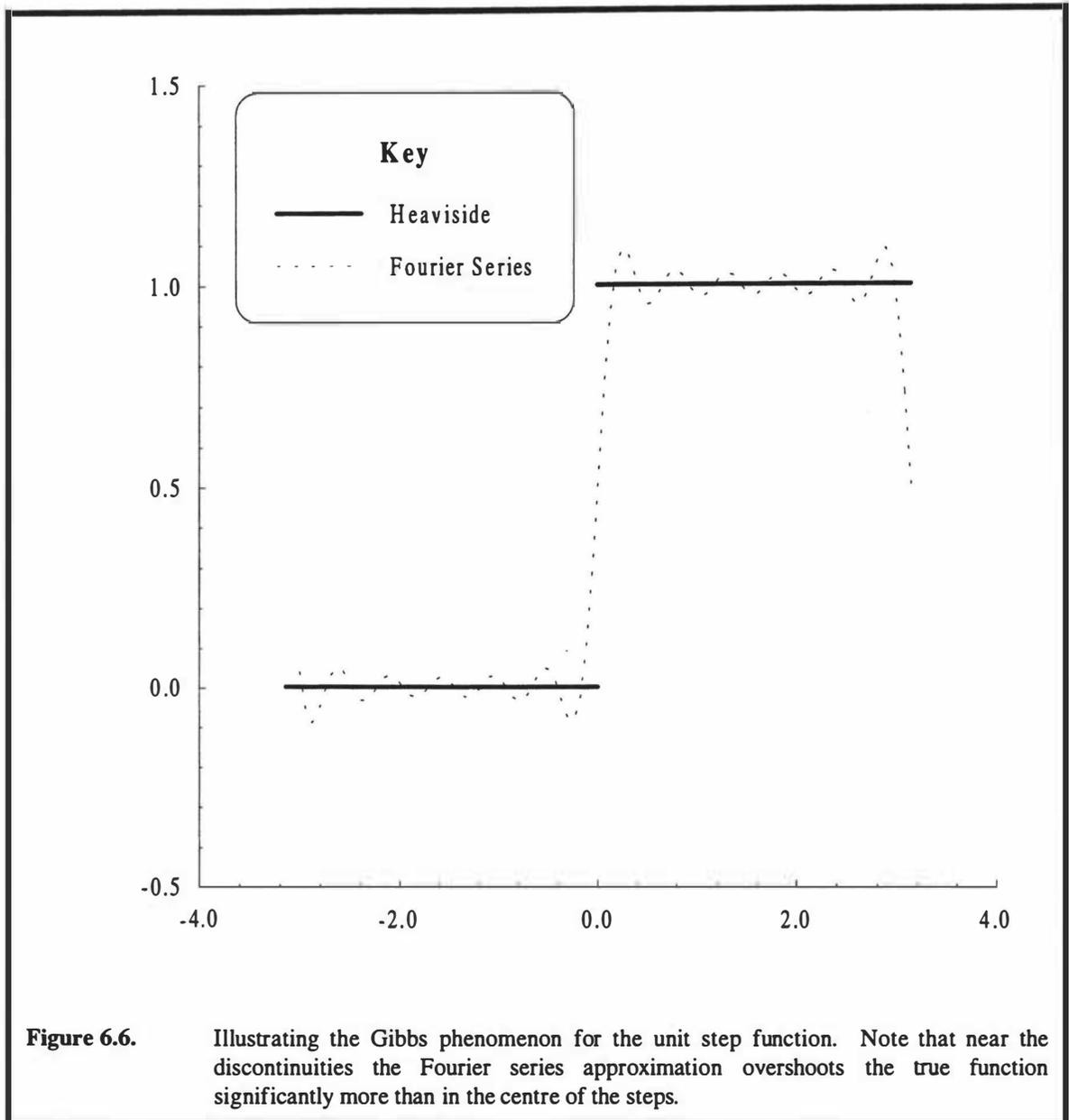
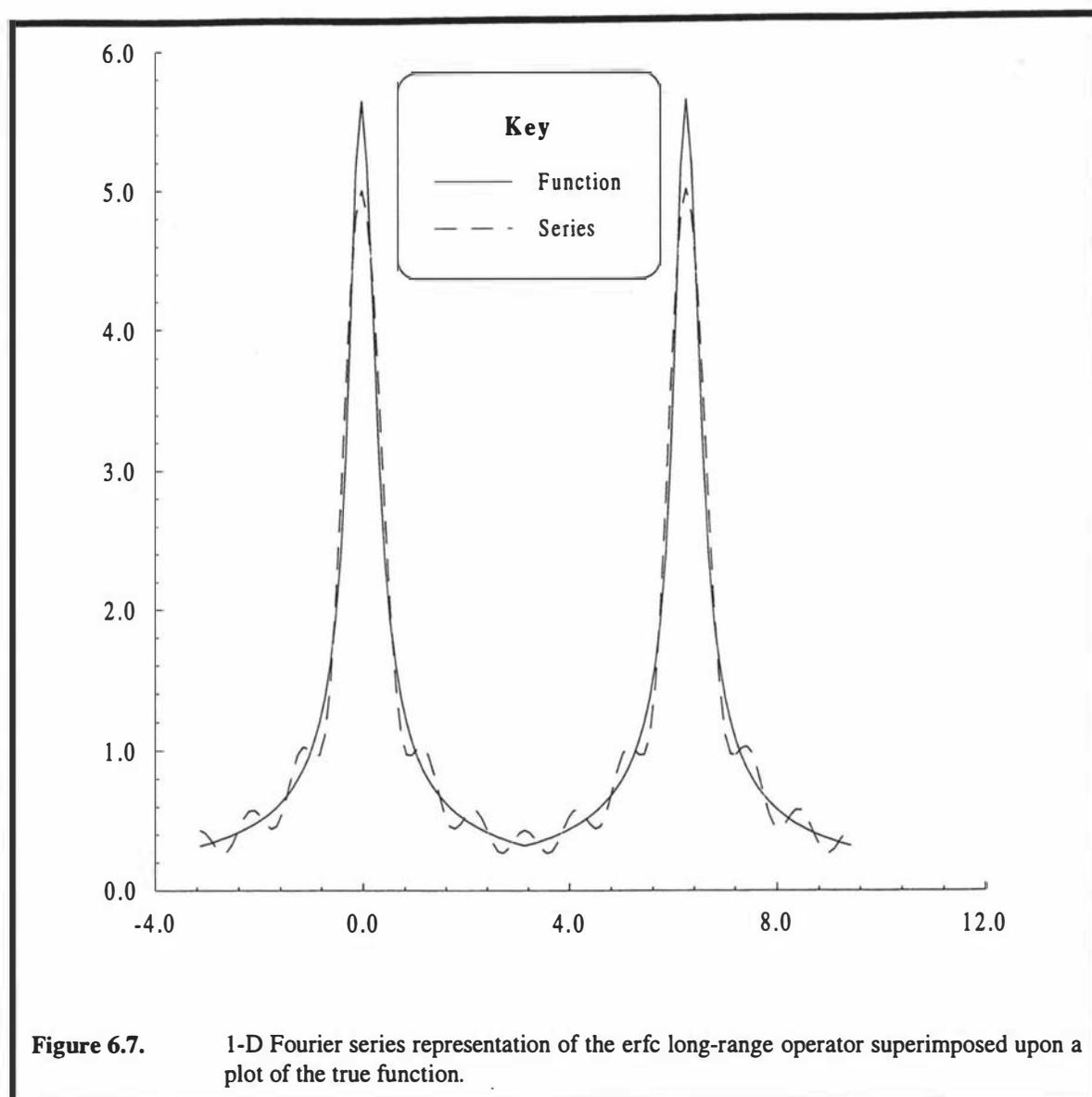


Figure 6.6. Illustrating the Gibbs phenomenon for the unit step function. Note that near the discontinuities the Fourier series approximation overshoots the true function significantly more than in the centre of the steps.



6.5. Summary

The long-range operator partition of the KWIK algorithm has been cast into two major forms - that of its Fourier transform and that of a Fourier series. The Fourier transform approach may lead to a more efficient methodology for accurately determining the long-range partition, but efforts in this work and in related approaches such as COP's and ROP's, have yet to indicate this potential. It is possible that crude approximations of the long-range energy may suffice in some applications and thus the Fourier transform may be a better starting point for such treatments.

The Fourier series representation of the long-range partition has been successfully implemented and offers the simple advantages of being both a well-defined approach in

all dimensions and having coefficients (weights) which are easily calculable. Computational advantages of the equi-spaced 'roots' of such an approach are significant and will be discussed in the next section.

CHAPTER SEVEN

APPLICATION OF KWIK TO DISTRIBUTIONS OF PARTICLES

7.0. Introduction

While the intention of the work undertaken for this thesis was to develop a new method for evaluating the inverse-square-law terms in self-consistent field calculations, the development of such a method can benefit greatly by considering the simplest possible generalisation. In doing so, the complexities of the final application are removed and one can concentrate solely on optimising the important characteristics of the algorithm. Hence, in the first instance, we consider the case of points charges rather than diffuse distributions of charge.

In Greengard's thesis [116], the Fast Multipole Method is developed by considering particles in a two dimensional world. The development of KWIK has begun at an even simpler level, by considering that of collinear unit point charges. In this Chapter, the equations for the application of KWIK to collinear point charges are derived followed by consideration of the linearity of the method for charged and neutral systems, and the effect of non-uniform distributions on the algorithm. This Chapter also aims to practically illustrate a number of theoretical concepts previously introduced.

7.1. Particles in One Dimension

The simplest generalisation of Coulomb interactions between charged distributions is that of collinear point charges. In contrast to Greengard's [116] derivation of the FMM, where he used the approach of a 2-D universe as a first generalisation, the 3-D Coulomb interaction in the special case where the charge distributions are collinear and points, has been chosen. Greengard's justification of his special case system was that "Many physical processes are adequately described by two-dimensional models ...". The justification here is that the interest lies with learning how the algorithm works so to obtain the best MQM implementation. It was considered

likely that this could be best achieved by first considering the case where we have linear molecules.

7.1.1. Derivation

The Coulomb energy of a system of N charged particles is given by

$$E = \sum_{l=1}^N \sum_{j<l}^N \frac{Q_l Q_j}{r_{lj}} \quad (7.1)$$

where Q_j is the charge on the j^{th} particle and r_{lj} is the inter-particle separation. Applying the Coulomb operator separation

$$E = E_{short} + E_{long} \quad (7.2)$$

$$E = \sum_{l=1}^N \sum_{l<j}^N \frac{Q_l Q_j \operatorname{erfc}(\omega r_{lj})}{r_{lj}} + \sum_{l=1}^N \sum_{l<j}^N \frac{Q_l Q_j \operatorname{erf}(\omega r_{lj})}{r_{lj}} \quad (7.3)$$

which for convenience can be rewritten

$$E = \sum_{l=1}^N \sum_{l<j}^N \frac{Q_l Q_j \operatorname{erfc}(\omega r_{lj})}{r_{lj}} + \frac{1}{2} \sum_{l=1}^N \sum_{j=1}^N \frac{Q_l Q_j \operatorname{erf}(\omega r_{lj})}{r_{lj}} - E_{SIC} \quad (7.4)$$

where the last term is the self interaction correction which, as written, is exactly

$$E_{SIC} = \sum_{j=1}^N \frac{Q_j^2 \omega}{\sqrt{\pi}} \quad (7.5)$$

By substituting the long-range operator by its Fourier series representation

$$\frac{\operatorname{erf}(\omega r_{lj})}{r_{lj}} = \sum_{k=-\infty}^{\infty} A(k, \omega) \exp(ik(r_i - r_j)) \quad (7.6)$$

where $A(k, \omega)$ are the associated Fourier coefficients, yielding

$$E_{long} = \frac{1}{2} \sum_{l=1}^N \sum_{j=1}^N \sum_{k=-\infty}^{\infty} A(k, \omega) Q_l Q_j \exp(ik(r_l - r_j)) - E_{SIC} \quad (7.7)$$

Inverting the sums and factorising the exponential function

$$E_{long} = \frac{1}{2} \sum_{k=-\infty}^{\infty} A(k, \omega) \sum_{j=1}^N Q_j \exp(ikr_j) \sum_{l=1}^N Q_l \exp(-ikr_l) - E_{SIC} \quad (7.8)$$

implies that E_{long} can be approximated to a required accuracy by

$$E_{long} \approx \frac{1}{2} \sum_{k=0}^M A(k, \omega) I(k) \quad (7.9)$$

where the Fourier coefficients have been adjusted for symmetry, M and ω are carefully chosen and

$$I(k) = \left| \sum_{j=1}^N Q_j \exp(ikr_j) \right|^2 \quad (7.10)$$

is the Fourier intensity of the total charge distribution. The self interaction correction now becomes

$$E_{SIC} = \sum_{j=1}^N Q_j^2 \sum_{k=0}^M A(k, \omega) \quad (7.11)$$

whose value is a poor approximation of (7.5) due to the relatively slow convergence of the Fourier series. With optimistic reference to the final four letters in (7.9) the algorithm is referred to as a KWIK algorithm.

7.1.2. Short-Range

7.1.2.0. Introduction

The evaluation of short-range interactions has been covered in detail in Chapter Five. Presented here are the practical aspects of the implementation. The first practical issue addressed is that of computing the inter-particle potential efficiently. This is followed by discussions on the determination of the point at which interactions become insignificant, memory requirements of 1-D short-range procedures, and a final note on distribution dependence.

7.1.2.1. Interpolation

The difference in determining the interaction potential for collinear particles and 3-D scattered particles lies only in a determination of the inter-particle separation. In 3-D, extra multiply-adds (MA's) and a square root are required. Otherwise the interpolation arguments presented here apply equally to both 1-D and 3-D.

The basic interaction to be calculated, once an inter-particle distance has been determined, is simply

$$\frac{\text{erfc}(\omega r_{12})}{r_{12}} \quad (7.12)$$

which requires a divide, a multiply and an erfc. Unfortunately, divides [133] are computationally expensive and are avoided, where possible when devising new efficient algorithms. The divide cannot be avoided in this case. Multiplies and adds have a (floating point operation) FLOP cost of one and an independent MA *i.e.*.

$$y = a * x + b \quad (7.13)$$

can also be evaluated in one cycle on RISC architectures [61]. The erfc function, however, was found, not surprisingly, to be computationally expensive relative to the more optimised trigonometric functions in the standard mathematical libraries.

In MQM, the basic integral from which higher integrals and derivatives are formed is a function involving an exponential function. Gill [128] uses a highly efficient interpolation scheme to evaluate the exponential. This approach has been adopted for evaluating erfc. The interpolation scheme described by Gill, and that implemented for purposes here, uses a cubic Chebychev interpolating polynomial

$$f(x) \approx a_0 T_0(t) + a_1 T_1(t) + a_2 T_2(t) + a_3 T_3(t) \quad (7.14)$$

where $T_k(t)$ is the k^{th} Chebychev polynomial,

$$t = \frac{x - X}{\Delta} \quad (7.15)$$

is a normalised variable and Δ is the greatest range of interpolation for some predetermined accuracy ε , given by

$$\Delta = \left[\frac{2^n (n+1)! \varepsilon}{\max |f^{(n+1)}(X)|} \right]^{1/(n+1)} \quad (7.16)$$

where the interpolation interval is $(X - \Delta, X + \Delta)$. The Chebychev polynomial is obtained as a least squares polynomial [128] from the Taylor expansion of the interpolating function at the nodes of interpolation X_j whereby the coefficients are expressed as

$$a_k = (2 - \delta_{k0}) \left(\frac{\Delta}{2} \right)^k \sum_{m=0}^{\infty} \left(\frac{\Delta}{2} \right)^m \frac{f^{(k+2m)}(X)}{m!(k+m)!} \quad (7.17)$$

which is a function of Δ and derivatives of the interpolated function. The derivatives of the error function can be conveniently determined using the Hermite polynomials [53].

The advantage in using the Chebychev polynomial is that the error is 'spread' over the whole region *c.f.* the Taylor polynomial, which is very accurate at the nodes and becomes poorer near the end-points. In computing $\text{erfc}(x)$, the same interpolation grid as used by Gill was chosen

$$X_j = (2j+1)\Delta \quad (7.18)$$

which allows the index j to be identified easily

$$j = \text{INT} \left(\frac{x}{2\Delta} \right) \quad (7.19)$$

so that the function can be evaluated to the predetermined accuracy in 3 FMA's *via* Gill's formula

$$f(x) \approx f_0 + \frac{x}{2\Delta} \left(f_1 + \frac{x}{2\Delta} \left(f_2 + \frac{x}{2\Delta} f_3 \right) \right) \quad (7.20)$$

This is illustrated in the partial FORTRAN code in Scheme 7.1.

```

R2DLTA = OMEGA / (TWO * DELTA)
.
.
.
SUME = ZERO
DO J=1,L
  RIJ = SQRT((R(1,i)-R(1,j))**2 +
             (R(2,i)-R(2,j))**2 +
             (R(3,i)-R(3,j))**2)
  Y = RIJ * R2DLTA
  K = INT(Y)
  ERFCWR = T(1,k) + Y*(T(2,k) + Y*(T(3,k) + Y*T(4,k)))
  SUME = SUME + Q(j) * ERFCWR
ENDDO
E = E + Q(i) * SUME
.
.
.

```

Scheme 7.1. Partial FORTRAN code illustrating the use of interpolating the erfc function in a 3-D simulation. The array T contains the interpolating constants f_k which are a function of the polynomial coefficients.

The effectiveness of this scheme is presented in Table 7.1 which illustrates the efficiency of using interpolation to evaluate the erfc function in three important situations. Note that interpolation is considerably faster than the standard mathematical libraries where savings of 77% are made. In cases encountered in evaluating the short-range energy partition in KWIK calculations, considerable savings are also obtained. Note the reduction in improved timings is due to the divides, and the square root, in the 3-D case.

	Standard	Interpolation	% Improve
Function	1.38	0.32	77
1-D	1.63	0.54	67
3-D	2.12	1.02	52

Table 7.1. CPU times (seconds) for evaluating 400,000 interactions in the interval (0,4) involving the erfc function. Key: Function - denotes evaluating the erfc function itself; 1-D - the one-dimensional interaction potential; 3-D - the three-dimensional interaction including a square root; Standard - is the time for evaluating the same potential using the standard mathematical library.

7.1.2.2. Cut-offs

In order to obtain a short-range energy to within an error of some given ϵ , in work that scales $O(N)$, we must determine the minimum cut-off radius, R_C , so that the sum of all interactions outside R_C , is less than ϵ for all distributions. Thus we require

$$\int_{R_c}^{\infty} \frac{\operatorname{erfc}(\omega x)}{x} dx \leq \varepsilon \quad (7.21)$$

While the primary interaction becomes insignificant, *i.e.*

$$\frac{\operatorname{erfc}(\omega R_c)}{R_c} \leq \varepsilon \quad (7.22)$$

before the cut-off satisfying (7.21), and was found to be satisfactory in preliminary investigations, (7.21) is a more rigorous upper bound than that satisfying (7.22).

7.1.2.3. Memory Requirements

Implemented were two generic short-range codes for collinear particles. The first used an ordering of particles within boxes so that binary searches can be employed, thus reducing the number of insignificant interactions evaluated. The second does not. In each case particles are sorted into respective boxes, irrespective of whether the particles are sorted within the boxes themselves.

The sorting routine is the most memory intensive of the short-range code. This is depicted in Scheme 7.2. The double precision $R(2, N)$ array contains the charge and particle position, where N is the number of particles. $ORD(N, 2)$ is a scratch array in which the particles are temporarily ordered into boxes (Scheme 5.2). The change in the way the scratch array is dimensioned is also necessary for the sorting within boxes, if required. The $INBOX$ array is another scratch array used to hold data for describing which box the j^{th} particle is in. $ICOUNT$ indicates the number of particles in the k^{th} box and $IPOINT$ is a pointer into the ORD (and a reordered R) array, to indicate the first particle in the k^{th} box. The variable $NBOX$, refers to the total number of boxes. Thus, the total memory requirements are roughly $5N-6N$ double precision words.

```
REAL*8  ORD(N, 2) , R(2, N)
INTEGER ICOUNT(NBOX) , IPOINT(NBOX) , INBOX(N)
```

Scheme 7.2. Illustration of memory requirements for the one dimensional sorting process.

The code for determining the short-range interactions requires the interpolation table, the ordered particle coordinates and charge (R), the pointer ($IPOINT$) into the

coordinate array and the counter arrays (ICOUNT). Total memory requirements are thus roughly $3N$ double precision words.

7.1.2.4. Distribution Dependence

In calculations using the generic codes described in section 7.1.2.3, the code using the binary search without box division, and thus affording theoretical efficiencies of 1.0, was always faster than codes with box division and/or without the binary search in the case of uniform distributions of charge. However, this was not the case for normally distributed particles. Table 7.2 contains illustrative timings for a KWIK short-range interaction calculation using the respective generic codes for a normally distributed set of particles and a uniformly distributed set of particles.

Distribution Division	Uniform		Normal	
	Ordered	Non-Ordered	Ordered	Non-Ordered
1	74.56	93.60	260.23	270.17
2	75.41	91.42	239.87	264.49
3	74.24	92.47	232.55	264.15
4	75.45	92.22	230.69	266.24
5	76.40	93.56	226.56	265.16

Table 7.2. CPU times (seconds) for a short-range KWIK calculation ($\omega=0.008$) on distributions of 100,000 unit point charges. Ordered implies a binary search has been used and division indicates the level of cell reduction.

In Table 7.2 it can be seen that when the normally distributed particles (obtained from using the `gaudev` [136] routine and scaled so that 10 standard deviations were within the region $(0,N)$) are investigated, considerable improvements in total time can be achieved by division of boxes and employing the binary search. Note that in other cases the division of boxes does not overly improve performance.

7.1.3. Long-Range

7.1.3.0. Introduction

The underlying theory has been developed in some detail in Chapter Six. However, unlike the short-range chapter, Chapter Six was concerned more with devising a generic scheme so that the required accuracy could be achieved by a sufficient expansion, as well as being extendable to higher dimensions. This contrasts somewhat with the short-range chapter which was concerned with maximising efficiency.

7.1.3.1. MOP's vs FLOP's

The true measure of the performance of an algorithm is best measured in CPU time, but is dependent on comparatively efficient implementations. Hence, a theoretical measure is often more useful. A theoretical measure requires not only consideration of the numbers of floating point operations (FLOP's) but also of memory operations (MOP's) [147]. FLOP's are multiplies, adds etc (usually one considers divides as a single FLOP also). A MOP is a retrieval or placement to and from memory. In our implementation of long-range KWIK (on the scalar machine) we have obtained decreased FLOP and MOP counts by the use of recursion and unrolling loops (Scheme 7.3). The recursion formula is simply

$$\exp(i(k + 1)r_j) = \exp(ir_j) * \exp(ikr_j) \quad (7.23)$$

Thus only the first term on the RHS of equation (7.23) need be evaluated, and all subsequent k terms can be obtained simply as a complex multiply (2 Multiply-Add's and 2 Multiply's) - significantly cheaper than evaluating a complex exponential. RISC technology currently allows an MA to be evaluated in a single machine cycle, the latest technology will perform 2 Multiply's and an Add in a single machine cycle. As an aside, care should be taken in determining the complex exponentials as some compilers don't recognise the pure imaginary status of the argument and evaluate an additional real exponential. In this case the complex exponential is best computed as its trigonometric components.

7.1.3.2. Memory Requirements

Memory requirements of 1-D long-range KWIK are $6N$ double precision words. $2N$ for particle charges and coordinates, $2N$ for the base complex exponential array (F) for recursion and a final $2N$ for the recursion update array (G). These are reasonable, given the gains in speed to be presented (*c.f.* with quadratic methods), and could be reduced to $2N$, by removal of recursion. The effect would be a loss in speed which would be relative to the efficiency of the platforms trigonometric functions, but would not effect the scaling of the algorithm.

```

REAL*8 R(2,N)
COMPLEX*16 A,TP,FC,F(N),G(N),S(5)
.
.
.
DO i=1,N
  A = DCMLPX(ZERO,R(2,i))
  F(i) = EXP(A)
  G(i) = DCMLPX(ONE,ZERO)
ENDDO
.
.
.
DO k=1,M-4,5
  S(1) = DCMLPX(ZERO,ZERO)
  .
  .
  DO i=1,N
    FC = F(i)
    CG = R(2,i)
    TP = G(i)*FC
    S(1) = CG*TP + S(1)
    TP = TP*FC
    S(2) = CG*TP + S(2)
    TP = TP*FC
    S(3) = CG*TP + S(3)
    TP = TP*FC
    S(4) = CG*TP + S(4)
    G(i) = TP*FC
    S(5) = CH*G(i) + S(5)
  ENDDO
.
.
ENDDO
.
.
.

```

Scheme 7.3. Illustrating the use of recursion to save FLOP's and unrolling the k loop to save on MOP's. Unrolling beyond 5 gave little improvement in speed-up on the IBM.

7.1.3.3. Cut-offs

As the extent of the total distribution increases with N , the density of k vectors decreases. Thus, to maintain accuracy the number of k vectors must also increase with N , *i.e.* for maintaining a constant ω for increasing N . This would imply cost scaling as $O(N^2)$. However, if we adjust ω to increase the short-range work, cost scaling is reduced to our empirically found value of $O(N^{1.5})$, $x \sim 1.4$. Similar cost scaling behaviour has been identified for the Ewald and PPPM methods [105,112]. However, if the required accuracy ϵ , and the size of the charged distribution N are related by

$$N \gg \frac{1}{\epsilon} \quad (7.24)$$

then error estimates are aided significantly by the random walk previously described. This will be elaborated further in section 7.1.4.2.

