

**A HIGH RESOLUTION ANALYSIS OF SPATIO-
TEMPORAL VARIATIONS OF GEOMAGNETIC
FIELD USING PARALLEL COMPUTING**

By:

ALE, Felix

BSc. (Hons), MSc (OAU, Ife)

(Matriculation Number: 089047017)

**Thesis submitted to the School of Postgraduate
Studies, University of Lagos in Partial Fulfillment of
the Requirements for the Award of the Degree of
DOCTOR OF PHILOSOPHY (Ph.D.)**

In

Systems Engineering

DEPARTMENT OF SYSTEMS ENGINEERING

FACULTY OF ENGINEERING

UNIVERSITY OF LAGOS

FEBRUARY 2014

**SCHOOL OF POSTGRADUATE STUDIES
UNIVERSITY OF LAGOS**

CERTIFICATION

This is to certify that the Thesis:









**“A HIGH RESOLUTION ANALYSIS OF SPATIO-TEMPORAL
VARIATIONS OF GEOMAGNETIC FIELD USING
PARALLEL COMPUTING”**

**Submitted to the
School of Postgraduate Studies
University of Lagos**

**For the award of the degree of
DOCTOR OF PHILOSOPHY (Ph.D.)
is a record of original research carried out
By:**

ALE, FELIX

In the Department of SYSTEMS ENGINEERING

<u>ALE FELIX</u>		<u>07-02-2014</u>
AUTHOR'S NAME	SIGNATURE	DATE
<u>Prof. O. Ibiodapo Oke</u>		<u>07-02-2014</u>
1 ST SUPERVISOR'S NAME	SIGNATURE	DATE
<u>Dr. T.A. FASHANU</u>		<u>07-02-2014</u>
2 ND SUPERVISOR'S NAME	SIGNATURE	DATE
<u>Dr. O. A. AGBoola</u>		<u>07/02/2014</u>
3 RD SUPERVISOR'S NAME	SIGNATURE	DATE
<u>Prof. O. Taniw</u>		<u>7/02/2014</u>
1 ST INTERNAL EXAMINER	SIGNATURE	DATE
<u>Prof. E.A. AYOLABI</u>		<u>07/02/2014</u>
2 ND INTERNAL EXAMINER	SIGNATURE	DATE
<u>Prof. A.B. RABIU</u>		<u>7/02/2014</u>
EXTERNAL EXAMINER	SIGNATURE	DATE
<u>Dr. G. O. Oyejemi</u>		<u>07/02/2014</u>
SPGS REPRESENTATIVE	SIGNATURE	DATE

DEDICATION

This work is primarily dedicated to the Almighty God, the creator of Heavens and Earth for giving me the required grace, might, wisdom, knowledge, understanding and wisdom to contribute to the fields of Earth Sciences and Computing.

I also dedicate this work to my late father, Pa Joshua Oluwasinmi Ale for his immense contribution to my existence and academics.

ACKNOWLEDGEMENTS

First of all, I am most grateful to the Almighty God, the creator of the universe (the Heavens and the Earth) for not only given me the gift of life but also for academic ability to study His divine creature – the Space and spatial phenomena!

I am sincerely indebted to my supervisor, Distinguished Professor Oyewusi Ibidapo-Obe for his efforts and endeared fatherly role. I am indeed grateful to Dr. T. A. Fashanu, the HOD, for his keen supervision of this work. His friendly laughter is quite commendable. I am also thankful to Dr. O. A. Agboola for his supervision, love, spiritual and financial supports.

I wish to acknowledge the support of the Dean of Faculty of Engineering, Prof. O. A. Fakinlede, Dr. S.O. Adeosun, Prof. M.A. Salau, Dr. J. Agunsoye, Dr. O.A. Alli, Dr. M. K. Ayomoh, Dr. T. T. Akano and Dr. Kamiyo. I wish to also thank all other members of staff of Department of Systems Engineering especially Emeritus Prof. Olunloyo, Dr. S.O. Asaolu, Dr. O.A. Ajofoyinbo, Dr. F. O. Ogunwolu and as well as the administrative staff.

I sincerely appreciate the supports of NASRDA staff especially Dr. S.O.Mohammed (DG), Dr. O. A. Agboola [D(ESS)], Dr. S. Onu [D(CSTD)], other Directors, Deputy and Assistant Directors as well as my staff colleagues for providing necessary support at different levels.

The results presented in this work rely on data collected at magnetic observatories. We thank the national institutes that support them and INTERMAGNET for promoting high standards of magnetic observatory practice (www.intermagnet.org). I particularly thank Ms. Uzzi A. and Mr. U. Tokula who assisted in pre-processing of the INTERMAGNET geomagnetic field raw data. I also wish thank Richard O., Kalu A., O. Ademola, A. Zachaeus and others for giving helping hands while typesetting the report of this work. I am indeed very grateful to Prof. A. B. Rabiou, Prof. K. Yumoto (PI, MAGDAS) and Dr. Y. Endawoke (AMBER) for exposing me to the field of geomagnetism in addition to my advanced computational skill.

I thank my beloved wife – Mrs. S. T. Ale, my children, my parents, family, siblings and relations for their moral support during my PhD programme. I am also indebted to Mrs. T. Agboola and my fellow pastors for their prayers and encouragement, may the Lord bless all of them in Jesus' Name.

ABSTRACT

This work presents a high resolution analysis of spatio-temporal variations of geomagnetic field using parallel computing. Geomagnetic field diurnal and hourly variation have been used extensively to study extant and emerging geophysical phenomena. Although, geomagnetic field variations occur even on nanosecond basis, however, to date, algorithm development and computing hardware architecture limit the resolution of monitoring solar quiet (Sq) daily variation to hourly and regional basis. This is obviously inadequate for detailed analyses of some geophysical phenomena with time scale less than hours. Also, regional analysis of this variation is insufficient for inter-regional phenomena. It therefore follows that any tool that facilitates a more detailed analysis of the earth's spatio-temporal magnetic field is bound to lead to a better understanding of related phenomena observed on earth. Thus, for the purpose of high resolution spatio-temporal analysis of geomagnetic field variations, a 366-day of 1-minute sampled geomagnetic field dataset was obtained from 64 observatories. These observatories are evenly distributed across the globe on the INTERMAGNET's network of 1996. The data was pre-processed and corrected for non-cyclic variations leading to a [366 x 276480] partitioned matrix. To obtain solar quiet (Sq) daily variation from the resulting large system, this work developed serial and parallel computing algorithms that were executed on a MatlabR2010a environment. It employed the well-known kriging method for geostatistical data gridding and mapping of solar quiet (Sq) daily variations. The results show that whereas sequential algorithm running on an Intel Xeon E5410 2.33GHz processor generates Sq conforming to international quiet day (IQD) standards in 18.5 minutes, the parallel algorithm running on 8 similar processors produces the Sq in 2.95 minutes. This translates to a speedup of 4.32 for the developed parallel algorithm using the benchmark performance metrics of Amdahl's. On logarithmic extrapolation, it was deduced that the parallel algorithm would run optimally on 13 Intel Xeon E5410 2.33GHz processors to generate solar quiet in less than a minute. However, the resultant relatively high index of performance called for an extension of the Amdahl's model to account for parallel overhead. Thus, in addition to enhanced analysis of geomagnetic field variations, the developed computational platform also enabled a high frequency availability of solar quiet daily variations. Consequently, this work provides a basis for early warning system and better understanding of some hitherto unclear geophysical phenomena such as travelling ionospheric disturbances, equatorial anomalies and high geophysical activities in the Polar Regions.

TABLE OF CONTENTS

	Pages
TITLE PAGE	i
CERTIFICATION	ii
DEDICATION	iii
ACKNOWLEDGEMENT	iv
ABSTRACT	v
TABLE OF CONTENTS	vi
LIST OF FIGURES	x
LIST OF TABLES	xiii
LIST OF ABBREVIATIONS	xiv
NOTATIONS	xv
CHAPTER 1: INTRODUCTION	1
1.1 Preamble	1
1.2 Background to the Study	2
1.2.1 The Concept of Geomagnetism	2
1.2.2 Historical Background of Geomagnetic Field	3
1.2.3 Geomagnetic Field Components	4
1.2.4 Overview of Geomagnetic Field Variations	6
1.2.5 Spatio-Temporal Variations of Geomagnetic Field	10
1.2.6 Geomagnetic Field Instrumentations	9
1.2.7 The Sun and Solar Activities	7
1.2.8 The Ionosphere	8
1.2.9 The Concept of Parallel Computing	11
1.2.10 Performance Evaluation of Parallel Systems	13
1.3 Statement of Problem	13
1.4 Aim and Objectives of the Study	14
1.5 Research Questions	15
1.6 Scope and Limitation of the Study	15
1.7 Significance of the Study	15

1.7.1	Space Weather Phenomena	16
1.7.2	Solar-Driven Geomagnetic Storms	16
1.7.3	Effects on Space-borne Technological Systems	18
1.7.4	Effects on Ground-based Technological Systems	20
1.7.5	Effects on Geologic exploration	22
1.7.6	Effects on Military Facilities	22
1.7.7	Effects on Geophysical Phenomena	22
1.7.8	The Merits of A High Resolution Analysis	22
1.8	Overview of Thesis	23
1.9	Operational Definition of Terms	23
 CHAPTER 2: LITERATURE REVIEW		 25
2.1	The Sun cycle and Solar Variations	25
2.2	Solar Quiet (Sq) Daily Variations	26
2.3	Variations of Geomagnetic Field	27
2.4	Evolution of Geomagnetic Field Observatories	29
2.5	Geo-statistical and Geo-Spatial Data Analysis	30
2.6	Numerical Interpolation Methods for Regionalized Variables	31
2.6.1	Polynomial Extrapolation Method	32
2.6.2	Triangulation Method	33
2.6.3	Cubic Spline Interpolation Method	33
2.6.4	Inverse Distance Weighting (IDW) Method	34
2.6.5	Kriging Interpolation Method	34
2.6.6	Comparison of Kriging and Triangulation Methods	36
2.7	The Concept of Advanced Computations	36
2.8	Architecture and Design of Parallel Computing	38
2.8.1	Single Instruction Single Data Stream (SISD)	39
2.8.2	Single Instruction, Single Data stream	39
2.8.3	Multiple Instructions, Single Data stream	40
2.8.4	Multiple Instructions Multiple Data streams	40
2.9	High-Performance Computing	41
2.9.1	Interconnect Hardware	44

2.9.2	Multi-core Technology	44
2.10	Parallel Algorithm and Programming	47
2.10.1	Message Passing Interface	49
2.10.2	Parallel Virtual Machine	50
2.11	Parallel Computation using MATLAB Compiler	51
2.12	Advances in Parallel Computing	53
2.13	Emerging Applications of Parallel Computing	54
2.14	Parallel Computing Benchmark Metrics	55
2.14.1	Scalability of Parallel Computations	56
2.14.2	Benchmark for Parallel Algorithms	57
2.15	Advanced Computation of Sq Daily variations	57

CHAPTER 3: ANALYSIS OF SPATIO-TEMPORAL VARIATIONS OF GEOMAGNETIC FIELD		58
3.1	Acquisition of Geomagnetic Field Data	58
3.1.1	Geomagnetic Data Quality Control Procedure	60
3.2	Geomagnetic Field Data Processing	60
3.2.1	Post-acquisition Processing of Geomagnetic Field Data	60
3.2.2	Interpolation of Sq Variations Using Kriging Method	64
3.2.3	Development of Scalable Algorithm for Simulation of Sq Variations	68
3.2.4	Development of Sequential Algorithm for Sq Variations	68
3.2.5	Development of Parallel Algorithm for Sq Variations	70
3.2.6	Resolving Tasks Allocation and Matrices Partitioning	70
3.3	NASRDA Advanced Computation Laboratory	71
3.4	The Design of Parallel Architecture for Sq Algorithm	72
3.5	Validation of the Parallel Computing Systems	74
3.6	Implementation of the Sq Parallel Algorithm	75
3.7	Efficiency Assessment of the Developed Computational Platform	75
3.7.1	Execution Time	76
3.7.2	Speedup	76
3.7.3	Scalability	76
3.7.4	Efficiency	77
3.7.5	Implication of the Parallel Overhead	77

3.7.6	Amdahl's Model for Evaluating Performance of Parallel Platform	77
3.7.7	Optimization of the Parallel Algorithm	78
3.7.8	Approximation of Optimum Processor Allocation	79
3.8	The Sq Program Codes Framework	80
CHAPTER 4: RESULTS AND DISCUSSION		82
4.1	Simulations of Sq Variations in 2-D	82
4.2	Simulations of Sq Variations in 3-D	89
4.3	Performance Analysis and Evaluation of the Sq Algorithm	94
4.3.1	Execution Time, Speedup and Efficiency Metrics	94
4.3.2	Parallel Overhead	97
4.4	Validation of the Parallel Computational Platform using Amdahl's Model	98
4.5	Improvement on Amdahl's Model for Better Performance	98
4.6	Performance Prediction using Least Squares and Pearson Method	99
4.7	Detailed Discussion of the Results	103
CHAPTER 5: SUMMARY OF FINDING, CONTRIBUTION TO KNOWLEDGE AND CONCLUSION		105
5.1	Summary of Findings	105
5.2	Conclusion	105
5.3	Contributions to Knowledge	106
5.4	Recommendations	106
REFERENCES		107
APPENDIX A: Detailed Distributions of INTERMAGNET Observatories		123
APPENDIX B: MATLAB PROGRAMS		125
B1:	Matlab (sequential) program for simulation of Sq variations	125
B2:	Matlab (parallel) program for simulation of Sq variations	136
B3:	Matlab program for Sq Graphical Users' Interface	148
APPENDIX C: High Resolution Sq(D) Variations in 2-D		152
APPENDIX D: High Resolution Sq(Z) Variations in 2-D		155

LIST OF FIGURES

	Page
Figure 1.1: Fluid flow and magnetic field in the liquid outer core	3
Figure 1.2: The Earth's Magnetic Field	3
Figure 1.3: The Earth's Magnetic field components	5
Figure 1.4: Types of Geomagnetic Field Variations	6
Figure 1.5: The Earth Atmospheric Regions	11
Figure 1.6: Schematic representation of Parallel Computation	12
Figure 1.7: Solar wind and Earth's magnetosphere	18
Figure 1.8: Space weather effects on Technology	20
Figure 1.9: In-orbit satellite failures mostly due to space weather effects	20
Figure 2.1: Abstraction of Geospatial Analysis for Regionalized Variables	31
Figure 2.2: Flynn's Taxonomy – Classification of computer architectures	39
Figure 2.3: Standard Computing Cluster Architecture	42
Figure 2.4: Interconnection of cluster components	43
Figure 2.5: Cluster Topology	43
Figure 2.6: Multi-cores systems	47
Figure 2.7: Parallel Programming Architecture	51
Figure 3.1: INTERMAGNET observatories	58
Figure 3.2: Stratification of INTERMANGET observatories to latitudinal zones	61
Figure 3.3: Sq Sequential Algorithm for Serial Computation	69
Figure 3.4: NASRDA Advanced Computation Laboratory	72
Figure 3.5: Layout of the Deployed Heterogeneous (Hybrid) Multicore systems	73
Figure 3.6(a): Simulations Execution Time versus Processors	74
Figure 3.6(b): Simulations Execution Time versus Processors	74
Figure 3.7: Sq Parallel Algorithm	75
Figure 3.8: Speedup Pattern in Scalable Parallel Algorithm	76
Figure 3.9: Optimization of the Parallel Algorithm	78

Figure 3.10:	Sq (H, D, Z) simulation process and program methodology	80
Figure 3.11:	A new approach for solving data-intensive geomagnetic data	81
Figure 4.1:	SqH Hourly variation in high latitude – January	83
Figure 4.2:	SqH Hourly variation low latitude – January	83
Figure 4.3:	High resolution of SqH across Low-, Mid- and High Latitudes – January	83
Figure 4.4:	High resolution of SqH across Low-, Mid- and High Latitudes – Februar	83
Figure 4.5:	High resolution of SqH across Low-, Mid- and High Latitudes – March	84
Figure 4.6:	High resolution of SqH across Low-, Mid- and High Latitudes – April	84
Figure 4.7:	High resolution of SqH across Low-, Mid- and High Latitudes – May	84
Figure 4.8:	High resolution of SqH across Low-, Mid- and High Latitudes – June	84
Figure 4.9:	High resolution of SqH across Low-, Mid- and High Latitudes – July	85
Figure 4.10:	High resolution of SqH across Low-, Mid- and High Latitudes – August	85
Figure 4.11:	High resolution of SqH across Low-, Mid- and High Latitudes – Sep.	85
Figure 4.12:	High resolution of SqH across Low-, Mid- and High Latitudes – Oct.	85
Figure 4.13:	High resolution of SqH across Low-, Mid- and High Latitudes – Nov.	86
Figure 4.14:	High resolution of SqH across Low-, Mid- and High Latitudes – Dec.	86
Figure 4.15:	Sq(H) variations in January, February and March along the latitudes	87
Figure 4.16:	Sq(H) variations in April, May and June along the latitudes	88
Figure 4.17:	Sq(H) variations in July, August and September along the latitudes	88
Figure 4.18:	Sq(H) variations in October, Nov and Dec. along the Latitudes	89
Figure 4.19a:	Dynamic SqH Variations Contour Map (Jan – Apr)	91
Figure 4.19b:	Dynamic SqH Variations Contour Map (May – Aug)	92
Figure 4.19c:	Dynamic SqH Variations Contour Map (Sep – Dec)	93
Figure 4.20:	GUI Dynamic Sq Variations Platform	94
Figure 4.21:	Execution time for Sequential Algorithm on one Processor	95
Figure 4.22:	Execution time versus number of Processors	95
Figure 4.23:	Speedup versus number of processors	96
Figure 4.24:	Efficiency of the parallel algorithm	96

Figure 4.25:	Execution time on parallel overhead	97
Figure 4.26:	Baseline Cost (\$) for Parallel processors	100
Figure 4.27:	Determination of predicted optimum number of processors	101
Figure C1:	High resolution of SqD across Low-, Mid- and High Latitudes – Jan.	152
Figure C2:	High resolution of SqD across Low-, Mid- and High Latitudes – Feb.	152
Figure C3:	High resolution of SqD across Low-, Mid- and High Latitudes – March	152
Figure C4:	High resolution of SqD across Low-, Mid- and High Latitudes – April	152
Figure C5:	High resolution of SqD across Low-, Mid- and High Latitudes – May	153
Figure C6:	High resolution of SqD across Low-, Mid- and High Latitudes – June	153
Figure C7:	High resolution of SqD across Low-, Mid- and High Latitudes – July	153
Figure C8:	High resolution of SqD across Low-, Mid- and High Latitudes – August	153
Figure C9:	High resolution of SqD across Low-, Mid- and High Latitudes – Sep	154
Figure C10:	High resolution of SqD across Low-, Mid- and High Latitudes – October	154
Figure C11:	High resolution of SqD across Low-, Mid- and High Latitudes – Nov.	154
Figure C12:	High resolution of SqD across Low-, Mid- and High Latitudes – Dec.	154
Figure D1:	High resolution of SqZ across Low-, Mid- and High Latitudes – January	155
Figure D2:	High resolution of SqZ across Low-, Mid- and High Latitudes – February	155
Figure D3:	High resolution of SqZ across Low-, Mid- and High Latitudes – March	155
Figure D4:	High resolution of SqZ across Low-, Mid- and High Latitudes – April	155
Figure D5:	High resolution of SqZ across Low-, Mid- and High Latitudes – May	156
Figure D6:	High resolution of SqZ across Low-, Mid- and High Latitudes – June	156
Figure D7:	High resolution of SqZ across Low-, Mid- and High Latitudes – July	156
Figure D8:	High resolution of SqZ across Low-, Mid- and High Latitudes – August	156
Figure D9:	High resolution of SqZ across Low-, Mid- and High Latitudes – Sep.	157
Figure D10:	High resolution of SqZ across Low-, Mid- and High Latitudes – October	157
Figure D11:	High resolution of SqZ across Low-, Mid- and High Latitudes – Nov.	157
Figure D12:	High resolution of SqZ across Low-, Mid- and High Latitudes – Dec.	157

LIST OF TABLES

	Page
Table 3.1: Excerpt of the 64 INTERMAGNET Observatory stations (1996)	59
Table 3.2: IQD and IDD Dataset (1996)	59
Table 3.3: Stratification of INTERMANGET observatories to latitude zones	61
Table 3.4: Work Breakdown Structure	62
Table 3.5: Sequential and Parallel execution time structure	70
Table 3.6: Parallel Computing System Specifications	73
Table 4.1: Benchmark Metrics for Performance Analysis of Parallel Sq Simulation	94
Table 4.2: Optimization Analysis of the Parallel algorithm	101
Table A: Detailed Distribution of INTERMAGNET Observatories	123

LIST OF ABBREVIATIONS

AMBER	African Meridian B – Field Education and Research
CPU	Central Processing Unit
FLOPS	Floating Point Operation Per Second
GMFV	Geomagnetic Field Variations
GPU	Graphic processing unit
GSM	Global System for Mobile communications
HPC	High Performance Computing
IAGA	International Association of Geomagnetism and Aeronomy
IDD	International Disturbed Days
IQD	International Quiet Days
MAGDAS	Magnetic Data Acquisition System
NASRDA	National Space Research & Development Agency (Nigeria)
NUMA	Non-Uniform Memory Access
Sq	Solar quiet
TID	Travelling Ionospheric Disturbances
UCLA	University of California, Los Angeles
ULF	Ultra-low frequency
SERC	Space Environment Research Centre, Japan
SSDSS	Space Systems Decision Support Service
WDC	World Data Centre

NOTATIONS

\vec{F}	Total intensity of geomagnetic field vector
T_s	Execution times for serial algorithm
T_p	Execution times for parallel algorithm
S_p	Speedup using number of processors (p)
Φ	Efficiency of the parallel systems
μ	Mean value
γ	Parallel overload factor
τ	Concurrency factor for parallel algorithm
λ	Lagrange multiplier
g	Code tuning factor
Θ	Load balance factor

CHAPTER ONE

1.0

INTRODUCTION

1.1 Preamble

The Earth is sometimes referred to as a big dipole magnet because of the magnetic field that originates from its outer core. Apart from facilitating navigation on seas and in air, the magnetic field also shields the Earth from ultra high energy wind that is continuously released from hydrogen explosions in the Sun. This magnetic field is not constant but varies in space and time because of the strong dynamic relationship between the Earth and the sun. Although the spatial scope of this variation is global, the rate is infinitesimal in time due to continual solar explosion. The variation is responsible for space weather and most other geophysical activities that are observed on Earth. Hence, the finer the time scales of computation and analysis of geomagnetic field variations, the better our understanding of many geophysical phenomena. Thus, this work presents a high (time) resolution analysis of geomagnetic field variations on a global scale using the benefits of parallel computing.

The central theme of this study is on the simulation of global spatio-temporal behavior of the large-scale ionospheric electric current system in the Earth's upper atmosphere. Detailed analysis of the variations caused by electric currents flowing in the ionosphere and magnetosphere require a subtraction of the dominant and slowly varying Earth main field from observed geomagnetic field data. In this regard, the study considers the occurrence of geomagnetic variations as a result of electric currents flowing in the ionosphere and magnetosphere. This differential results in the dominant and slowly varying Earth main field. For this purpose, the magnetic dataset acquired through the ground-based magnetometer observatories was used to analyse geomagnetic field variations during quiet time condition due to its nearly global and continuous representation of the ground level magnetic field perturbations. Sequential and parallel algorithms were developed to solve for solar quiet (Sq) daily variation of geomagnetic field using computational-intensive geomagnetic dataset from 64 observatory stations. The developed Sq parallel algorithm was optimized and further evaluated using parallel performance metrics established in Amdahl's and Gustafson's laws.

Modern applications in science, engineering and commercial fields involve problems with relative large and complex dataset that are impractical to solve on a single CPU-based computer. Data size and performance demands are increasing daily, thus there is requirement for high processing speed with thousands of gigaFLOPS and gigaBytes of computing resources (Zenios, 1999). Thus, monitoring of transient variations of geomagnetic field time series data is of much scientific importance consequent upon solar activities that vary from milliseconds to millions of years. This study therefore seeks to investigate an efficient application of parallel computing in solving solar quiet daily variation of geomagnetic field using one-minute (60-second) time series dataset.

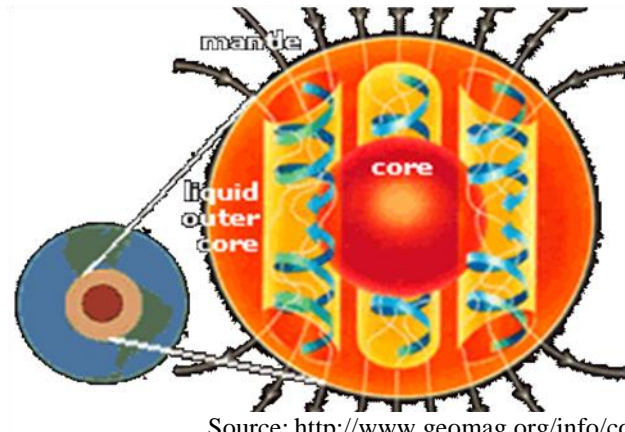
1.2 Background to the Study

Geomagnetic field is the Earth's magnetic field that envelopes its outer space. It shields the Earth from solar-driven energetic particles that could strip off the upper atmosphere. The geomagnetic field varies in space and time. This variation has been of significant interest to scientists since the beginning of the 18th century (Chapman & Bartels, 1940). Comprehensive literature review has further revealed that application of advanced computational methods would support detailed analyses, understanding and monitoring of ionospheric current which accounts for geomagnetic field variations. Parallel computing is the thrust of high-performance systems which involve parallel algorithms, parallel architectures and fast interconnection networks for parallel execution of large and complex problems to produce optimal results within a reasonable time frame (Michael, 1994). The shift in computing paradigm from sequential to parallel computations was necessitated by large computationally-intensive applications.