7.1.4. Charged Systems

7.1.4.0. Introduction

A charged system of particles, that is, a system of charged particles of the same sign interacting Coulombically, is analogous to any particle system interacting *via* an inverse-square-law, gravitational forces, for example. The energy of such an interaction will converge for infinite systems if and only if the interaction of the type r^{-p} , has a $p > 1$. The charged systems considered in these calculations are all finite.

7.1.4.1. Illustrative Timings

The KWIK algorithm has been used to compute the Coulomb energies of a number of uniformly distributed collinear unit point charges over a variety of predetermined accuracies. Presented here are those timings previously reported [148], and a more recent study over a smaller range of system sizes. The latter serves to illustrate the recent improvements, such as, the use of interpolation and better screening techniques.

Uniform random deviates for the interval (0,N) were generated by utilising the FORTRAN 77 subroutine `ran1` [136] and multiplying the resulting uniform random deviate by N to yield a distribution of particles with a mean separation of unity (Figure 7.4). Timings were optimised by determining the decay parameter, ω , and degree of Fourier expansion M , such that the times for determining the short and long-range components of the KWIK Coulomb energy were approximately the same. This was found empirically to be the optimal split. Timings are tabulated in Table 7.3.

N	10 ⁻²	10 ⁻⁴	10 ⁻⁶	10 ⁻⁸	10 ⁻¹⁰
100,000	1.59	15.35	55.57	108.15	135.26
200,000	3.24	37.05	165.26	295.64	374.75
300,000	4.99	57.00	295.35	532.74	655.78
400,000	6.79	84.89	396.31	723.45	982.05
500,000	8.35	110.08	488.01	1069.07	1389.17

Table 7.3. CPU times (seconds) for determining the Coulomb energy using KWIK for collinear, uniformly distributed, positive unit point charges for various predetermined accuracies.

Energy contributions and other details of the calculations for the timings in Table 7.3 are tabulated in the Appendix.

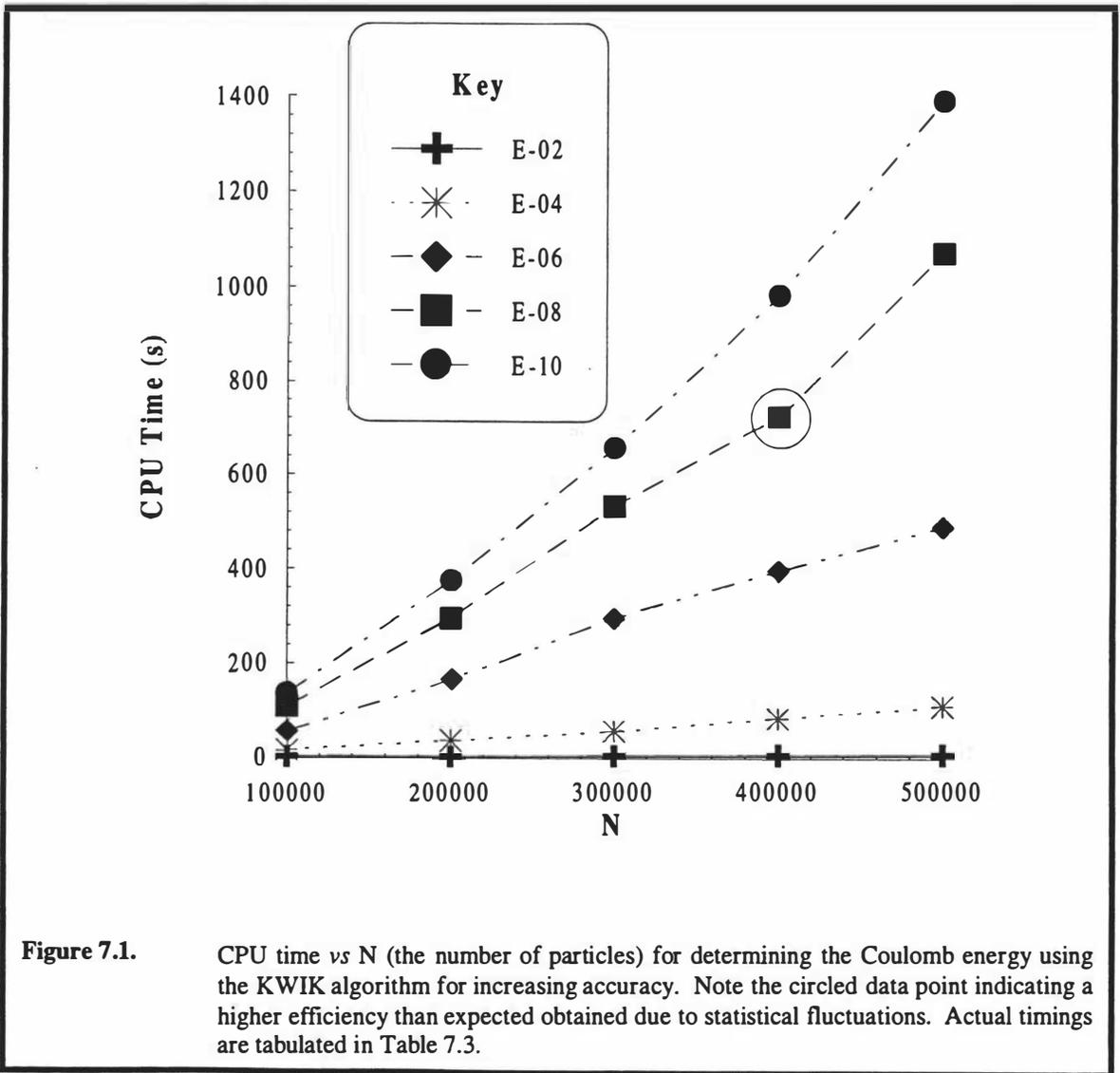


Figure 7.1 illustrates several important features of KWIK. Firstly, for sufficiently large N , where $N \gg 1/\epsilon$, the algorithm scales linearly as is illustrated by the $\epsilon=10^{-2}$ and

$\varepsilon=10^{-4}$ curves. Secondly, as N increases the algorithm tends towards linearity - for $\varepsilon=10^{-6}$ we see at $N=300,000$ the slope of the line decreases. Finally, for increasing accuracy, it is apparent that the amount of extra work required is dependent on N . If $N < 1/\varepsilon_1$, then the extra effort required is moderate (consider moving vertically from the x -axis at $N=200,000$). If $1/\varepsilon_2 > N > 1/\varepsilon_1$ then the extra effort is large (consider an increase inaccuracy from 10^{-6} to 10^{-8} for $N=500,000$). The extra effort required where $N \gg 1/\varepsilon_2 > 1/\varepsilon_1$ is small - consider an increase in accuracy from 10^{-2} to 10^{-4} for $N=500,000$

N	10^{-3}	10^{-5}	10^{-6}	10^{-8}	10^{-10}
1,000	0.5	0.5	0.6	0.7	0.8
5,000	1.8	2.8	3.1	3.7	4.2
10,000	3.6	5.2	6.5	8.7	9.7
50,000	18.5	46.2	48.3	69.7	85.2
100,000	37.2	92.8	121.6	179.1	219.6
500,000	199.1	560.9	778.3	1784.7	2488.6
1,000,000	414.2	958.1	1847.4	4763.3	-

Table 7.4. CPU times (seconds) for determining the Coulomb energy using KWIK for collinear, uniformly distributed, positive unit point charges for various predetermined accuracies. (As reported in [148]).

The timings in Table 7.4 should be compared (where possible) with Table 7.3 to illustrate the effect of incorporating interpolation, binary searches and numerical stabilisation in short-range code, and FLOP and MOP savings in long-range code. Note that for an accuracy of 10^{-10} and $N=1,000,000$ KWIK, had problems with numerical instability. KWIK is not a numerically unstable algorithm, the instability here was purely as a result of a poor implementation which has since been dramatically improved.

Quadratic timings are presented in Table 7.5. For other than the most trivial calculations with timings of a few seconds all the KWIK times are significantly (orders of magnitude) faster than their quadratic counterparts, offering a significant advantage for obtaining Coulomb energies without necessarily having to consider comparative cost scaling characteristics.

7.1.4.2. The Random Walk and $I(k)$

As described in Chapter Six, the function $I(k)$ can be likened to a drunkard taking unit steps on the complex plane. It was found that for large N , $I(k)$ is an exponential

variable with mean of N and standard deviation N . This is illustrated by considering $I(k)$ for large and small uniform distributions. The effect of polarised distributions on the Random Walk Advantage and the assumption of the random phase is also demonstrated.

N	Time (s)
1,000	0.15
5,000	4.15
10,000	18.00
50,000	750.00
100,000	3,000.00
200,000	12,000.00
300,000	27,000.00
400,000	48,000.00
500,000	75,000.00
1,000,000	260,000.00

Table 7.5. CPU times (seconds) for determining the Coulomb energy for various system sizes (N) using the classical double sum quadratic method.

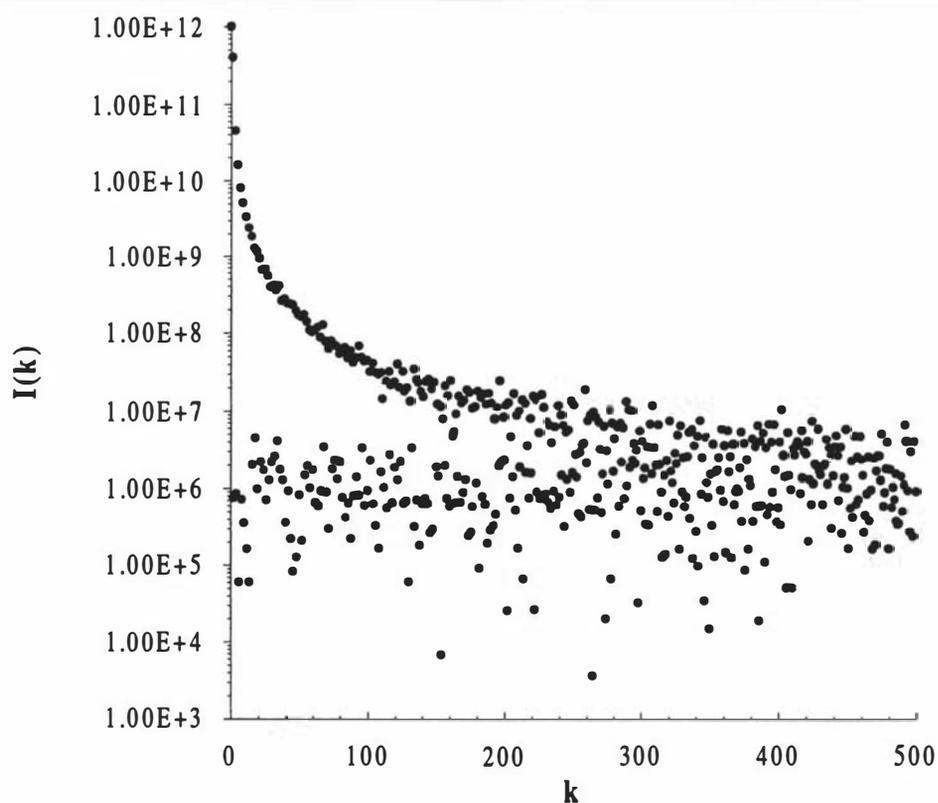


Figure 7.2. $I(k)$ vs k for discrete values of k for 1,000,000 collinear uniformly distributed, positive, unit point charges.

Figure 7.2 begins at the origin at 10^{12} (N^2) and decays extremely rapidly (note the logarithmic scale of the y-axis) to a noisy value of 10^6 (N). This can be rationalised in terms of the random walk, where the initial assumption of kr_j being a random phase is invalid (refer Chapter Six, section 6.3.5). But as k increases, the assumption becomes valid and the random walk can be seen to be a mean of N with standard deviation of N .

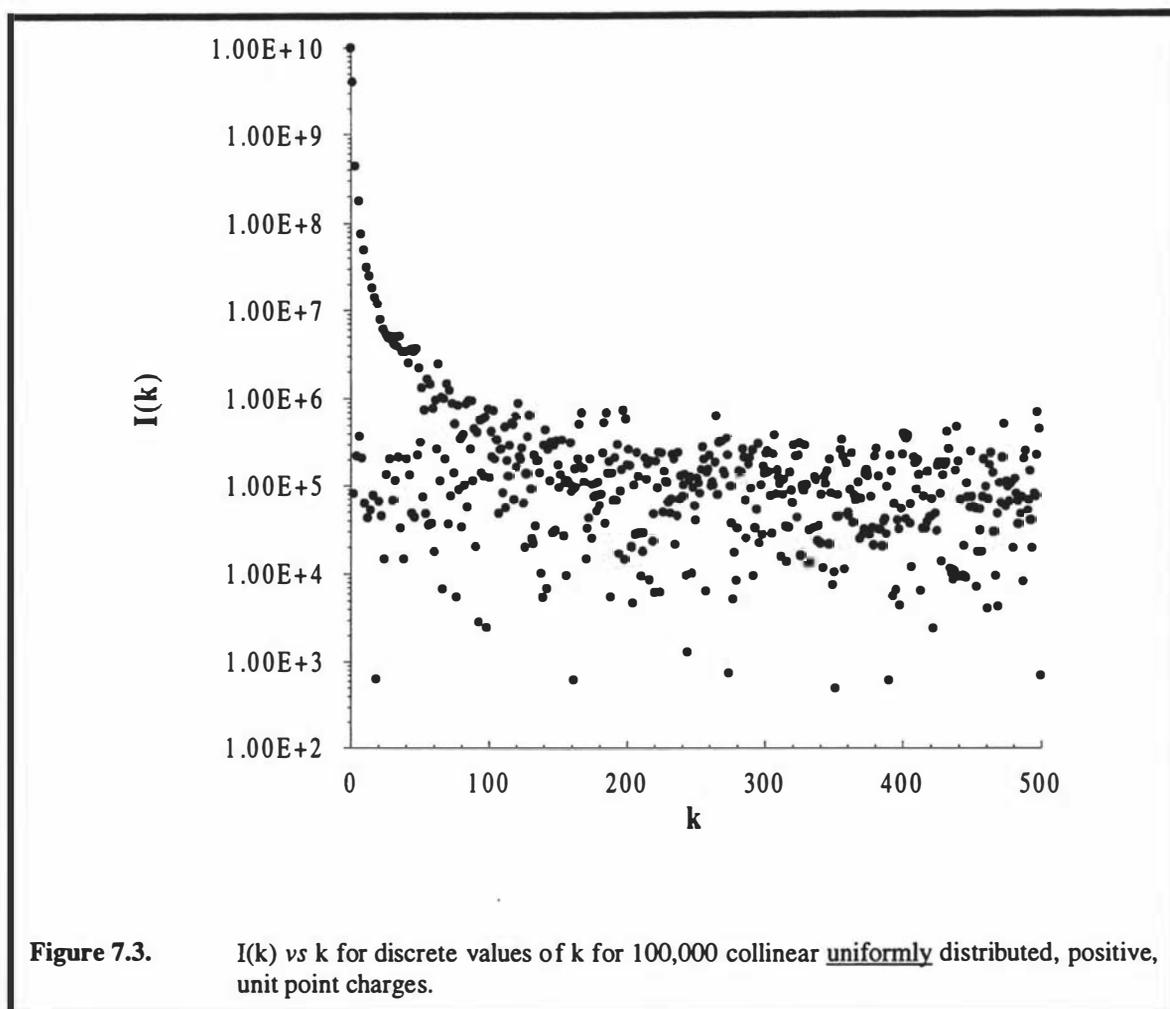
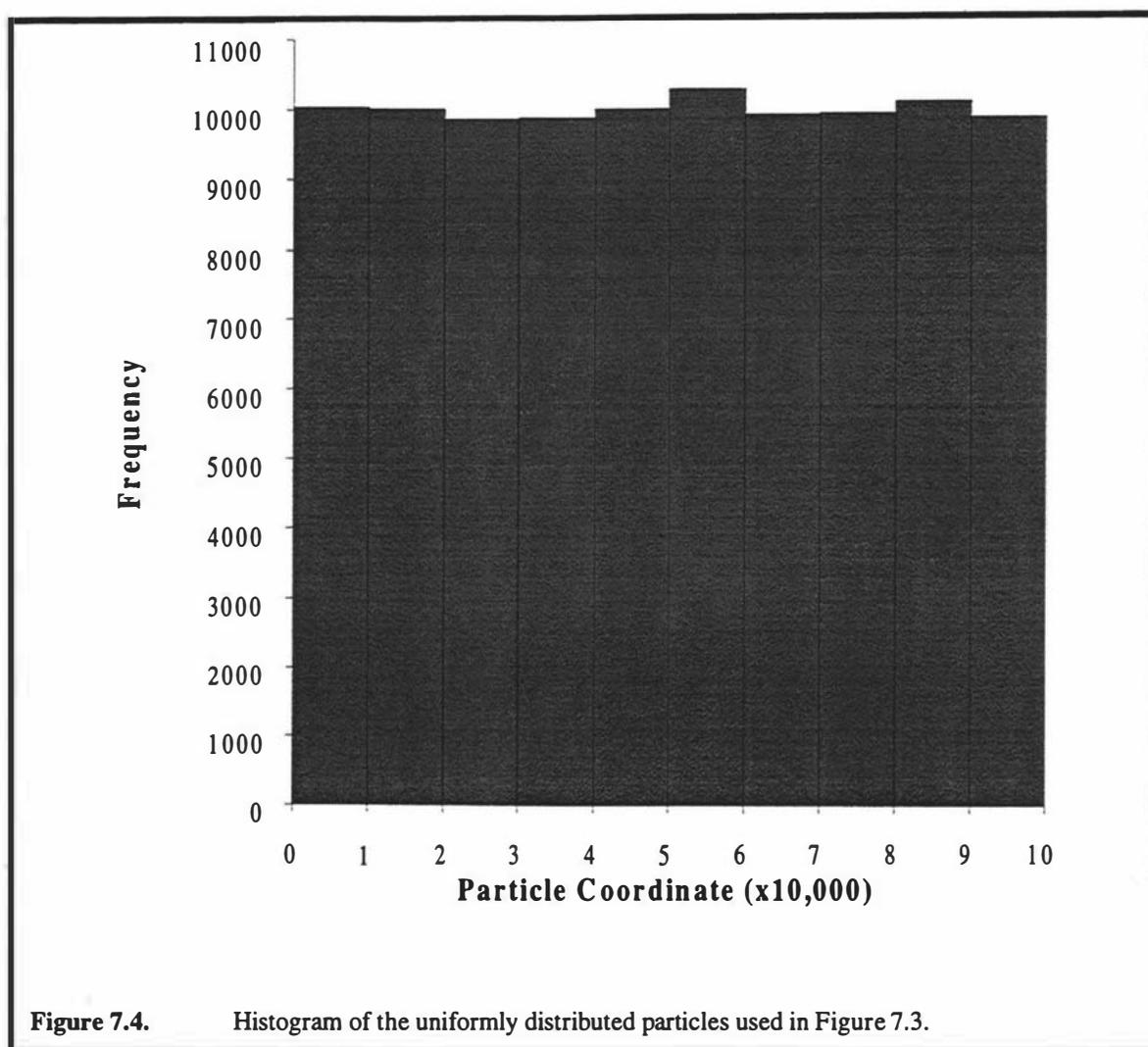


Figure 7.3 is analogous to Figure 7.2, but with fewer unit point charges. It is clear from these two plots that the rapid decay of $I(k)$ will aid significantly in cases where $N \gg 1/\epsilon$. For example, a calculation requiring an accuracy of 10^{-3} in both cases can be obtained for the long-range partition simply by increasing the Fourier series to around $k=20$. The distribution of particles used to form the data for Figure 7.3 is plotted in Figure 7.4.



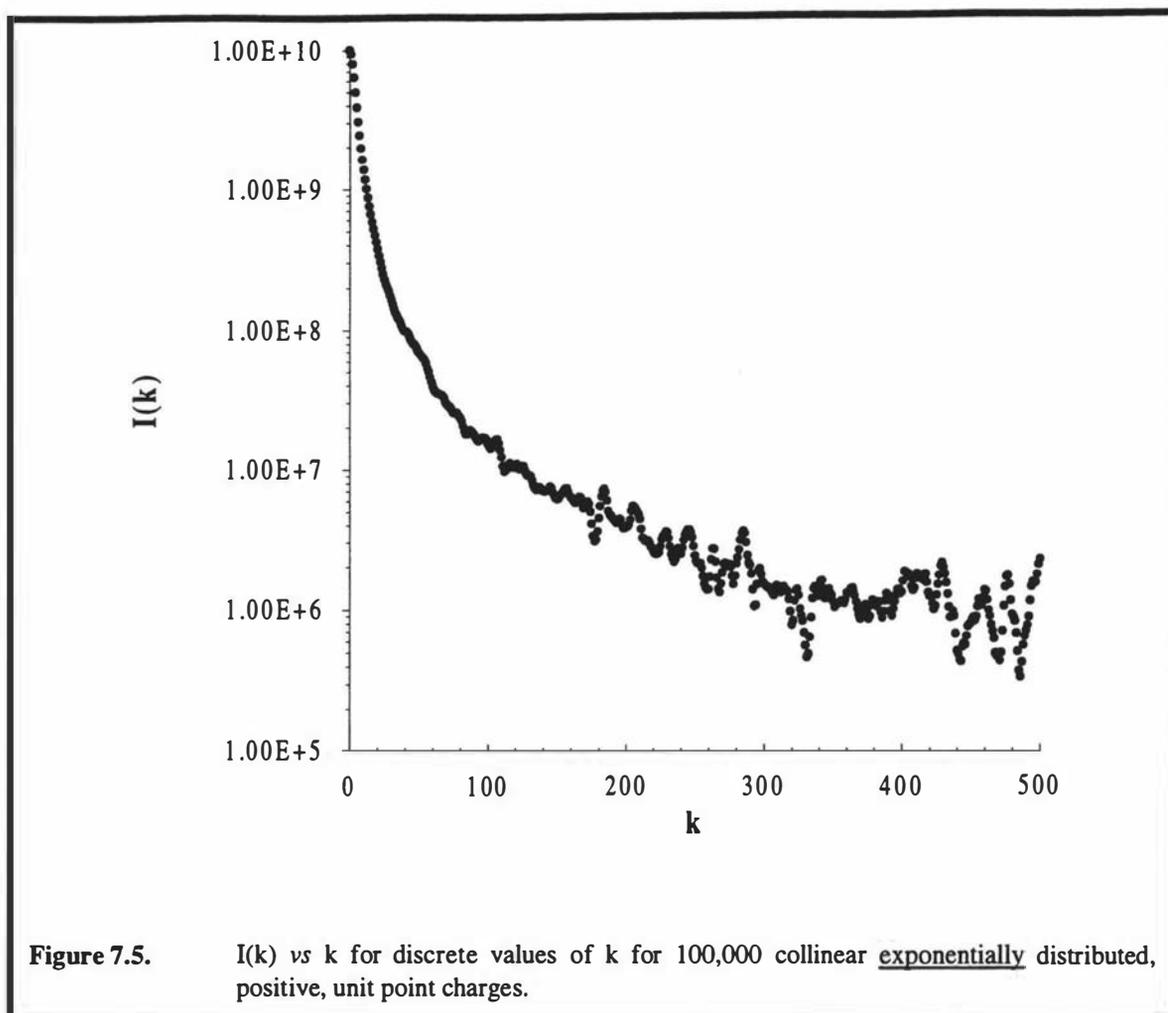
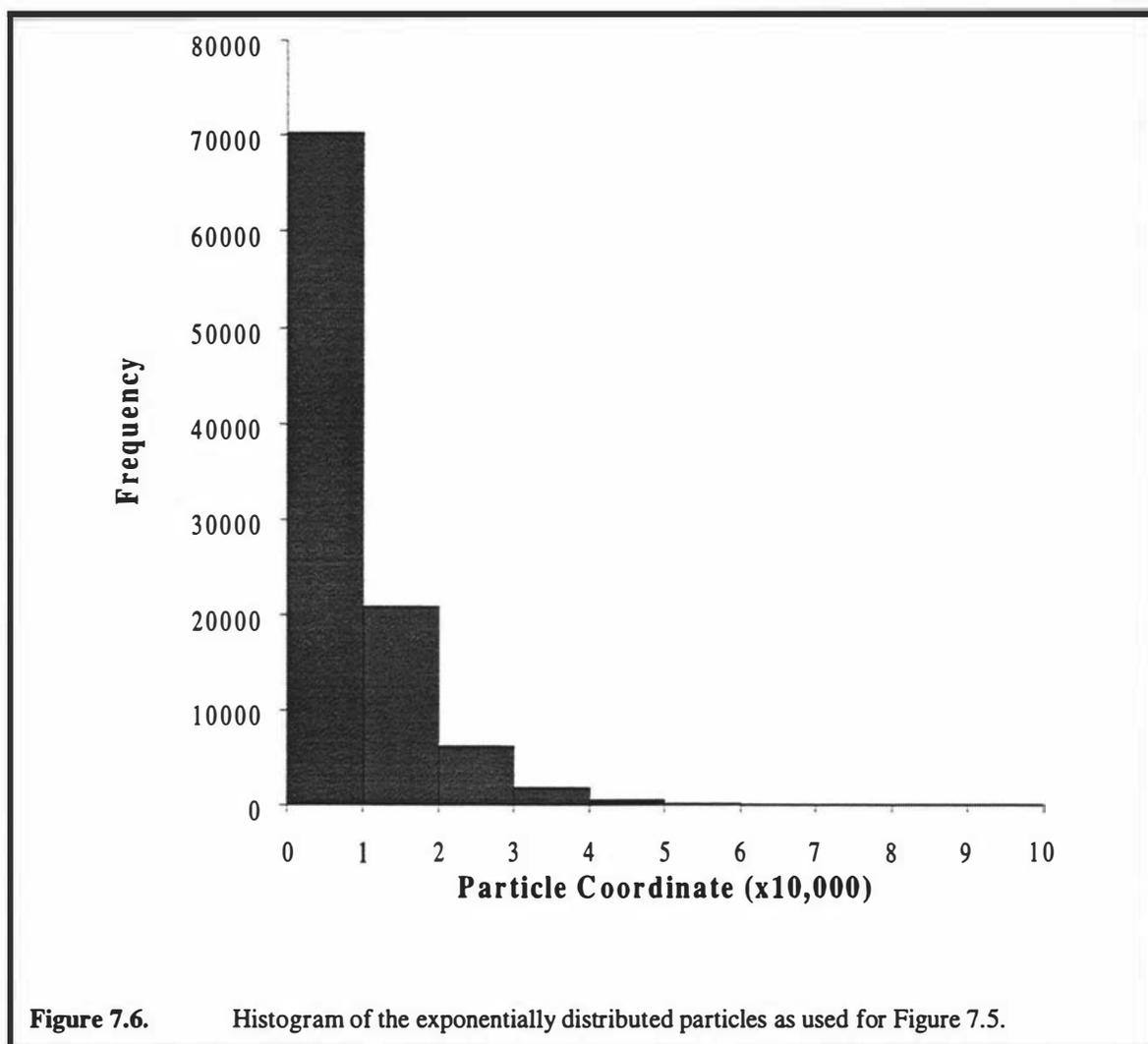
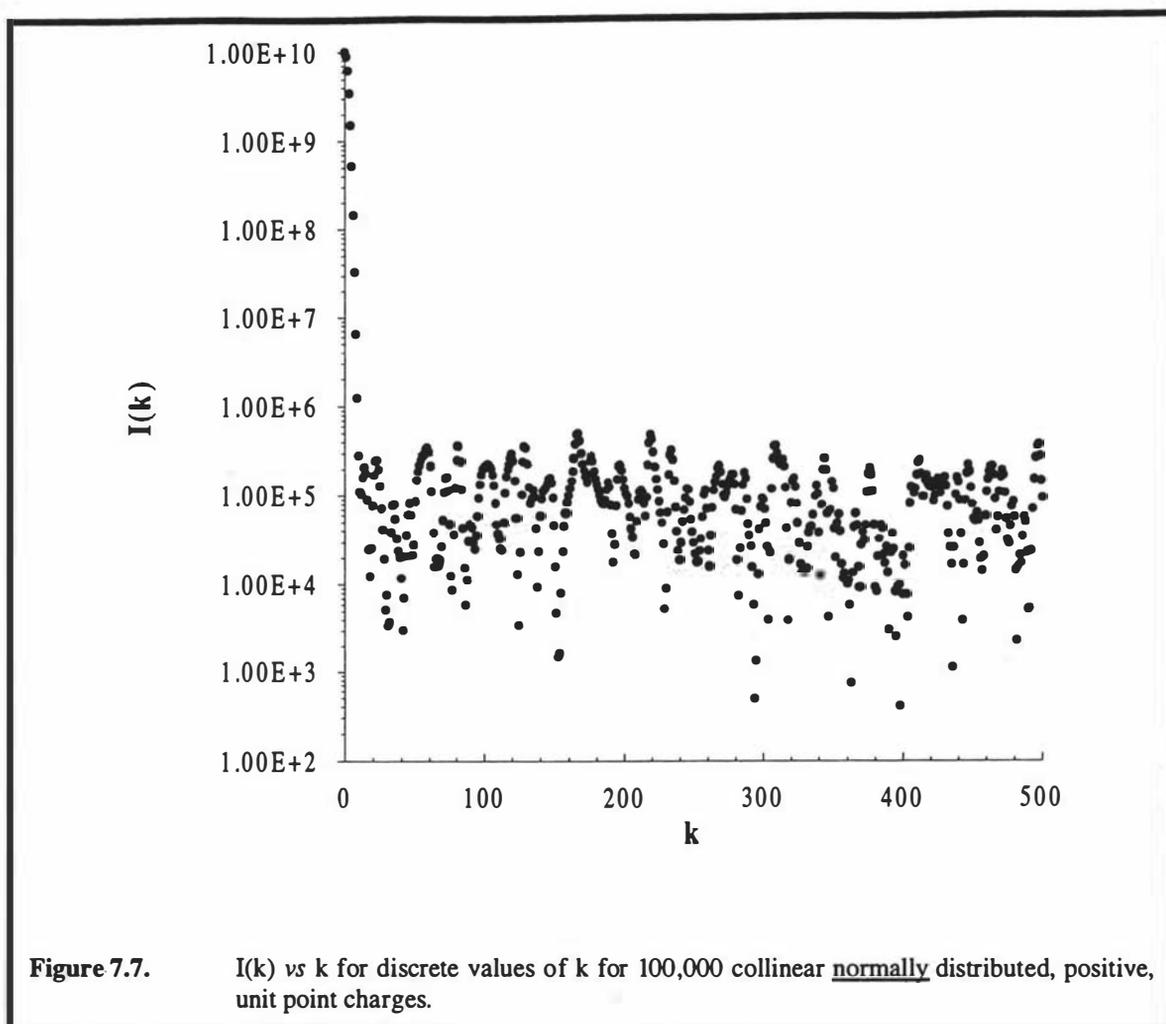
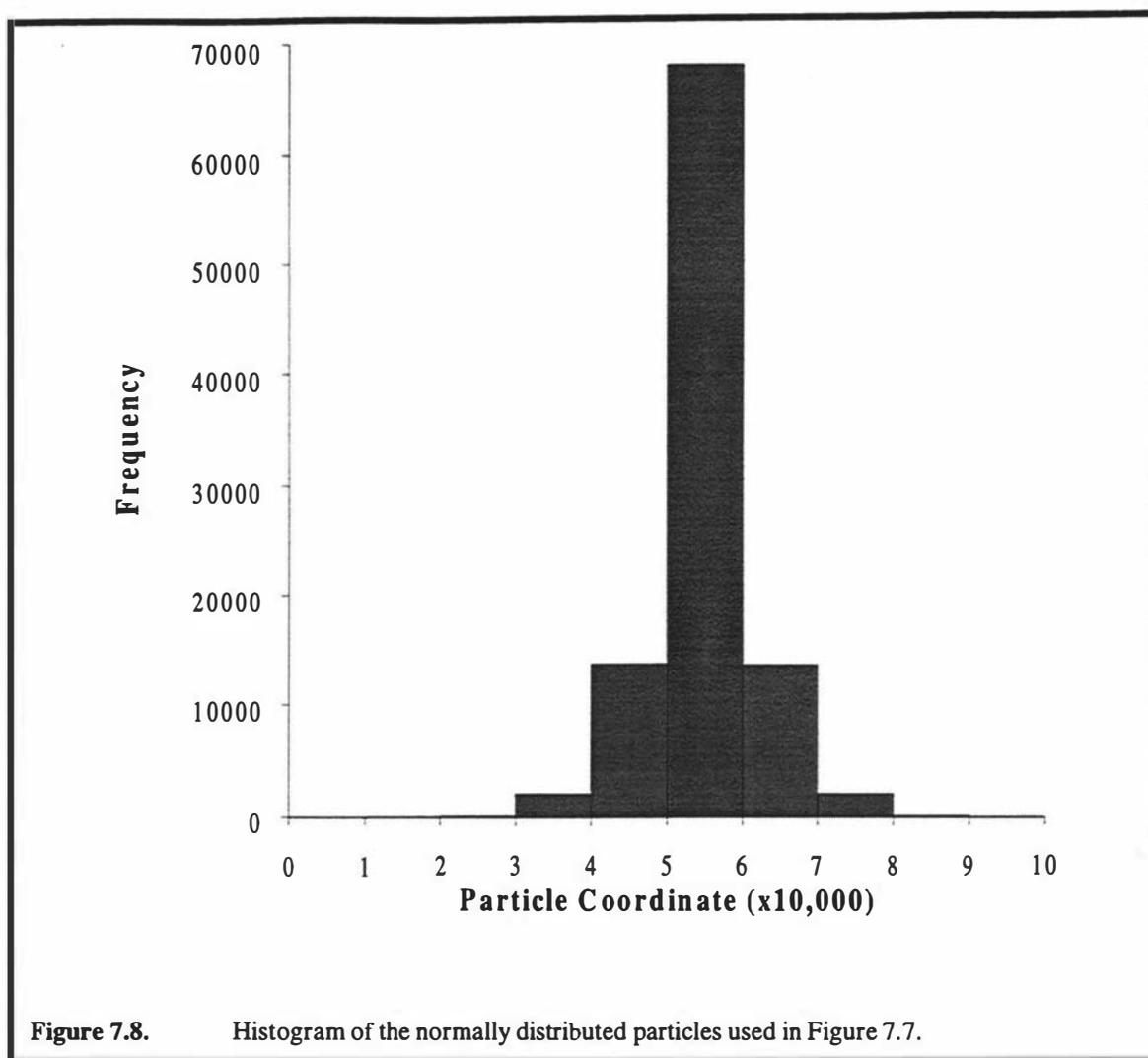


Figure 7.5 shows that the effect of a highly polarised distribution on the random phase assumption is to reduce the rate of relaxation to the expectation value. Thus, much larger k must be considered before the random walk begins to apply. The exponentially distributed random deviates were generated by utilising the FORTRAN 77 subroutine `expdev` [136] and scaling the resultant exponential random deviate to yield a distribution of maximum range N . A histogram is given in Figure 7.6 to illustrate the extent of the polarisation.





While Figure 7.5 illustrated a negative effect from an exponential distribution in terms of the random walk feature, Figure 7.7 shows that a normally distributed (Figure 7.8) set of particles (derived as outlined in section 7.1.2.4.) yields an $I(k)$ where the random phase assumption applies earlier than for the uniform distribution. Clearly the validity of the random phase for relaxation to a random walk is an extremely important aspect when considering non-uniform distributions and highlights the statistical nature of the long-range part of the algorithm.



7.1.5. Neutral Systems

7.1.5.0. Introduction

Neutral systems of charge, in particular, infinite periodic systems, have received much attention by the PPPM and Ewald summation techniques where the aim has been to study forces and bulk properties. Isolated molecules, that is a gas phase molecule in its own universe, receive a great deal of attention in molecular quantum mechanics - which is where we eventually wish to apply our new algorithm. Isolated molecules tend to either have a net neutral charge, or a small charge relative to the total number of electrons, and while the investigations to date have focused on charged systems, electron-electron repulsion and forming the \mathbf{J} matrix, consistency with the treatment of the Coulomb operator within the Hamiltonian for MQM calculations must be maintained. That is, it will be necessary to consider the replacement of the Coulomb

operator with the split operator throughout the Hamiltonian, not only for electron-electron interactions, but also electron-nuclei and nuclei-nuclei. Presented in this section are timings for net-neutral 1-D collinear particles using two alternative approaches. The random walk phenomenon is also considered, but a detailed discussion on this and scaling characteristics is left until Chapter Eight where the CASE approximation is introduced.

7.1.5.1. A Random Walk Advantage?

Figures 7.9 and 7.10 illustrate the random walk nature of the $I(k)$ function for neutral systems. Unlike charged systems, the random walk holds for all values of $k \neq 0$. This is not due to the random phase assumption of kr_j , but the effect of having a mixture of positive and negative charges corrects for the initial non-random phases.

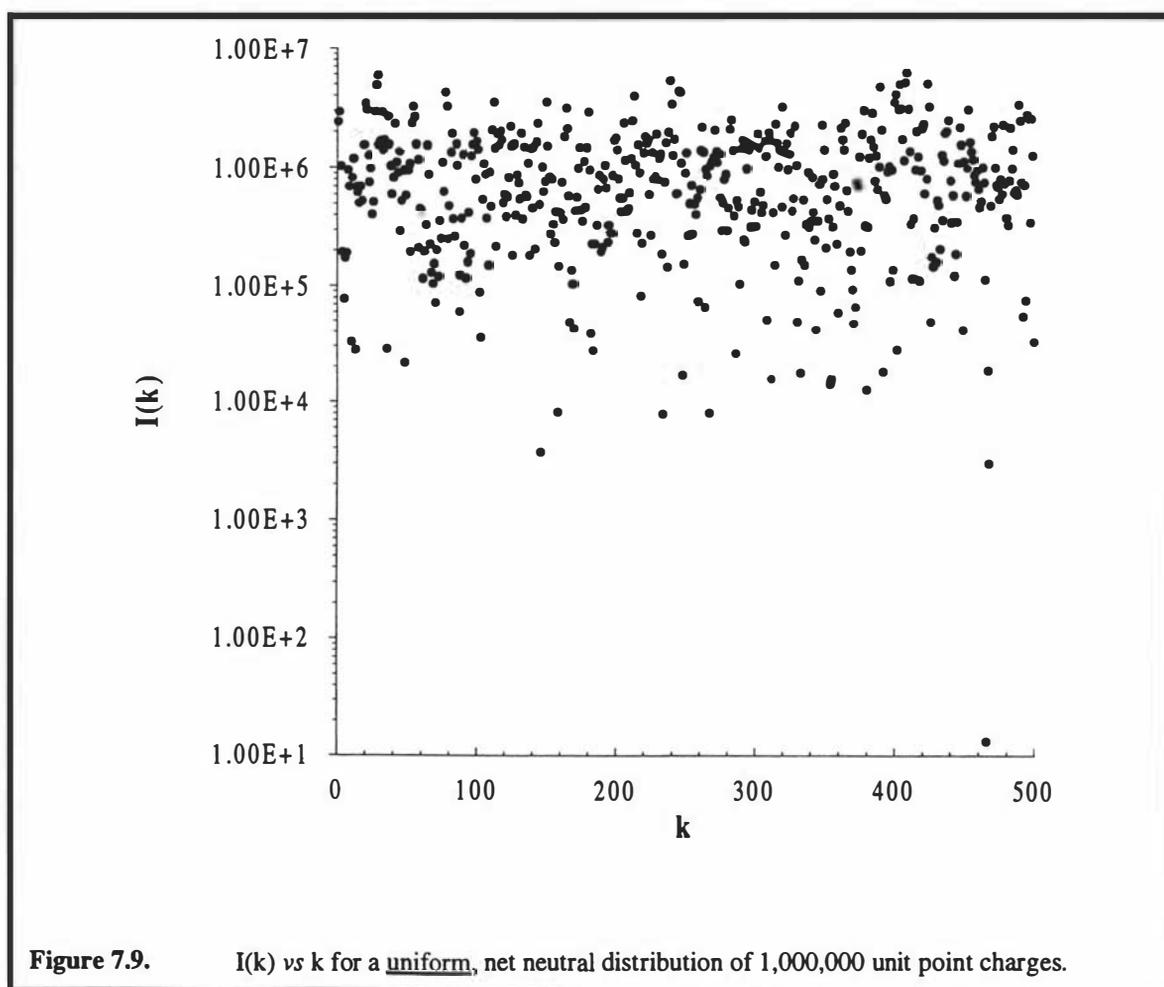


Figure 7.9 shows that the random walk holds for all values of $k \neq 0$, having an initial value of zero at $k=0$. The question which must be addressed is what advantage is this random walk behaviour for neutral systems?

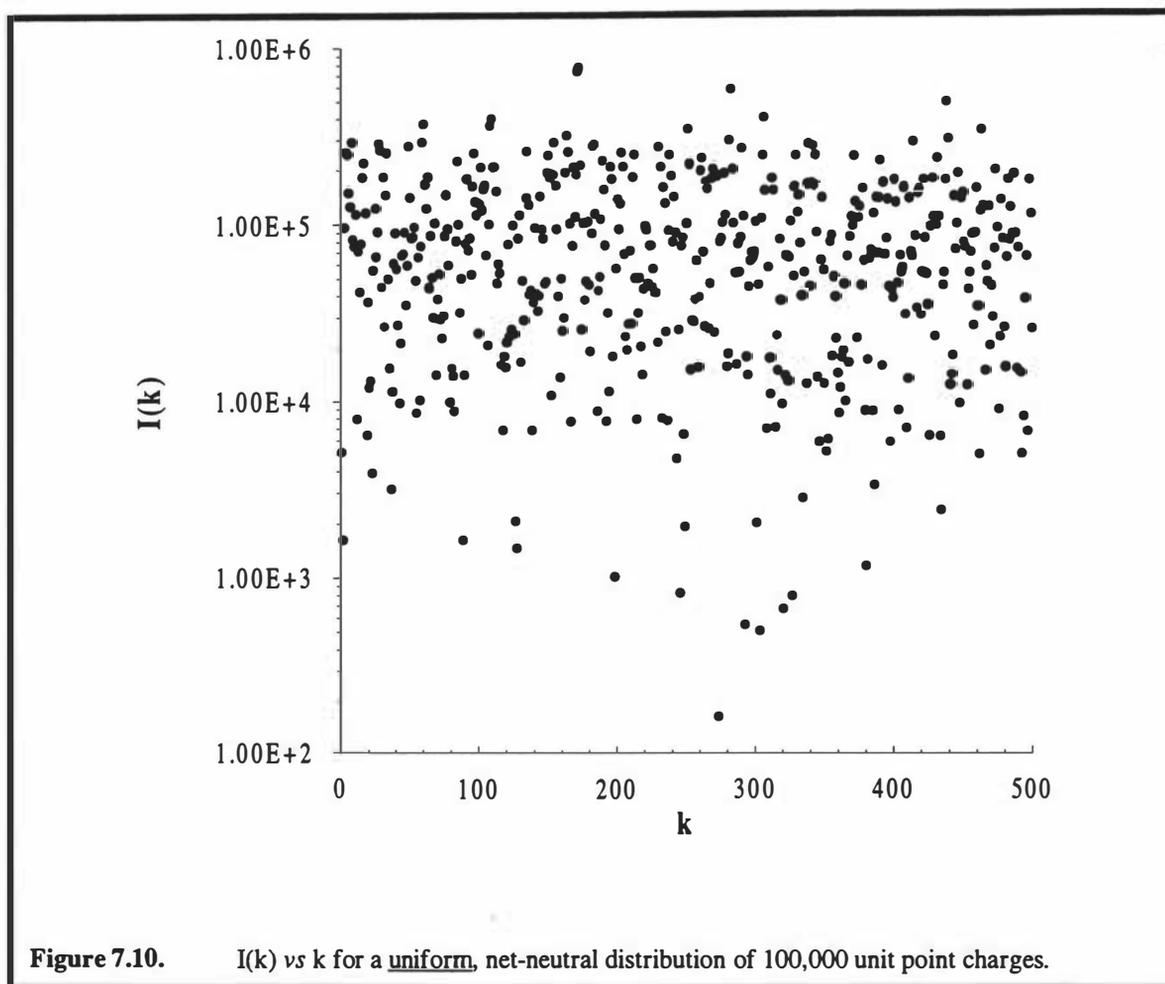
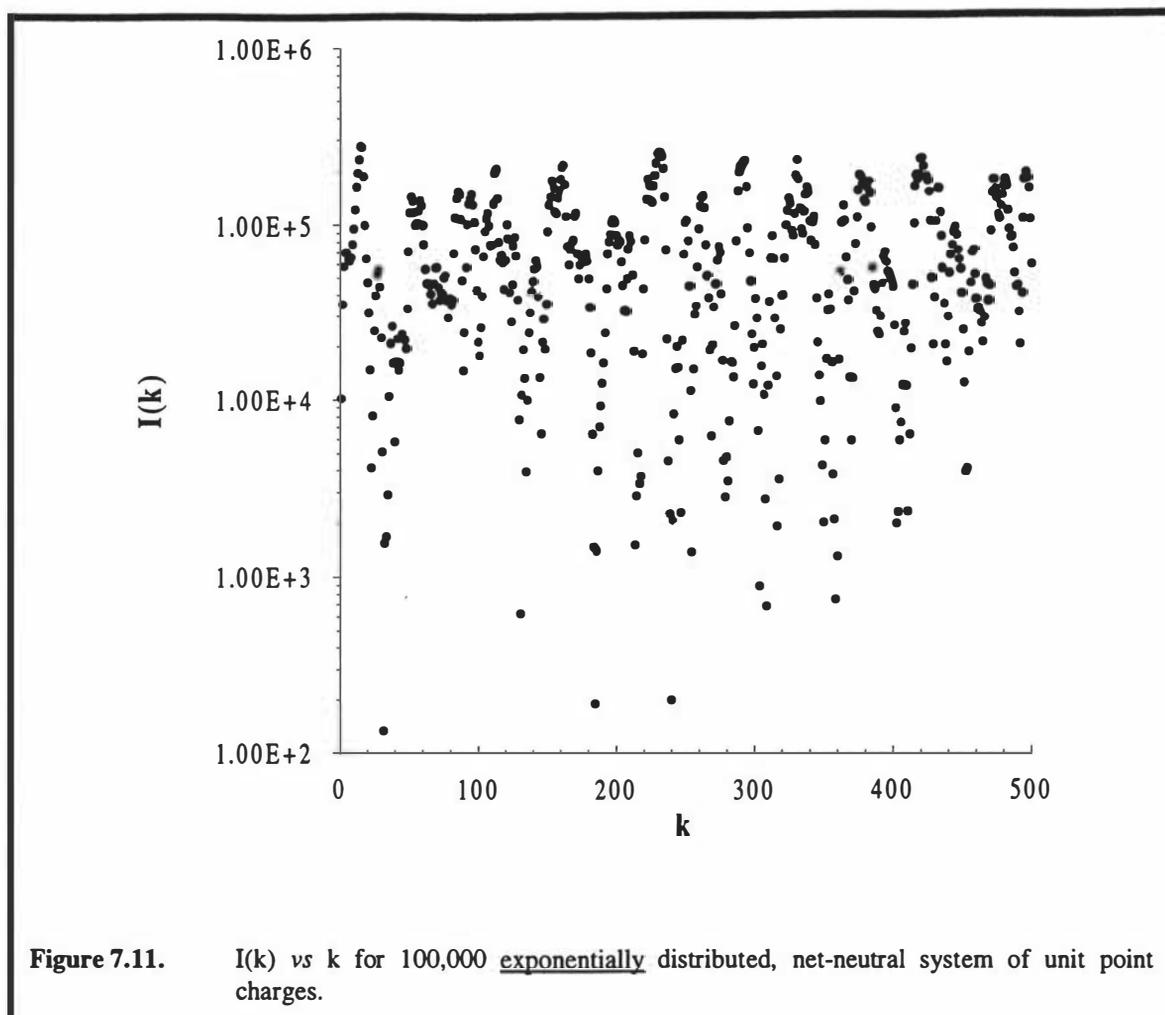


Figure 7.10 is analogous to Figure 7.9 where we have an $I(k)$ which is a 'noisy' value of N , where the noise is approximately $10N$. The random walk appears to be valid for all k .



While for charged systems the effect of polarising the distribution had a catastrophic effect on the rapid relaxation of $I(k)$ to the random walk value, for a net neutral system no such effect is observed, as can be seen in Figure 7.11.

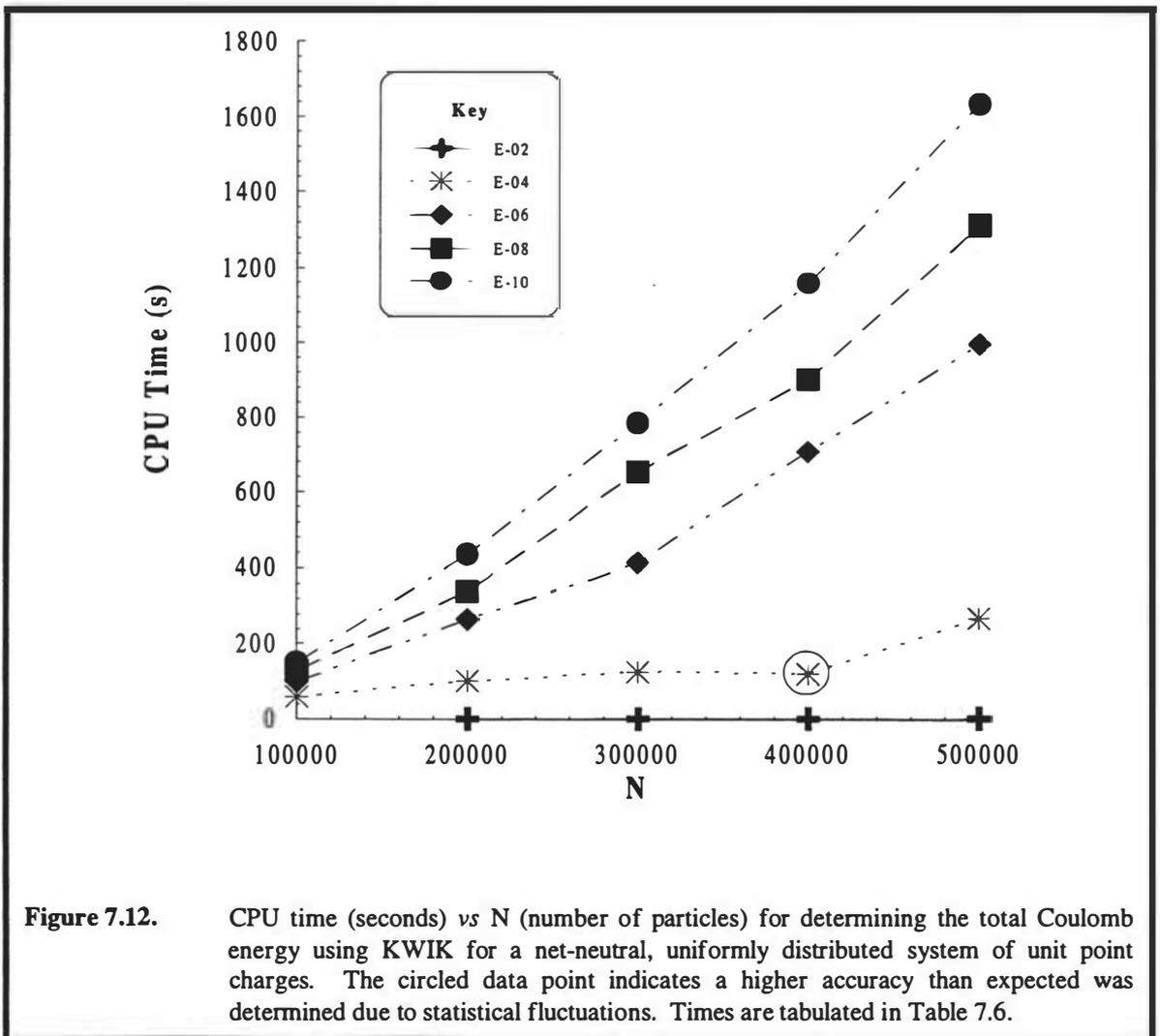
7.1.5.2. Illustrative Timings

The first of the two approaches taken for determining the Coulomb energy of net-neutral systems of charge was exactly analogous to the charged case. Uniform deviates were obtained in the same manner, with each deviate assigned a positive or negative charge to afford a net-neutral system. The decay parameter and degree of the Fourier expansion were varied to afford the desired accuracy, maintaining the even split of work between short- and long- range partitions - exactly the approach taken for the charged systems. The timings are tabulated in Table 7.6.