1.2.1 The Concept of Geomagnetism

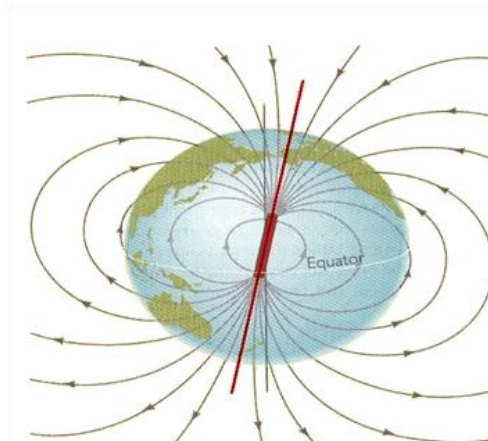
Chapman & Bartels (1940), Jacobs (1987) and Kelley & Heelis (1989), described geomagnetism as the study of the Earth's magnetic field. Merrill, et al. (1996), Campbell (1997) and Stern (2002) attributed geomagnetic field to the combined effects of the planetary rotation and the movement of molten iron in the Earth's core. Riktake (1961), Fleagle & Businger (1963) and Rishbeth (1971) used Figures 1.1 and 1.2 to illustrate the configuration of fluid flow and magnetic field in the liquid outer core as well as the imaginary lines that envelope the Earth. Campbell (1997) further illustrated the Earth's magnetic field and its directional components with north and south poles reaching about

36,000 miles into Space. This creates a protective bubble around the Earth that deflects ultra high energy solar wind.



Source: <http://www.geomag.org/info/coreflow.html>

Fig. 1.1: Fluid flow and magnetic field in the liquid outer core



Source: <http://csep10.phys.utk.edu/astr161/lect/earth/magnetic.html>

Fig.1.2: The Earth's Magnetic Field

1.2.2 Historical Background of Geomagnetic Field

In addition to the shape of the earth, the magnetic field was known to be among the initial properties attributed to the Earth as a whole according to the study by William Gilbert, physician to Queen Elizabeth I. Gilbert (1958) thereafter published his findings in “*De Magnete*” in the 16th century. This predates Newton’s gravitational Principia by

about 87 years. William Gilbert observed that the compass needle always pointed north-south and therefore inferred that the Earth itself was a huge magnet. Needham (1962) also reported that the magnetic compass had been in use by the Chinese since about the second century B.C, but it did not find its way to Europe until years later, where it became an indispensable tool for maritime navigation. According to Taylor (1951), the first scientific work on magnetism, about the discovery of magnetic meridians, the dipolar nature of the magnet and description of compass magnet was credited to Petrus Peregrinus. The *Epistola de Magnete* was later written in 1269 and subsequently widely circulated in Europe but was not published until the 16th century and later reviewed by Kane (1971a) and Hibberd (1985).

Although Gilbert (1958) placed the source of magnetism within the Earth in 1600, the temporal variations in the magnetic field were not formally documented until the middle of the seventeenth century when Henry Gellibrand appreciated that the differences among repeated measurements were more serious than inaccuracy in the observations. In 1680 Edmund Halley published the first contour map of the geomagnetic variation as the declination was then known (Bernard, 1981). Although Bernard did not know of the existence of the fluid outer core, he investigated the secular variation of the field as being caused by a collection of magnetic dipoles deep within the earth drifting westward with time in about a 700-year period. Also, a formal separation of the geomagnetic field into parts of internal and external origin was first achieved by the German mathematician Karl Friedrich Gauss in the nineteenth century (Chapman, 1967). Gauss also initiated the measurement of the geomagnetic field on global scale via distributed observatories. Some of those early observatory stations such as Kew observatory are still functional till today (Robert, 1885).

1.2.3 Geomagnetic Field Components

According to Vestine (1967), geomagnetic field is defined as a vector quantity (\vec{B}) that can be measured in magnitude (strength) and direction on the surface of the Earth as shown in Figure 1.3.

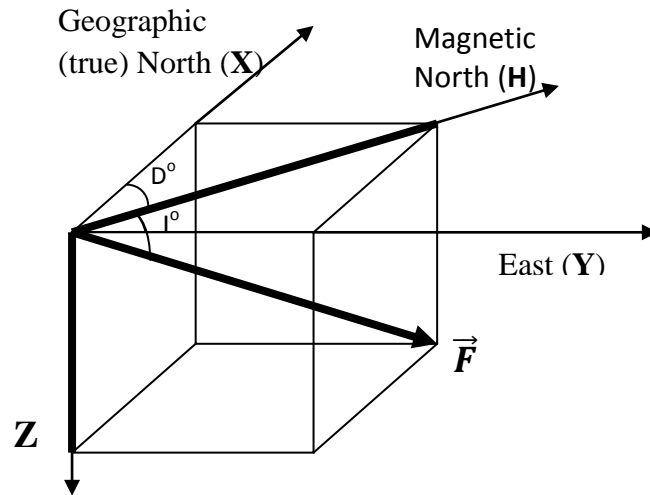


Fig.1.3: The Earth's Magnetic Field Components

From Figure 1.3, the Earth's magnetic field is thus described by the following seven components:

- (i) **Declination (D)** indicates angular difference between the true north and magnetic north
- (ii) **Inclination (I)** is the angle, in degrees, of the magnetic field above or below horizontal.
- (iii) **Horizontal Intensity (H)** defines the horizontal component of the total field intensity.
- (iv) **Vertical Intensity (Z)** defines the vertical component of the total field intensity.
- (v) **Geographic north / northerly intensity (X)** and **Geographic eastward component (Y)**
- (vi) **Total field Intensity (\vec{F})** is the strength of the magnetic field

The geomagnetic field vector (\vec{B}) at each point in space can be described completely either by three orthogonal strength components (X,Y,Z), total strength with an angle (F,D,I) or two strength components with one angle (H,D,Z). The field is often represented either in geographic XYZF or magnetic HDZF coordinate depending on the sensor orientation. The orthogonal relationship that exists between the geographic and magnetic planes enables easy conversion from one orientation to the other as required in the analyses.

1.2.4 Overview of Geomagnetic Field Variations

Due to continuous dynamic interaction between the exploding plasma of the sun, geomagnetic field varies continuously in space and time. The time scale of this variation ranges from milliseconds to millions of years. Variations of shorter time scales often arise from strong currents in the ionosphere and magnetosphere. Some of these changes can be traced to geomagnetic storms or daily variations in currents. Secular variation occurs over time scales of years and mostly reflects changes in the Earth's interior. Hibberd (1981) observed that geomagnetic field is so irregular that it must be measured in many locations for better visualization and accuracy of its distributions. The differential in solar-driven ions and electrons in the ionosphere forms current system which is responsible for variations in the intensity of the Earth's magnetic field. Figure 1.4 shows types of geomagnetic field variations.

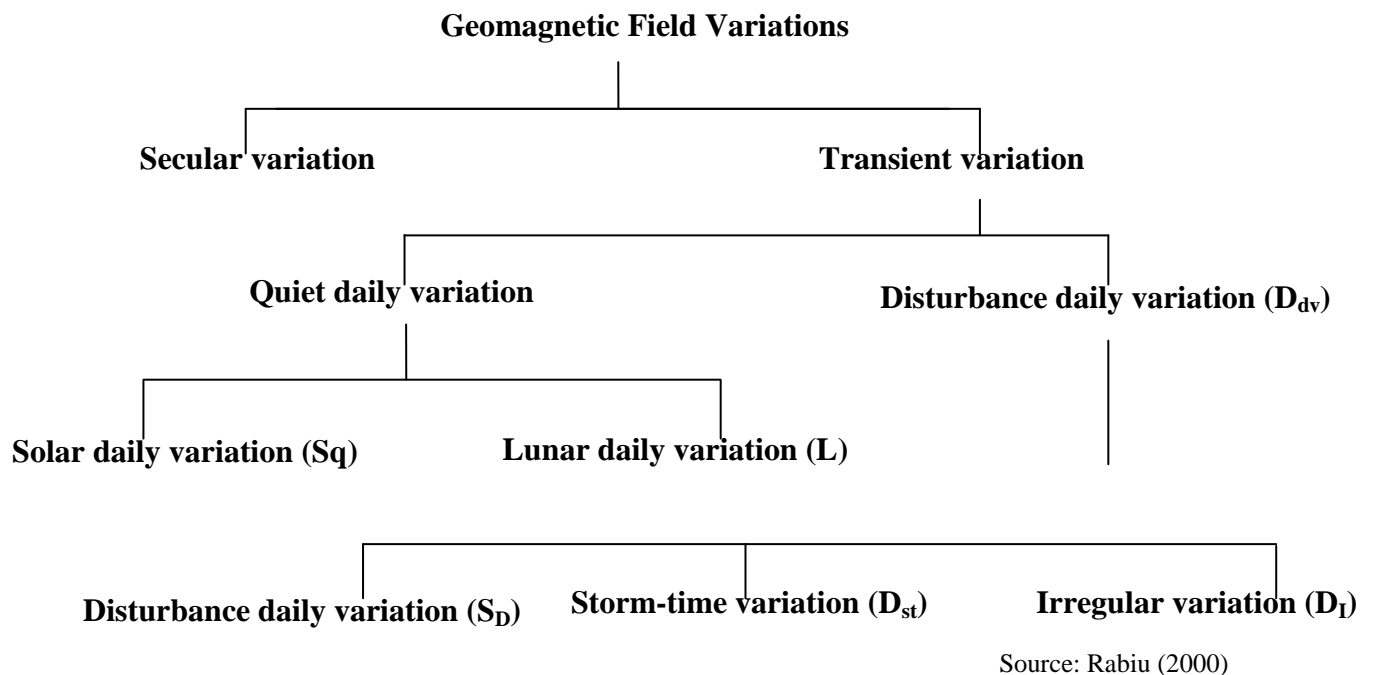


Fig. 1.4: Types of Geomagnetic Field Variations

1.2.5 Spatio-Temporal Variations of Geomagnetic Field

According to Matsushita & Maeda (1965a), Matsushita (1967) and Matveyenkov (1983), spatial variation is about the field variation from one place to another while a time dependent variation in geomagnetic field is known as temporal variation. This type of variations has periods ranging from 10^{-5} seconds to about 100 million years. In general,

temporal variations are classified into transient variations and secular variation as shown in Figure 1.4. Transient variations include the spectrum of variations having periods from seconds up to days while the secular variation refers to slow changes that are detectable in years or even centuries. Hasegawa (1960), Last (1976) and Hibberd (1981) examined the secular variations using hourly mean dataset of geomagnetic field horizontal intensity for Alibag during the international quiet days for the years 1934 to 1953 but split into two 10-year segments. However, the result based on H-component magnetogram alone was not sufficient to express a representation of ionospheric current sheet for Alibag area (Matsushita, 1960; Mayaud, 1976; Onwumechili et al., 1989).

Transient variations are further divided into quiet daily variations and disturbance daily variations. On some days the magnetic records change with primarily 24-, 12-, 8- and 6-hour period with spectral components dominating the field composition. On such days the records follow a pattern of gradual change through the seasons of the year and show a clear repetition from day to day. Such days are called magnetic quiet days and the associated variations are regarded as the quiet daily geomagnetic variations. Irregular and erratic changes characterized the records of magnetic disturbed days and the variations are called the disturbance day variation as reported in the work by Egedal (1947), Matsushita & Maeda (1965b), Rajaram & Mitra, (1981) and Onwumechili (1992a).

The quiet daily variations consist of the solar quiet daily variation (Sq) which is regular diurnal variations related closely to solar activities as well as some ionospheric phenomena, and the lunar daily variation (L), which is a regular semi-diurnal variation ascribed to the lunar-tidal current system. The solar quiet daily variation, Sq, can easily be recognized at sight from the magnetograms while the lunar daily variation L cannot because its range is small, rather it can be determined only by averaging over many days. True magnetic signatures are detected when the Earth's field is quiet (Matsushita, 1969; Harper, 1981; Onwumechili, 1985; Takeda, 1985; Sizova, 2002; Schuster, 1989).

In Earth sciences, a magnetogram displays measurement of temporal variation of the local strength and direction of the geomagnetic field. Such magnetograms have existed since Victorian times, and the British Geological Survey has preserved records from the 1850s, showing the effects of the 1859 Carrington Event, the worst magnetic storm known to have hit Earth. The disturbance day variations (D_{dv}) are variations due to the

disturbance in the geomagnetic field. Forbush & Casaverde (1961), Osborne (1966), Matsushita (1975), Forbes (1981) and Campbell (1989) observed in their findings that the disturbance daily variations are of three forms as follows:

- (i) The disturbance daily variation which is the difference between the mean of magnetic elements for a disturbed day and the monthly mean for all the days.
- (ii) The storm-time variation Dst is the regular part of the disturbance day variation and is estimated with reference to the hour of commencement of the storm.
- (iii) The irregular disturbance variation is the remainder of D_{dv} which is completely irregular being neither related to the storm-time nor to solar time.

In the studies carried out by Ogbuchi et al. (1967), Olson (1974) and Onwumechili & Ezema (1977), it was observed that all geomagnetic field components when recorded on magnetograms on some days showed very smooth and regular traces that follow clear patterns of gradual daily variations with respect to solar time. These days are called International Quiet (geomagnetic) Days (IQD) and the records are known as solar quiet daily variations (Sq). The authors ascertained that the variation is attributed to ionospheric current systems which are being driven by ionospheric tides that blow the ionized air across the lines of force of the geomagnetic field thereby generating electric fields to drive the electric currents. Therefore, the observed solar quiet daily variation is a reflection of the effects of the solar influences operating to produce ionization and the strength of upper air winds. Kane (1971c), Suzuki (1978), Onwumechili & Agu (1981), Mann & Schlapp (1988), Campbell (1997) and Onwumechili (1992f) explained that when the small but persistent effects due to the lunar-tidal current system (L) have been removed, the changes are called Sq for solar quiet fields, referring to their local time changes when solar terrestrial disturbances are absent.

The quiet day solar daily variation, Sq, according to Chapman & Bartels (1940) was derived from five quiet days per month. This estimation was considered to be more reliable in low and middle latitudes. Most old and recent studies on Sq variability of geomagnetic field were carried out on daily and annual basis, and their results were analyzed and reported on sectors or regional basis (Chapman & Stagg, 1931; Kane, 1975; Geener & Schlapp, 1979; Phillips & Briggs, 1991; Mazauder, 1993; Okeke et al.,

1998). Although, the results of similar regions were co-herent, there was no standard platform to benchmark their findings with respect to time and space. In order to obtain accurate estimation of Sq, Price & Wilkins (1963), Vestine (1967) and Lilley (1975) suggested the correction of non-cyclic variation, a phenomenon in which the values at times transition differ (e.g 01hour and 24hr or 01 minute and 1440 minute).

1.2.6 Geomagnetic Field Instrumentations

Magnetometers are common type of scientific instruments that measure magnetic field intensity and direction. Ground-based observatories are usually installed in locations free from magnetic materials and artifacts so as to prevent magnetically induced errors. Some institutes that operate magnetic arrays across the globe include Canadian Geospace Monitoring, Greenland Costal Array, MAGIC, CPMN/MAGDAS, Russian Artic, IMAGE, INTERMAGNET among others. Also, space-borne magnetometers are available as payload for THEMIS, GOES, CHAMP, ØERSTED, MAGSAT and SWARM low-orbiting satellites. Acquisition of geomagnetic field data from ground-based observatories is more convenient. Unlike satellite data, ground-based magnetometers do not have mission lifespan, thus they are more affordable, durable and deployable especially in the developing countries.

1.2.7 The Sun and Solar Activities

Friedman (1960) described the Sun as a spherical visible and magnetically active star at the center of our Solar System. The Sun consists of hot plasma (ionized gas) interwoven with magnetic fields. The Sun has an estimated diameter of about 1,392,000 km, about 109 times that of Earth, and its mass (about 2×10^{30} kilograms, 330,000 times that of Earth). Hargreaves (1979) further observed that the sun chemical composition consists about three quarters of hydrogen while the rest is mostly helium. Thus, the Sun's startling radiance provides the warmth and light that bathes the entire earth as established in the observations and studies carried out by Kirchoff & Carpenter (1976), NASA (1993) and Lang (2000).

Kelly & Heelis (1989) found out that the Sun's magnetic field and solar explosions often lead to many effects that are collectively known as solar activities. These activities include sunspots, solar flares, coronal mass ejection, geomagnetic storms and variations

in solar wind that carry materials through the Solar System. The solar activities change over time and solar magnetic field extends well beyond the Sun itself therefore causing perturbation on the geomagnetic field beginning at the magnetosphere. The perturbation leads to geomagnetic field variations which are being felt on the ground and recorded by the ground-based magnetometer instrumentations. Thus, investigation of these geomagnetic field variations during solar quiet time enables better understanding and monitoring of induced ionospheric current in the Earth upper atmosphere. These solar explosions are continuous in time and take few days or hours to reach the Earth, leading to some adverse effects on Earth systems. Furthermore, the impact of solar activities stratifies the Sun-Earth environment into troposphere, stratosphere, mesosphere, ionosphere and magnetosphere layers as shown in Figure 1.5. Solar variation is therefore due to varying amount of radiation emitted by the Sun which has a spectral distribution that spans over several years to millennia. These variations may be periodic or aperiodic.

Axford (1967) also found that the differential rotation of the Sun's latitudes causes twisting of geomagnetic field lines over time resulting to magnetic field loops from the Sun's surface. This action triggers the formation of the Sun's active region involving sunspots, solar prominences and coronal holes at its visible surface. This twisting action also creates the solar dynamo and an 11-year solar cycle of varying magnetic activity in which the Sun's magnetic field reverses itself about in a period of approximately 11 years. This solar cycle is also known as sunspots cycle.

1.2.8 The Ionosphere

The ionosphere is a region in the upper atmosphere where high solar-driven energetic (charged) particles exist in such a sufficient proportion that can affect radio waves. The energetic particles interaction with the Earth's magnetic field produces magnetic disturbances and thereby increases ionization in the ionosphere.

According to Akasofu & Chapman (1972), the ionosphere layer is located between the altitudes of 50 km and 1000 km in the Earth upper atmosphere where most activities of space-borne technological systems take place. Some of these technological systems include satellites, space shuttle, rockets missiles, radar, lidar and many others. The signal and radio waves from such systems can be reflected, deflected or diffracted depending on

the intensity of solar impact and ionospheric current at different latitudes, longitudes and time period of the Earth. The ionospheric current effect is more intensive along the equatorial region due to much exposure to the Sun in most time of the year. This peculiar effect has resulted to equatorial anomaly, equatorial electrojet and plasma fountain.

Campbell (1997) also established that solar quiet daily variation (Sq) current intensity has direct influence on terrestrial climate and seasons. Maynard (1995) therefore described space weather as the change in ambient plasma, magnetic fields and ionospheric currents or simply as the general condition of Sun-Earth environment.

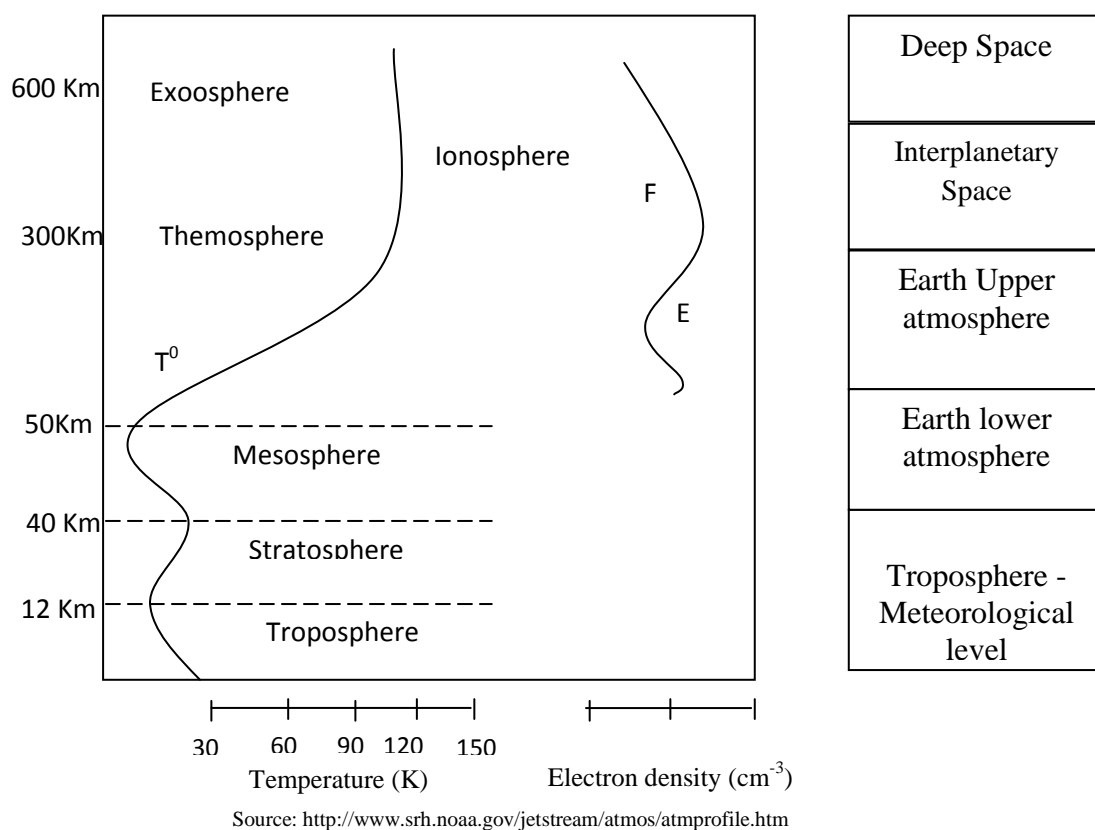


Fig. 1.5: The Earth Atmospheric Regions

1.2.9 The Concept of Parallel Computing

Modern advanced computation systems involve the use of parallel supercomputers, parallel algorithms, high and faster computing resources to concurrently solve a computational-intensive, large or complex but parallelizable problem in order to achieve optimized solution in a shorter execution time. This advanced computation technique also known as parallel computing is efficient at proffering solutions to most scientific

and engineering design problems. Examples of such problems include the solution of full *Navier–Stokes* equations for aerodynamics, hydrodynamics, energy equations, space weather modeling, genetic modeling, broad spectrum motion/energy modeling, geomagnetic field simulations, financial modelling among several others.

According to Ananth et al. (2005), parallel computing is the simultaneous use of multiple processing units to solve a large computationally-intensive problem. These processing units include array of processors, large memories and faster interconnection networks. Conventional software have been developed for serial computation which executes only one instruction at time. Today, data-intensive and grand challenge problems in commerce, science, engineering and industrial sectors have provided an equal or greater driving force in the development of supercomputers with parallel processing functionalities. Figure 1.6 shows schematic representation of parallel computation of a large problem that is split among multiple processors. Thus, parallel computation shortens execution time, saves cost, produces optimal solutions and also increases efficiency as well as yielding overall performance gain when compared to sequential systems.

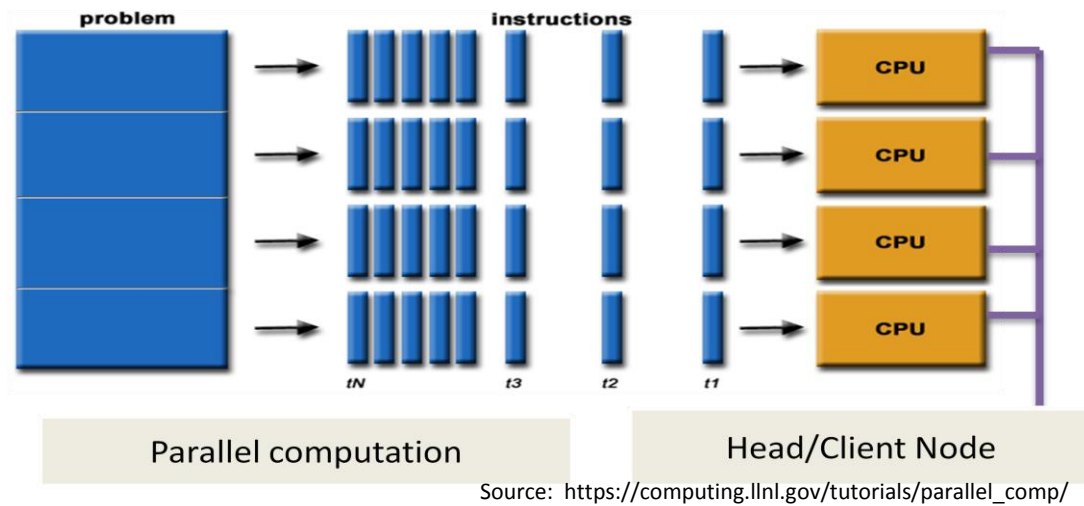


Fig. 1.6: Schematic representation of Parallel Computation

Although, parallel computers were introduced in the 1970's, development of efficient parallel algorithms, codes and software has been a major challenge that limits versatility of parallel applications. Michael (1994) observed that retarded expertise and non-effective industry standard models in parallel programming were due to late commercial availability of parallel computers. Problems associated with parallel computing,

therefore, involve load imbalance, synchronization of tasks, data or instructions partitioning and data dependence. Other details include message passing interface, parallel overload and absence of open architecture as well as diversities of incompatible parallel programming and hardware models. The commonly available parallel computing architectures are multiprocessors, multi-cores, many-GPU core, high-performance clusters, Beowulf, grid and cloud computing architectures. Multi-processing is an earlier form of parallel computation that involves processing of multiple distinct problems (jobs) on a parallel system with multiple processors. However, the algorithms were still executed sequentially.

1.2.10 Performance Evaluation of Parallel Systems

Amdahl (1967) and Gustafson (1988) developed models and performance metrics for evaluating parallel programming on parallel systems with respect to execution time. Amdahl's law established that speedup is limited by fraction of a code that cannot run completely in parallel. Other derivatives of performance metrics include speedup, efficiency and scalability. The speedup is simply the ratio of serial execution time and parallel execution time. Scalability of a parallel system was defined by Ananth et al. (2005) as the ability to maintain a proportionate performance gain with relative increase in problem size and computing resources. Efficiency is a measure of parallel performance that is closely related to speedup.

1.3 Statement of Problem

Geomagnetic field varies in space and time on a nanosecond basis. However, to date, algorithm development and architecture limit the time resolution of monitoring solar quiet (Sq) daily variation to hourly and regional basis. This is obviously inadequate for detailed analyses of some geophysical phenomena with time scales less than hours. Also, regional analysis of this variation is insufficient for inter-regional phenomena. This work therefore computes Sq daily variation across the globe on second basis.

Till date, comprehensive literature reviews show that analyses of solar quiet variations of geomagnetic field have been carried out using seasonal, annual, diurnal and hourly dataset on regional basis. The restriction to large time frames and regional Sq analyses are often due to limited availability of advanced computing resources, mainly, efficient algorithms for real-time processing of large and computational-intensive geo-magnetic

dataset (with finer time resolution). Computations of Sq using 1-minute sampled data will give more insight to the variations of geomagnetic fields (namely, Horizontal (H), angle of declination (D) and Vertical (Z) components). Due to limited availability of high performance computing resources and efficient algorithm, it has been impracticable to investigate Sq variations of the three (3) geomagnetic fields i.e. SqH, SqD and SqZ using 1-minute data on global scale. The problem under investigations generates large matrices which involves several iterations that will take a comparative long time to be processed depending on the speed and storage capacity of the computing resources.

For instance, if a simulation involving data- and computationally-intensive work such as this takes two (2) weeks to be completed on a single computer, any error or mistake realized at the end of the simulation will result to loss of time and waste of computing resources. Therefore, a study of this type requires parallel processing to obtain efficient and time relevant result. The use of 1-minute dataset will generate large matrices that cannot be processed efficiently on single-processor based computer in a reasonable time.

1.4 Aim and Objectives of the Study

The aim of this work is to present a high resolution (i.e. order of minutes) analysis of spatio-temporal variations of geomagnetic field on global scale using parallel computing.

The objectives of the research are:

- a) To improve on the time resolution for the analysis of geomagnetic field variations;
- b) To extend the scope of spatial analysis of geomagnetic field variations from regional to global basis;
- c) To develop an efficient computational algorithm able to achieve objectives (a) and (b); and
- d) To develop a 3-D visualization platform for real-time monitoring of geomagnetic field variation.

1.5 Research Questions

This study intends to answer the following set of questions:

- (i) What is the significance of processing geomagnetic field variations using one-minute time series magnetic dataset?

- (ii) How do we develop a scalable parallel solar quiet daily (Sq) variation algorithm that can take advantage of multiple processors with large geomagnetic dataset?
- (iii) How do we implement kriging methods as geostatistical data gridding tool to enhance parallel computation of solar quiet variations of geomagnetic field?
- (iv) Can we effectively use approximation methods to extrapolate runtimes of parallel algorithm in view of Runge's phenomenon?
- (v) Is there any area of improvement on Amdahl's laws of parallel systems performance considering emerging evolution of parallel computing technologies?

1.6 Scope and Limitation of the Study

This research is concerned with the development of parallel computation platform for analysis of spatio-temporal variations of geomagnetic field during quiet condition, that is, when solar wind input to magnetosphere is minimal. In addition, this work is limited to the following:

- (i) Acquisition of geomagnetic field data from ground-based observatory stations.
- (ii) The work excludes the simulations of solar systems, sunspots, secular variation and magnetic reconnection.

1.7 Significance of the Study

Although, there are diversities of space weather research areas, Sq variation analysis for ionospheric current simulation is an essential component. Adverse condition of the ionosphere can cause disruption of satellite operations, communications networks, GSM navigation, and electric power distribution grids, resulting into a variety of socio-economic problems (James, 2007). This severe impact on society is a major reason for scientific interest on Space weather research and other heliospherical studies. In addition, this work is motivated by the need to assess space weather impact on failure of some sensitive technological systems. One-minute digital data is also important for a close study of rapid impulses and variations of the geomagnetic field. This is useful for analysis of geomagnetic phenomena with time scale less than hours, such phenomena include travelling ionospheric disturbance, skywaves, ultra low frequency waves among others (Chi & Russel, 2005). Since the ULF waves which are observable in geomagnetic

field variations have been confirmed to be a precursor to earthquakes, it is therefore expected that the results of this study will provide space systems decision support service (SSDSS) for generating effective early warnings on possible occurrence of catastrophic geotechnical phenomena such as earthquakes, typhoon, geomagnetic storms and satellite orbital decay. This will enable relevant decision makers and other stakeholders to ensure safety of space-borne and ground-based technological systems as well as to initiate development of local competence in the areas of geomagnetic field instrumentations.

1.7.1 Space Weather Phenomena

Mendillo et al. (1981) and Maynard (1995) defined Space weather as the concept of changing environmental conditions in near-Earth space or the space from the Sun's atmosphere to the Earth's atmosphere. It is distinct from the concept of weather within the Earth's planetary atmosphere (troposphere and stratosphere). Space Weather is the description of changes in the ambient plasma, magnetic fields, radiation and other matter in space. Much of space weather is driven by energy carried through interplanetary space by the solar wind from regions near the surface of the Sun and the Sun's atmosphere namely the chromospheres and corona (Chalmers, 1962). The term space weather was not used until the 1990s. Prior to that time, activities now known as space weather were considered to be part of physics of aeronomy or space exploration. Scientific research and relevant applications have significantly influenced the study of Space weather (Matsushita & Tarpley, 1970; Onwumechili, 1992b, 1992c, 1992d).

1.7.2 Solar-Driven Geomagnetic Storms

Geomagnetic storm was defined by Langel et al. (1996) as a momentary disturbance of the Earth's magnetosphere as a result of solar wind penetration. A geomagnetic storm is a major component of space weather that provides the input for many other constituents of space weather. A geomagnetic storm is caused by a solar wind shock wave and/or cloud of magnetic field which interacts with the Earth's magnetic field to form magnetosphere region.

1.7.2.1 Solar Wind

Bhargava & Yacob (1971), Rastogi et al. (1971), Randall & Randall (1991) and Rastogi (1996) described solar wind as a supersonic flow of plasma that carries energetic

particles from the sun through the sun atmosphere into space forming the interplanetary magnetic field (IMF). The solar wind wraps Earth's geomagnetic field lines, flattening them on the sun-ward side and stretching them out on the downstream side. The influence of this distortion of the geomagnetic field is quite small near Earth's surface and becomes larger with increasing distance from Earth. However, its significant effect can be felt during solar eruptions as illustrated by Burda et al. (2009).

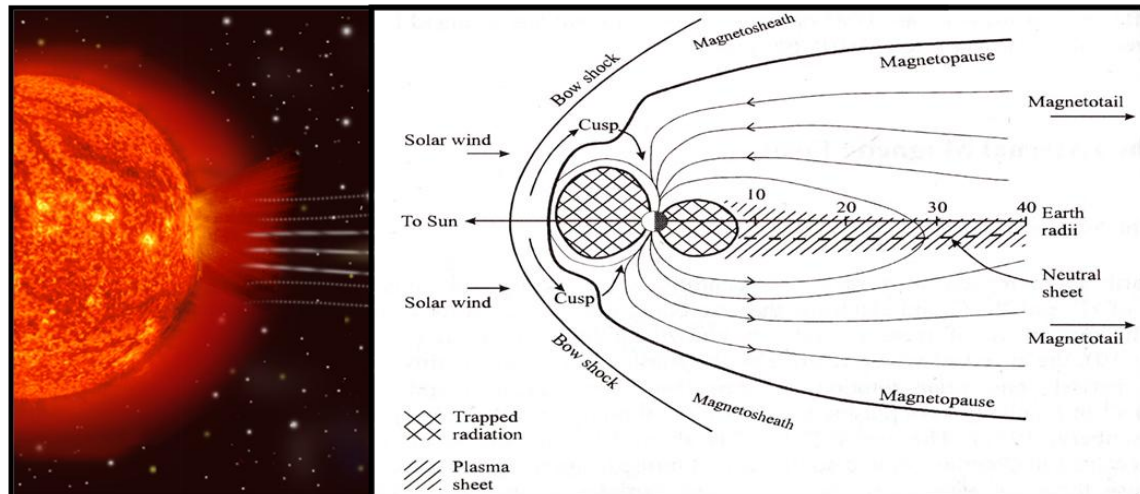
1.7.2.2 Solar Flares

Solar flares is a type of solar storms that were first observed on the Sun by Richard Christopher Carrington and independently by Richard Hodgson in 1859 as localized visible brightening of small areas within a sunspot group. Richard (1859) and Kopp & Rottman (2005) explained solar flare as a sudden (timescales of minutes, hours and days) outburst of radiation and energetic particles. The flare ejects clouds of electrons, ions, and atoms through the corona into space. The frequency of occurrence of solar flares varies, from several cycles per day when the Sun is particularly active to less than one every week when the Sun is quiet due to 11-year solar cycle.

1.7.2.3 Coronal Mass Ejection

Sarabhai & Nair (1969) described coronal mass ejection (CME) as a larger scale violent ejection of high energetic particles into space. If directed towards the Earth, CMEs can reach it within a few days and cause serious disruptions of the Earth's magnetic field resulting to geomagnetic storms. Salen (1968), Reddy & Devasia (1981) and Schmucker (1999a) observed that solar storms are more frequent near the peak of solar activity.

The increase in the solar wind pressure initially compresses the magnetosphere making the solar wind magnetic field to interact with the Earth's magnetic field and thereby transfer an increased amount of energy into the magnetosphere. Both interactions cause an increase in movement of plasma through the magnetosphere, this is driven by increased electric fields inside the magnetosphere, thus an increase in electric current in the magnetosphere and ionosphere. During the main phase of a geomagnetic storm, electric current in the magnetosphere create magnetic force which pushes out the boundary between the magnetosphere and the solar wind as shown in Figure 1.7.



Source <http://sec.gsfc.nasa.gov/popscise.jpg>

Fig. 1.7: Solar wind and Earth's Magnetosphere

The disturbance in the interplanetary medium which drives the geomagnetic storm may be due to a solar coronal mass ejection or a high speed stream co-rotating interaction region (CIR) of the solar wind originating from a region of weak magnetic field on the Sun's surface. The frequency of geomagnetic storms varies with the Sunspot cycle. Therefore, CME driven storms are more common during the maximum of the solar cycle while CIR driven storms are more common during the minimum of the solar cycle as demonstrated by Tarpley (1970; Stening, 1973; Sastri, 1981; Schmucker 1999b). Behnke & Hagfors (1974), Balan & Iyer (1983) and Brekke et al. (1994) discovered in their various studies that vast explosions, flares and CMEs activities often alter the Sun's radiation and solar wind over time scales from minutes to hours and to days.

1.7.3 Effects on Space- borne Technological Systems

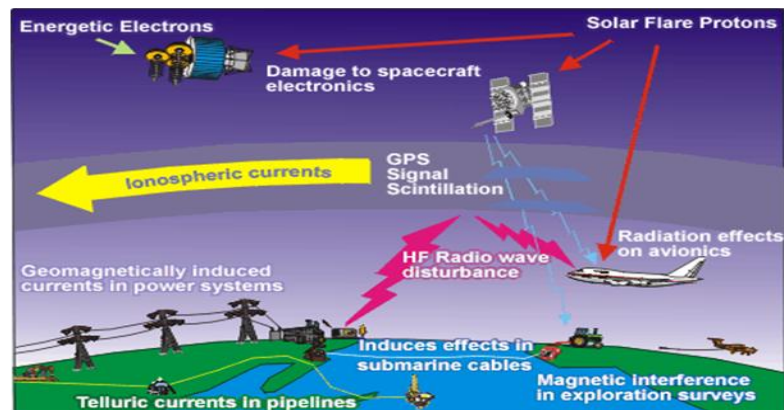
Although solar activities with geomagnetic interactions have some benefits, their adverse effects on human and technological infrastructure are sometimes devastating. Human dependence on satellites for earth observations and information communication is increasing by day, therefore Space weather will continue to have more impact on society than in the past. The effects of geomagnetic storms are obvious on technological systems as expressed by Suzuki (1979), Soicher & Gorman (1981) and Takeda (1984). According to Lanzerotti (2001), space weather phenomena have a variety of effects on technological systems as shown in Figure 1.8. Energetic particles from the sun interact with the earth's magnetic field producing magnetic disturbances and increased ionization in the

ionosphere. The high energy particles effect on satellites may cause single event upset, single point failure or equipment damage that can knock satellites out of mission. Space weather events can cause damage on-board data handling satellite subsystem which can lead to a single point failure as a result of single event upset. Avionics can be damaged through exposure to highly energetic particles thereby reducing a satellite's lifetime. Radio waves used for satellite communications or GPS navigation can be affected by the increased ionization with disruption of the communication or navigation systems.

Obviously, the top layer of the Earth's atmosphere heats and expands at times of increased solar activity. The resulting increased friction often causes atmospheric drag which slows down satellites in orbit and can thereafter drop satellites to lower altitudes and sometimes out of orbit due to orbital decay. Active satellites need to use up costly fuel to maintain orbit maneuvering and station keeping (Meyer-Vernet, 2007). Geomagnetic storm disturbance on the ionosphere may temporarily degrade or disrupt broadband and TV communications, navigation signals and other aviation instrumentations (such as GPS for timing signals). It also has impact on Internet links on which society increasingly depends. Similarly, radio operators using HF bands rely upon solar and geomagnetic alerts to keep their communication circuits up and running.

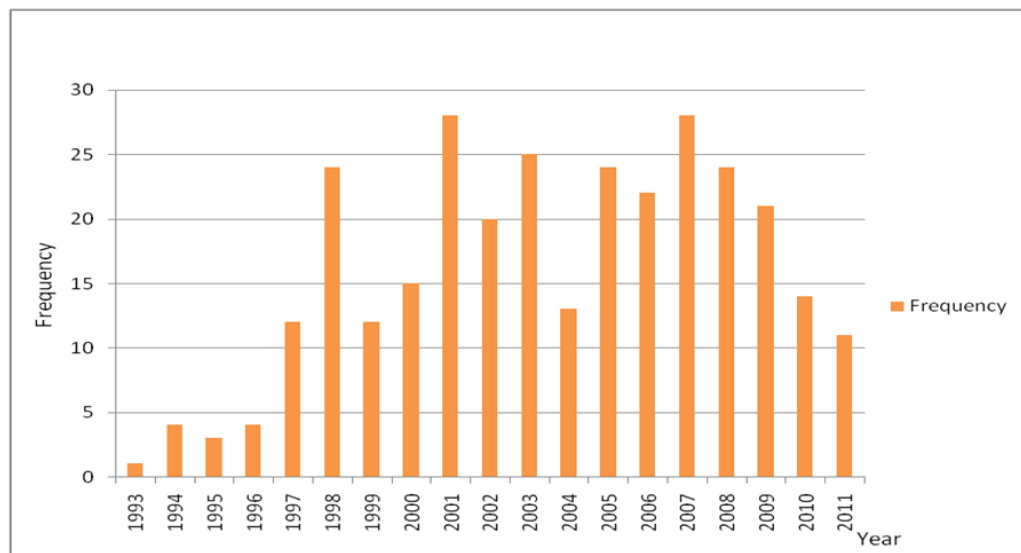
These effects can be particularly significant on high-altitude and high-latitude flights. Although the Earth's atmosphere and magnetic field offer some protection, unexpected solar events may still cause severe damage. If not inadequately shielded, the exposure of astronauts, air crews and space-bound passengers to ionizing radiation can cause chromosome damage, cancer or immediate death of living cells. In addition, the South Atlantic Anomaly (SAA) which refers to the area where the Earth's inner Van Allen radiation belt comes closest to the Earth's surface is a dangerous zone for orbiting spacecrafts. As technology has allowed spacecraft components to become smaller, the miniaturized systems have become increasingly vulnerable to the more energetic solar particles. These particles can significantly cause physical damage to microchips and can change software commands in on-board computers (Meyer-Vernet, 2007).

Figure 1.9 shows the graphical representation survey of satellites failure from 1993 – 2011 carried out by Satellite News Digest (2011). Most of the failures are as a result of Space weather adverse effects. The understanding and monitoring of Space weather is therefore necessary for prediction of safe launch window and guidance of spacecraft during launch and subsequently during in-orbit mission life.



Source: <http://www.spaceweather.gc.ca/tech/se-eng.php>

Fig.1.8: Space weather effects on Technology



Data Source: Digest (2011)

Fig.1.9: In-orbit satellite failures mostly due to space weather effects

1.7.4 Effects on Ground-based Technological Systems

Magnetic disturbances also induce electric currents in long conductors such as power lines and pipelines causing power system outages or pipeline corrosion (Hofmann et al., 1994). Geomagnetic storms can affect ground-based systems by causing geomagnetically induced currents (GIC) in conducting networks electrically grounded in the Earth's surface (for example pipelines, power systems or railways). GIC also affect high-voltage electricity transformers especially those located near the extremities of electric grids. Power outages may occur if a transformer is damaged enough to require repair or replacement. Operations that use the Earth's magnetic field as a directional reference can also be affected. Geomagnetic storms affect also long-haul telephone lines, including undersea cables except fiber optic (Gneryshev et al., 1977; Campbell, 1986; Ogunade, 1986).

The largest recorded geomagnetic storm occurred in 1859 (Onwumechili & Ogbuchi, 1962; Olson, 1970; Kane, 1971b). It significantly disrupted telegraph systems around the world for about eight hours. Others include Quebec Blackout Storm of March 1989, Halloween Storms of October/November 2003 in Sweden and Japan. These occurrences resulted to huge economy losses and disturbance of human activities (Walker et al., 1998; Wu et al., 2004). Skylab is an example of a spacecraft that re-entered the Earth's atmosphere prematurely in 1979 as a result of unexpected high solar activity (Heppenheimer, 1998).

When magnetic fields move about in the vicinity of a conductor such as a wire, a geomagnetically induced current is produced in the conductor. This happens on a grand scale during geomagnetic storms on all long transmission lines. Power companies which operate long transmission lines are thus subject to damage by this effect. Geomagnetic storm alerts and early warnings can help power companies to minimize damage to power transmission equipment by momentarily disconnecting transformers or by inducing temporary blackouts. Preventive measures can also be applied by digging transmission cables into the soil, placing lightning rods on transmission wires, reducing the operating voltages of transformers, and using cables that are shorter than 10 km (Ivanov & Kostomaroy, 1963). Osborne (1962) observed that rapidly fluctuating geomagnetic fields could also produce geomagnetically induced currents in pipelines. Flow meters in the pipeline can transmit erroneous flow information, and the corrosion rate of the pipeline is

dramatically increased. If engineers incorrectly attempt to balance the current during a geomagnetic storm, corrosion rates may increase even more. Engineers and pipeline managers will need space weather alerts and warnings to allow them to implement defensive measures.

1.7.5 Effects on Geologic exploration

According to Hutton (1967), Earth's magnetic field is used by geologists to determine subterranean rock structures. This helps geodetic surveyors to search for oil, gas, or mineral deposits. They can accomplish this only when Earth's field is quiet, so that true magnetic signatures can be detected. Some geophysicists prefer to work during geomagnetic storms, when strong variations in the Earth's normal subsurface electric currents allow them to sense subsurface oil or mineral structures. This technique is called magnetotellurics. For these reasons, many surveyors use geomagnetic alerts and predictions to schedule their mapping activities. In addition, the magnetic disturbances directly affect operations that use the magnetic field, such as magnetic surveys, directional drilling or compass for navigation.

1.7.6 Effects on Military Facilities

Some military detection or early warning systems are also affected by solar activity. The over-the-horizon radar bounces signals off the ionosphere to monitor the launch of aircraft and missiles from long distances. During geomagnetic storms, this system can be severely hampered by radio clutter. Some submarine detection systems use the magnetic signatures of submarines as one input to their locating schemes. Geomagnetic storms can mask and distort these signals (Kane, 1971b; Akasofu & Chapman, 1972).

1.7.7 Effects on Geophysical Phenomena

Studies by Smirnova et al. (1999) and Hayakawa et al. (2000) have shown that ultra low frequency (ULF) wave is a precursor to occurrence of earthquakes, tremor and typhoons. Since frequency is directly related to geomagnetic field with gyromagnetic ratio as the constant of proportionality, therefore, a high spatio-temporal resolution geomagnetic field variation will enable clear visibility, detection and monitoring of ULF waves. Thus, this would provide an effective platform for early warning alerts.

1.7.8 The Merits of A High Resolution Analysis

The main aim of a high resolution analysis of spatio-temporal variations of geomagnetic field is to simultaneously reduce time scale from order of diurnal and hourly to minute, and as well enlarge scope of space from order of regional to global. The microscopic view or detailed Sq variations will enhance monitoring of ionospheric current effects on geophysical phenomena such as earthquakes, storms, tremors, equatorial anomalies, typhoon among others. Furthermore, a high spatio-temporal resolution analysis of geomagnetic field variations will:

- (i) Facilitate a near real-time monitoring of Sq variations.
- (ii) Create a robust early warning system against natural disasters.
- (iii) Improve our ability to predict some undesirable space weather related phenomena.
- (iv) Enhance studies of atmospheric electricity.

1.8 Overview of Thesis

This study is reported in five chapters beginning with the introductory chapter that expressed the background to study, statement of problem and aim and objectives. The concepts of geomagnetism, solar quiet variations and parallel computing were reviewed in chapter two. The methodology of the study which involves acquisition and processing of geomagnetic field, development and implementation of parallel algorithm, configuration of parallel architecture were covered in Chapter three. High resolution analyses of SqH, SqD and SqZ in 2D and 3D display were reported and discussed in chapter four. Chapter five contains summary of findings, conclusion and recommendations.

1.9 Operational Definition of Terms

Computational platform-	Sequential / parallel algorithms and heterogeneous multicore systems
Core -	(i) Processor or Central Processing Unit (ii) Inside Earth
Coronal mass ejection -	a massive outburst of solar wind from solar corona of the sun
Geotechnical activities -	Geophysical, Equatorial anomaly, ULF waves, TID, geomagnetic storm
GPU	Graphics Processing Unit
High-resolution	Short time scale (one-minute), global spatial distribution

HPC	High Performance Computing
Ionosphere -	ionized layer in upper atmosphere
Latitudinal zones -	Low, Mid and High latitudes
Magnetosphere -	region of intense solar wind and geomagnetic field interaction
Minute -	60-second time
Node	A networked computer with single or many cores
Serial algorithm -	Sequential algorithm
Spatio-temporal -	Concerned with space and time

CHAPTER TWO

2.0

LITERATURE REVIEW

2.1 The Sun Cycle and Solar Variations

The number of sunspots visible on the Sun is not constant, but varies over an 11-year cycle. At a typical solar minimum, few sunspots are visible, and occasionally none at all can be seen such as the case may be at high solar latitudes. As the sunspot cycle progresses, the number of sunspots increases and they move closer to the equator of the Sun as illustrated by Spörer's law, (Maed, 1964; Hopkins, 1980) and the sunspots usually exist as pairs with opposite magnetic polarity. The magnetic polarity of the leading sunspot alternates every solar cycle, so that it will be a north magnetic pole in one solar cycle and a south magnetic pole in the next.

The solar cycle has a great influence on space weather and consequently on the Earth's climate due to luminosity that has a direct relationship with magnetic activity (Maugh, 1982). Solar activity minima tends to be correlated with colder temperatures and maxima solar cycles tend to be correlated with hotter temperatures, thus the sun modifies the earth's atmosphere. The physics of geomagnetic phenomena is therefore attributed to the sun. Solar terrestrial interactions have to do with the interaction of the solar radiation with the solar terrestrial system. Solar-terrestrial activity refers to the changes of energetic particles and electromagnetic fields that originate from the sun, propagates to the earth's magnetosphere and ionosphere causing drastic effect on the earth's atmosphere and geomagnetic field (Yacob & Rao, 1996).

Solar variation is the change in the amount of radiation emitted by the Sun and in its spectral distribution that spans years to millennia. These variations which occur even in milliseconds time scale, also have periodic time frame, the principal period is approximately 11-year solar cycle (or sunspot cycle). It therefore follows that the pattern of ionospheric current intensity is largely related to geomagnetic field variations which can be monitored and predicted with respect to solar cycle, as it is done globally today for meteorological weather forecast.

2.2 Solar Quiet (Sq) Daily Variations

The geomagnetic field has a regular variation with a fundamental period of 24 hours. This variation is dependent on local time, latitude, season and solar cycle. It is often caused by electrical currents in the ionosphere layer of upper atmosphere. At these altitudes the ionosphere is significantly ionized by solar radiation and the ions are moved by winds and tides arising from the heating effects of the Sun and the gravitational pull of the Sun and the Moon. This creates the required condition for a dynamo to operate (i.e. motion of a conductor in a geomagnetic field), and form ionospheric current. The magnetic effect of this current system is observed on the ground at observatories as solar daily variation during magnetospheric quiet condition (Nosé et al., 2001). Conventionally, the Sq algorithm involves removal of secular variation and correction of non-cyclic variation using the computed baseline and mid-night departure values. This is also a necessary step to filter out inherent errors in the data due to high noise level resulting from artifacts, temperature drift as well as magnetic field from other sources (Price, 1963). Most previous and existing studies concentrated on analyses of solar quiet daily variations based on individual geomagnetic field component at different regions using large time frame such as seasonal, annual or hourly magnetic dataset and reported accordingly. Such results are often represented on 2D plot visualization.

In the available literature, Onwumechili & Alexander (1959) and Oko et al. (1996) among others studied Sq variations along the low and middle latitudes. Similarly, Masahiko (1999), Sillanpa et al. (2004) and Love & Gannon (2010) carried out investigations on Sq variations along the low and high latitudes. In addition, Xu (1989), Barbatsi (1990), Cueto et al. (2003) as well as Stening et al. (2007) studied Sq variations in the American sector, Indian sector, Australia sector, Asian sector, African sector and also along the equatorial region (Onwumechili, 1967). Further literature review showed that Kane (1971b) and Olsen (2007) extended study area on more than one region but with relatively fewer observatory stations. On the other hand, Brown & Williams (1969), Schlapp (1976) and Alex et al. (1992) as well as McDonald (2007) observed Sq variations using diurnal geomagnetic field dataset. Furthermore, Okeke et al. (1998) and Love & Gannon (2010) used hourly geomagnetic field dataset to study Sq variations along the low latitude. In general, the authors observed consistent variability of Sq with respect to local time of the study regions. The precision of phenomena under investigation was limited by spatio-temporal resolution applied. However, effective space weather

programme relies on processing of near real-time geomagnetic data to monitor and predict the earth's response to changes in solar activity and conditions in the near earth space environment (David, 2001).

2.3 Variations of Geomagnetic Field

The strength and the direction of the Earth's magnetic field varies in time and space owing to changes of the electrical currents in the inner Earth as well as in the outer space. Aravindan and Iyer (1990) inferred that currents flowing in the ionosphere and magnetosphere normally induce magnetic field variations on the ground. These field variations generally fall into the categories of "quiet" and "disturbed". The solar quiet daily variation (Sq) results principally from currents flowing in the electrically-conducting E-layer of the ionosphere. Malin & Gupta (1977) and Briggs (1984) indicated that the increased particle flux from the sun may lead to large variations in the Earth's magnetic field. Shorter term variations in the external part of the geomagnetic field induce secondary variations in Earth's crust and mantle which are used to study the electrical properties of the Earth, giving insight into porosity, temperature, and composition in these regions.