N	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-10}
100,000	-	59.05	98.56	127.90	148.77
200,000	1.22	100.72	263.98	336.80	432.68
300,000	1.95	126.93	412.06	652.38	783.63
400,000	2.58	119.28	707.12	901.37	1158.69
500,000	3.73	269.08	994.70	1312.94	1632.99

Table 7.6. CPU times (seconds) for determining the Coulomb energy using KWIK for collinear, uniformly distributed, net-neutral systems of unit point charges for various predetermined accuracies. Decay parameter and degree of Fourier expansion determined as for charged systems.

Energy contributions and other details of the calculations for the timings in Table 7.6 are tabulated in the Appendix. To illustrate the scaling of net-neutral systems, the times tabulated in Table 7.6 have been plotted in Figure 7.12. Note the similar scaling behaviour of net-neutral distributions to that of charged distributions.



The second approach considered was to fix ω and vary M to obtain the required accuracy. Timings obtained by fixing ω at 0.01 and 0.005 and are tabulated in Tables 7.7 and 7.8.

N	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-10}
100,000	21.35	60.37	96.14	128.64	150.63
200,000	42.25	134.02	273.17	384.33	454.77
300,000	63.56	126.34	449.77	747.26	968.46
400,000	84.18	198.88	762.32	1314.66	1634.33
500,000	105.10	629.34	1336.75	1928.45	2445.92

Table 7.7. Total KWIK timings for various predetermined accuracies for net-neutral, uniformly distributed particles using a fixed decay parameter of $\omega = 0.01$.

The timings presented in Table 7.7 show a similar, if not less linear scaling behaviour than those tabulated in Table 7.6 and plotted in Figure 7.12, *i.e.* the scaling appears non-linear in Table 7.6 and quadratic in Table 7.7. The same quadratic scaling is observed in Table 7.8.

N	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-10}
100,000	40.53	82.11	115.94	140.71	175.17
200,000	83.40	176.91	274.93	347.16	414.84
300,000	124.79	251.79	463.10	607.83	748.61
400,000	167.85	317.22	720.23	945.53	1151.48
500,000	206.87	496.31	961.96	1355.56	1680.45

Table 7.8. Total KWIK timings for various predetermined accuracies for net-neutral, uniformly distributed particles using a fixed decay parameter of $\omega = 0.005$.

The concept of holding the decay parameter constant was based on the observation that the short-range partition of a KWIK calculation on a net-neutral distribution of charge captured the majority of the energy, as opposed to charged systems where the energy was more evenly split between the partitions. The comparison between charged and net-neutral systems can be seen in Table 7.9 for a fixed decay parameter.

N	Neutral			Charged		
	$\omega = 0$	$\omega = 0.01$	%Error	$\omega = 0$	$\omega = 0.01$	%Error
1000	-4314.8	-4311.7	7.2E-02	18712	16337	12.7
10000	-36144.2	-36144.6	9.1E-04	206651	160798	22.2
100,000	14368.1	14393.7	1.8E-01	2108919	1419757	32.7
1,000,000	-108906.9	-109004.5	9.0E-02	20988938	11795926	43.8

Table 7.9. Short-range KWIK energies for net-neutral and charged uniformly distributed unit point charges. A zero decay parameter corresponds to the total coulomb energy. The decay parameter of $\omega = 0.01$ corresponds to a real space cut-off of ≈ 302 for $\epsilon = 10^{-6}$.

The question was posed - could the desired accuracy for net-neutral systems be obtained by increasing the cut-off radius? That is, decreasing ω in such a way that more time is spent on calculating short-range interactions than long range, to better reflect the contributions of each partition to the total energy.

N	$\omega = 0$	$\omega = 0.01$	%Error	$\omega = 0.005$	%Error
100,000	14368.1	14393.7	1.8E-01	14377.5	6.5E-02
200,000	-46810.3	-46731.1	1.7E-01	-46747.2	1.3E-01
300,000	-78086.2	-78044.0	5.4E-02	-78046.4	5.1E-02
400,000	-163722	-163686	2.2E-02	-163690	2.0E-02
500,000	-126458	-126361	7.7E-02	-126393	5.1E-02

Table 7.10. Short-range KWIK energies for net-neutral, uniformly distributed unit point charges. The decay parameters of $\omega = 0.01$ and 0.005 have real space cut-offs of ≈ 302 and 603 respectively for $\epsilon = 10^{-6}$.

In Table 7.10 the cut-off radius is doubled to determine the effect on relative contributions to the total Coulomb energy. The effect of doubling the short-range work does not result in any significant increase in the relative contribution of the short-range partition to the total energy. Thus, long-range work must be considered for improvement of accuracy.

For a fixed ω , doubling the size of the system requires double the amount of short-range work. However, doubling the system size also doubles the range of the distribution, reducing the density of k vectors, and thus an increase in k must be required to afford similar accuracy. This would imply scaling of $O(N^2)$. This is apparent in Tables 7.7 to 7.8. By adjusting the decay parameter and simultaneously adjusting the degree of the Fourier expansion this scaling can be reduced to that of a high accuracy

relative-to- N -charged-system scenario, with an empirical scaling of $O(N^{1.X})$, which is similar to that described by Luty [105,112].

7.2. Particles in Three dimensions

7.2.0. Introduction

While Greengard began his development in 2-D then progressed to 3-D, the development of the KWIK algorithm began in 1-D. The aim was to simplify the approach, to learn as much as possible about the algorithm before generalising to higher dimensions. Preliminary investigations into 2-D and 3-D algorithms were made initially, but it was found that considering both would not add greatly to the understanding of the algorithm, and 2-D investigations were discontinued. This section illustrates that KWIK in 3-D is a simple extension of the 1-D analogue and presents the slight modifications required on increased dimension, 3-D KWIK timings previously reported, and discussions regarding cut-offs in the long-range partition. In the latter preliminary findings from the 2-D investigations are referred to for clarity.

7.2.1. Derivation

The derivation of the KWIK equations for 3-D point charges follows equations (7.1) to (7.5) and differs only where the long-range operator substituted by its Fourier series representation

$$\frac{\text{erf}(\omega r_{ij})}{r_{ij}} = \sum_{\mathbf{k}} A(\mathbf{k}, \omega) \exp(i\mathbf{k} \cdot (\mathbf{r}_i - \mathbf{r}_j)) \quad (7.25)$$

where the summation is over 3-D k -space and $A(\mathbf{k}, \omega)$ are the associated Fourier coefficients. Thus the new long-range energy expression is

$$E_{long} = \frac{1}{2} \sum_{l=1}^N \sum_{j=1}^N \sum_{\mathbf{k}} A(\mathbf{k}, \omega) Q_l Q_j \exp(i\mathbf{k} \cdot (\mathbf{r}_l - \mathbf{r}_j)) - E_{SIC} \quad (7.26)$$

Inverting the sums and factorising the exponential allows E_{long} to be approximated to the required accuracy by

$$E_{long} \approx \frac{1}{2} \sum_{|\mathbf{k}| < M} A(\mathbf{k}, \omega) I(\mathbf{k}) \quad (7.27)$$

and

$$I(\mathbf{k}) = \left| \sum_{j=1}^N Q_j \exp(i\mathbf{k} \cdot \mathbf{r}_j) \right|^2 \quad (7.28)$$

is again the Fourier intensity of the total charge distribution. The self interaction correction is

$$E_{SIC} = \sum_{j=1}^N Q_j^2 \sum_{|\mathbf{k}| < M} A(\mathbf{k}, \omega) \quad (7.29)$$

Equation (7.27) still contains the letters of the algorithm and note the little has changed from 1-D to 3-D.

7.2.2. Short-Range

7.2.2.1. Interpolation

The use of interpolation tables have been employed for evaluating the erfc function as for 1-D systems. However, not only is a divide still required but there is also an extra requirement of a square root for determining inter-particle distance. The expense of the square root reduces the impact of applying interpolation techniques (Table 7.1). It has been suggested [103] that the square root function can be calculated more efficiently if the system is appropriately scaled, but this introduced efficiency requires r_{ij}^2 to lie in the range (0.1,1.0) which for very large systems does not lead to a more efficient algorithm.

7.2.2.2. Cut-offs

The distance at which the primary interaction becomes insignificant (7.22) was initially used as the cut-off radius. However, the more rigorous upper bound of

$$4\pi \int_{R_c}^{\infty} r \operatorname{erfc}(\omega r) dr \leq \varepsilon \quad (7.30)$$

which is

$$\pi \operatorname{erfc}(\omega R_c) \left[2R_c^2 + \frac{1}{\omega^2} \right] + \frac{2\sqrt{\pi}}{\omega} R_c \exp(-R_c^2 \omega^2) \leq \varepsilon \quad (7.31)$$

and can be determined easily using a Newton-Rhapson procedure was implemented for consistency.

7.2.2.3. Memory Storage

Memory storage in 3-D has a total requirement of roughly $11N$, which is an increase from 1-D only by the extra requirements for storing the extra coordinates. This storage is depicted in a similar manner to the 1-D case (Scheme 7.2) in Scheme 7.4. (The sorting methodology was presented in Chapter Five.)

```
REAL*8   ORD(N,4),R(4,N)
INTEGER  ICOUNT(NBOX),IPOINT(NBOX),INBOX(N)
```

Scheme 7.4. Illustration of memory requirements for the three dimensional sorting process.

7.2.3. Long-Range

7.2.3.0. Introduction

The 3-D long range energy partition is handled in a manner analogous to 1-D. As before, FLOP and MOP savings are made using recurrence relations and loop unrolling. Also there is a random walk in the complex plane where the phases, $\mathbf{k} \cdot \mathbf{r}_j$ relax towards the random phase assumption rapidly. However, during our investigations into 2-D KWIK we observed an exciting trend in the decay of the Fourier coefficients to enable further improvements to the computation of the long-range energy.

7.2.3.1. Memory Requirements

A semi-memory-intensive long-range code was implemented for the purposes of maximising speed. The memory requirements are depicted in Scheme 7.5. The R array holds particle position and charge, the F array holds the base complex exponential values of two of the coordinates, and G holds the corresponding coordinate recursion updates. It was rationalised that the z-coordinate should not be evaluated in a recursive

manner for two reasons. Firstly, the Fourier sum is over half of k -space due to the even nature of the long-range operator, and it was arbitrarily chosen to consider only positive z -components of the k sum. Therefore, fewer z -coordinate complex exponentials need be evaluated. Secondly, the cost of the non-recursive complex exponential for the z -coordinate can be evaluated such that it is done only once for the xy plane of that z -component of the k sum. The extra $4N$ double precision words required for recursive evaluation of this component was deemed excessive, given the small speed up that may be achieved.

```
REAL*8 R(4,N)
COMPLEX*16 A,F(N,2),G(N,2),S(5)
```

Scheme 7.5. Array storage for long-range KWIK.

From Scheme 7.5 it can be seen that the memory requirements of 3-D long-range KWIK are $12N$ double precision words.

7.2.3.2. 2-D Fourier Coefficients

The exciting feature of the 2-D Fourier coefficients is that the near-diagonal elements converge faster than the off-diagonal elements, which results in near diagonal Fourier coefficients significantly smaller than off-diagonals. A small number of these are presented in Tables 7.11 and 7.12 for two decay parameter values.

k_x/k_y	0	1	2	3	4
0	0.471306				
1	0.0920439	0.0399114			
2	0.00715780	0.00715930	0.00261430		
3	0.00424720	0.00175870	0.00044460	0.00012310	
4	-0.00121160	-0.00014550	0.0000717	0.00000090	0.00000840

Table 7.11. Two dimensional KWIK Fourier coefficients for $\omega = 1$. k_x and k_y correspond to the x and y components of the k -space sum.

The near-diagonal elements of Tables 7.11 and 7.12 are significantly smaller than the off-diagonal counterparts of the same row. This was an important discovery to aid in determining Fourier space cut-offs.

k_x/k_y	0	1	2	3	4
0	-0.0336836				
1	0.0157290	0.0042100			
2	-0.00535970	-0.00094930	0.00005400		
3	0.00244910	0.00048300	-0.00003150	0.00002180	
4	-0.00139770	-0.00028250	0.00001600	-0.00001210	0.0000065

Table 7.12. Two dimensional KWIK Fourier coefficients for $\omega = 0.1$. k_x and k_y correspond to the x and y components of the k-space sum.

The significance of this near-diagonal convergence is best observed by comparing with the $\omega=0$ decay parameter coefficients tabulated in Table 7.13. This table also highlights the importance of splitting the Coulomb operator to obtain rapidly decaying Fourier coefficients. The Fourier coefficients for the Coulomb operator, as opposed to the long-range operator, decay very slowly.

k_x/k_y	0	1	2	3	4
0	0.561100				
1	0.174884	0.116741			
2	0.0742178	0.0702324	0.0563238		
3	0.0555007	0.0508122	0.0441101	0.0375350	
4	0.0383910	0.0383182	0.0356041	0.0318189	0.0281415

Table 7.13. Two dimensional KWIK Fourier coefficients for $\omega = 0$. k_x and k_y correspond to the x and y components of the k-space sum.

The trend depicted in Tables 7.11 and 7.12 is observed in an analogous fashion for 3-D Fourier coefficients. On this basis a check could be implemented for near diagonal Fourier coefficients below a required threshold. A crude cut-off technique based on this idea was implemented for the 3-D timings presented in Table 7.15, and significantly sped up the overall KWIK algorithm.

This cut-off technique needs to be investigated further as some problems were encountered when initial k-space cut-offs were too lean and the Fourier expansion required extension to reach the desired accuracy. Furthermore, as the long-range energy is the sum of the Fourier intensity, $I(\mathbf{k})$, multiplied by a Fourier coefficient in which the Fourier intensity is an exponential deviate, cut-offs based on the magnitude of the Fourier coefficient, maybe problematic when 'outlying' over-large $I(\mathbf{k})$'s, are

correspondingly not computed. *i.e.* the product may not be insignificant when statistically unlikely, over-large values of the Fourier intensity are encountered.

7.2.4. Charged Systems

7.2.4.0. Introduction

KWIK must be able to solve applications where three dimensionally distributed point charges interact *via* Coulombs law. This type of interaction is analogous to gravitational systems, is found in MQM calculations of the nuclear repulsion energy and is a generalised form of the electron-electron repulsion energy, the true electron-electron repulsion being between diffuse distributions of charge. The aim in the previous sections was to illustrate features of KWIK and some theoretical features discussed in previous chapters by considering collinear point charges. And while it is true that an algorithm's speed can be improved by allowing for the collinear assumption, the purpose was to illustrate fundamental features of how the KWIK algorithm worked. In section 7.2.4.1 timings are presented to show that the scaling behaviour persists exactly analogously to one dimension.

7.2.4.1. Illustrative Timings

Timings for 3-D unit point charges presently previously [148] are presented in Table 7.14, with the effect of interpolation and crude long-range cut-offs with significantly improved timings illustrated in Table 7.15.

N	10^{-3}	10^{-4}	10^{-6}	Exact
500	0.8	1.2	2.2	0.3
1,000	1.7	2.7	9.0	1.4
5,000	7.1	11.4	47.3	27.8
10,000	12.3	23.1	148.7	105.1
50,000	64.8	125.1	628.6	2532.1
100,000	139.5	361.2	1500.2	10145.7
500,000	769.4	2065.5	13515.2	252170.6

Table 7.14. KWIK timings for charged systems of three dimensional, uniformly distributed unit point charges as previously reported [148].

Initial investigations yielded similar qualitative scaling characteristics for the 3-D systems as for the 1-D analogues. The earlier reported timings (Table 7.14) have since been improved dramatically with refinements to the algorithm components. This is

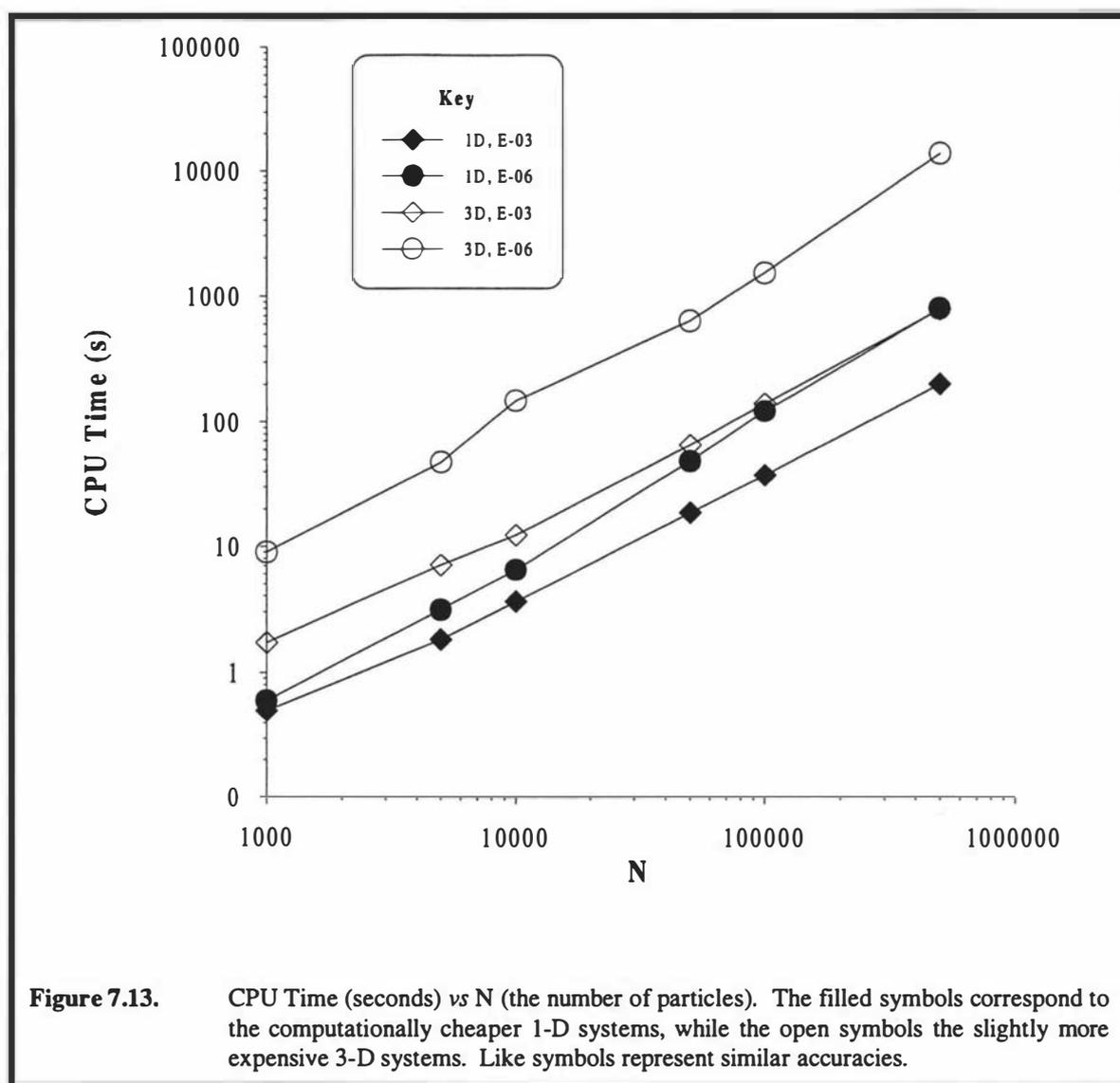
evident by comparing the times for the $N=100,000$ case, which for the higher accuracies has improved the total time by a factor of two. The greater improvement for the lower accuracy was due to improved timing optimisation and I/O time neglect.

N	10^{-3}	10^{-4}	10^{-6}	Exact
500	0.12	0.27	0.46	0.24
1,000	0.29	0.75	2.81	1.00
5,000	1.43	3.20	26.00	25.10
10,000	2.95	9.63	74.39	100.00
50,000	14.56	64.31	357.20	2510.00
100,000	29.53	151.38	817.84	10145.70
500,000	167.69	657.61	885.80	252170.60

Table 7.15. KWIK timings for charged systems of three dimensional, uniformly distributed unit point charges with incorporation of interpolation and a crude implementation of long-range cut-offs.

Unlike the FMM [114,116] which is considerably more expensive in 3-D than in 1-D, the cost of KWIK increases moderately with system dimensionality. This is illustrated for two predetermined levels of accuracy in Figure 7.13.

There are several important points to note from Figure 7.13. Firstly, the increase in cost from 1-D to 3-D is not excessive and is approximately an order of magnitude cheaper than the FMM [116]. Secondly, the data used to plot this graph used the earlier unoptimised timings - the effect of the recent refinements will simply be a downward shift in the lines. Finally, the scaling characteristics are the same in 1-D and 3-D. The two 10^{-3} relative accuracy lines are unit slope and obviously parallel. The higher accuracy lines are also parallel following the trend discussed in section 7.1.4.1.



7.3. Summary and Discussion

Illustrative timings have been presented for a number of theoretical developments presented in previous chapters. The effect of the theoretical developments was simplified by considering collinear distributions of charge, thus the fundamental features of the KWIK algorithm could be easily illustrated, in particular, the distribution dependence of the long-range partition. KWIK has been found empirically to scale linearly with the system's size for sufficiently large N , in the case of charged systems. Neutral systems however, appear to scale slightly less favourably than linearly, although the total time requirements for calculations on very large systems ($N=1,000,000$) has still been reduced significantly by utilising this algorithm. It was also indicated that for a neutral system the short-range energy contributes significantly more to the total energy

than charged counterparts. The validity of our emphasis on describing features of simpler 1-D systems has been justified by the similar scaling characteristics of 3-D charged systems, and the small increase in cost on increasing dimensionality from 1-D to 3-D.

The decay parameter, ω and the degree of the Fourier expansion have been determined through an optimisation procedure for the timings presented. It is apparent from these timings that the KWIK algorithm has a statistical probabilistic base, which stems from the random walk property of the Fourier intensity of the charge distribution, $I(k)$. This has led to cases where a higher accuracy was achieved at a computational cost which contrasted with the observed trend. Predetermining the optimal decay parameter and degree of Fourier expansion is likely to be linked to statistical properties of the distribution. The amount of work the short-range partition will require will be dependent on the density of the particles. The study of uniformly distributed net-neutral distributions with fixed decay parameter indicates that the range of the distribution is important for convergence of the long-range sum, with the plots of $I(k)$ for various charged distributions indicating that higher order distribution moments may be significant and indicate features of $I(k)$.

CHAPTER EIGHT

KWIK APPLIED TO QUANTUM CHEMISTRY

8.0. Introduction

The results of the previous chapters equip us with a toolbox of algorithms for computing Coulomb interactions using the KWIK method. With this established, it is of great interest to speculate on the possible future uses of such a toolbox. In this chapter, KWIK will be applied to arbitrary continuous charge distributions to illustrate the generality of the algorithm. Given a number of assumptions, KWIK will then be used to redefine the time dependent, non-relativistic Schrödinger equation, and subsequently KWIK formulae for self-consistent field calculations will be obtained. Emphasis will be placed on the formation of Coulomb (**J**) and exchange (**K**) matrices.

Two further sections have been added to indicate the usefulness of the algorithm. The first is the use of Slater functions in quantum chemistry and how KWIK may aid in an implementation and the second is a brief discussion on the possibilities KWIK has in correlated techniques.

The main aim of this thesis has been “to enable the rapid evaluation of the Coulomb field in Molecular Quantum Mechanical calculations.” If by ‘rapid’ we imply linear cost or $O(N)$ work we firstly require that the distribution is localised. No linear method will be applicable to systems that are significantly delocalised. While MO’s are often significantly delocalised, they consist of a sum of localised AO’s. In turn, the electron density is written as a product of a density matrix element and basis function pair (1.27), which is a localised representation.

8.1. KWIK for $O(N)$ Fock Matrix Formation

8.1.0. Introduction

In order to afford $O(N)$ SCF calculations, it is necessary to assume the localised nature of the molecular electron density. Given this assumption it can be illustrated how

the KWIK algorithm can be used to construct a Fock matrix in linear work. Individual elements of the Fock matrix will be considered, firstly the one electron core Hamiltonian matrix, then the Coulomb and Exchange matrices. To more clearly illustrate the Quantum Chemical KWIK, KWIK is firstly re-derived using the compact Dirac notation

8.1.1. KWIK in Dirac's Notation

Reconsider the general separation of the Coulomb operator into a short-range singular piece and a long-range piece

$$\frac{1}{r} \equiv \frac{f(r)}{r} + \frac{1-f(r)}{r} \quad (8.1)$$

The KWIK approach is to then replace the long-range operator with its Fourier series representation (incorporating the ω decay parameter)

$$\frac{1-f(\omega r_{12})}{r_{12}} = \sum_{\mathbf{k}} A(\mathbf{k}, \omega) \exp(i\mathbf{k} \cdot (\mathbf{r}_1 - \mathbf{r}_2)) \quad (8.2)$$

which can be recast concisely adopting Dirac's notation and the Einstein summation convention [149]

$$\frac{1-f(\omega r_{12})}{r_{12}} = |e_j(\mathbf{r}_1)\rangle \langle e_j(\mathbf{r}_2)| \quad (8.3)$$

where

$$|e_j(\mathbf{r})\rangle = \sqrt{A(\mathbf{k}, \omega)} \exp(i\mathbf{k}_j \cdot \mathbf{r}) \quad (8.4)$$

and the $A(\mathbf{k}, \omega)$ are the Fourier coefficients. The necessary KWIK matrix elements can now easily be determined by substituting the Coulomb operator with the following

$$\frac{1}{r_{12}} \equiv \frac{f(\omega r_{12})}{r_{12}} + |e_j(\mathbf{r}_1)\rangle \langle e_j(\mathbf{r}_2)| \quad (8.5)$$

8.1.2. Core Hamiltonian Matrix

The core Hamiltonian matrix consists of the one electron kinetic energy integrals and nuclear attraction integrals. The kinetic energy matrix \mathbf{T} , consisting of kinetic energy integrals, formally scales as $O(N^2)$, and can be afforded in linear work in a similar fashion to the overlap integral matrix \mathbf{S} . The nuclear attraction matrix \mathbf{V} on the other hand, formally scales as $O(N^3)$ which is reduced to $O(N^2)$ again by virtue of overlap as discussed in Chapter Three. While the latter ‘formally’ scales quadratically, the $O(N^2)$ coefficient is significantly smaller than that of the electron-electron Coulomb interaction as it is the interaction of the electrons, which are represented by a very large $O(N)$ number of small pieces of charge (the basis function pairs), with nuclei, whose $O(N)$ number is significantly less than that of electronic representation. Thus, the total time required to evaluate the \mathbf{V} matrix is significantly less than the \mathbf{J} or \mathbf{K} matrices. Notwithstanding, for significantly large systems with an $O(N)$ \mathbf{J} and \mathbf{K} matrix algorithm, \mathbf{V} will become rate limiting.

KWIK can readily be used to evaluate the \mathbf{V} matrix components. Consider the nuclear attraction matrix elements

$$V_{\mu\nu} = \int \phi_{\mu}(\mathbf{r}) \left[-\sum_A \frac{Z_A}{|\mathbf{R}_A - \mathbf{r}|} \right] \phi_{\nu}(\mathbf{r}) d\mathbf{r} \quad (8.6)$$

As a simplification let us represent the basis function pair as

$$\phi_{\mu}(\mathbf{r})\phi_{\nu}(\mathbf{r}) = Q_{\mu\nu}(\mathbf{r}) \quad (8.7)$$

and to enable the use of Dirac’s notation, rewrite the integral as the double integral using

$$\delta(\mathbf{R}_A - \mathbf{r}) = Q_A(\mathbf{r}) \quad (8.8)$$

and by adopting the Einstein summation convention the nuclear attraction matrix elements can be written as

$$V_{\mu\nu} = \iint Q_{\mu\nu}(\mathbf{r}_1) \left[-\frac{Z_A}{r_{12}} \right] \delta(\mathbf{R}_A - \mathbf{r}_2) d\mathbf{r}_1 d\mathbf{r}_2 \quad (8.9)$$

and adopting Dirac's notation

$$V_{\mu\nu} = \langle Q_{\mu\nu} | \frac{-Z_A}{r_{12}} | Q_A \rangle \quad (8.10)$$

now replacing the Coulomb operator with the KWIK operator (8.5) yields

$$\begin{aligned} V_{\mu\nu} &= \langle Q_{\mu\nu} | \frac{-Z_A}{r_{12}} | Q_A \rangle \\ &\approx \langle Q_{\mu\nu} | \frac{-Z_A f(\omega r_{12})}{r_{12}} | Q_A \rangle - Z_A \langle Q_{\mu\nu} | e_j \rangle \langle e_j | Q_A \rangle \\ &= \int Q_{\mu\nu}(\mathbf{r}) \left[\frac{-Z_A}{|\mathbf{R}_A - \mathbf{r}|} \right] d\mathbf{r} - \hat{Q}_{\mu\nu}^j Z_A \hat{Q}_A^j \end{aligned} \quad (8.11)$$

The first term of the last line is obtained by integrating the delta function term from the previous line. While the sum over A in the final line appears to make the \mathbf{V} matrix formation remain quadratic, this A summation need only be determined once for all elements and is thus $O(1)$. The first term is $O(1)$ from proximity arguments.

8.1.3. Coulomb and Exchange Matrices

The Coulomb, \mathbf{J} , and Exchange, \mathbf{K} , matrix elements are given by, respectively

$$J_{\mu\nu} = \sum_{\lambda\sigma} P_{\lambda\sigma}(\mu\nu|\lambda\sigma) \quad (8.12)$$

$$K_{\mu\nu} = \frac{1}{2} \sum_{\lambda\sigma} P_{\lambda\sigma}(\mu\lambda|\nu\sigma) \quad (8.13)$$

where

$$(\mu\nu|\lambda\sigma) = \int \phi_\mu(\mathbf{r}_1) \phi_\nu(\mathbf{r}_1) \left[\frac{1}{r_{12}} \right] \phi_\lambda(\mathbf{r}_2) \phi_\sigma(\mathbf{r}_2) d\mathbf{r}_1 d\mathbf{r}_2 \quad (8.14)$$

\mathbf{P} is the density matrix and $\{\phi\}$ is the basis. Using the simplification of the representation of the basis function pair the Coulomb matrix elements can now be written (adopting the Einstein summation convention) as

$$\begin{aligned}
J_{\mu\nu} &= P_{\lambda\sigma} (\mu\nu|\lambda\sigma) \\
&= P_{\lambda\sigma} \langle Q_{\mu\nu} | \frac{1}{r_{12}} | Q_{\lambda\sigma} \rangle
\end{aligned} \tag{8.15}$$

where on replacement of the Coulomb operator with the KWIK operator (8.5)

$$\begin{aligned}
J_{\mu\nu} &= P_{\lambda\sigma} \langle Q_{\mu\nu} | \frac{1}{r_{12}} | Q_{\lambda\sigma} \rangle \\
&\approx P_{\lambda\sigma} \langle Q_{\mu\nu} | \frac{f(\omega r_{12})}{r_{12}} | Q_{\lambda\sigma} \rangle + P_{\lambda\sigma} \langle Q_{\mu\nu} | e_j \rangle \langle e_j | Q_{\lambda\sigma} \rangle \\
&= P_{\lambda\sigma} (\mu\nu|\lambda\sigma)_S + P_{\lambda\sigma} \hat{Q}_{\mu\nu}^j \hat{Q}_{\lambda\sigma}^j
\end{aligned} \tag{8.16}$$

Similarly for the Exchange matrix elements, firstly adopting the Einstein summation convention

$$\begin{aligned}
K_{\mu\nu} &= P_{\nu\sigma} (\mu\nu|\lambda\sigma) \\
&= P_{\nu\sigma} \langle Q_{\mu\nu} | \frac{1}{r_{12}} | Q_{\lambda\sigma} \rangle
\end{aligned} \tag{8.17}$$

and replacing the Coulomb operator with the KWIK representation

$$\begin{aligned}
K_{\mu\nu} &= P_{\nu\sigma} \langle Q_{\mu\nu} | \frac{1}{r_{12}} | Q_{\lambda\sigma} \rangle \\
&\approx P_{\nu\sigma} \langle Q_{\mu\nu} | \frac{f(\omega r_{12})}{r_{12}} | Q_{\lambda\sigma} \rangle + P_{\nu\sigma} \langle Q_{\mu\nu} | e_j \rangle \langle e_j | Q_{\lambda\sigma} \rangle \\
&= P_{\nu\sigma} (\mu\nu|\lambda\sigma)_S + P_{\nu\sigma} \hat{Q}_{\mu\nu}^j \hat{Q}_{\lambda\sigma}^j
\end{aligned} \tag{8.18}$$

where the quantities $(\mu\nu|\lambda\sigma)_S$ are the two electron integrals over the short-range operator and the $\hat{Q}_{\mu\nu}^j$ are the Fourier transforms of the charge distribution $Q_{\mu\nu}$ evaluated at \mathbf{k}_j .

Note again, while the KWIK equations for Coulomb and Exchange matrices contain terms with $O(N)$ summations

$$J_{\mu\nu} = P_{\lambda\sigma} (\mu\nu|\lambda\sigma)_S + P_{\lambda\sigma} \hat{Q}_{\mu\nu}^j \hat{Q}_{\lambda\sigma}^j \tag{8.19}$$

$$K_{\mu\nu} = P_{\nu\sigma} (\mu\nu|\lambda\sigma)_S + P_{\nu\sigma} \hat{Q}_{\mu\nu}^j \hat{Q}_{\lambda\sigma}^j \tag{8.20}$$

they are simple matrix multiply's [56,57] which need only be computed once. Thus, KWIK can construct the Exchange matrix almost as easily (the summations in (8.20) are somewhat more difficult than for the Coulomb case) as the Coulomb matrix. This is not true for the CFMM and its variants, which can only be used to linearise density functional theory calculations (the DFT exchange term calculable in linear work).

8.2. Evaluating Short-Range KWIK

8.2.0. Introduction

During investigations into the optimal separator, the aim was to obtain a separator that would also allow the exploitation of current ERI technology for evaluating the short-range ERI's. In this section the use of the erfc separator in two electron integrals, how the integrals are best evaluated through exploitation of current integral technology, and how one may screen out the $O(N^2)$ insignificant integrals in $O(N)$ work is illustrated.

8.2.1. Integrals Over Gaussian S Type Functions

The Coulomb repulsion energy between two normalised Gaussian charge distributions is given by

$$\left(\frac{\alpha\beta}{\pi^2}\right)^{3/2} \iint \exp(-\alpha r_1^2) \left(\frac{1}{r_{12}}\right) \exp(-\beta|\mathbf{r}_2 - \mathbf{B}|^2) d\mathbf{r}_1 d\mathbf{r}_2 = \frac{1}{|\mathbf{B}|} \operatorname{erf}\left(\frac{|\mathbf{B}|}{\sqrt{\frac{1}{\alpha} + \frac{1}{\beta}}}\right) \quad (8.21)$$

From this basic integral recurrence relations can be applied to yield integrals of higher angular momentum [128]. In order to compute the short-range KWIK energy, we replace the Coulomb operator in (8.21) with the short-range KWIK operator. It can be easily shown that the short-range KWIK (erfc) energy, between two normalised Gaussians is given by

$$\begin{aligned} & \left(\frac{\alpha\beta}{\pi^2}\right)^{3/2} \iint \exp(-\alpha r_1^2) \left(\frac{\operatorname{erfc}(\omega r_{12})}{r_{12}}\right) \exp(-\beta|\mathbf{r}_2 - \mathbf{B}|^2) d\mathbf{r}_1 d\mathbf{r}_2 \\ &= \frac{1}{|\mathbf{B}|} \left[\operatorname{erf}\left(\frac{|\mathbf{B}|}{\sqrt{\frac{1}{\alpha} + \frac{1}{\beta}}}\right) - \operatorname{erf}\left(\frac{|\mathbf{B}|}{\sqrt{\frac{1}{\alpha} + \frac{1}{\beta} + \frac{1}{\omega^2}}}\right) \right] \end{aligned} \quad (8.22)$$

which is very similar to the Coulomb term (8.21).

8.2.2. Screening Insignificant Integrals in $O(N)$ Work

The screening of insignificant integrals follows closely from the techniques outlined in Chapter Five, in which the significance of an interaction between a pair of charges is based purely upon proximity. However, the continuous and diffuse nature of basis function pairs requires more care in how such techniques are implemented. The CFMM, for example, while not being a proximity based method, uses a well-separated (WS) index to handle diffuse basis function pairs that overlap a number of low level boxes. It is likely that such an approach should not be adopted for short-range proximity based techniques, such as KWIK and CASE. The short-range interaction between a small number of boxes already limits vectorisability, WS indices serve only to further decrease loop lengths and integral batch sizes. It is also likely that the mother cell dimension, R_C , should be chosen such that the most diffuse basis function pair need only interact with near neighbours.

While non-concentric shell-pairs may ‘spill’ over the sides of a cell, that is the centres of original shells reside in separate cells, the amplitude and diffuseness of such a pair will always be significantly less than that of the corresponding concentric shell-pairs from the same functions. Therefore, their interaction need only be with near neighbour boxes.

Alternatively, the methodology adopted in the CFMM case could be maintained if the approach of Rapaport is adopted. This will also aid in maintaining vectorisation. If cell division is applied, the WS indices will be altered and further increase efficiency. For example, if a shell-pair was only slightly over-diffuse and thus necessarily assigned a WS of 2, it would interact with many shell-pairs it need not. However, if cell size is reduced using the division of boxes approach, the WS indices become less coarse. So for a cell reduction of say $k=3$ ($WS=1 \equiv WS_{k=3}=3$) our slightly over-diffuse shell-pair would be assigned a new index of $WS_{k=3}=4$ c.f. $WS_{k=3}=6 \equiv WS=2$.

8.3. Evaluating Long-Range KWIK

8.3.1. CASE and CAP's

The most basic way to approximate the long-range partition is not to evaluate it, *i.e.* assume it to be zero. Thus, we have

$$\frac{1}{r} \approx \frac{\text{erfc}(\omega r)}{r} \quad (8.23)$$

This approximation has become known as the CASE (Coulomb Attenuated Schrödinger Equation) approximation [150]. The rationale behind such an apparently crude approximation is that it is well known that molecules are essentially non-polar except on small distance scales. We should expect the attractive and repulsive Coulomb interactions between widely separated regions of a molecule to approximately cancel, or, in the words [151] of Clementi, “the electrons on the nose of Professor Karplus do not interact with the electrons on the nose of Professor Eyring”.

Consider the potential due to a hydrogen atom nucleus at a distance x from its position at the origin

$$\begin{aligned} V_n(x) &= \int \frac{\delta(\mathbf{r})}{|\mathbf{r} - \mathbf{x}|} d\mathbf{r} \\ &= \frac{1}{x} \end{aligned} \quad (8.24)$$

and that of the potential due to the exact electron probability charge distribution

$$\begin{aligned} V_e(x) &= -\int \frac{\pi \exp(-2r)}{|\mathbf{r} - \mathbf{x}|} d\mathbf{r} \\ &= -\frac{1}{x} (1 - \exp(-2x) - x \exp(-2x)) \end{aligned} \quad (8.25)$$

The potential due to a hydrogen atom thus decays exponentially but individual contributions as $1/x$ (the latter requiring us to consider all interactions). This kind of reasoning can explain the success of CASE for determining the energetics of chemistry but the observation that wavefunctions are little affected by neglect of the long-range term requires deeper consideration.

Consider the average total energy E , of a system in state Ψ , which can be written concisely as

$$E[\Psi] = \frac{\langle \Psi | H | \Psi \rangle}{\langle \Psi | \Psi \rangle} \quad (8.26)$$

Now, the variation principle says that the best wavefunction is that which minimises the total energy, *viz.*

$$\delta E[\Psi] = 0 \quad (8.27)$$

so that if we add (or subtract) a flat potential, c , to the Hamiltonian we yield a total energy term augmented by that constant

$$\begin{aligned} E[\Psi] &= \frac{\langle \Psi | H + c | \Psi \rangle}{\langle \Psi | \Psi \rangle} \\ &= \frac{\langle \Psi | H | \Psi \rangle}{\langle \Psi | \Psi \rangle} + \frac{\langle \Psi | c | \Psi \rangle}{\langle \Psi | \Psi \rangle} \\ &= \frac{\langle \Psi | H | \Psi \rangle}{\langle \Psi | \Psi \rangle} + c \end{aligned} \quad (8.28)$$

However, the function that minimises the total energy (8.27) is completely unaffected by the addition of this constant, flat term as the wavefunction Ψ does not appear with c in the energy term. In our construction of the separator we aimed to afford the flattest possible long-range operator, thus the selection of the smooth, rapidly decaying CASE operator was far from arbitrary.

While CASE was reasonably successful, there are simple chemical problems in which the so-called background (the long-range partition) plays an important role. CASE also yields very poor total energies. While it is generally accepted that quantum chemistry relies heavily on cancellation of errors [41], total energies are not meaningless and CASE is best considered the zeroth order approximation which we perturb to improve its realm of applicability and accuracy. Higher order improvements could be KWIK or as recently reported, CAP's. The Coulomb Attenuated Potentials [152] model the long-range operator by a sum of Gaussian functions which yield ERI's easily handled by slight modifications to current integral code.

Gill [153] has recently indicated that the crude neglect of the long-range partition by CASE also neglected significant short-range character which causes the large errors in total energies (the long-range operator containing short-range character). He has suggested that a better zero order CASE would be that of the CAP(1/2) approximation, which is still short-ranged and is evaluated in closed form. The result is dramatically improved total energies and ionisation energies for no increase in computational cost.

8.3.2. Approximating the Coulomb Operator

Firstly, it is necessary to define what is meant by “Approximating the Coulomb Operator”. By approximate we mean that the total Coulomb energy is determined to within some predetermined level of accuracy, ϵ . That is, if the total Coulomb energy was determined using the Coulomb operator as written, the same total energy would be obtained using the alternative method, to within some level of tolerance. It is important to emphasise here that a computer has limited finite accuracy, and no real number is ‘exactly’ (to infinite precision) represented by a computer.

While we can guarantee that the total Coulomb energy will be accurate to some predetermined level of accuracy, no guarantee is placed on the accuracy of the derivatives, *i.e.* the Coulomb matrix elements. Therefore, this section is concerned with determining the total long-range KWIK energy to within some predetermined level of accuracy. The long-range KWIK energy between two Gaussians can be written as follows (adopting the Einstein summation convention and Dirac notation)

$$\begin{aligned}
 & \left(\frac{\alpha\beta}{\pi^2}\right)^{3/2} \iint \exp(-\alpha r_1^2) \left(\frac{\text{erf}(\omega r_{12})}{r_{12}}\right) \exp(-\beta|\mathbf{r}_2 - \mathbf{B}|^2) d\mathbf{r}_1 d\mathbf{r}_2 \\
 &= \left(\frac{\alpha\beta}{\pi^2}\right)^{3/2} \left\langle \exp(-\alpha r_1^2) \left| \frac{\text{erf}(\omega r_{12})}{r_{12}} \right| \exp(-\beta|\mathbf{r}_2 - \mathbf{B}|^2) \right\rangle \\
 &= \left\langle \left(\frac{\alpha}{\pi}\right)^{3/2} \exp(-\alpha r_1^2) \left| e_j \right\rangle \left\langle e_j \left| \left(\frac{\beta}{\pi}\right)^{3/2} \exp(-\beta|\mathbf{r}_2 - \mathbf{B}|^2) \right\rangle \right\rangle
 \end{aligned} \tag{8.29}$$

which can be expressed more concisely as

$$\left(\frac{\alpha\beta}{\pi^2}\right)^{3/2} \iint \exp(-\alpha r_1^2) \left(\frac{\text{erf}(\omega r_{12})}{r_{12}}\right) \exp(-\beta|\mathbf{r}_2 - \mathbf{B}|^2) d\mathbf{r}_1 d\mathbf{r}_2 = \hat{Q}_\alpha^j \hat{Q}_\beta^j \tag{8.30}$$

as we have written for the matrix elements in section 8.1. The latter are simply the products of the Fourier transforms of the two Gaussians evaluated over the Fourier sum. For a single term this is

$$\hat{Q}_\beta^j = \exp\left(-|\mathbf{k}_j|^2/4\beta\right)\exp(i\mathbf{k}_j \cdot \mathbf{B}) \quad (8.31)$$

8.4. Self Consistent Field Calculations

KWIK offers many possibilities for further improving the speed of SCF calculations far beyond the reduction from $O(N^2) \rightarrow O(N)$ cost. The Q-Chem program [154], for example, increases the error tolerance in early SCF cycles and reduces it as the density reaches self consistency. KWIK could further reduce the computation time for early cycles by use of CASE or CAP's. Given the quality of MO's obtained using the CASE approximation [150], higher order corrections may not be required until the last few cycles, at which time the long-range partition could then be slowly built up. For example, say at a DIIS [155-157] error of $1.0e-4$ the discrete FT's of basis function pairs could be formed, saved to disk and incorporated into the \mathbf{J} matrix for (say) up to $|\mathbf{k}|=0.5|\mathbf{k}_{\max}|$. While this would be 50% of the long-range intermediates formed, from the random walk argument of Chapter Six and the slow convergence of the Fourier Series, the quality of the resulting \mathbf{J} matrix would be significantly improved from the CASE approximation, and it is likely that the addition of a single extra cycle on reaching DIIS convergence of 10^{-6} , incorporating the remaining long-range intermediates would acquire the total energy to within the required accuracy.

8.5. Quantum Chemistry Using Slater Functions

8.5.1. STO's

Slater Type Orbitals have long been considered as the more favoured basis in which to expand MO's displaying the features of exact SCF MO's with cusps at the nuclei and exponential decay far from the nuclei. In addition, many fewer STO's are needed [19] than GTO's to achieve a given accuracy. While contracted GTO's [158] are able to remedy much of the functional deficiencies of primitive GTO's, they present a new computational problem. Furthermore, while GTO's have been shown to be capable

of yielding satisfactory molecular geometries and relative molecular energies, certain molecular properties, such as those involving the core electrons are significantly less effectively modelled by GTO's than STO's.

As discussed in Chapter Three, STO's have effectively been discarded as basis functions due to the intractability of the ERI's in non-linear molecules. The nature of the function, however, has always maintained prominence and directly led on to the STO-KG basis functions. However, to model a single Slater AO to 10^{-10} accuracy requires a sum of 27 Gaussians [159].

More recently, Adams *et al* [160] considered the approach of modelling the STO products

$$\rho(\mathbf{r}) = \phi_\mu \phi_\nu \quad (8.32)$$

by a sum of Gaussians

$$\hat{\rho}(\mathbf{r}) = \sum_{i=1}^K \beta_i \exp(-\alpha_i r^2) \quad (8.33)$$

in a least-squares sense

$$Z_\theta = \iint e(\mathbf{r}_1) \theta(r_{12}) e(\mathbf{r}_2) d\mathbf{r}_1 d\mathbf{r}_2 + \lambda \int e(\mathbf{r}) d\mathbf{r} \quad (8.34)$$

where

$$e = \rho - \hat{\rho} \quad (8.35)$$

the second term is a Lagrange multiplier to ensure conservation of charge and the $\theta(r_{12})$ determines the sense in which least-squares is minimised. Minimisation of (8.34) with respect to the β_i 's and α_i 's will yield models (8.33) which are minimal with respect to the residual density, electric field and potential of (8.35) for $\theta(r_{12})$'s corresponding to the delta, Coulomb and anti-Coulomb operators [68] respectively. It was found that the minimisation with respect to the potential of (8.35) led to the superior models where a

sum of just 12 Gaussians modelled the STO product (8.32) to afford an error of less than 10^{-10} in the ERI, significantly less than 27^2 for the density modelling of the AO's.

How does KWIK aid in this matter? Firstly, the use of STO's would still exhibit an $O(N^2)$ Coulomb problem in SCF calculations. If KWIK was used to overcome this, the short-range integrals over the Slater products will remain intractable, thus requiring the models outlined above. The long-range piece however, would not require this modelling as only the Fourier transform of the Slater product is required. The decay parameter could be adjusted so that the work rate of the short-range partition was decreased, as the integrals over 12 contracted Gaussian functions require more work than that of evaluating the FT of a single Slater product.

8.5.2. Slater Function Density

The advocates of Density Functional Theory claim that the ground state energy can be obtained as the minimum of the energy functional

$$E[\rho] = \int \rho(\mathbf{r})v(\mathbf{r}) d\mathbf{r} + F[\rho] \quad (8.36)$$

where v is the external potential and

$$F[\rho] = T[\rho] + V_{ee}[\rho] \quad (8.37)$$

where $T[\rho]$ is the kinetic energy. However, the kinetic energy term has proven a difficulty in that models such as the Thomas-Fermi one are crude approximations. The solution to this problem was the introduction of an orbital basis, as suggested by Kohn and Sham [42]. This remains the basis of DFT methodology at present, but given a more suitable kinetic energy functional it is likely that nuclear centred Slater-type densities will be adopted as a basis for the molecular density and the Kohn-Sham formalism will disappear. Remaining however, will be the $O(N^2)$ Coulomb bottle-neck to which KWIK will still be applicable.