According to circuital law formulated by Andre-Marie Ampere in 1826, a current will induce a magnetic field and conversely a time-changing magnetic field will induce a current to flow in a conductor (Heinz, 2000; Daniel, 2003). Currents flowing in the ionosphere induce a magnetic field variation observable on the ground. This changing magnetic flux links the conducting earth thereby causing currents to flow on the ground. These currents, in turn, induce a changing magnetic field on the ground which could be measured by ground magnetometers. These induced earth currents contribute about 25-30% of the total measured Sq field in H, D and Z components as Sq(H), Sq(D) and Sq(Z) respectively (Afanasyeva, 1961; Butcher, 1982; Campbell, 1982).

Olson (1989) work proved that the understanding of temporal geomagnetic variations was important due to its usefulness and application in producing magnetic maps of different times and for the navigation by compass. Campbell (1989) and Rabiou (2000) illustrated the geomagnetic variations as shown in Figure 1.4. Lunar daily variation is a type of quiet current system, L, which exists due to lunar tidal winds in the ionospheric E-region. These are gravitational tides, as opposed to solar-driven (i.e. thermally-driven)

atmospheric tidal oscillations. The L variation is about 10-15% of the Sq variation. In addition to Sq and L variations, the geomagnetic field often undergoes irregular or disturbance variations connected with solar disturbances. Severe magnetic disturbances are called magnetic storms. Such storms often begin with a sudden storm commencement (SSC), after which a repeatable pattern of behavior ensues (Anandarao & Raghavarao, 1987).

The currents in the core of the Earth that create its magnetic field started up at least 3,450 million years ago. The intensity and structure of the Earth's magnetic field are always changing, slowly but erratically, reflecting the influence of the flow of thermal currents within the iron core. This variation is reflected in part by the wandering of the North and South Geomagnetic Poles (Ashour & Price, 1965). Secular variations are slow and steady variations of geomagnetic field (Egedal, 1948; Last, 1976; Butcher & Schlapp, 1992).

Schlapp & Mann (1983) and Thakur & Rao (2003) evaluated the ranges of variations of the geomagnetic field vertical component, Z. The authors used the hourly values of the horizontal component, H, on the five international quiet days in the month of October 1975 for the stations Trivandrum, Kodaikanal, Annamalainagar, Hyderabad, Alibag, Ujjain, and Jaipur in determining the latitudinal variation of Sq(H). Other studies carried out used hourly magnetic data to compute Sq in either H or Z components of certain sectors (Osborne, 1964; Dunford, 1970; Emilia & Last, 1977; Butcher & Brown, 1981; Rajaram, 1983; Okeke, 1996). The studies by Brown & William (1969), Brown (1975) and Rastogi (1996) described the daily variation of H at equatorial and tropical latitude stations during different levels of solar activity especially about the asymmetries in the daily variations and the continuation of ionospheric currents during the night time hours.

Chapman (1951), Hines (1959), Campbell & Schiffmacher (1988) and Schlapp et al. (1998) investigated the traveling ionospheric disturbances (TIDs) and the ionospheric motions as the manifestation of Atmospheric Gravity Waves (AGWs) in the ionosphere. The gravity waves have an influence on the plasma distribution (especially in the F layer). Hargreaves (1979, 1989) and Harvey (2007) among others established that other sources of geomagnetic field are electrical currents flowing in the ionosphere and the

magnetosphere. These are related to the coupling between the primary geomagnetic field and the particle flux from the sun.

2.4 Evolution of Geomagnetic Field Observatories

Jankowski & Sucksdorff (1996), Oslen (1996) and Rieger & Leitinger (2002) defined geomagnetic ground-based observatory as a location on the earth surface where absolute vector observations of the Earth's magnetic field are being recorded regularly at a particular sampling rate or time resolution. The site of the observatory must be magnetically clean so as to prevent interference of external localized magnetic field. The larger the metallic object, the more it distorts the magnetic field. The earliest magnetic observatories where continuous vector observations were made began operation in the 17th century. For instance Kew observatory is one of the oldest in history. Kew Observatory was an old astronomical and terrestrial magnetic observatory founded by King George III (1760-1820), located within the Old Deer Park of the former Richmond Palace in Richmond, Surrey, now within Greater London (Osborne, 1968; Schlapp, 1968).

The two main categories of instruments installed in a typical observatory include variometers and magnetometers. Variometers make continuous measurements of elements of the geomagnetic field vector but in arbitrary units, for example millimeters of photographic paper in the case of photographic systems or electrical voltage in the case of fluxgates. A fluxgate sensor comprises a core of easily saturable material with high permeability (Carroll & Ostlie, 1995). Meanwhile, magnetometers are electric devices that senses slight changes in magnetic fields and is used to measure the strength and direction of magnetic field. Magnetometers detect slight deviations in the Earth's magnetic field due to the flowing currents in the ionosphere that induce magnetic perturbations on the ground. An important use of magnetometers is in measuring the earth's magnetic field. Ground-based magnetometers record variations of geomagnetic fields. Magnetometers are also used by treasure hunters to detect metallic objects underground or underwater (Campbell, 1973). By detecting irregularities in the earth's magnetic field, a magnetometer can indicate the location of deposits of magnetic ore, such as iron ore, or of geological formations associated with petroleum. Other possible sources of geomagnetic observations are satellites and ships (Campbell & Schiffmacher,

1987) and (Hibberd & Davidson, 1988). Earth is an extremely large magnet, which accounts for compasses pointing towards north. The invisible magnetic field that surrounds the planet forms different layers, called lines of flux. These lines vibrate at different frequencies and amplitudes depending on their location around the world.

Some notable and world known federations of institutions of geomagnetic observatories include International Real-time Magnetic Observatory Network (INTERMAGNET), MAGDAS, AMBER and Institution of Earth Physics, Paris (IPGP). The INTERMAGNET programme establishes a global network of cooperating digital magnetic observatories, adopting modern standard specifications for measuring and recording equipment, in order to facilitate data exchanges and the production of geomagnetic products in close to real time.

2.5 Geo-statistical and Geo-Spatial Data Analysis

Geostatistics is a branch of statistics that deals with spatio-temporal datasets for prediction of spatial distributions of random phenomena where it is impractical to obtain data values at grid points. Cressie (1990) referred geostatistics as an extension of interpolation methods that observes a phenomenon at unknown locations as a set of correlated random variables. Geostatistics is efficient at working on regionalized variables, such variables fall between random and completely deterministic values. Regionalized variables describe phenomena with geographical distribution such as geomagnetic field, elevation of ground surface among others. Such phenomena exhibit spatial continuity and variability. Obviously, it is not always possible to sample random variables at every location, therefore, unknown values must be estimated from data taken at specific locations that can be sampled. The size, shape, orientation, and spatial arrangement of the sample locations are useful in predicting the unknown samples. A number of simpler interpolation algorithms such as inverse distance weighting, cubic-splines, Lagrange polynomial, triangulation and nearest-neighbor were already well known before geostatistics.

Spatial data is a type of data attributed to a point line area or region defined by some geometrical system. Geographical data is spatial data that also relates to some period or periods in time. The spatial reference relates to some three-dimensional region on or near the surface of the Earth that is abstracted in a simplified form. Ale (2005) used Figure

2.1 to depict the ideal model of geostatistical analysis and framework of regionalized problems in 3-dimensional perspective.

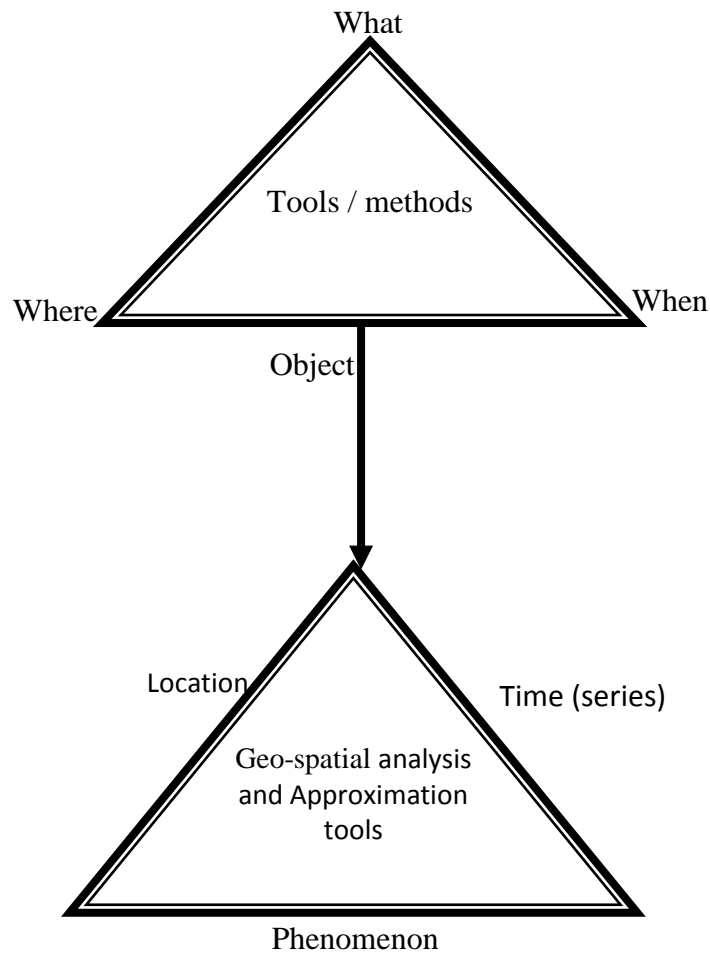


Fig. 2.1: Abstraction of Geospatial Analysis for Regionalized Variables

2.6 Numerical Interpolation Methods for Regionalized Variables

In scientific and engineering applications, primary data collected from the field are usually discrete and the physical meanings of the data are not always well known. An analytical unbiased function that fits the field is therefore necessary in order to estimate the outcomes and as well interpret the physical phenomenon. The process of finding the coefficients for the fitting function is popularly known as curve fitting while the process of estimating the outcomes in between sampled data points is known as interpolation. Similarly, the process of estimating the outcomes beyond the range covered by the existing data is called extrapolation (Cressie, 1990).

Extrapolation and Interpolation are important concepts in numerical analysis for approximation of functions or given system. The interpolation method chosen in contour mapping can affect the accuracy of the resulting surface considering the spread of original data. Interpolation methods are programmed to make different assumptions based on surrounding information. For this reason, some datasets or mapping objectives lend themselves better to different interpolation methods. Therefore, the goal of choosing the right interpolation method is to generate surfaces to a degree of accuracy for representation, analysis or assessments.

Approximation of results is a means of finding an analytic function, $f(x)$, that passes through a given N points. Sometimes functions may not be available explicitly but only the values of the function at a set of points. The values for $f(x_i)$ may be the results from a physical measurement such as geomagnetic field data worldwide. It may also be from some long numerical computation which cannot be put into a simple equation. Some common approximation methods include polynomial, triangulation, kriging, cubic splines among others.

2.6.1 Polynomial Extrapolation Method

A polynomial curve can be created through the entire known data or just near the end. The resulting curve can then be extended beyond the end of the known data. Polynomial extrapolation is typically done by means of Lagrange interpolation or using Newton's method of finite differences to create a Newton series that fits the data. The resulting polynomial may be used to extrapolate the data. High-order polynomial extrapolation must be used with due care because anything above order one (linear extrapolation) will possibly yield unusable values. Usually, an error estimate of the extrapolated value will grow with the degree of the polynomial extrapolation which is known as Runge's phenomenon. However, a straightforward approach is the use of Lagrange polynomials. The Lagrange Polynomial may be used where the data set is unevenly spaced.

Basically two criteria employed for expression of polynomials as follows:

- (i) Least squares fit criterion:
The variants of least squares fit criterion include linear least squares and non-linear squares (such as exponential, powers, logarithmic, Lagrange polynomial, divided difference)
- (ii) Exact fit criterion:

Other approximations by interpolation or extrapolation algorithms include Splines, Cubit and Piecewise Hermite.

Computer software can model terrain surfaces for contour mapping by using interpolation to estimate and derive information between data points provided by a survey.

2.6.2 Triangulation Method

Kriging and triangulation are two interpolation methods in mapping simulation software, with various merits and demerits for different mapping situations. Interpolation by triangulation is a very flexible method that can create a surface quickly from different types of data sources. Rather than using a grid, the triangulation method defines a surface from irregularly spaced points. This characteristic often results in more accurately defined features. Since triangulation does not generate average information during interpolation, it is better equipped to deal with data sets with abrupt peaks or changes. The triangulation method can be computed quickly because it uses fewer points than kriging, but it will not create a smooth surface. If appearance is a main mapping objective, triangulation might not be the best option.

2.6.3 Cubic Spline Interpolation Method

In numerical analysis, spline interpolation adopts a piecewise polynomial. Spline interpolation is preferred to polynomial interpolation because the interpolation error can be made small even when using low degree polynomials for the spline. Therefore, spline interpolation minimizes the problem of Runge's phenomenon which occurs when interpolating between equidistant points with high degree polynomials. The disadvantage of cubic splines is that they could oscillate in the neighborhood of an outlier. This characteristic makes cubic spline method inappropriate for interpolating geomagnetic field data and parallel processors runtimes.

The cubic spline interpolation is capable of using third degree polynomials to connect the data points which often results in strikingly smooth curve fits. The method of least squares is most commonly used to calculate the coefficients of the interpolating polynomial.

2.6.4 Inverse Distance Weighting (IDW) Method

IDW weighs the value of each point by its distance to the cell being analyzed and averages the values. The weight (influence) of a sampled data value is inversely proportional to its distance from the estimated value. It assumes that unknown value is influenced more by nearby than far away points, but we can control how rapid that decay is. Influence diminishes with distance. Unfortunately, IDW has no method of testing for the quality of predictions, so validity testing requires taking additional observations. IDW is also sensitive to sampling, with circular patterns often around solitary data points (Faith & Sheshinski, 1979). Therefore, IDW is inefficient for interpolating geomagnetic field data at unknown grid points.

2.6.5 Kriging Interpolation Method

Krige (1951) and Matheron (1963) developed kriging geostatistical interpolation method for estimation of regionalized random variables at unobserved locations using observations at nearby locations. Kriging method corresponds to collection of generalized regression techniques to minimize variable estimation defined from a prior model for covariance. This involves solving large set of computationally-intensive equations. Golden Surfer application and MATLAB Mathworks high level technical programming language are two efficient software tools for multi-dimensional visualization and contour mapping of geophysical simulations. Although, Surfer is more effective than MATLAB at data gridding and contour mapping due to its in-built kriging interpolation method, however it does not have compiler and programming interface for simulating dynamic system such as solar quiet daily variation of geomagnetic field.

The kriging interpolation method uses grid of equally spaced data points to discretize surfaces. This method uses weighted moving averaging, in assuming that one pattern of variation can be applied across the surface it defines from the set of data. The goal of kriging is not to directly mimic reality, but to follow the general trends shown by data points. Although kriging method was originally developed specifically for ore reserve estimation, it has been used for other spatial estimation applications, such as analyzes and modeling of air quality data (Faith & Sheshinski, 1979; Grivet, 1980). In its simplest form, kriging can be thought of as a way to interpolate spatial data much as an automatic contouring program would. In a more precise manner, kriging method can be defined as

the best linear unbiased estimator of a spatial variable in a particular domain. Kriging assigns low weights to distant samples and vice versa, but also takes into account the relative position of the samples to each other and the site or area being estimated.

Kriging algorithm can be used to derive an empirical model for the stochastic component of the geomagnetic field observed data. This model allows estimating the value of variable points of the observations. Kriging method assumes that the trend is a linear combination of known functions of the location and unknown parameters which can be formally written as the explored data was collected by a ground-based survey magnetometer. There are three basic types of kriging considered in literature. The simple kriging method assumes that the trend is known. The ordinary kriging method assumes that the trend is constant but unknown. The most general method is the universal kriging. Kriging method uses variogram to express spatial variation and it minimizes the error of predicted values which are estimated by spatial distribution of the predicted values. Oliver & Webster (1990) ascertained that the use of kriging method for interpolation ensured the estimation of the confidence in any interpolated value in a way better than the earlier methods would do. Kriging is an alternative method for interpolating a regionalized dataset that does not require an exact fit at each tabulated data point. It applies a weight to each of the tabulated data points based on spatial variance and trends among the points. Weights are computed by combining calculations of spatial continuity and anisotropy within either an exponential or spherical semivariogram model.

Kriging as a stochastic model provides estimates of accuracy and certainty in predictions. The kriging is an application of least squares method (weighted average) that uses trends in the map to extrapolate into areas of no data, sometimes resulting in minimum and maximum Z values in the grid that are beyond the values in the data file. In principle, the simple kriging algorithm estimates the unknown value using weighted linear combinations of the available regionalized data with their covariance function $c(x,y)$:

Kriging algorithm also can be described clearly by a set of equations for computation of the unobserved estimated values. The set of such equations can be represented by the system of linear kriging equations as described by Journel & Huijbregts (1978), Hohn (1988), Liebhold et al. (1993) and Jensen et al. (1997). Ideally, the main aim of kriging algorithm is to compute the optimal value of weighting coefficient so as to obtain a

minimal estimation error. Other variants of kriging algorithm address the more optimal weighting coefficient by adding Lagrange multiplier and also make the total weighting coefficient to be equal one so as to generate unbiased estimations.

2.6.6 Comparison of Kriging and Triangulation Methods

Kriging is more efficient at handling regionalized random variables like geomagnetic field dataset. It has several advantages over triangulation and other interpolation methods due to its better accuracy and unbiased of the resultant surface. Estimates in kriging have a known minimum degree of variation, so a calculation of confidence across surfaces can be determined. Further, the method is beneficial to data analysis by showing gaps in information. The resultant surface from kriging is averaged, so while it is generally accurate, it does not necessarily give credence to original data points. Triangulation surfaces accurately follow original data, but are less capable of extrapolating information or trends beyond an original data set.

However, one of the major drawbacks of kriging method is that it is more computationally-intensive due to constant variance or errors estimation correction at every step and computation of the covariances as well. Thus, most existing Sq serial algorithms on serial processors have not sufficiently explored kriging method using high resolution time scale dataset of geomagnetic field.

2.7 The Concept of Advanced Computations

There is an ever increasing demand for computational power to improve the speed and accuracy of solutions to real world problems through faster computations and simulations techniques. Advanced computations in science and engineering facilitates analysis and discovery that are difficult to achieve by theory and experiment (Mark & Michael, 1997).

The advanced computation system involves the use of well-configured massively distributed and parallel computers to provide increased computing capability. Essentially, high processing speed for complex and large problems in wide range of engineering and scientific simulations cannot be over-emphasized (Erich et. al. 2005). The main issue in parallel computing is to build efficient parallel system that is capable

of exploiting greater or smaller degrees of parallelism in order to produce optimized result in the fastest possible time. According to Intel co-founder Gordon Moore in 1965, he observed that the number of transistors per square inch on integrated circuits had doubled every year since their invention. Therefore, Moore's law predicted that this trend would continue into the foreseeable future. Although the pace has slowed, the number of transistors per square inch has since doubled approximately every 18 months. However, the experts believed that the Moore's law of exponential growth was unlikely to continue indefinitely (Duncan, 1990). The experts also envisaged that Moore's law might hold for another two decades while some studies showed that physical limitations would be attained by 2017 (Ethan & Randy, 1997). Thus, the most possible and alternative means of improvement in computing is on hardware and software parallelism. The concept of parallelism through multiple processors has constituted the major technique of high-performance computing over some past decades, however interest in it has grown lately due to the physical constraints preventing frequency scaling. Meanwhile, Amdahl's law also suggested efficient parallelism of programs so as to take the best advantage from multiples CPUs / cores (Amdahl, 1967).

According to Almasi & Gottlieb (1989) and Ananth et al. (2005), parallel computing is an advanced technique of computation in which large and/or complex problems broken into smaller tasks are solved concurrently. Different kinds of parallel computing include bit-level, instruction level, data and task parallelism. Some other popular variants of parallel computing include distributed computing, cloud computing, grid computing and high-performance computing.

A well designed (or configured) parallel computers, parallel algorithms and programming, as well as performance evaluation would ensure effective advanced computations. Parallel computing should be scalable and robust in terms of the number of processors, fault tolerance, redundant supporting high-speed communication interfaces and inter-processor data sharing (Gilan, 1994; George et al., 2009). James & Alan (2003) and Antonio et al. (2008) agreed that the resources of parallel computing would not be used effectively unless there were available parallel algorithms, languages, and methods to evaluate the performance. Parallel programming methods and libraries are also very much important in helping parallel program development by shielding users from low-level machine characteristics. Parallel applications are the final target of

parallel computing. Therefore efficiency and portability of a parallel application are the most to date research topics of parallel computing society.

Parallel computers are also classified based on granularity, that is, the ratio of time required for a basic communication operation to the time required for a basic computation. The granularity of a parallel computer is defined by the number of processors and the speed of each individual processor in the system. Parallel computers with small granularity are suitable for algorithms requiring frequent communication and the ones with large granularity are suitable for algorithms that do not require frequent communication. Computers with large number of less powerful processors would have small granularity and are called fine-grain computers or Massively Parallel Processors (MPP). In contrast computers with small number of very powerful processors have large granularity and are called coarse-grain computers or multi-computers. Serial computers are not capable of implementing parallel algorithms, and therefore cannot solve many tasks of the same problem simultaneously.

2.8 Architecture and Design of Parallel Computing

Parallel computing architectures include multi-core processors, many core-GPU, clusters and cloud processors. Multi-core and multi-processor computers comprise of multiple processing elements within a single machine, while clusters, MPPs, and grids use multiple computers to work on the same task (Patterson & John, 1998). In 1966, Michael J. Flynn further classified computer architecture into the following four categories:

- i. Single Instruction, Single Data stream (SISD)
- ii. Single Instruction, Multiple Data streams (SIMD)
- iii. Multiple Instruction, Single Data streams (MISD)
- iv. Multiple Instruction, Multiple Data streams (MIMD)

The classification which was later tagged Flynn's taxonomy was based upon the number of concurrent instructions (or control) and data streams available in the architecture. Flynn (1972) explained the configurations of SISD, SIMD, MISD and MIMD as shown in Figure 2.2.

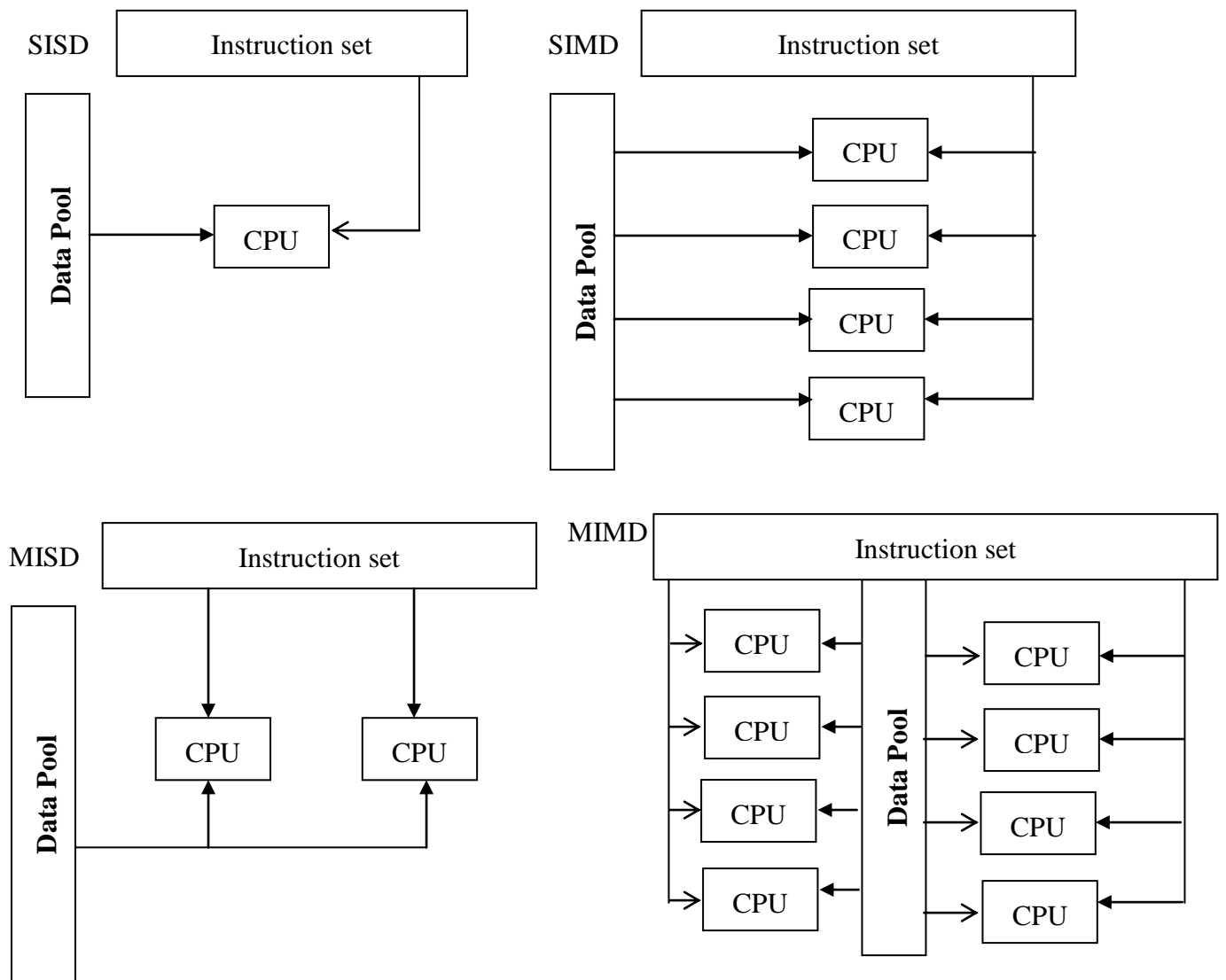


Fig.2.2: Flynn's Taxonomy – Classification of computer architectures

2.8.1 Single Instruction, Single Data stream (SISD)

SISD is a sequential computer that exploits no parallelism in either the instruction or data streams. In this architecture, a single CPU fetches single Instruction Stream from memory. The CPU then generates appropriate control signals to direct single processing element to operate on single Data Stream, that is, one operation at a time. Examples of SISD architecture are the traditional uni-processor machines such as a stand-alone PC. The speed of SISD computer is limited by the execution rate of instructions and the speed at which information is exchanged between the memory and CPU.

2.8.2 Single Instruction, Multiple Data streams (SIMD)

SIMD was defined as a computer which exploits multiple data streams against a single instruction stream to perform operations which may be naturally parallelized such as an array of processors or CPUs. A SIMD architecture system has a single control unit that dispatches instructions to each processing element in the system. For certain classes of problems called data-parallel problems, this architecture is perfectly suited to achieving very high processing rates, as the data can be split into many different independent pieces, and the multiple instruction units can all operate on them at the same time. SIMD architecture is synchronous, deterministic and very well-suited to instruction and operation level parallelism.

2.8.3 Multiple Instructions, Single Data stream (MISD)

MISD computer architecture involves multiple instructions operate on a single data stream. This architecture is quite rare but very useful for fault tolerance. Heterogeneous systems operate on the same data stream and must agree on the result. Such architecture was used in the design of Space Shuttle flight control computers.

2.8.4 Multiple Instructions Multiple Data streams (MIMD)

This class of computers involves multiple autonomous processors simultaneously executing different instructions on different data. Distributed systems are generally recognized to be MIMD architectures which either implements a single shared memory space or a distributed memory space. A MIMD architecture system has each CPU in each processing element such that each processor is capable of executing a different program independent of the others in the same system.

According to Patterson & John (1998), MIMD architecture can be classified into shared memory and distributed memory architectures. In shared memory architecture, the same memory is accessible to multiple processors and synchronization is achieved by controlling tasks, reading from and writing to the shared memory. The advantage of this type of architecture is the ease of programming but there are many disadvantages to pay off for this. Since multiple processors can access to the same memory, the scalability of the system is limited by number of access provided with a local memory and the global memory is shared among all the processors in the system. Depending on the speed of

memory accessing, shared memory computers can be classified into the Uniform Memory Access (UMA) or Non-Uniform Memory Access (NUMA) computers.

The SIMD and MIMD computers belong to parallel computer category. In a distributed memory parallel computer each processor element has its own local memory and synchronization is achieved by the communication between processors via inter-connection network. The major concern in distributed memory architecture is the data decomposition, that is, how to distribute data among processors (local memories) to minimize inter-processor communications. Distributed memory systems operate message-passing and virtual-shared or distributed-shared memory models.

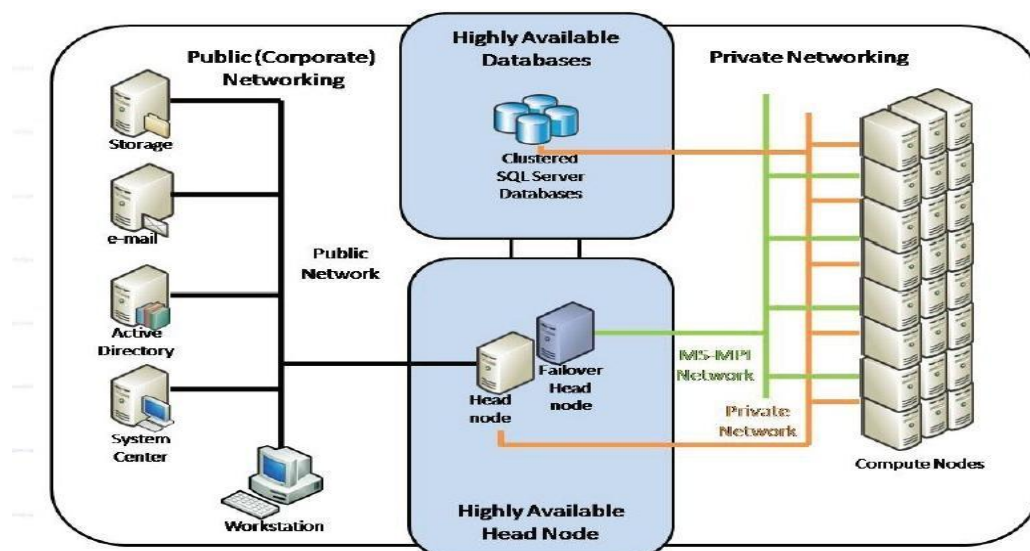
The other class of MIMD computers that is based on both shared- and distributed-memory architectures is the Symmetric Multi-Processor (SMP) clusters. A typical SMP system is a cluster of large groups of nodes as a distributed-memory computer where each node behaves as a processor. Each node, in turn, has small number of processors (typical 4 or 8 processors) with shared memory. In practice, SMP clusters should be treated as distributed-memory computers. Programming in SMP clusters is very similar to programming in distributed-memory computers where each processor acts as vector processors. Both the shared- and distributed-memory computers are constructed by connecting processors and memory units using variety of interconnection networks. In shared-memory systems, interconnection network is for the processors to communicate with the global memory. The most applicable programming model for a parallel architecture is the Multiple Instruction Stream, Multiple Data Stream (MIMD) (Patterson & John, 1998).

2.9 High-Performance Computing

Lars et al. (1999) and Olaf et al. (2001) described high-performance computing (HPC) as a new branch of computer science that deals with development of supercomputers, parallel processing algorithms and software. HPC also refers to the use of parallel supercomputers and computer clusters including software and hardware components to speed up and optimize computations results. The HPC architecture relies upon hardware level parallelism to improve performance with the aid of multiple execution units, pipelined instructions and multi-core systems. Cluster architectures are designed to enable high performance parallel computing which often consist of one or more master

nodes and compute nodes interconnected by a private network system. Most High Performance Computing Cluster stacks are built around standards-based commodity components. The master node is the architecture gateway to external resources that supports the Network File System. George et al. (2009) reviewed some benefits of clusters which include scalability, vendor independence, reliability, cross-platform and availability. In many computational fields, the problem workload is so large that it simply could not be processed on a single processor-based system due to time constraints. Clusters also provide an easier path for increasing the computational resources as the workload increases over time.

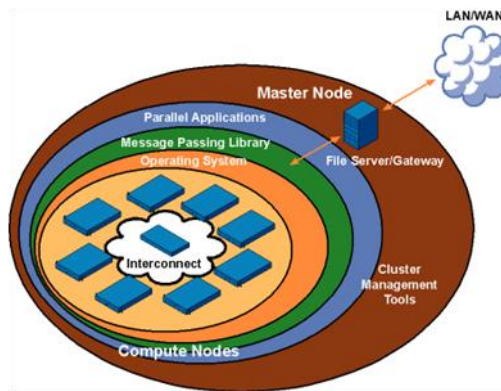
The compute nodes are the cluster work engines that execute parallel jobs. Access and management of compute nodes are provided through remote interfaces such as network connections to the master node. Compute nodes commonly use private IP addresses for accessing other computing resources within the local area network (Bahman et al., 2008). The use of high-density rack mounted servers is the most popular configuration for HPC cluster environment. Besides the compute nodes, each rack could be equipped with network switches, powervault (high capacity storage array) and power distribution unit (PDU). For some types of applications where communication bandwidth between nodes is critical, low-latency and/or high-bandwidth interconnects such as Gigabit Ethernet (Hipper & Tavangarian, 1998), Myrinet and Infiniband are common choices for interconnecting among compute nodes (Mitsuhsa et al., 1998; Barney, 2011). Figure 2.3 shows a typical architecture of a high-performance computing cluster.



Source: https://computing.llnl.gov/tutorials/parallel_comp/

Fig.2.3: Standard computing cluster architecture

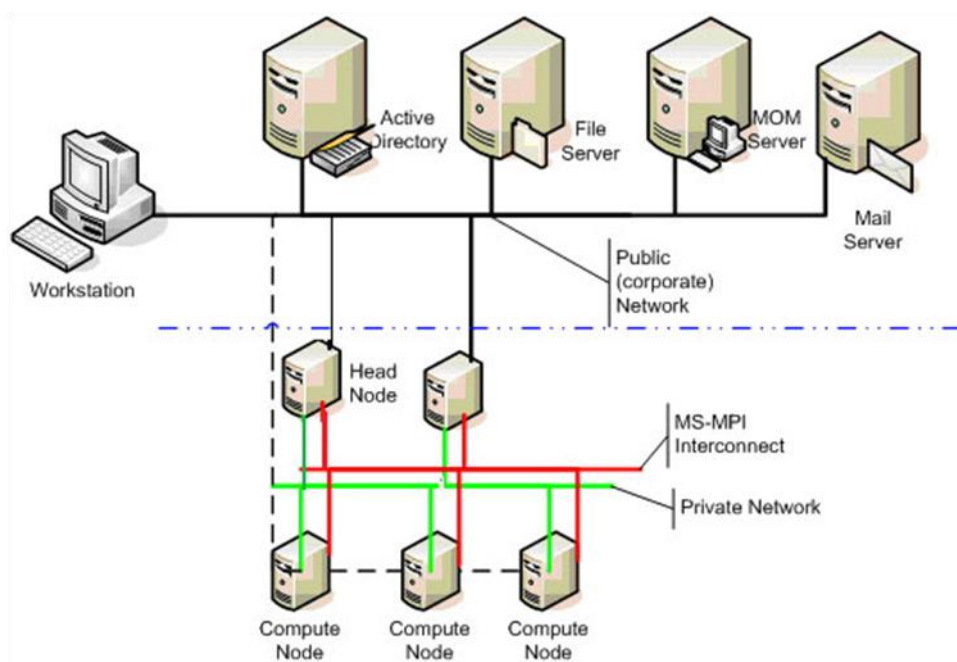
Figures 2.4 and 2.5 show the hierarchical structure and layout of massively parallel cluster architecture. The job scheduler submits users' job obtained through the workstations to the head nodes which thereafter break the job into tasks that will be processed by the nodes in parallel (Patterson & John, 1998).



Source: https://computing.llnl.gov/tutorials/parallel_comp/

Fig.2.4: Interconnection of cluster components

The message passing interface is a library protocol that enables interconnection and exchange of message among the nodes (Melissa & Gregory, 2005; Plaza, 2008).



Source: https://computing.llnl.gov/tutorials/parallel_comp/

Fig.2.5: Cluster Topology

2.9.1 Interconnect Hardware

High Performance Computing Cluster stacks are built around standards-based commodity components in terms of hardware, interconnects and software. Some of the sub-components are listed below:

- (i) Network infrastructure and services to provide a solid backbone
- (ii) Remote management specifications
- (iii) Network, power, floor space design and layout
- (iv) Cluster management selection
- (v) Storage Area Network / Network Attach Storage
- (vi) Stateless or Diskless clusters
- (vii) NIS, DNS, NTP, AD
- (viii) Fast Ethernet / fibre optics and cluster monitoring

The compute and head (master) nodes are often interconnected through a single local area network (LAN), and so the network treats each node as a separate one. Message (data) can be passed throughout these networks with very little lag, ensuring that data does not bottleneck between nodes. The head node controls and monitors all access to the cluster resources and is the single point of management, deployment and job scheduling for the compute cluster. It utilizes the existing corporate Active Directory infrastructure for security and account management.

For some types of applications especially where communication bandwidth between nodes is critical, low-latency and high-bandwidth interconnects such as Gigabit Ethernet, Myrinet, Infiniband or equivalent are common choices for interconnecting compute nodes (Almasi & Gottlieb, 1989). As demonstrated in Figure 2.5, the design and configuration of HPCC involve master nodes, compute nodes, farm servers, LAN / WAN and workstations. Farm servers comprise of Active Directory, File server, mail server and print server.

2.9.2 Multi-core Technology

Darema et al., (1988) and Bossomaier & Green (2000) and defined a multi-core system as a processing system which involve integration of two or more independent cores (or CPUs). The cores are typically assembled on a single integrated circuit die (chip multiprocessor), or they may be integrated onto multiple dies in a single chip package. While a multi-core processor implements multiprocessing on a single physical package,

a many-core processor is one in which the number of cores is large enough that traditional multi-processor techniques are no longer efficient. The increase in computational power directly relates to the increase in number of connected processors and the interconnection networks.

In contrast to multi-core systems, the term multi-CPU refers to multiple physically distinct processing units (which often contain special circuitry to facilitate communication between each other). Hack (1989) opined that many-core and massively multi-core terms are sometimes used to describe multi-core architectures with an especially high number of cores. The motivation for multi-core processors comes from greatly diminished gains in processor performance from increasing the operating frequency (frequency scaling). The instruction level parallelism (ILP) and memory wall are major causes of speed limitation for system performance despite continued processor frequency increases. The memory wall refers to the increasing gap between processor and memory speeds, which pushes cache sizes larger to mask the latency to memory. This helps only to the extent that memory bandwidth is not the bottleneck in performance. The ILP wall refers to increasing difficulty to find enough parallelism in the instructions stream of a single process to keep higher performance processor cores busy.

The proximity in the circuitry of multiple CPU cores on the same die allows the cache coherency circuitry to operate at a much higher clock rate than is possible if the signals were to travel off-chip. Combining equivalent CPUs on a single die significantly improves the performance of cache snoop operations since the signals between different CPUs travel shorter distances, and therefore those signals degrade less. This design will be appreciated when running CPU-intensive processes such as weather modeling, computational fluid dynamics among others. The result of a study carried out by Hummel (1999) inferred that the existing serial software/codes must be re-compiled in parallel so as to maximize utilization of the computing resources provided by multi-core processors. Conversely, the efficiency of multi-core processors to increase application performance depends largely on the use of multiple threads within applications.

An instance of a dual-core processor contains two cores while a quad-core processor contains four cores. Cores in a multi-core system may be coupled together tightly or

loosely. For example, cores may or may not share caches, and they may implement message passing or shared memory inter-core communication methods. Common network topologies to interconnect cores include bus, ring, 2-dimensional mesh, and crossbar. All cores are identical in homogeneous multi-core systems and they are not identical in heterogeneous multi-core systems.

The general trend in processor development has been from multi-core to many-core: from dual-, tri-, quad-, octo-core chips to ones with tens or even hundreds of cores. In addition, multi-core chips mixed with simultaneous multithreading, memory-on-chip, and special-purpose "heterogeneous" cores promise further performance and efficiency gains, especially in processing multimedia, recognition and networking applications. Software benefits from multi-core architectures where code can be executed in parallel. Under most common operating systems this requires code to execute in separate threads or processes. Each application running on a system runs in its own process so multiple applications will benefit from multicore architectures. Each application may also have multiple threads but, in most cases, it must be specifically written to utilize multiple threads. Operating system software also tends to run many threads as a part of its normal operation. Running virtual machines will benefit from adoption of multiple core architectures since each virtual machine runs independently of others and can be executed in parallel.

Multi-core processors are widely used across many application domains involving general-purpose, embedded, network, digital signal processing and graphics. The amount of performance gained by the use of a multi-core processor is strongly dependent on the software algorithms and implementation. In particular, the possible gains are limited by the fraction of the software that can be parallelized to run on multiple cores simultaneously. This effect was described by Amdahl's law (Gustafson, 1988). In the best case, so-called embarrassingly parallel problems may realize speedup factors near the number of cores. Many typical applications, however, do not realize such large speedup factors and thus, the parallelization of software becomes very significant.

Most application software is not written to use multiple concurrent threads intensively because of the challenge of doing so. A frequent pattern in multithreaded application design is where a single thread does the intensive work while other threads do much less.

Programming truly multithreaded code often requires complex co-ordination of threads and can easily introduce subtle and difficult-to-find bugs due to the interleaving of processing on data shared between threads. Consequently, such code is much more difficult to debug than single-threaded code when it breaks. There has been a perceived lack of motivation for writing consumer-level threaded applications because of the relative rarity of consumer-level multiprocessor hardware.

Parallel programming techniques can benefit from multiple cores directly. Some existing parallel programming models such as Cilk++, OpenMP and MPI can be used on multi-core platforms. Intel introduced a new abstraction for C++ parallelism called Threading Building Block which has also been implemented in Matlab compiler.

Managing concurrency acquires a central role in developing parallel applications. Figure 2.6 shows a typical layout of multi-core system suitable for parallel computing.

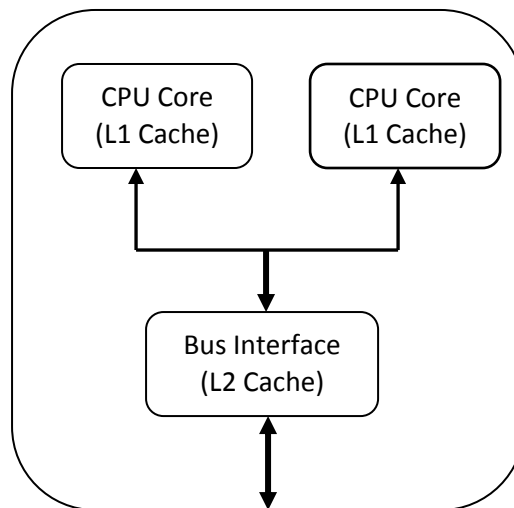


Fig.2.6: Multi-cores systems

2.10 Parallel Algorithm and Programming

Mattson et al. (2005) explained that the development of a parallel program largely depends on finding independent computations that can be executed simultaneously. Currently most of the parallel programmers produce a parallel program in the following three steps:

- (i) developing and debugging a sequential program
- (ii) transform a sequential program into a parallel program, and
- (iii) optimizing the parallel program.

This approach is called implicit parallel programming in which mostly data is paralleled. This is because of the sequential nature of the human and also because of the long time training in sequential program developments. The right way for developing an effective parallel program is initial parallel concept, that is, the programmers have to explicitly program the code in parallel and there is no sequential-to-parallel step. This requires the development of parallel mathematics, numerical methods, schemes, as well as parallel computational algorithms which is currently under developments.

With the existence of well recognized computer codes, the term parallel programming currently includes the conversion of sequential programs into the parallel ones. For many years from the first time when parallel systems were developed, the parallelization of existing codes has been done largely by manual conversion. Till date, many research and development projects are under funding to develop parallel tools for automatic parallelization. Many current compilers attempt to do some automatic parallelization, especially of do-loops, but the bulk of parallelization is still the realm of parallel programmer. The parallelization of a serial code can involve several actions such as the removal of inhibitors for parallelization, the insertion of constructs or calls to library routines and the restructure of serial algorithms to the parallel ones. There are different aspects for developing a parallel program on a shared-memory and distributed-memory systems. In the shared-memory programming, programmers view their programs as a collection of processes accessing a central pool of shared variables. In the message-passing programming, programmers view their programs as a collection of processes with private local variables and the ability to send and receive data between processes by passing messages. Since most of the shared memory computers are NUMA architectures, using message-passing programming paradigm on the shared-memory machines in fact improves the performance for many applications. In some problems many data items are subjected to identical processing and such problems can be parallelized by assigning data elements to various processors, each of which performs identical computations on its data.

2.10.1 Message Passing Interface

In term of software requirements, the current industry trends generally do not satisfy most of scientific and engineering design problems. Parallel software technologies are too far behind the hardware and as such the hardware resources are used wastefully in most of the cases. The types of software needed for the cluster are operating systems, utilities software (libraries, compilers, batch schedulers, and monitoring software), and application software (Hummel, 1999). The operating systems commonly used for HPC clusters are Unix system (e.g. Red Hat Linux Enterprise server) and Microsoft Windows HPC cluster server (Bradford & Mauget, 2001).

Message passing interface (MPI) is the industry standard library for developing message passing programs on HPC platform. The major objective of the MPI is to establish a portable, efficient, and flexible standard for message passing that will be widely used for developing message passing programs. According to Gropp et al. (1999), MPI is known to be the first standardized, vendor independent, message passing library. The advantages of developing message passing software using MPI closely match the design goals of portability, efficiency, and flexibility. MPI is not an IEEE or ISO standard, but has in fact, become the "industry standard" for writing message passing programs on HPC platforms.

MPI layer or Application Programming Interface(API) enables the nodes to work together. MPI is a software library for C, C++, and Fortran programs that allows the sub-programs send messages to each other (Matthias et al., 1999). An MPI program is therefore a collection of sub-programs that work together to solve a complex and computational-intensive problems. Such problems are too usually complex computationally to complete in a reasonable time without utilizing multiple compute nodes, or it may be too large in size to fit onto one compute node, or a combination of both cases (Gregorio et al., 1997) and (Alberto et al., 2010). Unless the program has been explicitly changed to run on a cluster (or across multiple cores) it will only ever run on one core. In message-passing systems, program tasks communicate by explicitly sending/receiving data packages to/from each other. MPI initiates inter- and intra-communication among the cores and node (Rolf & David, 1999). The process of sending/receiving data is specified and controlled by the application program via

message passing libraries (MPI, PVM, etc.) or languages (VPP-Fortran, Linda, etc). Message Passing Interface allows message (data) to be passed between processes in a distributed memory environment. MPI's prime goals are to provide source-code portability and allow efficient implementation. It offers a great deal of functionality and also supports heterogeneous parallel architectures.

The current message passing programming models require application developers to explicitly manage data layout across memory hierarchies from on-chip caches to archival storage. There is a need for portable higher level abstractions that simplify the process of parallel programming by automating data layout and hiding low level architectural details such as memory hierarchies. Applications should be able to adapt quickly and reliably as new computational requirements arise and are scalable and easily portable to successive generations of platforms. High level programming models and abstractions are required to facilitate code reuse, reduce code complexity and abstract away low level details necessary to achieve performance on a particular architecture. The current models for message passing are too low level to achieve this. Four programming models that are currently of interest are:

- (i) the message passing (MPI) everywhere (shared and/or distributed memory)
- (ii) the standard language extensions or compiler directives for shared memory parallelism and the message passing for distributed
- (iii) the explicit threads for shared and the message passing for distributed
- (iv) the high level language and run-time (object) constructs to present a serial programming model and a global data space over shared and/or distributed memory

Figure 2.7 shows a typical parallel programming architecture with respect to MPI.

2.10.2 Parallel Virtual Machine

Parallel virtual machine (PVM) is a portable software package that permits a heterogeneous collection of Unix and/or Windows computers connected together by a network to be used as a single large parallel computer (Gropp et al., 2001). The PVM system transparently handles message routing, data conversion for incompatible

architectures, and other tasks that are necessary for operation in a heterogeneous, network environment (Al Geist et al., 1994). PVM can be used on homogeneous or heterogeneous platforms. It handles transparently all message routing, data conversion and task scheduling across a network of massively parallel processors or computers.

There are many types of computers available from single processor or scalar computers to machines with vector processors to massively parallel computers with thousands of microprocessors. Each platform has its own unique characteristics. Understanding the differences is important to understanding how best to program each. However, the best approach is to develop programs that will run reasonably well on a wide range of computers and architectures.

User Applications		Cluster Management Tools
Programming Libraries and Development Tools		
Interconnecting Messaging Protocol (MPI)		
Physical setup (On racks)	Interconnect Hardware (switches & network boards)	
	Cluster Nodes	

Fig.2.7: Parallel Programming Architecture

2.11 Parallel Computation using MATLAB Compiler

Some compilers and software suites developers have begun to convert their serial codes and algorithms to parallel-aware so as to take advantage of multi-cores and multi-processors technology. In engineering, few among such software include ANSYS and MATLAB. Some professional software and applications such as COMSOL MultiPhysics are still working on serial to parallel conversion which would be released for commercial purpose when ready.

Among these software, MATLAB compilers of versions 2008 till date, are most suitable for implementation of parallel computation of parallel algorithm for spatio-temporal

variations of geomagnetic field and other space weather simulations. MATLAB compiler is a high level technical programming language that was originally developed by Mathworks Incorporation to be a MATtrix LABoratory. It provides easy access to matrix software developed by the LINPACK and EISPACK projects. Over the years, Matlab has evolved into an interactive system and programming language for general scientific and technical computation and visualization. The basic MATLAB data element is a matrix, therefore, MATLAB commands are developed on the first principles and theories available in mathematics and engineering (Hanselman & Littlefield, 2000).

MATLAB is enriched with toolboxes that were developed mostly by third-party programmers to solve certain specific computational problems in science and engineering. The command “*ver*” if typed (without the quote) on the MATLAB command line will display the version with the list of accompanied toolboxes. Parallel computing toolbox (PCT) and Matlab distributed computing server (MDCS) are the toolboxes and functionality that enable Matlab to support parallel applications developed using the compiler programming compiler. MATLAB of versions 2008 till date have PCT and MDCS support for parallel computing.

The MATLAB compiler comprises of libraries and functions that are capable of solving computationally and data-intensive problems on multi-cores platforms, GPUs, distributed and clusters platforms. The high-level constructs, parallel constructs, special array types and in-built parallelized numerical algorithms enable the parallelization of MATLAB applications with associated MPI construct. The PCT toolbox provides workers (compute nodes of MATLAB computational engines) to execute applications locally on multi-core systems or on networked clusters with some modification of code, the same applications can be run on computer cluster or a grid computing service using the MATLAB Distributed Computing Server.

The MATLAB function “*matlabpool*” is used to connect MATLAB sessions to a pool of MATLAB workers that can run either locally on a multi-core system or on a computer cluster to configure a dedicated interactive parallel execution environment. This function reserves a number of MATLAB workers for executing a subsequent parallelizable statements. The scheduler who handles jobs scheduling determines whether the workers

would run remotely on a cluster or locally on MATLAB client machines. The parallel constructs and libraries could execute loops with independent variables in parallel on dynamically allocated cluster resources with interleaved serial and parallel code.

Software compilers are also required to generate highly optimized code that effectively utilizes multiple levels of memory. Run-time libraries that provide efficient and scalable methods for numerical analysis, adaptive mesh refinement, load balancing, mesh partitioning and parallel I/O are also necessary. Performance measurement, debugging, quality control, verification, and validation of parallel codes become increasingly difficult as code size and complexity increases and so the process of code development for complex parallel architectures needs to be made simpler.

2.12 Advances in Parallel Computing

Although, parallelism in computing is a concept that reduces turnaround time, it could also increase the CPU time due to overhead and thereby enlarge the required memory owing to duplicated data and instructions. Developing and debugging a parallel program is much more complicated than a sequential program, at least an order-of-magnitude. Different architectures and diverse programming models are suitable for various applications. Thus, the characteristics of applications should make decision for the selection of parallel hardware architecture and also the parallelization of the applications.

Dongarra et al. (2009) indicated some of the major challenges of parallel computations. He therefore recommended that good programming techniques must be applied to ensure efficient and scalable parallel processing of computationally-intensive applications. The frameworks of these challenges are precipitated on hardware and programming design considerations. Essentially, the following details were observed to be properly considered:

- (i) Identifying possibility of concurrency in algorithms;
- (ii) Scheduling tasks at the right granularity onto the processors in parallel;
- (iii) Data locality and data association with tasks difficulty;
- (iv) Scalability support in hardware with respect to bandwidth and memory latencies, and interconnects between processing resources;

- (v) Scalability support in software in view of libraries, algorithms and adaptive runtimes;
- (vi) Synchronization constructs and protocols that enable development of programs devoid of deadlock and race conditions;
- (vii) Tools, application program interfaces, auto-tuning and methodologies to support the debugging process;
- (viii) Support for fault tolerance and error recovery which involves techniques, framework and methodologies to guarantee successful completion or restore of the computations in an event of single point of failures of a node or core;
- (ix) Support for good software engineering practices such as composability, incremental parallelism, portability, code reuse among others;
- (x) Adaptive response to load balance is significant so as to optimize utilization of computing resources with respect to execution times. Load imbalance in parallel algorithms is largely due to non-uniformity in data distribution and poorly partitioning of the workload for the parallel processors so as to minimize idle time while synchronizing parallel runtimes. Load balancing is a means of distributing sub-tasks to each processor so as to be sure the workload is balanced and that no processor is idle while other processors are overworking.