If the density is constructed as a linear combination of Slater densities

$$\rho(\mathbf{r}) = \sum_{j=1}^M P_j (\alpha_j^3 / 8\pi) \exp(-\alpha_j |\mathbf{r} - \mathbf{A}_j|) \quad (8.38)$$

then the density self repulsion energy can be written

$$E_J = \langle \rho(\mathbf{r}_1) | \frac{1}{r_{12}} | \rho(\mathbf{r}_2) \rangle \quad (8.39)$$

substituting the KWIK operator (8.5) into (8.39)

$$\begin{aligned} E_J &= \langle \rho(\mathbf{r}_1) | \frac{\text{erfc}(\omega r_{12})}{r_{12}} | \rho(\mathbf{r}_2) \rangle + \langle \rho(\mathbf{r}_1) | e_j(\mathbf{r}_1) \rangle \langle e_j(\mathbf{r}_2) | \rho(\mathbf{r}_2) \rangle \\ &= \langle \rho(\mathbf{r}_1) | \rho(\mathbf{r}_2) \rangle_S + \hat{\rho}_j \hat{\rho}_j^* \end{aligned} \quad (8.40)$$

where

$$\hat{\rho}_j = \frac{P_l \alpha_l^4}{(\alpha_l^2 + |\mathbf{k}_j|^2)^2} \exp(i\mathbf{k}_j \cdot \mathbf{A}_l) \quad (8.41)$$

is the FT of the Slater density at \mathbf{k}_j .

If an appropriate kinetic energy functional is obtained, the CFMM will also be able to aid in rapidly determining the Coulomb energy. Unfortunately the GvFMM has been limited to Gaussian functions and will require re-working.

8.5.3. Stewart-Slater Atoms

A problem frequently encountered by MQM is the chemist's desire to consider molecules in terms of their constituent atoms. Schrödinger's equation makes no reference to these atoms and it has been generally accepted that it is not possible to extract atoms out of molecules *via* the basic postulates of quantum mechanics. Recently, Gill [161] has devised a method of extracting nuclear centred spherical functions (Stewart atoms) whose sum best fits a molecular density in a least-squares sense. Due to difficulties encountered in extracting the exact Stewart atoms, approximate Stewart-Slater atoms are employed such that the density is represented as a sum of Slater functions.

The possibility arises of replacing the true electron density by the Stewart-Slater density for purposes of calculating the Coulomb energy. If the Stewart-Slater Coulomb

energy is a good approximation to the true Coulomb energy, the work required to calculate this will be significantly reduced and KWIK will also be able to be used to further reduce computational cost.

8.6. KWIK in Correlated methods

The initial CASE investigations [150] not only considered Coulomb attenuation in HF calculations, but also that in determining MP2 correlation energies. The MP2 correlation energy is given by

$$E_0^{(2)} = \frac{1}{4} \sum_{abrs} \frac{\left| \langle \Psi_0 | \sum_{i<j} r_{ij}^{-1} | \Psi_{ab}^{rs} \rangle \right|^2}{\varepsilon_a + \varepsilon_b - \varepsilon_r - \varepsilon_s} \quad (8.42)$$

and the approach adopted was to use the true HF MO's, the MP2 energy was obtained by replacement of the Coulomb operator in (8.42) by the short-range KWIK operator, affording correlation energies in good, but systematically lower, agreement with the true MP2 energies. However, as for the HF CASE energies, no computational advantage had been implemented. A computational advantage is readily obtained for the HF case *via* the employment of boxing techniques. The matter of how to apply the short-range cut-off to improve the scaling of the attenuated MP2 energy calculation is a little more difficult due to the energy denominators. Laplace transform techniques have been proposed [162,163] as a method of removing this denominator, but a numerical integration is then required.

The attenuation of the Coulomb operator is an exciting prospect for perturbation theory and for localised treatments in general [164,165] especially considering the small absolute error invoked. However, the computational advantage is somewhat more difficult to implement and will require significant effort.

In terms of the KWIK algorithm, disregarding the long-range term in the attenuated approaches, such as CASE and CAP's, is moving away from where the KWIK algorithm holds its computational advantage - the relaxation of functions like $I(k)$ to its random walk expectation value. And in the case of correlation energy it is

likely this advantage holds, as although the molecule remains net-neutral, all the interactions are purely electron-electron.

Does KWIK have a place in correlated techniques? The first response would be to answer no, because it appears that correlation is a localised phenomenon and in the Coulomb energy application KWIK was used to handle the long-range piece. However, let us return to the statistics problem described in Chapter Four. "... the underlying success of the method can be attributed to the fact that although the total number of possible A_i 's grows as $O(n^m)$, they are highly dependent as they have been generated from only n independent values." A split was applied for Coulomb energies because the Fourier series for the Coulomb operator decayed slowly and the "... total number of interactions ... generated from the independent values" was nowhere near as great as in the statistics problem, where this number was enormous. Therefore, convergence was necessarily aided by a more rapidly decaying long-range separator.

We should now recall that the conventional approach to correlation is to obtain the HF wavefunction and consider electron excitations in a configuration interaction approach, coupled cluster approach or perturbatively. All methods consider a large number of contributions, but derived from a much smaller number of independent quantities. Given that there are likely to be a very large number of contributions, the decay of the associated Fourier series (whatever that may be) will be much less important. Exactly how the KWIK philosophy may be applied to correlation is yet to be addressed.

CHAPTER NINE

CRITICAL ANALYSIS

9.0. Introduction

The KWIK algorithm has been derived for Coulomb energies for arbitrary distributions, applied to one and three dimensionally distributed point charges and to bottle-necks in self-consistent field calculations, and implemented for one and three dimensional point charges. In this final Chapter, the aim is to give a critical account of the algorithm, note deficiencies and how it may be improved, as well as comparing and contrasting KWIK with its peers. Particular emphasis is placed on its application to quantum chemistry.

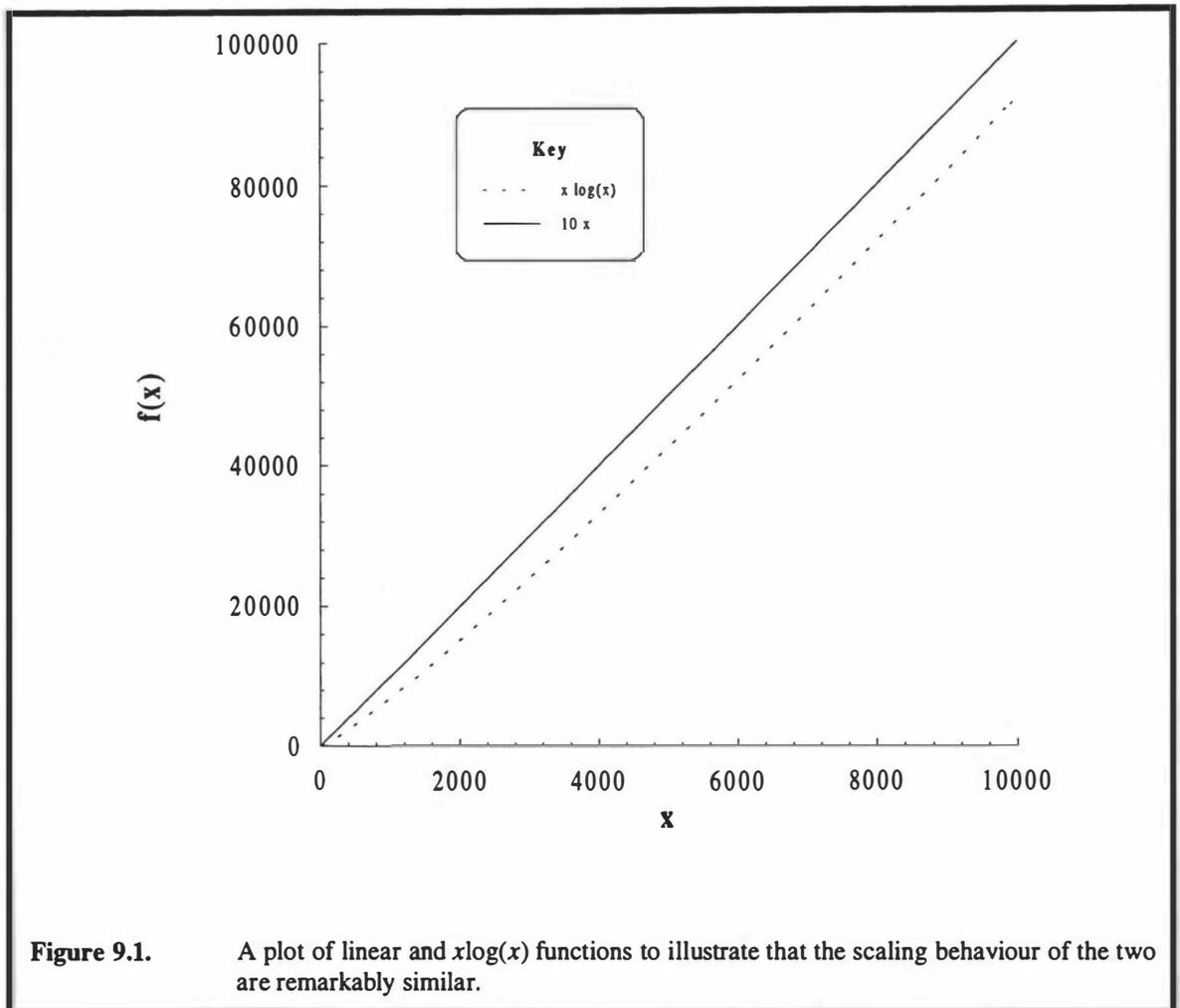
9.1. Scaling

The KWIK algorithm originates from a combinatorial statistical problem [126] (see also Chapter Four) in which the desired probability was written as a Fourier series. Whilst the Fourier series coefficients decay slowly, the inner sum was found to decay extremely rapidly. Initially, the inner sum begins at $({}^n C_m)^2$, but as the Fourier series is expanded this inner sum rapidly decreases to a noisy value of ${}^n C_m$, obtaining the required probability to high accuracy for small expansions of the Fourier series, thus in rapid time. The phenomenon of this rapid convergence observed by Gill has, in this thesis, been linked to a random walk in the complex plane.

In the application of the methodology of Gill to the Coulomb problem for charged systems, it has been found empirically for relatively large distributions, the algorithm scales linearly with system size else, for smaller systems, it scales super-linearly. Whatever the size of the system, total Coulomb energies can be afforded significantly more rapidly using KWIK than with the classical quadratic method for other than the most trivial system size. No formal proof for the scaling observation (*i.e.* that it is rigorously linear) has been presented with the linearity argument based upon decay characteristics of the Fourier intensity $I(k)$. However, it *may* be possible that the rate of

relaxation from N^2 to the noisy random walk value of N is dependent on the size of the system, which may lead to ‘formal’ super linear scaling such as $O(N\log(N))$.

The argument of formal cost scaling has also been directed towards the FMM [166], where it is explicitly stated in the title that “Greengard’s N -body algorithm is not order N .” Whatever the case may be for KWIK and FMM, the difference between an $O(N)$ algorithm and an $O(N\log(N))$ algorithm is difficult to judge empirically at present (see Figure 9.1), where the range of application is ‘limited’ to perhaps millions of particles due to machine hardware restraints, rather than time constraints, which is the case for quadratic methods.



The concern with respect to cost scaling behaviour for the sub-quadratic approaches (FMM, KWIK) should be the magnitude of the coefficient. The plot in Figure 9.1 has been carefully constructed to imply $O(N\log(N))$ is similar, if not superior to an $O(N)$ function by careful choice of coefficients. Equally, the plot could have

assigned a large coefficient to the $O(N \log(N))$ slope to indicate that an algorithm with this scaling is completely undesirable. Conversely, $O(N^2)$ ERI code may maintain superiority over $O(N)$ code over a suitable range of applicability. That is, the coefficient of the linear method is so large that the linear method is computationally more expensive than the quadratic method for all suitably sized molecules.

To summarise this argument, cost scaling must be held in context. KWIK is undoubtedly faster than the classical quadratic approach to Coulomb energies, as is the FMM. The aim of both the algorithms is to rapidly evaluate the Coulomb field of large systems of charged particles. If they do formally scale non-linearly, the success of these algorithms has not been affected by this apparent unfavourable formal scaling.

In the case of net-neutral systems, the auxiliary function $I(k)$ is not a rapidly decaying inner sum. For these systems we have found that the KWIK algorithm scales in a similar fashion to that of moderately sized charged systems and the PPPM algorithm. The Ewald summation technique scales weakly as $O(N^2)$ as the 'split' is applied so that the real space work considers only particles within a single simulation cell and not its images. That is, the decay parameter is fixed. Similar $O(N^2)$ scaling was noted for KWIK when the decay parameter was fixed for neutral systems rather than optimised with the degree of Fourier expansion. While KWIK's empirical scaling is worse for net-neutral systems than for charged systems, its total cost remains substantially cheaper than the quadratic alternative.

9.2. Significant Findings

KWIK has been shown to be an excellent method for rapidly obtaining the Coulomb energy of large systems of particles exhibiting orders of magnitude speed improvement over the classical quadratic method. High accuracy calculations can now be conducted routinely for systems upwards of millions of particles. The swift evaluation of energies of such systems is attributable to the rapid decay of the Fourier intensity of the charge distribution which is linked to a random walk in the complex plane. This is irrespective of the lack of a rigorous proof of linear cost scaling.

On application of KWIK to quantum chemistry it has been clearly illustrated that KWIK can be used to rapidly determine the Coulomb and Exchange matrices. The Exchange matrix has not previously been calculable by non-approximate sub-quadratic methods. (An approximate sub-quadratic method would be one which makes an assumption about some feature of the exchange reducing its scaling behaviour. A non-approximate method would calculate the matrix elements to within some predetermined accuracy.)

The results of the application of the CASE approximation is one of *the* most significant findings in quantum chemistry in recent times. Disregarding such a large part of the Coulomb operator with massive errors induced in total energies to afford chemistry to such high accuracy, is in the least, surprising.

CASE is clearly linear.

9.3. KWIK vs The Rest

9.3.1. KWIK vs Ewald

KWIK is not a competitor of the Ewald summation technique. Ewald is designed for net-neutral, periodic systems. KWIK is not designed for periodic systems and can be used to consider charged systems (infinite systems cannot be charged - they have infinite energy). Furthermore, the success of KWIK is based on the relaxation to a random walk expectation value, not the random walk expectation value itself. This relaxation phenomenon is not observed for neutral systems.

The major difference between KWIK and the Ewald summation technique is that KWIK uses a Fourier series to model the long-range operator, Ewald on the other hand uses a Fourier Series to model the long-range potential. Ewald then exploits the periodicity of the Fourier series to aid in overcoming surface effect problems. KWIK makes no use of the Fourier series periodicity and, in fact, avoids it.

Delley [110] has indicated that the Ewald k-space cut-off is aided by the random walk feature (the k-space partitions of KWIK and Ewald have similar features - $I(\mathbf{k})$). This is not true for non-regular arrangements as has been shown for neutral systems in

Chapter Seven. As a final remark on Delley's paper, Luty [105,112] has shown for electrostatics on a grid within Ewald summation applications, the scaling is not linear (mildly sub-quadratic) which contrasts with Delley's claim of linearity.

9.3.2. KWIK vs FMM

The most important comparison to be made is that of KWIK with the FMM as both have been directed towards solving similar problems. To date, a direct timing comparison has not been made, but as mentioned in Chapter Three, it is important to consider realms of applicability and prospects of complementary behaviour. Complementary behaviour has often been found for $O(N^2)$ ERI evaluation which has led directly to the PRISM [167] algorithm. For example, some methods are better for high degrees of contraction, others large angular momentum. No single approach is superior for all cases.

All short-range interactions are significant for the FMM. The separation is discontinuous and with the recent advances by White [81], this separation can be made close to optimal. KWIK on the other hand, uses a proximity based short-range separation, which (see Chapter Five) leads to large inefficiencies. That is, all FMM short-range computations are significant and contribute significantly to the total energy, KWIK wastes time computing quantities which do not contribute significantly to the total energy.

In terms of the long-range partition, FMM is truly long-ranged with all contributions arising from distant particles, on the other hand, the long-range KWIK operator is named solely due to its slow decay, and contains short-range character. KWIK's long-range efficiency is dependent on the range of the distribution, so if an outlier particle is added the overall efficiency will be reduced, by virtue of reduced k-vector density [105,112]. The FMM, on the other hand, can easily add in outliers without significant adverse effects on efficiency [116].

Factors which favour KWIK are memory requirements and simplicity. The memory requirements for the long-range FMM requires the storage of high degree multipole expansions, which is memory intensive. Long-range KWIK, if so desired, can

have its memory requirements reduced to that of only the particle coordinates. The disadvantage is a reduction in speed.

KWIK is a very simple algorithm in concept. It requires particle interactions *via* a short-range interaction potential, and long-range contributions requiring only Fourier coefficients with the remaining long-range components constructed from trigonometric functions and built up using simple recursion relations. The FMM requires multipole expansions, conversions to a localised Taylor expansion with complicated (but ingenious) passes and translations. In summary, a crude KWIK algorithm could be coded up significantly faster and more efficiently than a crude version of the FMM.

FMM also has the advantage of being a more mature algorithm - at least ten years senior. Thus, many aspects of the algorithm have been refined and re-refined by a number of workers to increase the efficiency and accuracy of the algorithm. However, Chapter Seven has indicated that KWIK has already gone through a number improvements since its inception - a direct comparison of the methods two years ago were far less favourable than at present (factors of two for 3-D systems) - and further improvements will come.

9.3.3. KWIK *vs* CFMM

The CASE approximation has shown virtue as a method in its own right and, as discussed in Chapter Eight, a means to reducing the cost of early SCF cycles. The CASE approximation is successful because the KWIK separation is continuous. Applying a similar neglect of long-range contributions from the CFMM (or FMM) will lead to massive errors which would imply that the CFMM would be more difficult to use to build up accuracy in the early iterations of SCF calculations in manner that KWIK can be.

KWIK can also be used to rapidly evaluate the Exchange matrix which cannot be done using the CFMM. In defence of the CFMM, it can and, unlike KWIK, has been used in DFT calculations which have the added advantage of having correlation incorporated into the calculation. While KWIK can also be used in such calculations, it may or may not add any significant speed-up.

Molecules have a (near) net-neutral charge, which from our particle studies indicate that KWIK will lack the ‘linearity’ for relatively large systems. However, the bottle-neck lies heavily towards evaluating the Coulomb and Exchange matrices, both of which involve interactions of species of a single charge sign. That is the number of $O(N^2)$ terms and the coefficient for evaluating Coulomb and Exchange terms is much greater than that of nuclear attraction and nuclear repulsion contributions. So, like the Hamiltonian diagonalisation, where the asymptotic cost scaling will be greater for these terms if constructed classically, the cost of evaluating Coulomb and Exchange terms using KWIK will still remain substantial for molecules that we wish to subject to calculation. CFMM on the other hand scales similarly for neutral and charged systems.

On the introduction of new algorithms for calculating Coulomb interactions, linear or non-linear, comparisons should be made in an attempt to identify the areas where the new algorithm can be used to strengthen weaknesses of current algorithms, so that new and old may be used to complement, rather than compete. For example, CASE could be used in early SCF cycles to get the MO’s converging to self-consistency, followed by CFMM for converging the energetics if CFMM is found to be superior to KWIK. Whether KWIK is or is not faster than CFMM, given the statistical basis of KWIK, it ought to be possible to fabricate a molecule whereby the CFMM is significantly faster than the KWIK algorithm and vice-versa.

9.4. Further Work Directions

Since Greengard’s thesis, the FMM has been applied in many areas previously restricted by an N -body bottle-neck and the algorithm has been through many radical changes which have improved its efficiency and generalisation. KWIK, on the other hand, is a young algorithm which has shown great potential and has areas where significant improvements could be made.

- A more detailed investigation into the relaxation of the $I(\mathbf{k})$ to its random walk expectation value needs to be considered. KWIK’s advantage lies in the rapid decay of $I(\mathbf{k})$, and as we have seen for highly polarised distributions, this advantage is lost.

Relationships need to be developed between the statistical distribution of a system and this relaxation phenomenon.

- The decay parameter and degree of Fourier expansion need to be predetermined. It was found empirically (similar to the PPPM methodology) that the total time was minimised by an approximate 50/50 split of CPU time between the short- and long-range partitions. However, if the relaxation of $I(k)$ is slow, this mix will require significantly different (optimal) parameters for similar sized distributions. Furthermore, the nature of the platform will also have a part to play. It was noted in a brief trial, that on an IBM RS/6000 model 43P, which has specifications implying that it is approximately 2.5 times faster than the IBM RS/6000 model 355, the short-range code was 2.5 times faster but the long-range code ran at similar times. Similarly, vectorisation of the KWIK algorithm will see that the long-range code is highly vectorisable, with long inner loops over N , whereas the short-range code will require careful tuning. The advantage with KWIK is that the short/long split is continuous so that the decay parameter, and degree of Fourier expansion can be carefully adjusted for differing platforms and architectures to minimise total time requirements.
- The formal scaling of the algorithm needs to be elucidated. Does the relaxation phenomenon have an N dependence for similarly distributed systems? In investigations undertaken, it was sought to illustrate for relatively low accuracies that the formal cost scaling was linear and from an empirical standpoint this is correct. However, the short-range code became 'choked' on the scalar machine, where increases in the decay parameter (reduced cut-offs) saw increased CPU times for short-range work. This would indicate very short inner loops and it is possible that the implementation of Rapaport's layer algorithm may improve matters. On the other hand, for relatively low accuracy, KWIK calculated total energies extremely rapidly. That is, the $O(N^2)$ coefficient is small, emphasising the context of algorithm scaling.

- The `erfc` function has been implemented as the separator, previously not having determined the ultimate separator. The improved efficiency on the use of the ultimate separator should be addressed and will serve as a useful benchmark.
- It is clear from Chapter Five how best to evaluate short-range interactions with interpolation of the potential and suitable boxing schemes. The long-range partition remains more open. The use of the Fourier series has been maintained after significant investigations presented in Chapter Six where the Fourier series was likened to a Simpson type quadrature rule. While this has allowed the implementation of efficiencies based on the even placement of roots, it is well known that such placement is inherently inefficient in terms of the number of roots required.
- It has been found with the CASE approximation and in KWIK calculations on net-neutral particle systems that the long-range contribution to chemistry (and total energy) is minimal. However, in the KWIK operator split, computational time requirements were apportioned equally for evaluating both the long- and short-range partitions. Either the manner in which relative contributions are measured in terms of their magnitude is incorrect or, the way energy contributions are separated is inefficient. My belief is that the former is incorrect. Consider, for example the correlation energy. Calculating the correlation energy more precisely in a conventional sense requires enormous computational effort for approximately 1% of the total energy. However, this energy has been found to be extremely important in many areas of chemistry. On the other hand, it has been found that neglect (HF calculations) or partial neglect (MP2) of correlation, introduces an acceptable error in the chemistry. Can the long-range partition be treated in a similar fashion? Already in the CASE and CAP approximations realms of applicability have been illustrated. CASE yielding acceptable chemistry for molecule energies, poor chemistry for ionisation potentials. The latter corrected acceptably by the addition of a constant.
- The long-range partition needs to be improved upon for the purposes of determining accurate Coulomb energies. It is possible that methods which encompass the

observations from the CASE approximation may lead to such improvements. The fact that the KWIK long-range partition has short-range character needs to be addressed - thus the first order approximations to the long-range operator should be short-range in character. Then look towards identifying a long range approach that on increasing complexity captures the parts of the long-range contribution which are chemically important. It may be that the cosine modelling of the long-range operator by matching derivatives is a suitable contender.

- The nature of the long-range partition, whether it be a Fourier series or Fourier transform with some ingenious quadrature scheme, requires evaluating $I(\mathbf{k})$ at some given value of \mathbf{k} . This requires looping over all particles repeatedly which is particularly vectorisable as mentioned, but is also amenable to parallelism. Effectively, very similar jobs are required to be completed, simultaneously, each with identical computational cost. This is a perfect contender for use on a parallel machine. For small numbers of processors, each processor could be assigned a certain root or number of roots at which to evaluate the $I(\mathbf{k})$ function or alternatively, for massively parallel machines, each processor assigned a particle, of which the FT need be evaluated for all roots.

9.5. Summary

The KWIK algorithm for Coulomb interactions has been presented with a number of applications. KWIK has been shown to be a useful tool for the rapid evaluation of the Coulomb field of large systems of charged matter, the nature of the method indicating that very large systems become an advantage for obtaining increased accuracy. The underlying success of the method is related to the relaxation of the Fourier intensity of the charge distribution to a random walk expectation value. This indicates a statistical quality which may allow an optimal short-long split to be tailor-made by exploitation of simple distribution properties of the charge density.

KWIK was found to be superior to alternative methods, such as the (C)FMM, in a number of ways. For example:

- The increase in work with increasing dimension is approximately an order of magnitude less than the FMM.
- Memory requirements of KWIK are moderate, whereas the FMM storage requirements are large and are dependent on the accuracy required.
- KWIK can be used to determine the non-classical exchange matrix elements in a HF calculation, which is not possible with the CFMM.

A number of important applications in quantum chemistry have been illustrated, including the ease of which matrix elements can be derived by employing the compact Dirac notation. In addition, the immaturity of the KWIK algorithm has been highlighted by indicating the scope for improvement in specific areas. In particular, linear scaling relying on the relaxation of the Fourier intensity to a random walk expectation value is not yet fully understood.

APPENDIX A

DATA FOR THE SHORT-RANGE INVESTIGATIONS

Presented in Table A.1 to A.3 are the efficiency data for Figures 5.4 and 5.5 as well as the number of significant near-neighbour cells.

k	Near Cells	Efficiency	Rel.
1	7	0.1727	0.38
2	19	0.2546	1.54
3	31	0.3511	3.46
4	37	0.5229	6.16
5	55	0.5496	9.62
6	73	0.5963	13.86
7	91	0.6511	18.86
8	121	0.6396	24.63
9	151	0.6486	31.18
10	163	0.7418	38.49
11	199	0.7352	46.57
12	235	0.7410	55.43
13	265	0.7711	65.05
14	301	0.7874	75.44
15	361	0.7537	86.60
16	379	0.8168	98.53
17	433	0.8071	111.24
18	475	0.8248	124.71
19	511	0.8542	138.95
20	583	0.8296	153.96
21	649	0.8217	169.74
22	685	0.8544	186.29
23	745	0.8586	203.61
24	817	0.8525	221.70
25	859	0.8798	240.56

Table A.1. Theoretical efficiencies and number of near neighbour cells for the use of Hexagons as the parent cell in 2-D short-range calculations.

k	Near Cells	Efficiency	Rel.
1	9	0.3491	1
2	25	0.5027	4
3	49	0.5770	9
4	77	0.6528	16
5	109	0.7205	25
6	157	0.7204	36
7	201	0.7659	49
8	265	0.7587	64
9	321	0.7927	81
10	385	0.8160	100
11	461	0.8246	121
12	533	0.8488	144
13	621	0.8550	169
14	713	0.8636	196
15	817	0.8652	225

Table A.2. Theoretical efficiencies and number of near neighbour cells for the use of Squares as the parent cell in 2-D short-range calculations.

k	Near Cells	Efficiency	Rel.
1	13	0.4186	1.73
2	37	0.5883	6.93
3	73	0.6709	15.59
4	121	0.7195	27.71
5	181	0.7516	43.30
6	253	0.7743	62.35
7	337	0.7912	84.87
8	421	0.8272	110.85
9	529	0.8332	140.30
10	649	0.8384	173.21

Table A.3. Theoretical efficiencies and number of near neighbour cells for the use of Triangles as the parent cell in 2-D short-range calculations.

APPENDIX B

DERIVATION OF NUMERICAL APPROXIMATION FOR DETERMINING THREE DIMENSIONAL FOURIER COEFFICIENTS

From Chapter Six, we have the problem of not being able to obtain an analytic solution to the integral (6.75)

$$\int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \frac{\exp(i\mathbf{k} \cdot \mathbf{r})}{|\mathbf{r}|} dx dy dz \quad (\text{B.1})$$

which we call I and rewrite as

$$I = \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \frac{\exp(i(k_1x + k_2y + k_3z))}{r} dx dy dz \quad (\text{B.2})$$

What is required is a highly accurate numerical approximation, which can be evaluated quickly. To do this [146] we begin by considering the double tetrahedron T_Z with vertices $(-1,-1,1)$, $(-1,1,1)$, $(1,1,1)$, $(1,-1,-1)$, $(0,0,0)$ for the upper part, and $(-1,-1,-1)$, $(-1,1,-1)$, $(1,1,-1)$, $(1,-1,-1)$, $(0,0,0)$ for the lower part. We now define

$$I_Z = \iiint_{T_Z} \frac{\exp(i(k_1x + k_2y + k_3z))}{r} dx dy dz \quad (\text{B.3})$$

Similarly, if we consider the tetrahedra T_X and T_Y aligned with the x and y axes with analogously defined I_X and I_Y . Then

$$I = I_X + I_Y + I_Z \quad (\text{B.4})$$

therefore

$$I_X(k_1, k_2, k_3) = I_Z(k_3, k_2, k_1) \quad (\text{B.5})$$

and

$$I_Y(k_1, k_2, k_3) = I_Z(k_1, k_3, k_2) \quad (\text{B.6})$$

thus we need only focus on I_Z .

It can be easily shown that our focus integral can be written as

$$\begin{aligned} I_Z &= \int_{-\pi}^{\pi} \int_{-|z|}^{|z|} \int_{-|z|}^{|z|} \frac{\exp(i(k_1x + k_2y + k_3z))}{r} dx dy dz \\ &= \int_{-\pi}^{\pi} \int_{-z}^z \int_{-z}^z \frac{\exp(i(k_1x + k_2y + k_3z))}{r} dx dy dz \end{aligned} \quad (\text{B.7})$$

By letting $x = Xz$ and $y = Yz$ yields

$$I_Z = \int_{-\pi}^{\pi} \int_{-1}^1 \int_{-1}^1 \frac{|z|}{\sqrt{1 + X^2 + Y^2}} \exp(iz(k_1X + k_2Y + k_3)) dY dX dz \quad (\text{B.8})$$

and by reordering the integration and a change in notation

$$I_Z = \int_{-1}^1 \int_{-1}^1 \int_{-\pi}^{\pi} \frac{|z|}{\sqrt{1 + x^2 + y^2}} \exp(iz(k_1x + k_2y + k_3)) dz dy dx \quad (\text{B.9})$$

We can then integrate over z to yield

$$\begin{aligned} I_Z &= \int_{-1}^1 \int_{-1}^1 \frac{1}{\sqrt{1 + x^2 + y^2}} \left[\frac{2\pi \sin(\pi(k_1x + k_2y + k_3))}{k_1x + k_2y + k_3} + \right. \\ &\quad \left. \frac{2(\cos(\pi(k_1x + k_2y + k_3)) - 1)}{(k_1x + k_2y + k_3)^2} \right] dy dx \end{aligned} \quad (\text{B.10})$$

It is then convenient to write

$$(k_1, k_2, k_3) = k(\cos \chi \sin \psi, \sin \chi \cos \psi, \sin \psi) \quad (\text{B.11})$$

so that

$$\left[-\sin \chi \frac{\partial}{\partial x} + \cos \chi \frac{\partial}{\partial y} \right] f(k_1x + k_2y) = 0 \quad (\text{B.12})$$

where f is arbitrary. Note also that

$$\begin{aligned} & \left[-\sin \chi \frac{\partial}{\partial x} + \cos \chi \frac{\partial}{\partial y} \right] \operatorname{arcsinh} \left(\frac{-x \sin \chi + y \cos \chi}{\sqrt{1 + (x \cos \chi + y \sin \chi)^2}} \right) \\ &= \frac{1}{\sqrt{1 + x^2 + y^2}} \end{aligned} \quad (\text{B.13})$$

therefore

$$\begin{aligned} I_z = \iint_R & \left[-\sin \chi \frac{\partial}{\partial x} + \cos \chi \frac{\partial}{\partial y} \right] \operatorname{arcsinh} \left(\frac{-x \sin \chi + y \cos \chi}{\sqrt{1 + (x \cos \chi + y \sin \chi)^2}} \right) * \\ & \left[\frac{2\pi \sin(\pi(k_1x + k_2y + k_3))}{k_1x + k_2y + k_3} + \frac{2(\cos(\pi(k_1x + k_2y + k_3)) - 1)}{(k_1x + k_2y + k_3)^2} \right] dx dy \end{aligned} \quad (\text{B.14})$$

where R is the unit square. Applying Green's theorem in the plane reduces the double integral in (B.14) to two single integrals

$$\begin{aligned} I_z = & \int_{-1}^1 -\sin \chi \left[\operatorname{arcsinh} \left(\frac{-x \sin \chi + y \cos \chi}{\sqrt{1 + (x \cos \chi + y \sin \chi)^2}} \right) * \right. \\ & \left. \left\{ \frac{2\pi \sin(\pi(k_1x + k_2y + k_3))}{k_1x + k_2y + k_3} + \frac{2(\cos(\pi(k_1x + k_2y + k_3)) - 1)}{(k_1x + k_2y + k_3)^2} \right\} \right]_{x=-1}^{x=1} dy \\ & + \int_{-1}^1 \cos \chi \left[\operatorname{arcsinh} \left(\frac{-x \sin \chi + y \cos \chi}{\sqrt{1 + (x \cos \chi + y \sin \chi)^2}} \right) * \right. \\ & \left. \left\{ \frac{2\pi \sin(\pi(k_1x + k_2y + k_3))}{k_1x + k_2y + k_3} + \frac{2(\cos(\pi(k_1x + k_2y + k_3)) - 1)}{(k_1x + k_2y + k_3)^2} \right\} \right]_{y=-1}^{y=1} dx \end{aligned} \quad (\text{B.15})$$

The second of these integrals can be written as

$$\begin{aligned}
J(k_1, k_2, k_3) = \int_{-1}^1 \frac{k_1}{\sqrt{k_1^2 + k_2^2}} & \left\{ \operatorname{arcsinh} \left(\frac{-xk_2 + k_1}{\sqrt{k_1^2(1+x^2) + 2k_1k_2x + 2k_2^2}} \right) * \right. \\
& \left(\frac{2\pi \sin(\pi(k_1x + k_2 + k_3))}{k_1x + k_2 + k_3} + \frac{2(\cos(\pi(k_1x + k_2 + k_3)) - 1)}{(k_1x + k_2 + k_3)^2} \right) - \\
& \operatorname{arcsinh} \left(\frac{-xk_2 - k_1}{\sqrt{k_1^2(1+x^2) - 2k_1k_2x + 2k_2^2}} \right) * \\
& \left. \left(\frac{2\pi \sin(\pi(k_1x - k_2 + k_3))}{k_1x - k_2 + k_3} + \frac{2(\cos(\pi(k_1x - k_2 + k_3)) - 1)}{(k_1x - k_2 + k_3)^2} \right) \right\} dx
\end{aligned} \tag{B.16}$$

and the first integral is simply $J(k_2, k_1, k_3)$. Hence we have

$$I_Z = J(k_1, k_2, k_3) + J(k_2, k_1, k_3) \tag{B.17}$$

and

$$\begin{aligned}
I = J(k_1, k_2, k_3) + J(k_2, k_1, k_3) + J(k_1, k_3, k_2) \\
+ J(k_3, k_1, k_2) + J(k_2, k_3, k_1) + J(k_3, k_2, k_1)
\end{aligned} \tag{B.18}$$

For our numerical approximation method it is best if we introduce an alternative to the function J which is

$$\begin{aligned}
Z(k_1, k_2, k_3) = \int_{-1}^1 \frac{2\pi}{\sqrt{2+x^2}} \frac{k_1x + 2k_2}{k_1^2 + k_2^2 + (k_1x + k_2)^2} & \left[g(\pi(k_1x + k_2 + k_3)) + \right. \\
& \left. g(\pi(k_1x + k_2 - k_3)) \right] dx
\end{aligned} \tag{B.19}$$

where the function g is defined as

$$g(x) = \begin{cases} 0, & x = 0 \\ \frac{1 - \cos(x)}{x}, & x \neq 0 \end{cases} \tag{B.20}$$

The integral (B.2) is then obtained as for the J function

$$\begin{aligned}
I = Z(k_1, k_2, k_3) + Z(k_2, k_1, k_3) + Z(k_1, k_3, k_2) \\
+ Z(k_3, k_1, k_2) + Z(k_2, k_3, k_1) + Z(k_3, k_2, k_1)
\end{aligned} \tag{B.21}$$

The key to evaluating (B.19) is replacement of the radical $1/\sqrt{2+x^2}$ by a rational function in x^2 valid for the interval $[-1,1]$. This is done by approximating $1/\sqrt{2+t}$ by a rational function in t over the interval $[0,1]$. Thus, the original radical can be approximated as follows

$$\frac{1}{\sqrt{2+x^2}} \approx a_{0,m} + \sum_{j=1}^m \frac{a_{j,m}}{b_{j,m} + x^2} \quad (\text{B.22})$$

The integral (B.19) can then be evaluated analytically to yield

$$\begin{aligned} Z(k_1, k_2, k_3) = 2\text{Re} & \left[\sum_{j=1}^m a_{j,m} A1(k_1, k_2, k_3, b_{j,m}, d) I1(k_1, k_2, k_3, b_{j,m}) + \right. \\ & \left. \sum_{j=1}^m a_{j,m} A2(k_1, k_2, k_3, b_{j,m}, d) I2(k_1, k_2, k_3, b_{j,m}) + \right. \\ & \left. \left\{ a_{0,m} B1(k_2, k_3, d) + \sum_{j=1}^m a_{j,m} A3(k_1, k_2, k_3, b_{j,m}, d) \right\} I3(k_1, k_2, k_3, d) + \right. \\ & \left. \left\{ a_{0,m} B2(k_2, k_3, d) + \sum_{j=1}^m a_{j,m} A4(k_1, k_2, k_3, b_{j,m}, d) \right\} I4(k_1, k_2, k_3, d) + \right. \\ & \left. \sum_{l=-k_3}^{k_3, 2k_3} \left\{ a_{0,m} B3(k_2, l, d) + \sum_{j=1}^m a_{j,m} A5(k_1, k_2, l, b_{j,m}, d) \right\} I5(k_1, k_2, l) \right] \end{aligned} \quad (\text{B.23})$$

where

$$d = \sqrt{k_1^2 + k_2^2} \quad (\text{B.24})$$

and the functions are defined as follows:

$$A1(k_1, k_2, k_3, a, d) = \frac{2(2k_2 - ik_1a)(k_2 - ik_1a)}{(-2ia)(d^2 + (k_2 - ik_1a)^2)((k_2 - ik_1a)^2 - k_3^2)} \quad (\text{B.25})$$

$$A2(k_1, k_2, k_3, a, d) = \frac{2(2k_2 + ik_1a)(k_2 + ik_1a)}{(2ia)(d^2 + (k_2 + ik_1a)^2)((k_2 + ik_1a)^2 - k_3^2)} \quad (\text{B.26})$$

$$A3(k_1, k_2, k_3, a, d) = \frac{-(k_2 - id)k_1^2}{((k_2 + id)^2 + a^2k_1^2)(k_3^2 + d^2)} \quad (\text{B.27})$$

$$A4(k_1, k_2, k_3, a, d) = \frac{-(k_2 + id)k_1^2}{((k_2 - id)^2 + a^2k_1^2)(k_3^2 + d^2)} \quad (B.28)$$

$$A5(k_1, k_2, k_3, a, d) = \frac{(k_2 - k_3)k_1^2}{((k_2 + k_3)^2 + a^2k_1^2)(k_3^2 + d^2)} \quad (B.29)$$

$$B1(k_2, k_3, d) = \frac{-k_2 + id}{k_3^2 + d^2} \quad (B.30)$$

$$B2(k_2, k_3, d) = \frac{-k_2 - id}{k_3^2 + d^2} \quad (B.31)$$

$$B3(k_2, k_3, d) = \frac{k_2 - k_3}{k_3^2 + d^2} \quad (B.32)$$

$$I1(k_1, k_2, k_3, a) = \log\left(\frac{ia + 1}{ia - 1}\right) - \exp(\pi(k_1a + i(k_2 + k_3))) * \\ 2i \left[\operatorname{Im}(-E_1(\pi k_1(a - i))) - \begin{cases} 0, k_1 > 0 \\ \pi, k_1 \leq 0 \end{cases} \right] \quad (B.33)$$

$$I2(k_1, k_2, k_3, a) = \log\left(\frac{ia - 1}{ia + 1}\right) - \exp(\pi(-k_1a + i(k_2 + k_3))) * \\ 2i \left[\operatorname{Im}(-E_1(-\pi k_1(a + i))) + \begin{cases} 0, k_1 < 0 \\ \pi, k_1 \geq 0 \end{cases} \right] \quad (B.34)$$

$$I3(k_1, k_2, k_3, d) = \frac{1}{k_1} \left[\log\left(\frac{k_1 + k_2 + id}{-k_1 + k_2 + id}\right) - \exp(\pi(ik_3 + d)) * \right. \\ \left. \{E_1(\pi(d + i(k_1 - k_2))) - E_1(\pi(d - i(k_1 + k_2)))\} \right] \quad (B.35)$$

$$I4(k_1, k_2, k_3, d) = \frac{1}{k_1} \left[\log\left(\frac{k_1 + k_2 - id}{-k_1 + k_2 - id}\right) - \exp(\pi(ik_3 - d)) * \right. \\ \left. \{E_1(-\pi(d + i(k_1 + k_2))) + \begin{cases} 2\pi i, k_1 + k_2 > 0 \\ 0, k_1 + k_2 \leq 0 \end{cases} + \right. \\ \left. E_1(-\pi(d + i(-k_1 + k_2))) - \begin{cases} 2\pi i, k_2 - k_1 > 0 \\ 0, k_2 - k_1 \leq 0 \end{cases} \right] \quad (B.36)$$

$$I5(k_1, k_2, k_3) = \frac{1}{k_1} \left[\begin{array}{l} -\log(\pi) - \gamma, k_1 + k_2 + k_3 = 0 \\ \log(i(k_1 + k_2 + k_3)) + E1(-i\pi(k_1 + k_2 + k_3)), k_1 + k_2 + k_3 \neq 0 \end{array} \right] \quad (\text{B.37})$$

$$\left[\begin{array}{l} -\log(\pi) - \gamma, -k_1 + k_2 + k_3 = 0 \\ \log(i(-k_1 + k_2 + k_3)) + E1(-i\pi(-k_1 + k_2 + k_3)), -k_1 + k_2 + k_3 \neq 0 \end{array} \right]$$

Within (B.24-37) i is the imaginary number and γ is the Euler-Gamma constant.

Coefficients in (B.22) and (B.23) are tabulated in Table B.1 for various minimum errors.

j,m	$a_{j,m}$	$b_{j,m}^2$	Error
0,1	0.21279948857675		
1,1	1.38968592022695	2.81161655043445	$4.1 \cdot 10^{-5}$
0,2	0.12759217738641		
1,2	0.69059184081206	2.25699934453100	
2,2	1.81578702500711	6.63818399576723	$2.7 \cdot 10^{-8}$
0,3	0.09111050578366		
1,3	0.46925779209278	2.12683144521070	
2,3	0.73237516225917	3.55372210183254	
3,3	2.38111230961923	12.5803427170724	$1.7 \cdot 10^{-11}$

Table B.1. Table of approximation coefficients for equation (B.22) as obtained using *Mathematica* [168].

APPENDIX C

QUADRATURE RULES AND POLYNOMIALS FOR THE EXPONENTIAL INTEGRAL WEIGHT FUNCTION

Presented in Table C.1. are the first set of 16 roots and weights over the exponential integral weight function for approximating integrals of the type

$$\frac{1}{4\sqrt{\pi}} \int_{-\infty}^{\infty} f(x) E_1(x^2/4) dx \approx \sum_{j=1}^M w_j f(x_j) \quad (\text{C.1})$$

where the x_j and w_j are the roots and weights respectively. The quadrature (C.1) is exact for polynomials $f(x)$, up to degree $2M-1$.

M	Roots	Weights
1	0.00000000E+00	1.00000000E+00
2	$\pm 0.81649658E+00$	0.50000000E+00
3	0.00000000E+0 $\pm 1.89736660E+00$	0.81481481E+00 0.925925923E-01
4	$\pm 0.61901463E+00$ $\pm 2.75043139E+00$	0.48026319E+00 0.19736817E-01
5	0.00000000E+00 $\pm 1.47338538E+00$ $\pm 3.53899365E+00$	0.72794476E+00 0.13235395E+00 0.36736750E-02
6	$\pm 0.52562035E+00$ $\pm 2.20250313E+00$ $\pm 4.21854877E+00$	0.45934097E+00 0.39949092E-01 0.70994320E-03
7	0.00000000E+00 $\pm 1.25634926E+00$ $\pm 2.90309801E+00$ $\pm 4.85530078E+00$	0.67258100E+00 0.15305082E+00 0.10531594E-01 0.12708160E-03
8	$\pm 0.46765442E+00$ $\pm 1.90283824E+00$ $\pm 3.52528890E+00$ $\pm 5.42906631E+00$	0.44152700E+00 0.55673325E-01 0.27764449E-02 0.23232486E-04

N	Roots	Weights
9	0.00000000E+00	0.63251186E+00
	± 1.11755679E+00	0.16511843E+00
	± 2.53756008E+00	0.17950771E-01
	± 4.11850982E+00	0.67083737E-03
	± 5.97431298E+00	0.40290865E-05
10	± 0.42690048E+00	0.42643715E+00
	± 1.70451624E+00	0.67674755E-01
	± 3.11226962E+00	0.57261575E-02
	± 4.66081281E+00	0.16123457E-03
	± 6.47794625E+00	0.70951775E-06
11	± 0.00000000E+00	0.60139972E+00
	± 1.01859480E+00	0.17263576E+00
	± 2.28864417E+00	0.24951366E-01
	± 3.66689410E+00	0.16765164E-02
	± 5.18097036E+00	0.36383422E-04
12	± 6.96120116E+00	0.12008017E-06
	± 0.39609730E+00	0.41347132E+00
	± 1.56012909E+00	0.76957335E-01
	± 2.82465860E+00	0.90801655E-02
	± 4.17947860E+00	0.48297931E-03
13	± 5.66534585E+00	0.81773835E-05
	± 7.41455027E+00	0.20575626E-07
	0.00000000E+00	0.57613576E+00
	± 0.94325209E+00	0.17749703E+00
	± 2.10403206E+00	0.31255608E-01
14	± 3.34651894E+00	0.30473839E-02
	± 4.67472250E+00	0.13035029E-03
	± 6.13274621E+00	0.17508224E-05
	± 7.85250785E+00	0.34160150E-08
	± 0.37168398E+00	0.40215984E+00
14	± 1.44866731E+00	0.84264765E-01
	± 2.60802363E+00	0.12538550E-01
	± 3.83287640E+00	0.10017689E-02
	± 5.13904051E+00	0.34704537E-04
	± 6.57345258E+00	0.37398577E-06
	± 8.26772200E+00	0.57315195E-09

N	Roots	Weights
15	0.00000000E+00	0.55497276E+00
	± 0.88330966E+00	0.18069442E+00
	± 1.95965707E+00	0.36837108E-01
	± 3.10198960E+00	0.46644345E-02
	± 4.30549209E+00	0.30880973E-03
	± 5.58921937E+00	0.87741475E-05
	± 7.00078174E+00	0.76832601E-07
	± 8.67082039E+00	0.93692566E-10
16	± 0.35167460E+00	0.39215887E+00
	± 1.35911522E+00	0.90112950E-01
	± 2.43667316E+00	0.15935537E-01
	± 3.56539142E+00	0.16969892E-02
	± 4.75106905E+00	0.93441500E-04
	± 6.01562508E+00	0.21950778E-05
	± 7.40730175E+00	0.15768352E-07
	± 9.05590925E+00	0.15458650E-10

Table C.1. Roots and weights for the Gauss type quadrature scheme over the weight function $E_1(k^2/4)$.

Equation (C.1) contains the weight function, over which the polynomials whose roots are those tabulated above, are orthogonal, *i.e.*

$$w(x) = \frac{1}{4\sqrt{\pi}} E_1(x^2/4) \quad (\text{C.2})$$

These orthogonal polynomials satisfy the three term recurrence relation

$$P_n(x) = xP_{n-1}(x) - A_n P_{n-2}(x), n = 2, 3, 4, \dots \quad (\text{C.3})$$

where

$$P_{-1}(x) = 0, P_0(x) = 1 \text{ and } P_1(x) = x \quad (\text{C.4})$$

of which the constant A_n was unable to be determined in closed form during our investigations. By multiplying (C.3) by $x^{2j}w(x)$ and integrating over the region of orthogonality we yield

$$\int_{-\infty}^{\infty} x^{n+2j} P_n(x) w(x) dx = \int_{-\infty}^{\infty} x^{n+2j+1} P_{n-1}(x) w(x) dx - A_n \int_{-\infty}^{\infty} x^{n+2j} P_{n-2}(x) w(x) dx \quad (\text{C.5})$$

If we define

$$d_{n,j} = \int_{-\infty}^{\infty} x^{n+2j} P_n(x) w(x) dx \quad (\text{C.6})$$

then we can compactly write (C.4) as

$$d_{n,j} = d_{n-1,j+1} - A_n d_{n-2,j+1} \quad (\text{C.7})$$

Now from ref. [140] we can show that

$$A_n = \frac{d_{n-1,0}}{d_{n-2,0}} \quad (\text{C.8})$$

which allows us to rewrite (C.7) as

$$d_{n,j} = d_{n-1,j+1} - \frac{d_{n-1,0}}{d_{n-2,0}} d_{n-2,j+1} \quad (\text{C.9})$$

Since

$$d_{0,j} = \frac{2^{2j+2}}{2j+1} \Gamma\left(j + \frac{1}{2}\right) \quad (\text{C.10})$$

and

$$d_{1,j} = \frac{2^{2j+4}}{2j+3} \Gamma\left(j + \frac{3}{2}\right) \quad (\text{C.11})$$

are easily afforded, we can use the recursion formula in (C.9) to build up elements necessary for determining A_n and thus recursively determine our polynomials *via* (C.3).

APPENDIX D

DETAILS OF KWIK TIMINGS PRESENTED IN CHAPTER SEVEN

N/E	Feature	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-10}
100,000	NBOX	21387	1648	486	259	202
2,108,918	M	10	145	605	1250	1555
	ω	0.3007	0.03770	0.01463	0.009373	0.008424
	R_c	4.676	60.69	206.2	387.4	495.1
	RLONG	48.2%	39.0%	34.5%	32.4%	31.9%
200,000	NBOX	40228	2927	762	388	288
4,136,409	M	10	200	1080	1785	2090
	ω	0.2791	0.03348	0.01148	0.007033	0.005997
	R_c	4.972	68.34	262.7	516.29	695.5
	RLONG	52.1%	42.5%	37.3%	35.0%	34.2%
300,000	NBOX	57297	4260	819	449	377
6,304,164	M	10	205	1090	1965	2550
	ω	0.2631	0.03249	0.008230	0.005434	0.005240
	R_c	5.236	70.43	366.5	668.2	795.9
	RLONG	52.9%	43.6%	37.1%	35.1%	34.9%
400,000	NBOX	73813	5070	1113	592	447
8,387,886	M	10	230	1135	2030	2775
	ω	0.2542	0.02901	0.008388	0.005367	0.004653
	R_c	5.419	78.90	359.6	676.6	896.5
	RLONG	54.2%	44.5%	38.6%	36.5%	35.8%
500,000	NBOX	90902	6092	1420	643	502
10,498,905	M	10	210	1140	2500	3155
	ω	0.2505	0.02788	0.008562	0.004667	0.004187
	R_c	5.500	82.08	352.3	778.1	996.1
	RLONG	55.1%	45.4%	39.7%	36.9%	36.3%

Table D.1.