2.13 Emerging Applications of Parallel Computing

High-performance computing (HPC) are vital tools for simulation of large and/or complex scientific and engineering design problems. Such problems include aircraft wing design, computational fluid dynamics for global climate change, heliophysical studies, solid mechanics for materials modeling, distributed Digital Signal Processing (DSP) systems, charged particle beams for accelerator design and virtual manufacturing (James & Alan, 2003). Some other computational and data-intensive application areas also include an automated algorithm for performing modal analysis of finite elements models (Oldehoeft, 2000), image processing, load balancing methods for mapping a computational grid to an array of processors, the development of iterative and direct solvers for sparse systems of linear equations, as well as scalable structural mechanics, shock physics models, genomics, geological and seismic studies (Gregorio et al,1997).

Modeling, simulation, visualization and analysis using HPC would help to reduce risk in engineering designs and total cost of ownership in software and hardware development (Bahman et al., 2008). In automobile industry, application of HPC is used to accelerate design cycles and minimize physical prototyping costs. Computer-aided engineering (CAE), including crash simulation, virtual wind tunnels and other computational fluid dynamics (CFD) models are among the application areas that run on large clusters of servers (Carre et al., 1999). The application of advanced computation systems can also be used to gain better physical understanding within the Earth, astrophysical, geophysical, geological and seismic studies as well as Space sciences (Yifeng & Hong, 2006). It has a significant role in achieving engineering design objectives and goals such in manufacturing, weather monitoring, spacecraft subsystems design, navigation paths, orbital predictions and also in financial modelling (Gropp et al., 2001), thus enhancing technological breakthrough, industrialization and socio-economic development (Haug et al., 1994). Computational mathematics and algorithms, numerical methods are needed to enable scientific simulations. Existing numerical methods, although adequate for most 2-D simulations, would not scale well for emerging 3-D problems. Such numerical methods include preconditioned iterative methods for sparse linear systems of equations, methods for large systems of nonlinear equations, methods for time-dependent differential equations, parallel adaptive mesh refinement techniques and support libraries, flexible code frameworks for building physics codes and application aware communication routines. The interests in analysis of problems of complex geometries also require sophisticated, intuitive and easy to use graphical mesh generation tools.

2.14 Parallel Computing Benchmark Metrics

Dongarra et al. (2009) observed that overall parallel computing performance depend on specific characteristics of nodes in HPC, such as memory, processor type and processor quantity as well as parallel algorithm. Li et al. (2003) compiled a set of benchmarks that clearly characterize cluster performance while Joseph et al. (2001) emphasized the importance of working knowledge of the hardware, operating system and low-level system libraries. A set of benchmarks capable of providing test performance of the major subsystems of the clusters include the measurement of the following parameters:

- (i) total network transfer capacity of the cluster using large blocks of data, performing a matrix transpose;
- (ii) floating point operations per second (FLOPS) across the whole cluster using High Performance Linpack (HPL)
- (iii) matrix-to-matrix multiply on separate nodes without communications between them
- (iv) Bandwidth between CPU and memory using vector-scalar multiply-add operations, update rate of integers in random memory locations.
- (v) CPU execution rate in FLOPS for Discrete Fast Fourier Transforms
- (vi) latency and bandwidth of the cluster's network using small data sets and MPI routines
- (vii) peer-to-peer network testing and I/O operations

2.14.1 Scalability of Parallel Computations

Scalability is the ability to retain performance levels when number of processors and/or job size is increased. Computational algorithms should scale well with different sizes of input dataset and computing resources (inter-process communication and coordination overheads). The transition from serial to parallel algorithm must be undertaken to improve scalability and the over performance gain. Numerical instability and lack of data locality are major causes of poor scalability of parallel algorithms.

A serial algorithm is a computation that runs on a single processor while a parallel algorithm can execute multiple instructions simultaneously in a typical parallel architecture. A parallel architecture is a computing environment that has multiple processing resources such as processors, interconnection network and large storage or memory devices. Dongarra et al. (2009) further gave more insight on the execution of parallel programs which requires the parallel architecture to make use of either shared memory or message-passing (distributed memory) communication model. The latter model is commonly used today due to its relative low cost and better availability. The breaking down (decomposition) of serial algorithm into a large number of small tasks is known as *fine-grained* while decomposition that consists of a small number of larger tasks, it is called *coarse-grained*.

2.14.2 Benchmark for Parallel Algorithms

The performance of an algorithm is usually determined and analyzed based on execution times (runtimes) and speedup. The speedup (S_p) of a parallel algorithm can be defined as the ratio of the time taken to solve a problem on a single processor to the time required to solve the same problem(s) on a given parallel computer with processors (p). It is the metric value that evaluates the relative speed of the parallel algorithm with respect to number of processors as compared with the serial algorithm runtime.

2.15 Advanced Computation of Sq Daily Variations

According to the literature reviewed, it was generally observed that benchmark for parallel algorithm has not been developed for parallel computation and analysis of worldwide Sq on higher time resolution. However, parallel computing has been applied for simulations of magnetic reconnection, magnetospheric events and models (James, 2007). In similar literature survey, it was observed that some applications of parallel computation on space weather modeling were executed on multi-processing platform.

Till recent time, geomagnetic field dataset were analyzed by using simple statistical methods on relatively small localized dataset. Therefore, complex computationally-intensive methods and algorithms were not applied. Most of the earlier geomagnetic field acquisition systems were also prior to the evolution of advanced computational platforms. Today, sophisticated instruments have been installed in most geomagnetic observatories to record magnetic field variations even on second basis. It was observed that the existing analysis techniques for Sq daily variation are not sufficient to describe some geotechnical phenomena that have inter-regional footprint. Meanwhile, Uozumi et al. (2000), Chi & Russel (2005), Manoj et al. (2006), Courtillot & LeMouel (2007), Courtillot & Chulliat (2008), Chulliat et al. (2009) identified the usefulness of even 1-second sampled data in studying ionospheric current, travelling ionospheric disturbance, ULF and the equatorial electrojet as precursors of some geophysical phenomena that occur in less than one-hour. Love & Carol (2011) modified the 50 years old Kyoto (WDC) algorithm to process one-minute data for 13 stations along the low-latitude zone. They suggested that their work would serve as a basis for research on Sq current dynamics if computational resources would be enhanced.

CHAPTER THREE

3.0 ANALYSIS OF SPATIO-TEMPORAL VARIATIONS OF GEOMAGNETIC FIELD

In this section, the procedures and tools employed for achieving the aim and objectives of this work are presented.

3.1 Acquisition of Geomagnetic Field Data

The primary data used for this research was obtained from International Real-time Magnetic Observatory Network (INTERMAGNET). This network hosts and regulates the geomagnetic array observatories. After magnetic data acquisition, INTERMAGNET ensures that certain level of data pre-processing and error correction are effected before distribution. For the purpose of the analysis in this work, one-minute time samples of geomagnetic field dataset of year 1996 were extracted from 64 observatory stations located in 20 countries worldwide. The detailed distributions and list of the observatories are shown in Figure 3.1 and APPENDIX A respectively.

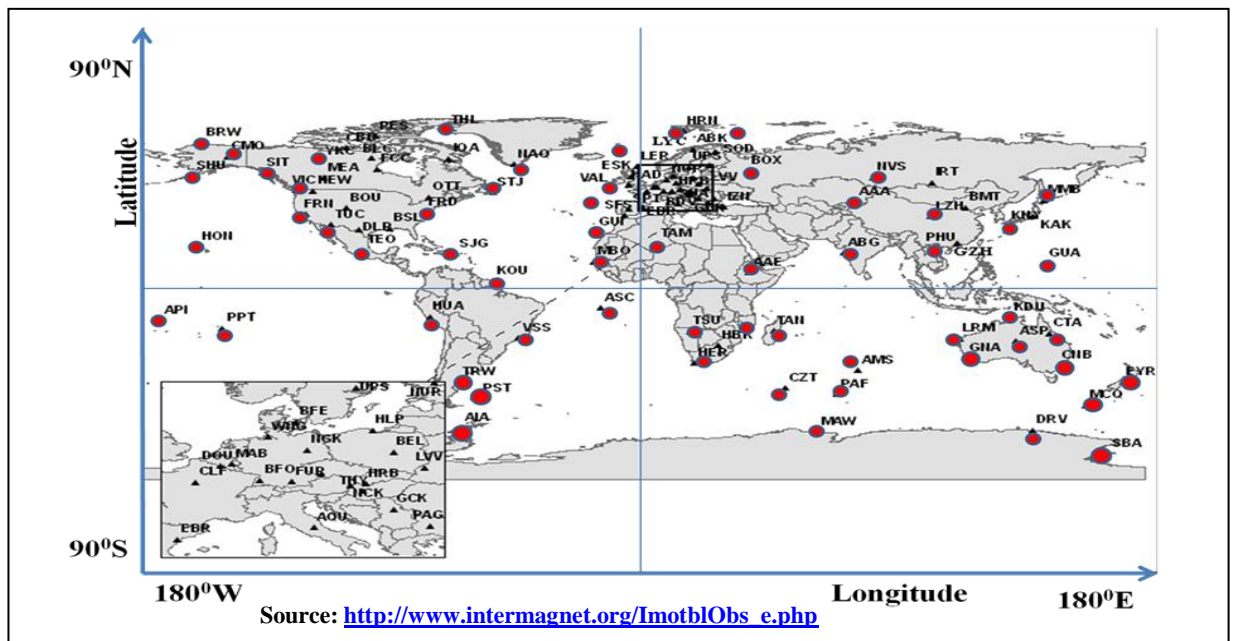


Fig. 3.1: INTERMAGNET observatories

1996 was the nearest solar minimum year that had larger volume of dataset required in this work. Other data used in this work include International Quiet Days (IQD) and geographic coordinates of the stations in latitude/ longitude for local time referencing. IQD were collected when solar activities were relatively quiet as against International Disturbed Days (IDD). Table 3.1 shows the excerpt of the distributions while Table 3.2 shows a list of IQD and IDD as provided by Geosciences Australia (2010). The five most quiet days were selected per month for the computation of solar quiet daily variations.

Table 3.1: Excerpt of the INTERMAGNET 64 Observatory stations (1996)

S/N	Station ID	Locality	Country	Orientation	Long ($^{\circ}$)	Lat ($^{\circ}$)
1	ABK	Abisko	Sweden	HDZF	18.823	68.358
2	ALE	Alert	Canada	XYZF	297.65	82.5
3	AMS	MartinDe Vivis	France	XYZF	77.574	-37.796
4	BDV	Budkov	Czech Republic	XYZF	14.015	49.08
5	BEL	Belsk	Poland	XYZF	20.79	51.84
6	BFE	Brofelde	Denmark	XYZF	11.67	55.63
7	BLC	Baker Lake	Canada	XYZF	263.97	64.33
8	BNG	Bangui	Central African Republic	XYZF	18.566	4.333
9	BOU	Boulder	USA	HDZF	254.76	40.14
10	BRW	Point Barrow	USA	HDZF	203.38	71.38
11	BSC	Bay St. Loius	USA	HDZF	270.6	30.4
12	CBB	Cambridge Bay	Canada	XYZF	254.97	69.12
13	CLF	Chambon la foret	France	XYZF	2.266	48.017
14	CMO	College	USA	HDZF	212.16	64.86
15	CNB	Canberra	Australia	XYZF	149.36	-35.31
..

Source: http://www.intermagnet.org/ImotblObs_e.php

Table 3.2: IQD and IDD Dataset (1996)

10 International Quiet Days (Q1-Q10) and 5 International Most Disturbed Days (D1-D5)																
Year	Month	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	D1	D2	D3	D4	D5
1996	JANUARY	09	08	10	11	07	23	04	01	25	06	13	14	29	15	20
1996	FEBRUARY	06	03	21	04	05	01	07	19	09	08	11	24	25	26	23
1996	MARCH	07	02	06	01	30	31	15	29	05	03	21	11	20	13	24
1996	APRIL	07	06	26	28	29	02	25	30	08	24	17	14	19	12	18
1996	MAY	18	28	07	23	06	12	11	09	10	02	13	14	20	04	16
1996	JUNE	13	14	25	22	01	03	09	12	26	02	06	19	29	17	18
1996	JULY	10	09	27	11	24	29	19	06	23	16	31	08	28	03	05
1996	AUGUST	08	10	19	12	11	18	22	21	07	04	29	25	30	27	28
1996	SEPTEMBER	03	30	02	06	01	05	08	29	07	09	20	10	21	26	12
1996	OCTOBER	06	05	07	26	27	15	01	17	04	25	19	22	23	18	09
1996	NOVEMBER	30	02	03	01	23	08	11	07	16	22	14	13	15	18	17
1996	DECEMBER	19	06	20	05	01	08	28	26	27	24	10	11	15	16	04
source: http://www.ga.gov.au/oracle/geomag/display_iqd.jsp																

3.1.1 Geomagnetic Data Quality Control Procedure

The INTERMAGNET organization was established as a global network of co-operating digital magnetic observatories in order to adopt best practices and modern standard specifications for measuring and recording equipment. This, as a result facilitates data integrity, data exchange and production of definitive geomagnetic datasets. The low fields, often voltage fluctuations are collected and oversampled via magnetometer (e.g. Narod fluxgate sensor) analog output. The sampled data are thereafter digitized by an Ethernet-based 24-bit A/D converter and Gaussian digital filter which generates magnetogram. Subsequently, the magnetogram are validated by correcting noise level due to external spikes, offsets, or artificial disturbances. This is often done by comparing minute-to-minute differences in the data values at observatories where two magnetometers are operated in parallel. The root mean square (RMS) of the differences is calculated within short durations on regular basis. Consequently, mean and standard deviation are computed over successive period. The refined data also known as definitive data are thereafter distributed at geomagnetic information nodes for the public domain. Most work reviewed in the literature used INTERMAGNET geomagnetic definitive dataset.

3.2 Geomagnetic Field Data Processing

The geomagnetic field data obtained has additional contributions from other sources as explained in the literature. These extraneous contributions are therefore filtered out in order to obtain only solar quiet daily variations of geomagnetic field.

3.2.1 Post-acquisition Processing of Geomagnetic Field Data

The magnetic dataset acquired were available in one-hour and one-minute sampling rates in IAGA2000/2002 format. The data from 44 observatories were represented in XYZF orientation while 20 observatories were in HDZF orientation. Equations (3.1) to (3.5) show the conversion of XYZ orientation to HDZ orientation using the orthogonal geometry with respect to horizontal plane as shown in Figure 1.3.

For the purpose of better analysis, the observatories were classified into low latitude ($0-30^{\circ}$), middle latitude ($31-60^{\circ}$) and high latitude ($61-90^{\circ}$) zones as shown in Table 3.3 and Figure 3.2. The details of the observatories are included in Appendix A.

Table 3.3: Stratification of INTERMANGET observatories to latitude zones

	Low Latitude (0-30 ⁰)	Middle Latitude (31-60 ⁰)		High Latitude (60-90 ⁰)
Station ID	<ul style="list-style-type: none"> • BNG • BSC • DLR • GUA • HON • KOU • MBO • PHU • PPT • SJG • TAM • TAN 	<ul style="list-style-type: none"> • AMS • BDV • BEL • BFE • BOU • CLF • CNB • CZT • ESK • EYR • FCC • FRD • FRN • FUR • GDH • GLN • GNA • HAD 	<ul style="list-style-type: none"> • HER • KAK • LER • LOV • MEA • MMB • NCK • NEW • NGK • NUR • OTT • PAF • PBQ • SIT • STJ • THY • TUC • VIC • WNG 	<ul style="list-style-type: none"> • ABK • ALE • BLC • BRW • CBB • CMO • DRV • IQA • MBC • NAQ • RES • SBA • SOD • THL • YKC
Total	12 (19%)	37 (58%)		15 (23%)

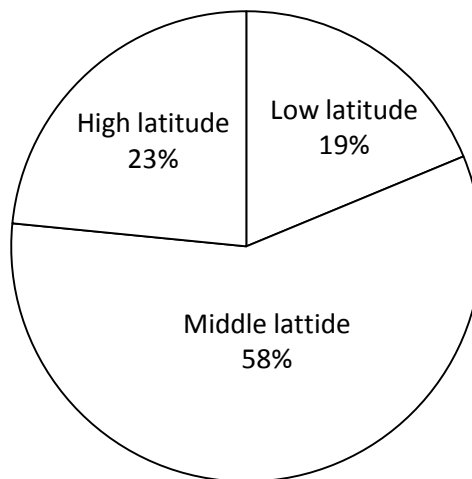


Fig. 3.2: Stratification of INTERMANGET observatories to latitudinal zones

Using their orthogonal geometry (as shown in Figure 1.3) for the conversion, we have:

$$F^2 = H^2 + Z^2 \quad (3.1)$$

Where H is given by

$$H = \sqrt{X^2 + Y^2} \quad (3.2)$$

In effect,

$$D = \cos^{-1}\left(\frac{X}{H}\right) = \sin^{-1}\left(\frac{Y}{H}\right) = \tan^{-1}\left(\frac{Y}{X}\right) \quad (3.3)$$

Furthermore, $\vec{F}, \mathbf{H}, \mathbf{I}, \mathbf{Z}$ can be expressed as follows:

$$H = F * \cos(I) \quad \text{and} \quad Z = F * \sin(I) \quad (3.4)$$

This implies that

$$I = \tan^{-1}\left(\frac{Z}{H}\right) \quad \text{and} \quad \vec{F} = \sqrt{H^2 + Z^2} \quad (3.5)$$

Tesla, Gamma and Gauss are the International System of Units (SI) unit for magnetic field intensity (flux density). 1 Gauss = 100,000 nT, the gamma (1 gamma = 1 nT).

The data was pre-arranged in 144 x 10 matrix with sensor orientations, locality and IAGA code metadata in the block headers. Thus, Matlab m-file functions and scripts were developed to re-arrange the dataset and discard the successive headers. The processing of one-minute (1440 minutes/day) geomagnetic field data with their respective three (3) components (i.e. H, D and Z) for 366 days in 1996 generated [366 x 276480] matrix which represents 101,191,680 measurements. Table 3.4 shows an order of magnitude and workload involved at solving the problem. The increased resolution of dataset enlarges the capacity of data processing and therefore requires more computing resources so as to have optimal results in a reasonable time.

Table 3.4: Work Breakdown Structure

S/N	PROBLEM	SIZE (Matrix)	CASE (n)	Task
1	64 stations with 3 Geomagnetic field components (H, D, Z)	64x3	192	Data Post-acquisition processing
2	Year 1996 = 366 days; Daily = 24hrs * 60 = 1440 minutes;	$\begin{Bmatrix} (3*64*366,1440) \\ 70272 \times 1440 \end{Bmatrix}$	101,191,680	Baseline and non-cyclic correction
3	2D Sq variations for H, D, Z	$\begin{Bmatrix} 36*64,1440 \\ 20304 \times 1440 \end{Bmatrix}$	20,304	2D SqH, SqD, SqZ
4	3D (Contour) Sq variations for H, D, Z; average mean (Jan-Dec)	$\begin{Bmatrix} 36*64,1440 \\ 20304 \times 1440 \end{Bmatrix}$	20,304	3D SqH, SqD, SqZ

The workload of computational dataset in Table 3.4 can be expressed in algebraic form as shown in Equation (3.6):

$$\left\{ \begin{array}{c} 64 \times 3 \\ 70272 \times 1440 \\ 20304 \times 1440 \\ 20304 \times 1440 \end{array} \right\} \left\{ \begin{array}{c} H \\ D \\ Z \end{array} \right\} = \left\{ \begin{array}{c} SqH \\ SqD \\ SqZ \end{array} \right\} \quad (3.6)$$

Equation (3.6) involves several iterations, loops and arrays that could take advantage of parallel computational platform so as to obtain optimal results at a reasonable time. In addition to the observatory-based data quality control methods, the following computational step-wise procedures were applied to separate Sq field from contributions due to other sources.

3.2.1.1 Daily Baseline values

The baseline values were computed and used to adjust components (H, D, Z) dataset so as to correct non-random error and thereby enhance confidence limit. The daily baseline value (M_0) is defined as the average of 240 minutes flanking local midnight, that is,

$$M_0 = \frac{\sum_{t=1}^{120} M_t + \sum_{t=1324}^{1440} M_t}{240} \quad (3.7)$$

Where M_0 = baseline value; $t = 1, 2, \dots, 1440$ (local time). In equation (3.7), M_t represents H_t , D_t or Z_t .

3.2.1.2 Minute-basis Departures

Conventionally, in the mid-night the ionospheric current should be minimal due to minimal solar impact. Therefore, departure (ΔM_t) (on minute basis) from midnight baseline was obtained by subtracting the midnight baseline values for a particular day from all *minute* values for that same day. Thus,

$$\Delta M_t = M_t - M_0 \quad (3.8)$$

3.2.1.3 Correction for Non-cyclic Variations

Non-cyclic variation due to geomagnetic storm was removed by making linear adjustment in the daily minute values of ΔM_t as follows:

$$\Delta c = \left(\frac{\Delta M_1 - \Delta M_{1440}}{1439} \right) \quad (3.9)$$

where Δc are mean values of the magnetic elements for two intervals of time in minute.

The linear adjustment therefore generates the Sq field as follows:

$$Sq = \Delta M_t + (t - 1)\Delta c \quad (3.10)$$

Specifically, we have:

$$Sq_H = \Delta H + (t - 1)\Delta c \quad (3.11)$$

$$Sq_D = \Delta D + (t - 1)\Delta c \quad (3.12)$$

$$Sq_Z = \Delta Z + (t - 1)\Delta c \quad (3.13)$$

3.2.2 Interpolation of Sq Field using Kriging Method

Kriging method was applied to compute unbiased estimation of unknown values at some grid points. This process enables data gridding for simulation of Sq field in 2D and 3D contour mapping and global visualization of Sq dynamics as expressed in Equations (3.14) to (3.38). The geomagnetic field data (H,D,Z) in variable $Z(s_i)$ is known to follow an intrinsically stationary process with constant unknown mean (μ), random noise function (ϵ), weighting coefficient (w) and known spatial covariance function $C(h)$ for spatial lags $h=s_i-s_j$, $s=(x,y)$, x =latitude (lat); y = longitude (lon), this can be expressed as:

$$Z(S_i) = \mu + \epsilon(s_i) \quad (3.14)$$

To estimate geomagnetic field values at unobserved locations, $S_o \in D$, $s = (\text{lat}, \text{lon})$ while D is the study region, kriging method was used to compute weighted average of the observations (S_i) as follows:

$$\bar{Z}(s_o) = \sum_{i=1}^n w_i Z(s_i) \quad (3.15)$$

To find the best linear unbiased predictor, we minimized the variance of the interpolation error, that is, minimize mean square prediction error, as follows:

$$Var(\bar{Z}(s_o) - Z(s_o)) = E[(\bar{Z}(s_o) - Z(s_o))^2] \quad (3.16)$$

For an unbiased estimation, it is required that:

$$E[\bar{Z}(s_o)] = E[Z(s_o)] = \mu \quad (3.17)$$

In compliance with Equation (3.14), this implies that:

$$E[\bar{Z}(s_o)] - E[Z(s_o)] = E\left[\sum_{i=1}^n w_i Z(s_i)\right] - E[Z(s_o)] \quad (3.18)$$

$$= \sum_{i=1}^n w_i E[Z(s_i)] - E[Z(s_o)] \quad (3.19)$$

$$= \sum_{i=1}^n w_i \mu - \mu = \mu \left(\sum_{i=1}^n w_i - 1 \right) \quad (3.20)$$

Thus $\sum_{i=1}^n w_i = 1$ for unbiasedness to hold and we have a minimization problem with a constraint that can be solved using Lagrange multipliers as follows:

$$L(\omega_i, \lambda) = E[(\hat{Z}(s_o) - Z(s_o))^2] + 2\lambda \left(\sum_{i=1}^n w_i - 1 \right) \quad (3.21)$$

$$= Var\left[\sum_{i=1}^n \omega_i Z(s_i)\right] + Var[Z(s_o)] - 2Cov\left[\sum_{i=1}^n \omega_i Z(s_i), Z(s_o)\right] + 2\lambda \left(\sum_{i=1}^n w_i - 1 \right) \quad (3.22)$$

$$\begin{aligned} &= \sum_{i=1}^n \sum_{j=1}^n \omega_i \omega_j Cov[Z(s_i), Z(s_j)] + Var[Z(s_o)] \\ &\quad - 2 \sum_{i=1}^n \omega_i Cov[Z(s_i), Z(s_o)] \\ &\quad + 2\lambda \left(\sum_{i=1}^n \omega_i - 1 \right) \end{aligned} \quad (3.23)$$

Using the variance of a linear combination, we write

$$Var\left[\sum_{i=1}^n \omega_i Z(s_i)\right] = \sum_{i=1}^n \sum_{j=1}^n \omega_i \omega_j Cov[Z(s_i), Z(s_j)] \quad (3.24)$$

To find the extremum we differentiate with respect to ω_i and λ and let the derivatives vanish. This implies that:

$$\frac{\partial L(\omega_i, \lambda)}{\partial \omega_i} = 2 \sum_{j=1}^n \omega_j Cov[Z(s_i), Z(s_j)] - 2Cov[Z(s_i), Z(s_o)] + 2\lambda = 0 \quad (3.25)$$

$$\sum_{j=1}^n \omega_j \text{Cov}[Z(s_i), Z(s_j)] + \lambda = \text{Cov}[Z(s_i), Z(s_o)] \quad (3.26)$$

$$\frac{\partial L(\omega_i, \lambda)}{\partial \omega_i} = 2 \sum_{i=1}^n \omega_i - 2 = 0 \quad (3.27)$$

Which gives

$$\sum_{i=1}^n \omega_i = 1 \quad (3.28)$$

In matrix notation, this can be expressed as: $C^*w = D$

Where C is the covariance matrix of the observed values; w is vector of weights and the Lagrange multiplier. D is the vector of covariances at the prediction location (S_o).