Details for KWIK calculations conducted on collinear charged uniformly distributed systems as presented in Table 7.3. Key: NBOX - the number of cells in the short-range calculation; M - the degree of the Fourier expansion; ω - the decay parameter; R_c - the real space cut-off radius; RLONG - the relative contribution of the long range energy to the total energy; E - the total Coulomb energy; N - the size (and range) of the distribution.

N/E	Feature	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-10}
100,000	NBOX	-	449	285	214	183
14,368.1	M	-	655	1155	1470	1690
	ω	-	0.01025	0.008579	0.007761	0.007616
	R _c	-	223.2	351.6	467.9	547.7
	R _{LONG}	-	0.176%	0.140%	0.112%	0.116%
200,000	NBOX	131564	1052	452	327	250
-46,810.3	M	0	555	1645	1935	2405
	ω	1.129	0.01203	0.006811	0.005930	0.005197
	R _c	1.520	190.2	442.8	612.3	802.6
	R _{LONG}	0.0142%	0.172%	0.149%	0.143%	0.136%
300,000	NBOX	181328	2154	597	374	317
-78,086.2	M	0	535	1565	2470	2960
	ω	1.024	0.01643	0.005999	0.004520	0.004405
	R _c	1.654	139.3	502.8	803.3	946.8
	R _{LONG}	0.00872%	0.0579%	0.0511%	0.0508%	0.0508%
400,000	NBOX	251817	3666	648	495	387
-163,722	M	0	335	2135	2660	3315
	ω	1.074	0.02097	0.004883	0.004485	0.004028
	R _c	1.588	109.1	617.7	809.5	1035
	R _{LONG}	0.00426%	0.0296%	0.0193%	0.0193%	0.0191%
500,000	NBOX	224195	2615	688	515	441
-126,458	M	0	635	2310	3015	3910
	ω	0.7245	0.01197	0.004148	0.003737	0.003677
	R _c	2.230	191.2	727.1	971.7	1134
	R _{LONG}	0.00545%	0.0781%	0.0482%	0.0469%	0.0466%

Table D.2.

Details for KWIK calculations conducted on collinear, net-neutral uniformly distributed systems as presented in Table 7.6. Key: NBOX - the number of cells in the short-range calculation; M - the degree of the Fourier expansion; ω - the decay parameter; R_c - the real space cut-off radius; R_{LONG} - the relative contribution of the long range energy to the total energy; E - the total Coulomb energy; N - the size (and range) of the distribution.

N	Feature	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-10}
100,000	R _c	137.8	228.8	301.6	363.1	417.1
	M	5	660	1290	1910	2250
	R _{TIME}	2.30%	43.1%	53.4%	58.6%	59.1%
	R _{LONG}	0.00854%	0.171%	0.178%	0.178%	0.178%
200,000	M	0	830	2325	3525	4215
	R _{TIME}	1.40%	48.9%	67.1%	72.3%	73.0%
	R _{LONG}	0.00917%	0.177%	0.169%	0.169%	0.169%
300,000	M	0	190	2655	4960	6585
	R _{TIME}	1.42%	18.5%	70.0%	78.6%	80.9%
	R _{LONG}	0.00576%	0.0627%	0.0540%	0.0540%	0.0540%
400,000	M	0	390	3695	7025	8830
	R _{TIME}	1.40%	31.2%	76.3%	83.8%	84.9%
	R _{LONG}	0.00283%	0.0291%	0.0217%	0.0217%	0.0217%
500,000	M	0	2335	5650	8460	10900
	R _{TIME}	1.40%	72.8%	83.1%	86.2%	87.4%
	R _{LONG}	0.00376%	0.0680%	0.0773%	0.0772%	0.0772%

Table D.3. Details for KWIK calculations conducted on collinear, net-neutral uniformly distributed systems for fixed decay parameter $\omega = 0.01$, as presented in Table 7.7. Key: R_c - the real space cut-off radius; M - the degree of the Fourier expansion; R_{TIME} - the relative contribution of the long-range time to the total time; R_{LONG} - the relative contribution of the long range energy to the total energy; N - the size (and range) of the distribution.

N	Feature	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-10}
100,000	R _c	275.5	457.7	603.2	726.2	834.2
	M	0	355	670	855	1255
	R _{TIME}	0.765%	16.6%	23.2%	24.3%	28.4%
	R _{LONG}	0.0250%	0.0562%	0.0656%	0.0656%	0.0656%
200,000	M	0	500	1210	1700	2065
	R _{TIME}	0.719%	22.7%	34.9%	38.9%	39.5%
	R _{LONG}	0.00843%	0.128%	0.135%	0.135%	0.135%
300,000	M	0	390	1635	2450	3170
	R _{TIME}	0.729%	18.7%	42.0%	47.6%	50.1%
	R _{LONG}	0.00531%	0.0606%	0.0511%	0.0510%	0.0510%
400,000	M	0	260	2270	3250	4105
	R _{TIME}	0.733%	13.3%	50.1%	54.6%	56.6%
	R _{LONG}	0.00262%	0.0124%	0.0193%	0.0194%	0.0194%
500,000	M	0	785	2625	4210	5400
	R _{TIME}	0.696%	31.3%	53.6%	61.0%	63.0%
	R _{LONG}	0.00348%	0.0587%	0.0519%	0.0518%	0.0518%

Table D.4. Details for KWIK calculations conducted on collinear, net-neutral uniformly distributed systems for fixed decay parameter $\omega = 0.005$, as presented in Table 7.8. Key: R_c - the real space cut-off radius; M - the degree of the Fourier expansion; R_{TIME} - the relative contribution of the long-range time to the total time; R_{LONG} - the relative contribution of the long range energy to the total energy; N - the size (and range) of the distribution.

REFERENCES

- [1] S. F. Boys, G. B. Cook, C. M. Reeves and I. Shavitt, *Nature* (1956), **178**, 1207.
- [2] W. J. Hehre, W. A. Lathan, M. D. Newton, R. Ditchfield and J. A. Pople, *GAUSSIAN 70*, Bloomington, Indiana, 1970.
- [3] S. R. La Paglia, *Introductory quantum chemistry*; Harper & Row, New York, 1971.
- [4] E. Rutherford, *Phil. Mag.* (1911), **21**, 669.
- [5] N. Bohr, *Phil. Mag.* (1913), **26**, 1.
- [6] L. de Broglie, *Ann. Phys.* (1925), **3**, 22.
- [7] A. Einstein, *Ann. Physik.* (1905), **17**, 132.
- [8] C. J. Davisson, *Congres Intern. Elec.* (1932), **2**, 593.
- [9] E. Schrödinger, *Ann. Physik.* (1926), **79**, 361.
- [10] W. Heisenberg, *Z. Physik.* (1926), **39**, 499.
- [11] M. Born and J. R. Oppenheimer, *Ann. Physik.* (1927), **84**, 457.
- [12] D. R. Hartree, *Proc. Cam. Phil. Soc.* (1928), **24**, 89.
- [13] D. R. Hartree, *Proc. Cam. Phil. Soc.* (1928), **24**, 111.
- [14] D. R. Hartree, *Proc. Cam. Phil. Soc.* (1928), **24**, 426.
- [15] V. Fock, *Z. Physik.* (1930), **61**, 126.
- [16] J. C. Slater, *Phys. Rev.* (1929), **34**, 1293.
- [17] J. C. Slater, *Phys. Rev.* (1930), **35**, 509.
- [18] A. Szabo and N. S. Ostlund, *Modern quantum chemistry: Introduction to advanced electronic structure theory*; First (revised) edition; McGraw-Hill, New York, 1989.
- [19] E. R. Davidson and D. Feller, *Chem. Rev.* (1986), **86**, 681.
- [20] J. P. Lowe, *Quantum chemistry*; Academic Press, New York, 1978.

- [21] R. McWeeny, *Methods of molecular quantum mechanics*; Second edition; Academic Press, London, 1992.
- [22] J. A. Pople and R. K. Nesbet, *J. Chem. Phys.* (1954), **22**, 571.
- [23] D. J. DeFrees, B. A. Levi, S. K. Pollack, W. J. Hehre, J. S. Binkley and J. A. Pople, *J. Am. Chem. Soc* (1979), **101**, 4085.
- [24] M. Head-Gordon, *J. Phys. Chem.* (1996), **100**, 13213.
- [25] K. Raghavachari and J. B. Anderson, *J. Phys. Chem.* (1996), **100**, 12960.
- [26] J. A. Pople, M. Head-Gordon and K. Raghavachari, *J. Chem. Phys.* (1987), **87**, 5968.
- [27] R. Krishnan, M. J. Frisch and J. A. Pople, *J. Chem. Phys.* (1980), **72**, 4244.
- [28] L. A. Curtiss, C. Jones, G. W. Trucks, K. Raghavachari and J. A. Pople, *J. Chem. Phys.* (1990), **93**, 2537.
- [29] L. Curtiss, K. Raghavachari, G. W. Trucks and J. A. Pople, *J. Chem. Phys.* (1991), **94**, 7221.
- [30] L. A. Curtiss, K. Raghavachari and J. A. Pople, *J. Chem. Phys.* (1995), **103**, 4192.
- [31] J. A. Pople, M. Head-Gordon, D. J. Fox, K. Raghavachari and L. Curtiss, *J. Chem. Phys.* (1989), **90**, 5622.
- [32] A. Nicolaidis and L. Radom, *J. Phys. Chem.* (1994), **98**, 3092.
- [33] R. J. Bartlett, *J. Phys. Chem.* (1989), **93**, 1697.
- [34] C. Møller and M. S. Plesset, *Phys. Rev.* (1934), **46**, 618.
- [35] P. M. W. Gill and L. Radom, *Chem. Phys. Lett.* (1986), **132**, 16.
- [36] E. A. Hylleraas, *Z. Physik.* (1928), **48**, 469.
- [37] E. A. Hylleraas, *Z. Physik.* (1929), **54**, 347.
- [38] E. A. Hylleraas, *Z. Physik.* (1930), **65**, 209.
- [39] H. M. James and A. S. Coolidge, *J. Chem. Phys.* (1933), **1**, 825.
- [40] Z.-C. Yan and G. W. F. Drake, *Chem. Phys. Lett.* (1996), **259**, 96.
- [41] B. J. Persson and P. R. Taylor, *J. Chem. Phys.* (1996), **105**, 5915.

- [42] W. Kohn and L. J. Sham, *Phys. Rev. A* (1965), **140**, 1133.
- [43] P. Hohenberg and W. Kohn, *Phys. Rev. B* (1964), **136**, 864.
- [44] R. G. Parr and W. Yang, *Density-Functional theory of atoms and molecules*; First edition; Oxford University Press, New York, 1989.
- [45] P. A. M. Dirac, *Proc. Cam. Phil. Soc.* (1930), **26**, 275.
- [46] B. G. Johnson, PhD Thesis. "Development, implementation and performance of efficient methodologies for density functional calculations" Carnegie Mellon University, Pittsburgh, 1993.
- [47] P. M. W. Gill, *submitted* (1996).
- [48] J. A. Pople, P. M. W. Gill and B. G. Johnson, *Chem. Phys. Lett.* (1992), **199**, 557.
- [49] B. G. Johnson, P. M. W. Gill and J. A. Pople, *J. Chem. Phys.* (1993), **98**, 5612.
- [50] J. Baker, J. Andzelm, M. Muir and P. R. Taylor, *Chem. Phys. Lett.* (1995), **237**, 53.
- [51] Q. Zhang, R. Bell and T. N. Truong, *J. Phys. Chem.* (1995), **99**, 592.
- [52] R. H. Landau and P. J. Fink, Jr., *A scientist's and engineer's guide to workstations and supercomputers. Coping with Unix, RISC, vectors and programming*; John Wiley & Sons Inc., New York, 1993.
- [53] M. Abramowitz and I. A. Stegun, *Handbook of mathematical functions*; M. Abramowitz and I. A. Stegun, Ed.; Dover Publications, New York, 1972.
- [54] J. J. Dongarra and D. W. Walker, *SIAM Review* (1995), **37**, 151.
- [55] J. J. Dongarra, J. Du Croz, S. Hammarling and R. J. Hanson, *ACM Trans. Math. Software* (1988), **14**, 18.
- [56] J. J. Dongarra, J. Du Croz, S. Hammarling and R. J. Hanson, *ACM Trans. Math. Software* (1988), **14**, 1.
- [57] C. L. Lawson, R. J. Hanson, D. R. Kincaid and F. T. Krogh, *ACM Trans. Math. Software* (1979), **5**, 308.
- [58] E. Anderson, Z. Bai, C. Bischof, J. W. Demmel, J. J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney and D. Sorenson, "LAPACK: A

portable linear algebra library for high performance computers," University of Tennessee, 1990.

- [59] D. C. Rapaport, *Comp. Phys. Rep.* (1988), **9**, 1.
- [60] S. Heath, *Microprocessors, architectures and systems. RISC, CISC and DSP*; Paperback edition; Newnes, London, 1993.
- [61] R. R. Oehler and R. D. Groves, *IBM J. Res. Develop.* (1990), **34**, 23.
- [62] K. A. Gallivan, R. J. Plemmons and A. H. Sameh, *SIAM Review* (1990), **32**, 1.
- [63] S. F. Boys, *Proc. Roy. Soc., A* (1950), **A200**, 542.
- [64] I. Shavitt and M. Karplus, *J. Chem. Phys.* (1965), **43**, 398.
- [65] W. J. Hehre, R. F. Stewart and J. A. Pople, *J. Chem. Phys.* (1969), **51**, 2657.
- [66] W. J. Hehre, L. Radom, P. v. R. Schleyer and J. A. Pople, *Ab initio molecular orbital theory*; Wiley, New York, 1986.
- [67] M. C. Strain, G. E. Scuseria and M. J. Frisch, *Science* (1996), **271**, 51.
- [68] P. M. W. Gill, *Advances in Quantum Chemistry* (1994), **25**, 142.
- [69] T. R. Adams, R. D. Adamson and P. M. W. Gill, *J. Chem. Phys.* (1997), in press.
- [70] K. Ishida, *Int. J. Quantum Chem.* (1996), **59**, 209.
- [71] R. D. Adamson, Honours Thesis. "Shell-pair economisation" Massey University, Palmerston North, 1995.
- [72] L. Greengard and J. Strain, *SIAM J. Sci. Stat. Comput.* (1991), **12**, 79.
- [73] J. Strain, *SIAM J. Sci. Stat. Comput.* (1991), **12**, 1131.
- [74] P. M. W. Gill, B. G. Johnson and J. A. Pople, *Chem. Phys. Lett.* (1994), **217**, 65.
- [75] D. L. Strout and G. E. Scuseria, *J. Chem. Phys.* (1995), **102**, 8448.
- [76] E. Schwegler and M. Challacombe, *J. Chem. Phys.* (1996), **105**, 2726.
- [77] W. Kohn, *Phys. Rev. Lett.* (1996), **76**, 3168.
- [78] C. A. White, B. G. Johnson, P. M. W. Gill and M. Head-Gordon, *Chem. Phys. Lett.* (1994), **230**, 8.

- [79] C. A. White, B. G. Johnson, P. M. W. Gill and M. Head-Gordon, *Chem. Phys. Lett.* (1996), **253**, 268.
- [80] C. A. White and M. Head-Gordon, *J. Chem. Phys.* (1994), **101**, 6593.
- [81] C. A. White and M. Head-Gordon, *Chem. Phys. Lett.* (1996), **257**, 647.
- [82] C. A. White and M. Head-Gordon, *J. Chem. Phys.* (1996), **105**, 5061.
- [83] R. Kutteh, E. Apra and J. Nichols, *Chem. Phys. Lett.* (1995), **238**, 173.
- [84] J. C. Burant, M. C. Strain, G. E. Scuseria and M. J. Frisch, *Chem. Phys. Lett.* (1996), **248**, 43.
- [85] J. C. Burant, M. C. Strain, G. E. Scuseria and M. J. Frisch, *Chem. Phys. Lett.* (1996), **258**, 45.
- [86] C. A. White and M. Head-Gordon, *J. Chem. Phys.* (1996), **104**, 2620.
- [87] B. Jawerth and W. Sweldens, *SIAM Review* (1994), **36**, 377.
- [88] G. Beylkin, *SIAM J. Numer. Anal.* (1992), **6**, 1716.
- [89] J.-L. Calais, *Int. J. Quantum Chem.* (1996), **58**, 541.
- [90] K. Cho, T. A. Arias and J. D. Joannopoulos, *Phys. Rev. Lett.* (1993), **71**, 1808.
- [91] Z. Li, A. Borrmann and C. C. Martens, *Chem. Phys. Lett.* (1993), **214**, 362.
- [92] S. Wei and M. Y. Chou, *Phys. Rev. Lett.* (1996), **76**, 2650.
- [93] W. Yang, *Phys. Rev. Lett.* (1991), **66**, 1438.
- [94] Z. Zhou, *Chem. Phys. Lett.* (1993), **203**, 396.
- [95] W. Yang, *Phys. Rev. A* (1991), **44**, 7823.
- [96] W. Yang and T.-S. Lee, *J. Chem. Phys.* (1995), **103**, 5674.
- [97] C. Lee and W. Yang, *J. Chem. Phys.* (1992), **96**, 2408.
- [98] Q. Zhao and W. Yang, *J. Chem. Phys.* (1995), **102**, 9598.
- [99] W. Thiel, *Tetrahedron* (1998), **44**, 7393.
- [100] M. J. S. Dewar, *The molecular orbital theory of organic chemistry*; McGraw-Hill, New York, 1969.

- [101] R. T. Gallant and A. St-Amant, *Chem. Phys. Lett.* (1996), **256**, 569.
- [102] S. K. Goh and A. At-Amant, *Chem. Phys. Lett.* (1997), **264**, 9.
- [103] M. P. Allen and D. J. Tildesley, *Computer simulation of liquids*; Clarendon Press, Oxford, 1987.
- [104] P. P. Ewald, *Ann. Physik.* (1921), **64**, 253.
- [105] B. A. Luty, I. G. Tironi and W. F. van Gunsteren, *J. Chem. Phys.* (1995), **103**, 3014.
- [106] J. W. Perram, H. G. Petersen and S. W. De Leeuw, *Mol. Phys.* (1988), **65**, 875.
- [107] Y. Rosenfeld, *Mol. Phys.* (1996), **88**, 1357.
- [108] V. Natoli and D. M. Ceperley, *J. Comput. Phys.* (1995), **117**, 171.
- [109] G. Hummer, *Chem. Phys. Lett.* (1995), **235**, 297.
- [110] B. Delley, *J. Phys. Chem.* (1996), **100**, 6107.
- [111] R. W. Hockney and J. W. Eastwood, *Computer simulation using particles*; Special Student edition; IOP Publishing, Bristol, 1988.
- [112] B. A. Luty and W. F. van Gunsteren, *J. Phys. Chem.* (1996), **100**, 2581.
- [113] J. Barnes and P. Hut, *Nature* (1986), **324**, 446.
- [114] L. F. Greengard and V. Rokhlin, *J. Comput. Phys.* (1987), **73**, 325.
- [115] L. Greengard, *Science* (1994), **265**, 909.
- [116] L. Greengard, *The rapid evaluation of potential fields in particle systems*; The MIT Press, London, 1987.
- [117] J. A. Board, Jr., J. W. Causey, J. F. Leathrum, Jr., A. Windemuth and K. Schulten, *Chem. Phys. Lett.* (1992), **198**, 89.
- [118] C. G. Lambert, T. A. Darden and J. A. Board, Jr., *J. Comput. Phys.* (1996), **126**, 274.
- [119] M. Unser, A. Aldroubi and S. J. Schiff, *IEEE Trans. Signal Processing* (1994), **42**, 3519.
- [120] D. N. S. Permann and H. Teitelbaum, *J. Phys. Chem.* (1993), **97**, 12670.

- [121] C. K. Chui, *An introduction to wavelets*; Academic Press, Boston, 1992.
- [122] C. K. Chui, J. C. Goswami and A. K. Chan, *Numer. Math.* (1995), **70**, 283.
- [123] J. C. Goswami, A. K. Chan and C. K. Chui, *IEEE Trans. Micro. Theory Tech.* (1995), **43**, 655.
- [124] H. S. Malvar, *Electron. Lett.* (1992), **28**, 1393.
- [125] M.-H. Yaou and W.-T. Chang, *IEEE Trans. Patt. Anal. Machine Intell.* (1994), **16**, 673.
- [126] P. M. W. Gill, *unpublished*.
- [127] J. F. James, *A student's guide to Fourier transforms with applications in physics and engineering*; Cambridge University Press, Cambridge, 1995.
- [128] P. M. W. Gill, B. G. Johnson and J. A. Pople, *Int. J. Quantum Chem.* (1991), **40**, 745.
- [129] C. Fox, *An introduction to the calculus of variations*; Dover, New York, 1987.
- [130] E. Kreyszig, *Advanced engineering mathematics*; Seventh edition; Wiley, Singapore, 1993.
- [131] A. M. Lee, S. W. Taylor, J. P. Dombroski and P. M. W. Gill, *Phys. Rev. A* (1997), in press.
- [132] F. J. Vesely, *Computational physics. An introduction*; Plenum Press, New York, 1994.
- [133] P. W. Markstein, *IBM J. Res. Develop.* (1990), **34**, 111.
- [134] L. Verlet, *Phys. Rev.* (1967), **159**, 98.
- [135] S. Plimpton, *J. Comput. Phys.* (1995), **117**, 1.
- [136] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical recipes in FORTRAN: The art of scientific computing*; Second edition; Cambridge University Press, Cambridge, 1992.
- [137] A. L. Loeb, *Space structures. Their harmony and counterpoint*; Addison-Wesley: Reading, Massachusetts, 1976.

- [138] R. L. Burden and J. D. Faires, *Numerical Analysis*; Fifth edition; PWS, Boston, 1993.
- [139] P. F. Davis and P. Rabinowitz, *Methods of numerical integration*; Second edition; Academic Press, London, 1984.
- [140] A. H. Stroud and S. D., *Gaussian quadrature formulas*; Prentice-Hall, London, 1966.
- [141] A. T. B. Gilbert, Honours Thesis. "Coulomb Orthonormal Polynomials" Massey University, Palmerston North, 1996.
- [142] K. Pearson, *Nature* (1905), **72**, 294.
- [143] G. H. Weiss, *Aspects and applications of the random walk*; North-Holland, New York, 1994.
- [144] P. M. W. Gill, B. G. Johnson and J. A. Pople, *Chem. Phys. Lett.* (1993), **209**, 506.
- [145] P. M. W. Gill, C. R. T. Maunder and S. W. Taylor, *Submitted* (1996).
- [146] S. W. Taylor, *Personal Communication*.
- [147] M. J. Frisch, B. G. Johnson, P. M. W. Gill, D. J. Fox and R. H. Nobes, *Chem. Phys. Lett.* (1993), **206**, 225.
- [148] J. P. Dombroski, S. W. Taylor and P. M. W. Gill, *J. Phys. Chem.* (1996), **100**, 6272.
- [149] D. C. Kay, *Theory and problems of tensor calculus*; McGraw Hill, New York, 1988.
- [150] R. D. Adamson, J. P. Dombroski and P. M. W. Gill, *Chem. Phys. Lett.* (1996), **254**, 329.
- [151] E. Clementi, *Proceedings of the Robert A. Welch Foundation Conferences on Chemical Research, XVI Houston*, 1972.
- [152] P. M. W. Gill and R. D. Adamson, *Chem. Phys. Lett.* (1996), **261**, 105.
- [153] P. M. W. Gill, *Personal Communication*.

- [154] B. G. Johnson, P. M. W. Gill, M. Head-Gordon, C. A. White, J. Baker, D. R. Maurice, R. D. Adamson, T. R. Adams, J. Kong, M. Oumi and I. Ishikawa, *Q-Chem 1.0*, Q-Chem Inc., Pittsburgh, PA, 1997.
- [155] P. Pulay, *J. Comput. Chem.* (1982), **3**, 556.
- [156] P. Pulay, *Chem. Phys. Lett.* (1980), **73**, 393.
- [157] H. Sellers, *Int. J. Quantum Chem.* (1993), **45**, 31.
- [158] E. Clementi, *IBM J. Res. Develop.* (1965), **9**, 2.
- [159] R. Lopez, G. Ramirez, J. M. Garcia de la Vega and J. F. Rico, *J. Chim. Phys.* (1987), **84**, 695.
- [160] T. R. Adams, J. P. Dombroski and P. M. W. Gill, *unpublished results* .
- [161] P. M. W. Gill, *J. Phys. Chem.* (1996), **100**, 15421.
- [162] M. Haser and J. Almlöf, *J. Chem. Phys.* (1992), **96**, 489.
- [163] J. Almlöf, *Chem. Phys. Lett.* (1991), **181**, 319.
- [164] S. Saebo, *Int. J. Quantum Chem.* (1992), **42**, 217.
- [165] S. Saebo, *Annu. Rev. Phys. Chem.* (1993), **44**, 213.
- [166] S. Aluru, *SIAM Journal on Scientific Computing* (1996), **17**, 773.
- [167] P. M. W. Gill and J. A. Pople, *Int. J. Quantum Chem.* (1991), **40**, 753.
- [168] S. Wolfram, et al, *Mathematica 2.0*, Wolfram Research Inc., Champaign, Illinois, 1991.