Solving for the weights gives:

$$W = C^{-1}D \quad (3.29)$$

Expanding, Equation (3.29) leads to the following set of linear equation:

$$\begin{aligned} C_{11}w_1 + C_{12}w_2 + \dots + C_{1n}w_n + \lambda &= C_{1o} \\ C_{21}w_1 + C_{22}w_2 + \dots + C_{2n}w_n + \lambda &= C_{2o} \\ \dots & \\ C_{n1}w_1 + C_{n2}w_2 + \dots + C_{nn}w_n + \lambda &= C_{no} \\ w_1 + w_2 + \dots + w_n + 0 &= 1 \end{aligned} \quad (3.30)$$

The overall task of the kriging method is to calculate the optimal value of weighting coefficient (w) so as to obtain a minimal estimation error. Equation (3.30) can then be expressed as:

$$\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \\ \lambda \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} & \dots & C_{1n} & 1 \\ C_{21} & C_{22} & \dots & C_{2n} & 1 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ C_{n1} & C_{n2} & \dots & C_{nn} & 1 \\ 1 & 1 & \dots & 1 & 0 \end{pmatrix}^{-1} \begin{pmatrix} C_{1o} \\ C_{2o} \\ \vdots \\ C_{no} \\ 1 \end{pmatrix} \quad (3.31)$$

Thus, C only needs to be calculated and inverted once but D is found for every prediction location. This inversion operation is computationally intensive for the large geomagnetic

dataset (n) as shown in Table 3.4. With these weights, we can solve for expected value of the missing and unknown geomagnetic field data at the next grid points.

So we have,

$$\bar{Z}(s_o) = \sum_{i=1}^n w_i Z(s_i) \quad (3.32)$$

The variance of the prediction is found via the mean squared error as follows:

$$\begin{aligned} & \sum_{i=1}^n \sum_{j=1}^n \omega_i \omega_j \text{Cov}[Z(s_i), Z(s_j)] + \text{Var}[Z(s_o)] \\ & - 2\text{Cov} \sum_{i=1}^n \omega_i \text{Cov}[Z(s_i), Z(s_o)] \end{aligned} \quad (3.33)$$

Where

$$\sum_{i=1}^n \sum_{j=1}^n \omega_i \omega_j \text{Cov}[Z(s_i), Z(s_j)] = \sum_{i=1}^n w_i \sum_{j=1}^n \omega_j \text{Cov}[Z(s_i), Z(s_j)] \quad (3.34)$$

$$= \sum_{i=1}^n w_i (\text{Cov}[Z(s_i), Z(s_o)] - \lambda) \quad (3.35)$$

So, the mean squared error gives the required kriging variance,

$$= \sigma^2 - \sum_{i=1}^n w_i (\text{Cov}[Z(s_i), Z(s_o)] - \lambda) \quad (3.36)$$

$$\sigma_k^2 = \sigma^2 - \sum_{i=1}^n w_i (\text{Cov}[Z(s_i), Z(s_o)] - \lambda) \quad (3.37)$$

In matrix notation as:

$$\sigma_k^2 = \sigma^2 - w'D \quad (3.38)$$

Thus, kriging provides σ_k^2 as a measure of precision.

To implement this algorithm in Matlab environment, the *krigingAPI* m-file was developed to invoke GridLib.dll from Surfer application so as to interface kriging interpolation method with MATLAB programming compiler. Syntax of the m-file codes is shown below:

```
[AI,BI] = meshgrid(t,t);
[AI,BI,MI] = krigingAPI(a,b,Mb, AI,BI, 'kriging')
% Mt = geomagnetic data ( HDZ components); (a,b) = (longitude, latitude) coordinate
% AI and BI = uniform grid points produced by meshgrid in-built function,
t=1,2,3,...1440
```

Where MI fits the 3D view Sq contour plots of the form $m = f(a, b)$ to the data in the non-uniformly spaced vectors (a, b, Mt) . The developed function, `krigingAPI` interpolates the 3D view at the points specified by (AI, BI) to produce the MI. The Sq contour passes through the data points. This algorithm segment invokes *srfKriging* from Surfer GridLib.dll library for further data gridding. The kriging approximation function was implemented in the optimization routines.

3.2.3 Development of Scalable Algorithm for Simulation of Sq Variations

The serial and corresponding parallel algorithms together with SurferAPI were fully developed and implemented using Matlab R2010a compiler. The execution times were observed as the parallel algorithm scales with varying number of processors.

3.2.4 Development of Sequential Algorithm for Sq Variations

Figure 3.4 shows the schematic flowchart of the developed sequential (serial) algorithm for analysis and simulation of solar quiet daily variations of geomagnetic field. The development of the algorithm generated large codebases which involve many processes and threads in Matlab R2010a environment. Implementation of kriging method was quite challenging and complex because Matlab compiler does not have kriging interpolation toolbox. Therefore, in this work, application programming interface (API) routine was developed as an interface between Matlab compiler and Surfer gridding library (Gridlib.dll). The integrated programs were thereafter compiled in Matlab Development Environment. This real time interfacing is one of the unique features of this work.

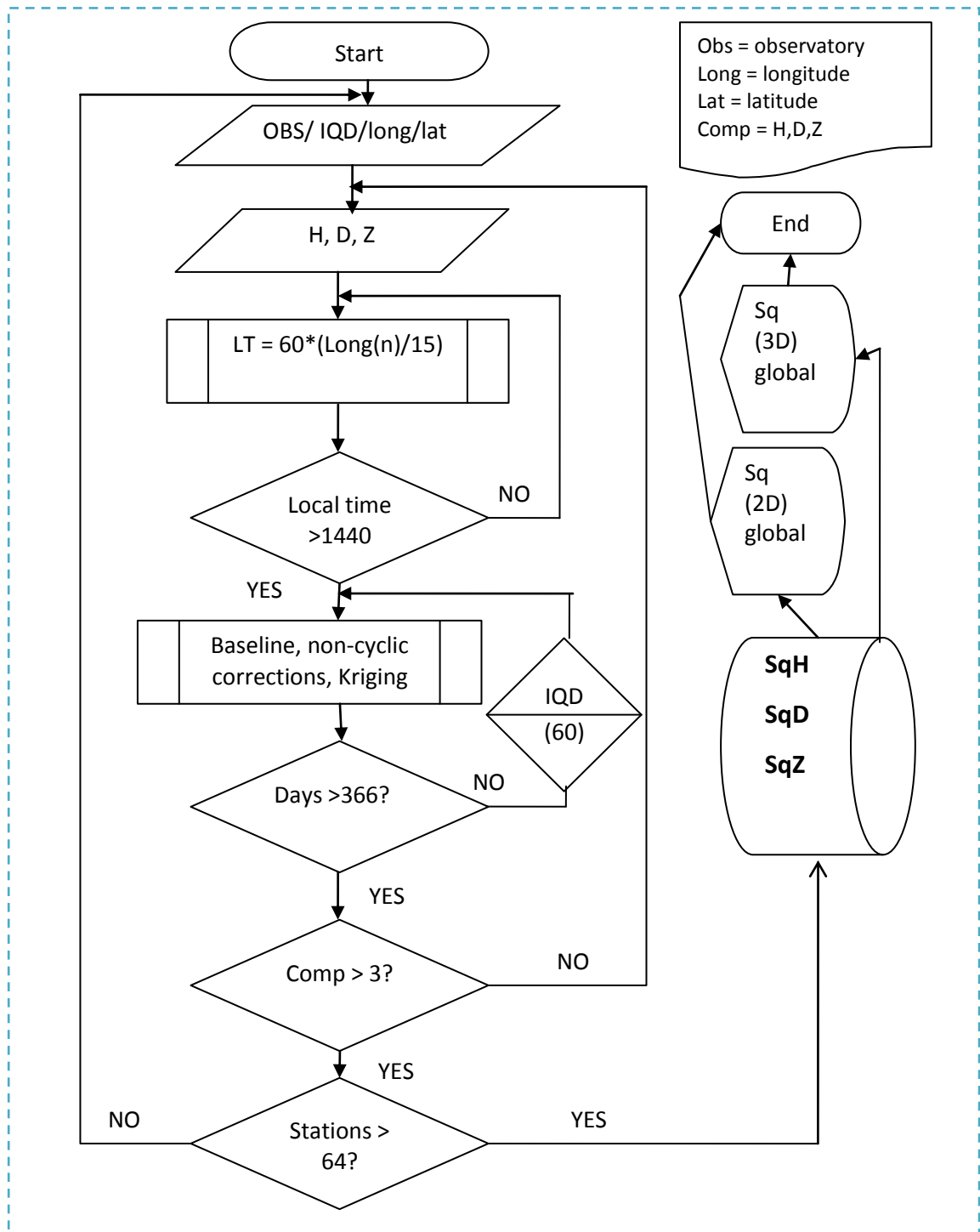


Fig. 3.3: Sq Sequential Algorithm for Serial Computation

3.2.5 Development of Parallel Algorithm for Sq Variations

The sequential algorithm was thereafter re-engineered to develop the corresponding parallel version for parallel computation of Sq daily variation of geomagnetic field. The tasks were run simultaneously on eight (8) processors as shown schematically in Table 3.5. Profiling in-built function was used to examine the code segments or functions in terms of CPU execution time. Data dependencies and recursive code segments were resolved to ensure parallelism. Critical variables such as local time (LT) that run 1440 iterations repeatedly were sliced and indexed logically so as to enable parallelization. Runtime and syntax errors generated were corrected and modified by re-designing programming structures and methods. The algorithm with kriging method was fully developed and compiled using Matlab R2010a and associated parallel computing libraries. The parallel algorithm was designed to be clairvoyant by implementing parallelism of the multiple processes and threads in codebase.

Table 3.5: Sequential and Parallel execution time structure

Sequential computation	Parallel computation
$\text{for } n = 1:8$ $\text{Simulation Time} = \sum_{n=1}^8 Sq_n$ End	$\text{Matlabpool } 8 \text{ open}$ $\text{parfor } n=1:8$ $\% \text{ slicing and indexing variables}$ $\text{Simulation Time} = \max_n(Sq_n)$ End Matlabpool close

Where p = number of processors used, in this initial step, $p=8$ (CPUs).

3.2.6 Resolving Tasks Allocation and Matrices Partitioning

Load balancing and concurrency were identified as major problems of the Sq algorithm optimization. One of the solutions to load balancing employed was by partitioning the large matrices for a fairly even assignment of subtasks to the eight (8) processors in parallel. The success of sub-tasks allocation minimized the processor idle time and time spent on inter-processor communication which was also facilitated by the parallel hardware design. The dimensions of acquired dataset, matrices (i.e. workload for processing) are highlighted below.

- Set of matrices used
 - i. 64 x 3 (for H, D, Z components by 64 observatories)
 - ii. 192 by 1440 (276480 cases)
 - iii. 366 x 276480 (leap year)
- For the partitioning procedure, we therefore employed the following format:

Let $V = (366\text{-by-}276480)$ matrix

Thus, V was distributed to the eight (8) CPUs, the algorithm was developed to partition the two-dimensional array horizontally into segments and assigns one segment of the array to each CPU. Consequently, the two-dimensional array was partitioned horizontally (by rows) by assigning columns of the V -array of 366-by- 34560 to each CPU. The pseudo code shown below was developed to balance the load of Sq processing on the eight (8) processors. It shows how a V -array was distributed to eight (8) CPUs.

```
S = zeros(366, 34560); T = codistributed(S);
# with the use of 8 processors (p)
computeNode 1: This node stores T(:,1: 34560);
computeNode 2: This node stores T(:,34561: 69120);
computeNode 3: This node stores T(:,69121: 103680);
computeNode 4: This node stores T(:,103681: 138240);
computeNode 5: This node stores T(:,138241: 172800);
computeNode 6: This node stores T(:,172801: 207360);
computeNode 7: This node stores T(:,207361: 241920);
computeNode 8: This node stores T(:,241921: 276480);
```

This above pseudo-code creates a 366-by-276480 replicated array locally and assigned it to variable S . The segment also distributes S and creates T as 366-by-34560 array for individual CPU. Each CPU has access to other segments of the array and locally holds copies of some global variables so as to run faster rather than sending and receiving data too frequently and thereby saving more execution times. Domain (data) decomposition technique was also applied to solve this problem in complement to functional (instructions set) decomposition due to configuration and nature of the problem.

The generic algorithm responsible for load balancing is given as:

```
matlabpool open config p
spmd { A = zeros(n, m);
       B = codistributed(S);
       # with the use of 8 processors (p)
       a=m/p; z=0;z1=1;
       For p=1:8
       z = z+a; computeNode p: This node stores B(:,z1:z);
       z1=z+1; end;      matlabpool close
```

3.3 NASRDA Advanced Computation Laboratory

The Advanced Computation Laboratory of National Space Research & Development Agency (NASRDA), Abuja, Nigeria was used as a test-bed for implementation of the parallel algorithm. NASRDA is a Nigeria Space Agency that has mandates to design, build and develop various kinds of satellites as well as capacity building in Space science and technology research for socio-economic benefits. In order to effectively achieve these mandates, in 2008, the Agency setup a world class laboratory for simulations of complex and/or large scientific and engineering designs problems.

The Advanced computation laboratory has a configuration of High Performance Computing Clusters (HPCC) as a shared research infrastructure for large-scale computations. The HPCC has 24 Dell PowerEdge master/slave servers as nodes with a total number of 192 cores. The Agency also has some other parallel computing resources in the laboratory such as 40 client workstations, 30TB dedicated powervaults, high capacity swappable storage drives, deployment of Unix and Windows server operating systems, fast Ethernet and fibre optics for LAN connectivity as shown in Figure 3.5.

Eight (8) cores, job scheduler, parallel computing toolbox, kriging algorithm and other libraries were configured in Matlab Windows-based platform to implement the Sq parallel algorithm developed.

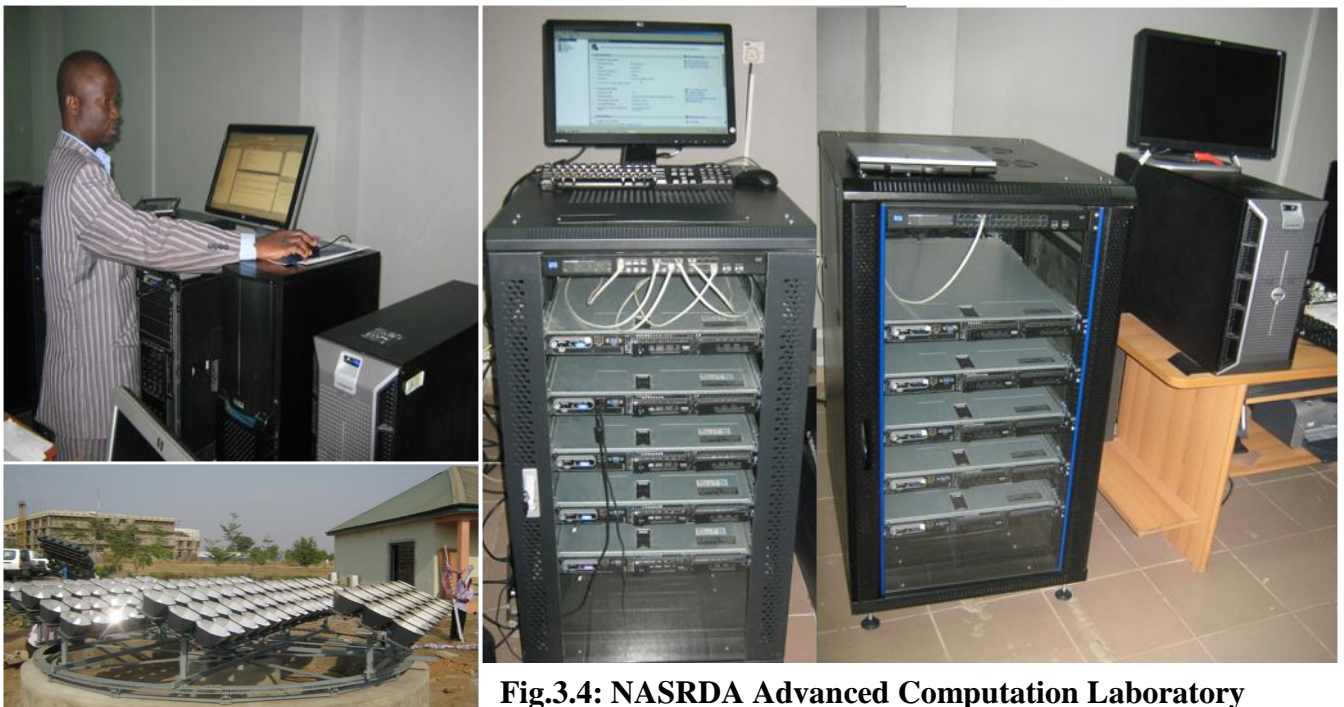


Fig.3.4: NASRDA Advanced Computation Laboratory

3.4 The Design of Parallel Architecture for Sq Algorithm

The design and implementation of the parallel computing platform involves eight parallel processors in a heterogeneous multicore architecture with shared and distributed (hybrid) memory model. The message passing interface protocol exchanged data among the nodes through high-speed front-side bus as shown in Figure 3.6. Table 3.6 shows the specifications of the parallel computing hardware deployed in this work. Thus, the parallel algorithm and parallel computing hardware constituted the parallel computational platform for analysis of Sq variations.

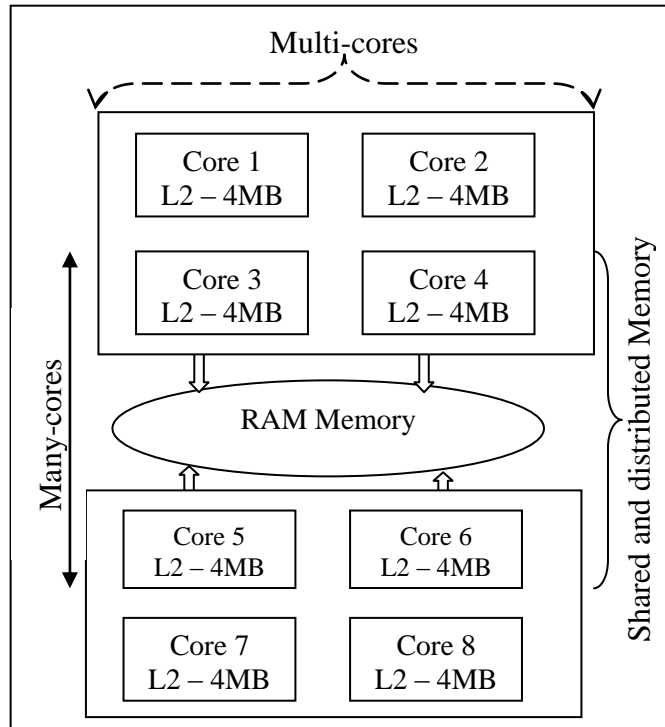


Fig.3.5: Layout of the Deployed Heterogeneous (Hybrid) Multicore systems

Table 3.6: Parallel Computing System Specifications

Feature	Specification
Model	Dell PowerEdge 2950
Processor (clock speed)	Intel(R) Xeon(R) CPU E5410 @ 2.33GHz (8 CPUs)
Front-side Bus speed	1333MHz
L2 Cache	4MB per core
Memory	16378MB RAM
Operating System	Windows Server® 2008 HPC Edition (6.0, Build 6001) (64-bit)

3.5 Validation of the Parallel Computing Systems

Serial and corresponding parallel algorithm segments for millions of array processing were developed to validate the parallel systems. Figures 3.7(a) and 3.7(b) show the execution times versus the number of processors. This is a sample case of embarrassingly parallel workload where there is no data dependency or inter-processor communication of intermittent results among the tasks. The curve of Figure 3.7a slopes down and begins to rise when seven (7) processors are used. This is known as parallel slowdown that indicates the optimal number of processors to solve the problem efficiently. However, the same algorithm scales well as the size of the problem increases in Figure 3.7b.

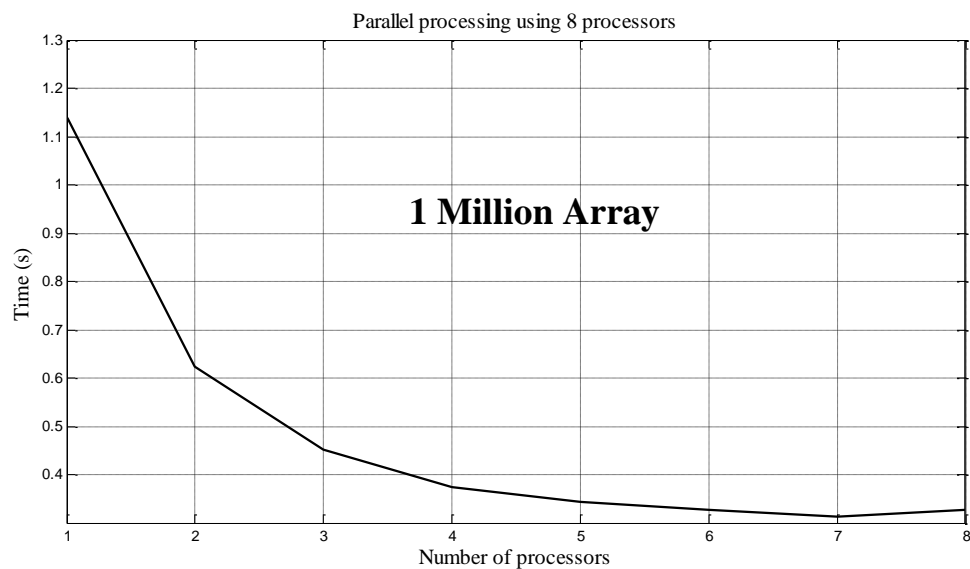


Fig. 3.6(a): Simulations execution time versus processors

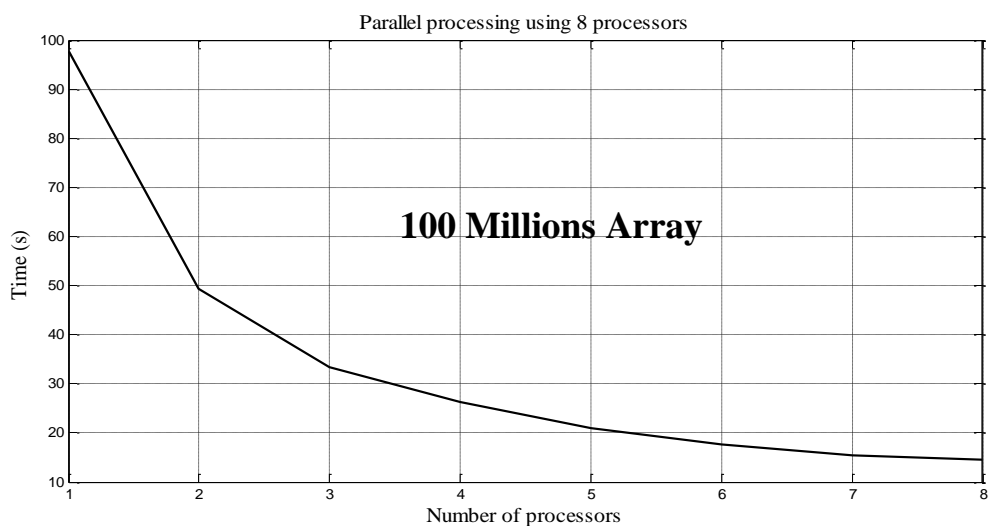


Fig. 3.6(b): Simulations execution time versus processors

3.6 Implementation of the Sq Parallel Algorithm

The developed parallel algorithm was implemented and validated on the configured heterogeneous multi-core systems as shown in Figure 3.8. The algorithm scalability was tested with increasing number of processors from 1 to 8 and it performed efficiently. Multithreaded technique was adopted as a means of dealing with recursive segments and data dependency. Some major segments were developed separately and thereafter interfaced together.

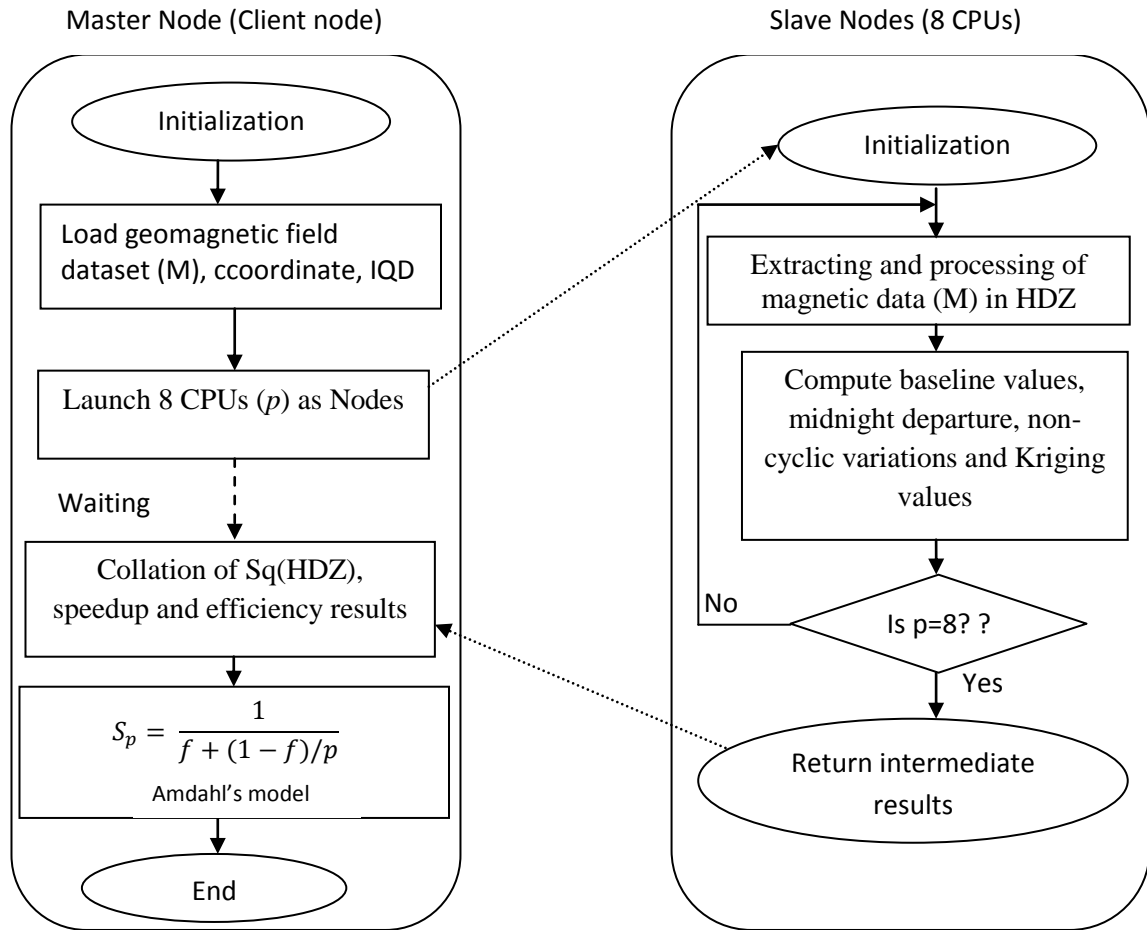


Fig. 3.7: Sq Parallel Algorithm

3.7 Efficiency Assessment of the Developed Computational Platform

In this work, the performance of the developed computational platform was evaluated using benchmark metrics that are generally based on wall clock execution time. The metrics include execution time, speedup, efficiency and other derivatives. Amdahl's model (1967) is an integrated benchmark was used to evaluate the scalability of the Sq

computational platform. A subroutine of the developed program generated the benchmark metrics for performance analysis alongside.

3.7.1 Execution Time

The execution time or run-time (in seconds) indicates the wall clock time from the start of the first process until the termination of the last process of program execution by processors(s). The run-times expressed in this work are times spent by the efficient sequential program (T_s), run-time for parallel program (T_p) and run-time when one processor (p) was used on the parallel system (T_1).

3.7.2 Speedup

Speedup (S_p) is parallel processing performance gain over sequential processing of the same problem. This gives:

$$S_p = \frac{T_s}{T_p} \quad (3.39)$$

Figure 3.9 shows three categories of speedup as a measure of scalability. These variations occur when $S_p > p$ (superlinear), $S_p = p$ (linear) and $S_p < p$ (sublinear). The superlinear and linear speedup are rarely obtained due to parallel overhead, therefore sublinear is the standard form that was used to measure how the algorithm scales as more processors are used.

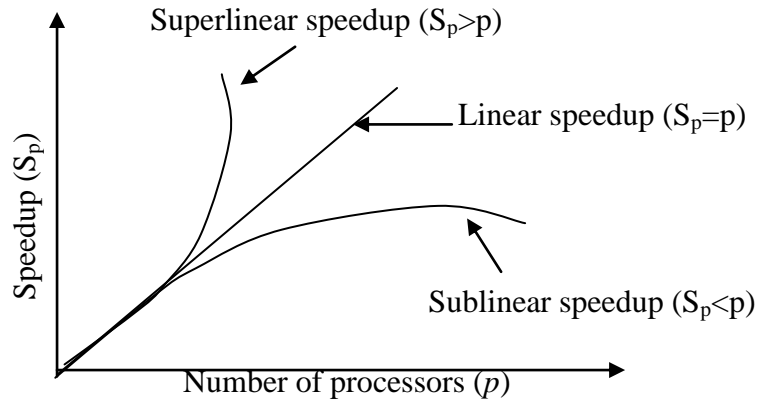


Fig. 3.8: Speedup Pattern in Scalable Parallel Algorithm

3.7.3 Scalability

The parallel processing gain was maintained when both problem size and number of processors were varied. Pre-allocation of array, global declaration of data, logical

indexing and vectorization of various geomagnetif field variables as well as multithreading were applied to enforce scalability of the developed parallel algorithm.

3.7.4 Efficiency

The Efficiency (ϕ) of a parallel system describes the average speedup with respect to the number of processors used. This is given as:

$$\phi = \frac{S_p}{p} * 100 \% \quad 0 \leq \phi \leq 1 \quad (3.40)$$

Efficiency is an estimation of how the processors were well-utilized in solving the Sq problem compared to how much effort was spent in inter-processor communications and synchronization. The efficiency of the Sq code parallelization measured percentage of simultaneously available resources.

3.7.5 Implication of the Parallel Overhead

Parallel overhead (T_o) is the difference between the total execution time (R_t) and the time spent (R_a) to process the actual job as shown in Equation (3.41).

$$T_o = R_t - R_a \quad (3.41)$$

This difference exists because each processor is involved in communication and synchronization with other processors while executing in parallel mode. These activities are absent in the sequential algorithm.

3.7.6 Amdahl's Model for Evaluating Performance of Parallel Platform

Amdahl's law further expanded Equation (3.39) as a function of T_s , T_p , f and p . Where f is the fraction of program that cannot be parallelized but must run serially on a single processor, as follows:

$$T_p = T_s * f + T_s * (1-f)/p \quad (3.42)$$

Substituting for T_p gives $S_p = T_s/T_p = T_s / (T_s * f + T_s * (1-f)/p)$

Therefore,

$$S_p = \frac{1}{f + \frac{1-f}{p}} \quad (3.43)$$

Hence, as $p \rightarrow \infty$, $S_p \rightarrow 1/f$ and $\phi \rightarrow 0$. $(1 - f)$ indicates fraction of codes parallelizable.

Thus, Amdahl's law indicates that the sequential fraction of code has much effect on speedup and overall efficiency of the processors application.

3.7.7 Optimization of the Parallel Algorithm

Figure 3.10 shows the process adopted in the tuning of the developed parallel algorithm. Profiling of the algorithm facilitated detailed observation of execution time on each segment, process and thread. The code segments where longer time was spent were reviewed, reversed and optimized.

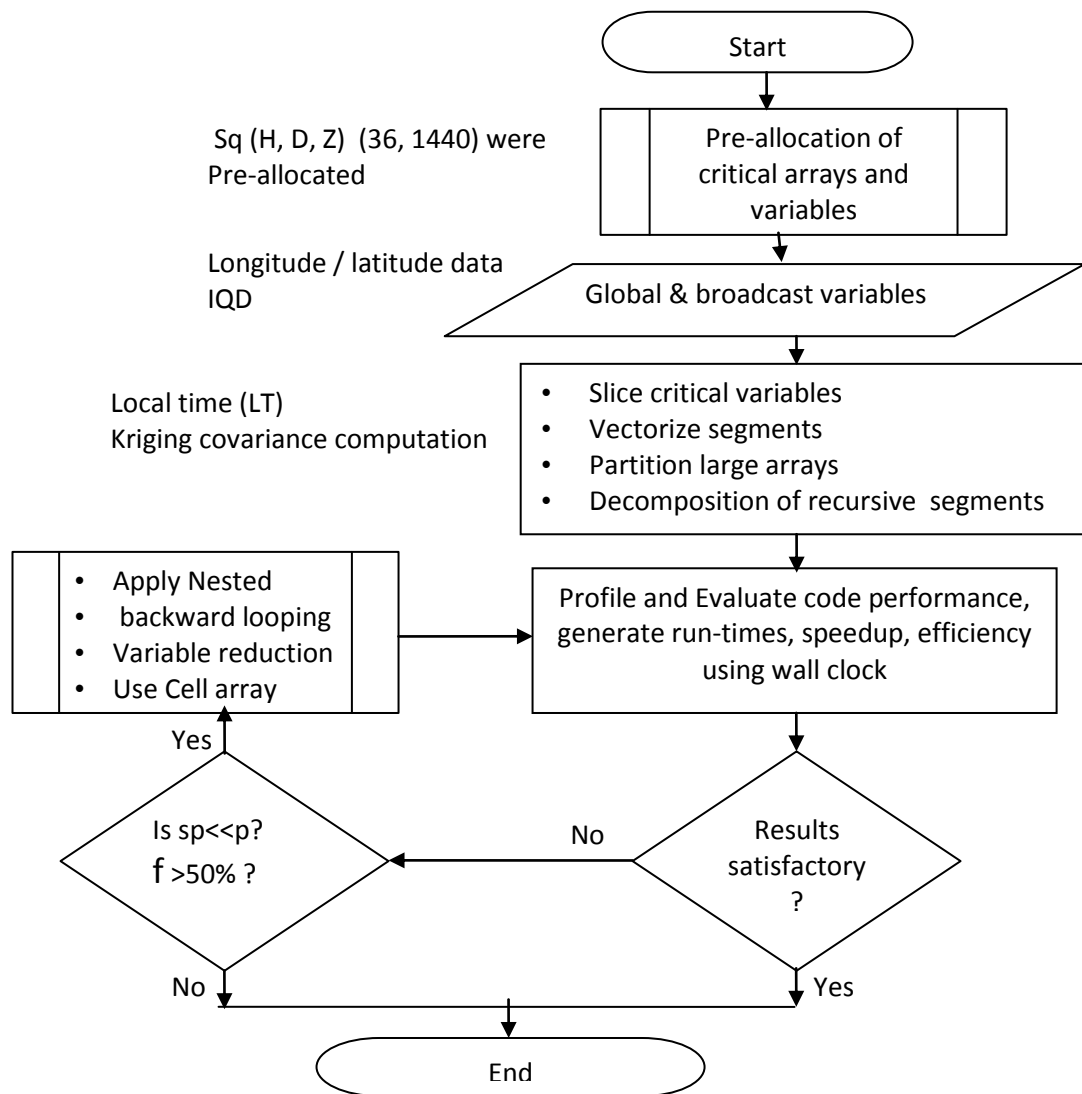


Fig. 3.9: Optimization of the Parallel Algorithm

3.7.8 Approximation of Optimum Processor Allocation

Regression analysis was applied using Least Squares and Pearson's methods for performance prediction of the number of processors that could process the Sq field in less than one minute. This was necessitated by the financial implication of the parallel architecture. The correlation coefficient (R) measures the strength of relationships between observed run-time (t_1) and predicted time (t_2). The coefficient of determination (R^2) measures the percentage of variation in the dependent variable as shown in the following equations.

$$R = \frac{\sum_{n=1}^8 (t_n - \bar{t}_n)(t'_n - \bar{t}'_n)}{\sqrt{\sum_{n=1}^8 (t_n - \bar{t}_n)^2 \sum_{n=1}^8 (t'_n - \bar{t}'_n)^2}} \quad 0 \leq R \leq 1 \quad (3.44)$$

Where t_n = observed runtimes; t'_n = predicted runtimes; n = number of processor used
The relationship between execution times and number of processors was found to be logarithmic in nature as shown in Equation (3.45).

$$t = a + b \cdot \ln(p) \quad (3.45)$$

Finding the sum of the squares of vertical deviations S^2 for $n=8$ processors (p), we have:

$$S^2 = \sum [t_i - f(\ln(p), a_1, a_2, \dots, a_n)]^2 \quad (3.46)$$

The condition for S^2 to be a minimum is that: $\frac{\partial S^2}{\partial a_i} = 0$; for $i = 1, 2, 3, \dots, n$ (3.47)

$$S^2(a, b) = \sum_{i=1}^8 [t_i - (a + b \cdot (\ln p_i))]^2 \quad (3.48)$$

$$\frac{\partial S^2}{\partial a} = -2 \sum_{i=1}^8 [t_i - (a + b \cdot (\ln p_i))] = 0 \quad (3.49)$$

$$\frac{\partial S^2}{\partial b} = -2 \sum_{i=1}^8 [t_i - (a + b \cdot (\ln p_i))] \cdot (\ln p_i) = 0 \quad (3.50)$$

These lead to the equations:

$$na + b \sum_{i=1}^8 \ln(p_i) = \sum_{i=1}^8 t_i \quad (3.51)$$

$$a \sum_{i=1}^8 \ln(p_i) + b \sum_{i=1}^8 \ln(p_i^2) = \sum_{i=1}^8 \ln(p_i) \cdot t_i \quad (3.52)$$

In matrix form, we have

$$\begin{bmatrix} n & \sum_{i=1}^8 \ln(p_i) \\ \sum_{i=1}^8 \ln(p_i) & \sum_{i=1}^8 \ln(p_i^2) \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^8 t_i \\ \sum_{i=1}^8 \ln(p_i) \cdot t_i \end{bmatrix} \quad (3.53)$$

Specifically, the coefficients of Equation (3.53) are:

$$b = \frac{n \sum_{i=1}^8 (\ln(p_i) * t_i) - \sum_{i=1}^8 t_i \sum_{i=1}^8 \ln(p_i)}{n \sum_{i=1}^8 (\ln(p_i))^2 - (\sum_{i=1}^8 \ln(p_i))^2} \quad (3.54)$$

$$a = \frac{\sum_{i=1}^8 t_i - b \sum_{i=1}^8 (\ln(p_i))}{n} \quad (3.55)$$

3.8 The Sq Program Codes Framework

Figure 3.11 shows the m-files developed in Matlab R2010a for simulation of the solar quiet daily variations of geomagnetic field. The *parfor* and *matlabpool* parallel m-files constructs and functions were used to program message passing interface and parallel virtual machine (PVM) for inter-processors communications among the nodes. *Matlabpool* function creates session and also enables clusters management such as nodes initialization, job submission, distribution and collection of intermediate results through the master (head) node.

The user-defined functions developed were able to identify and evaluate the level of concurrency in the solution methodology and also gave insight to a method by which the task could be shared and process among the cluster nodes (workers in Matlab environment). The main program, functions and libraries developed are shown in Figure 3.11. The Matlab m-file codes for the Sq in sequential and parallel versions are included in appendices B.1 and B.2 respectively.

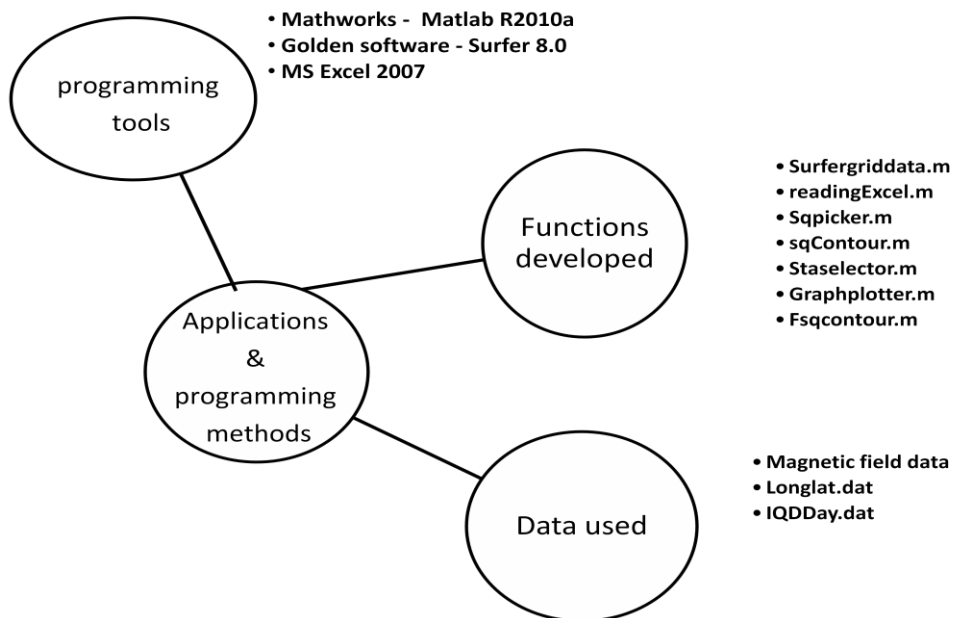


Fig.3.10: Sq (H, D, Z) simulation process and program methodology

The user-functions developed and used for the simulations include the following:

- (i) readingExcel.m - pre-processing of raw data to get rid of or filter out the headers. Matlab cannot directly process the data with the inclusion of headers. The headers give the identification of the data in terms of orientation, station name, the magnetic field components and dates;
- (ii) SqPlanePlot.m - This function imported the computed the {36,1440} Sq results;
- iii) fSqgraph2D – This function generates the Sq results in 2D plots using the stations local time in minutes and fsqcontour provided the 3D contour plot in animation form. Figure 3.12 shows the Sq algorithm modules and functions.

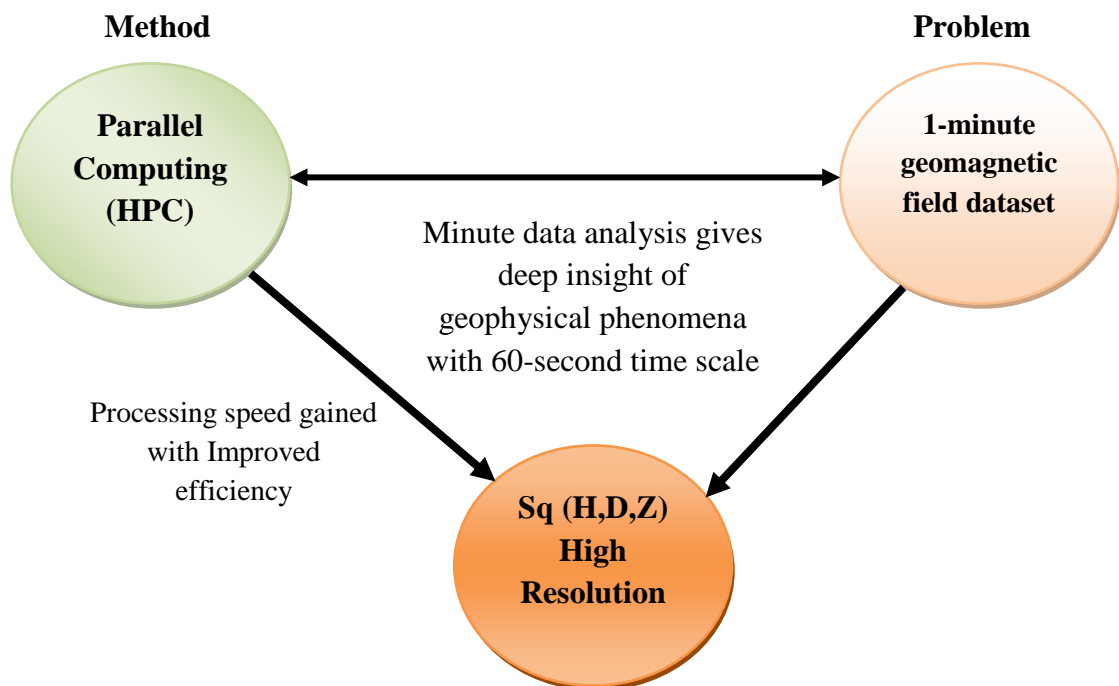


Fig.3.11: A new approach for solving data-intensive geomagnetic data

The serial and parallel versions of the Sq Matlab m-files were implemented on a single CPU and also on eight (8) multi-cores respectively in the following order:

- (i) Sequential Sq Algorithm
 - a. Single CPU
 - b. Multiple CPUs in parallel
- (ii) Parallel Sq Algorithm
 - a. Single CPU
 - b. Multiple CPUs in parallel

CHAPTER FOUR

4.0 RESULTS AND DISCUSSIONS

5.5 Simulations of Sq Variations in 2-D

The Sq variations results of the serial and corresponding parallel algorithms implementation were compared and found the same. This indicated that the parallel version was free of any logical error. In order to compare and interpret conveniently the results of Sq variations in 2-dimensional scale, BNG, KAK and ABK observatories were randomly selected along the low, middle and high latitudes respectively. Figures 4.1 and 4.2 show SqH hourly variations for the month of January in the high and low latitudes at Abisko (ABK) and Bangui (BNG) stations respectively. In addition, Figures 4.3 to 4.14 show 1-minute SqH variations in 2D form for mean monthly values of January to December over BNG, KAK and ABK stations across low, mid and high latitudinal zones. The amplitudes of Sq (nT) variations are shown on the vertical axis while the horizontal axis indicates time (minutes) values.

In comparison, it is observed that geomagnetic spikes and rapid variations are not obvious in the hourly variations, therefore, monitoring of geophysical phenomena with time scale less than one-hour would be impossible. It is also apparent that vital geotechnical information are hidden in the poor hourly resolution of Sq variations. Till date, the hourly variation of worldwide Sq has been the ultimate resolution so far as available in most current literatures of geoscientific field. Some other selected simulation results of SqD and SqZ variations over BNG, KAK and ABK for months of January to December are shown in Appendix C1-C12 and Appendix D1-D12 respectfully. Further interpretation of the details derived from the results of this study will facilitate prediction and early warning alert of any pending disaster or other related geophysical phenomena. The detailed discussion of results is included in subsequent sections.

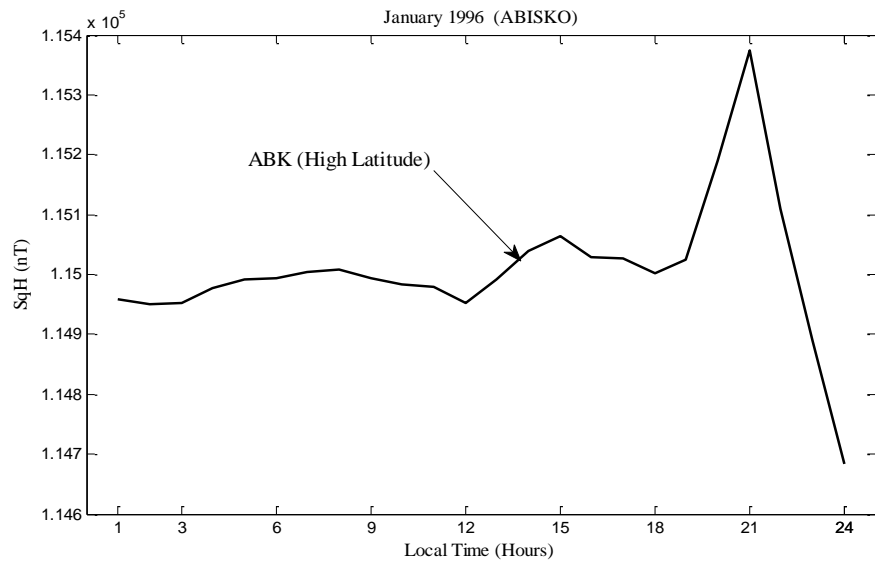


Fig. 4.1: SqH Hourly variation in high latitude – January

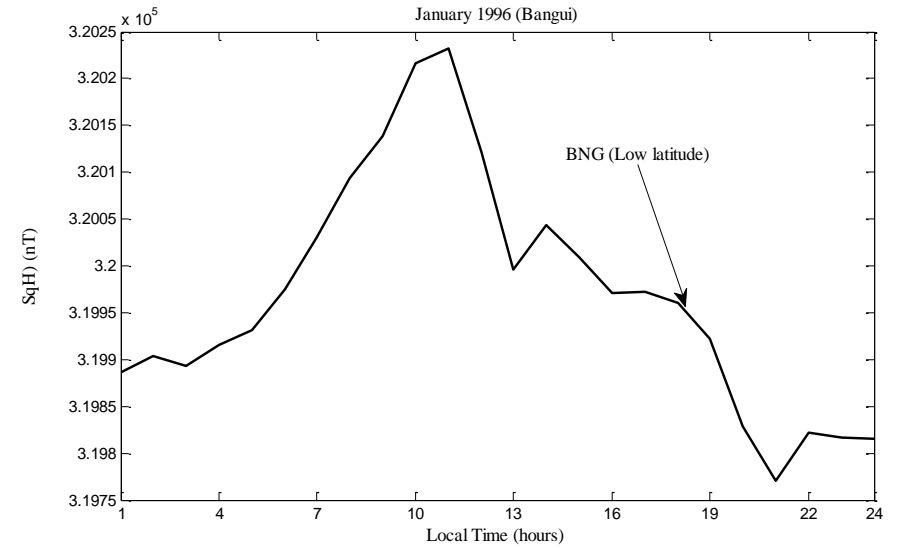


Fig. 4.2: SqH Hourly variation low latitude – January

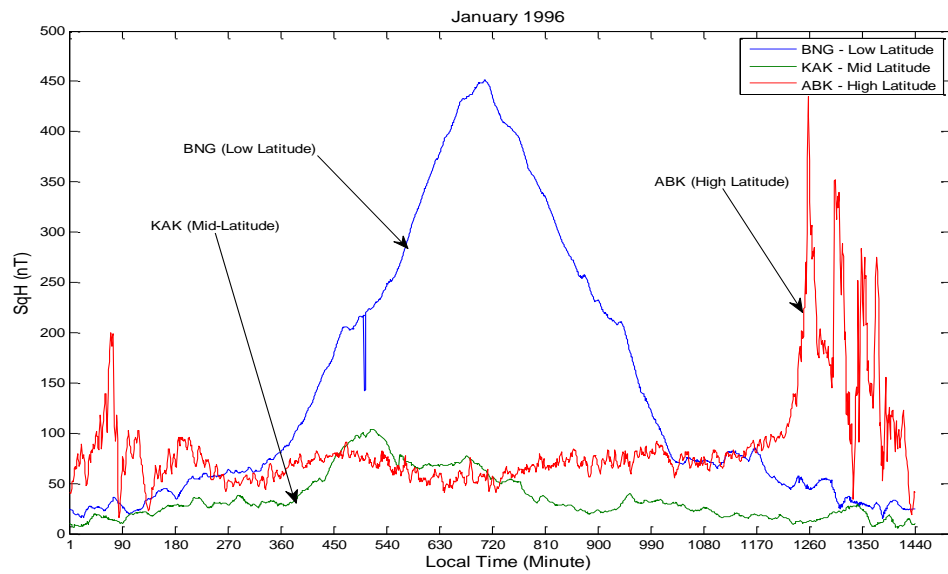


Fig. 4.3: High resolution of SqH across Low-, Mid- and High Latitudes - January

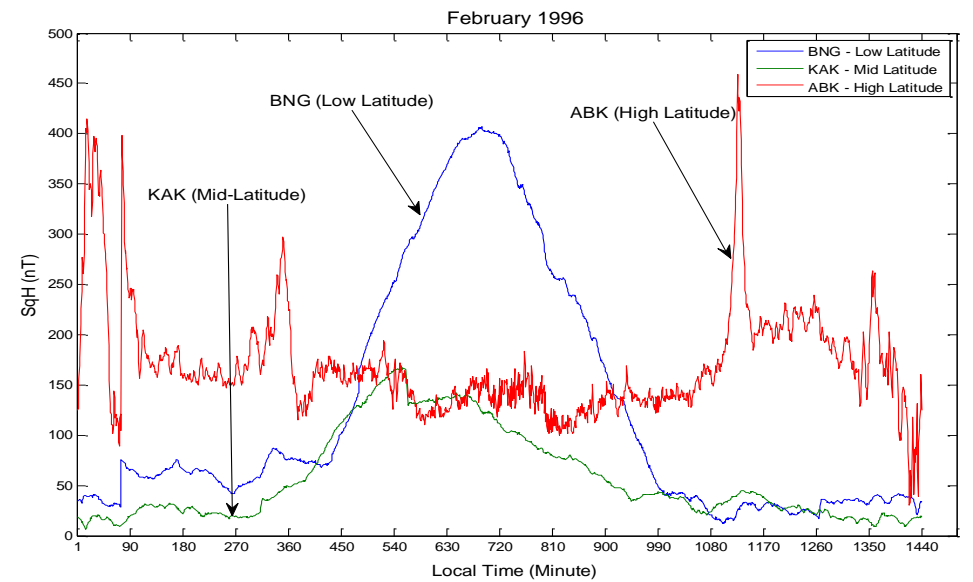


Fig. 4.4: High resolution of SqH across Low-, Mid- and High Latitudes - February

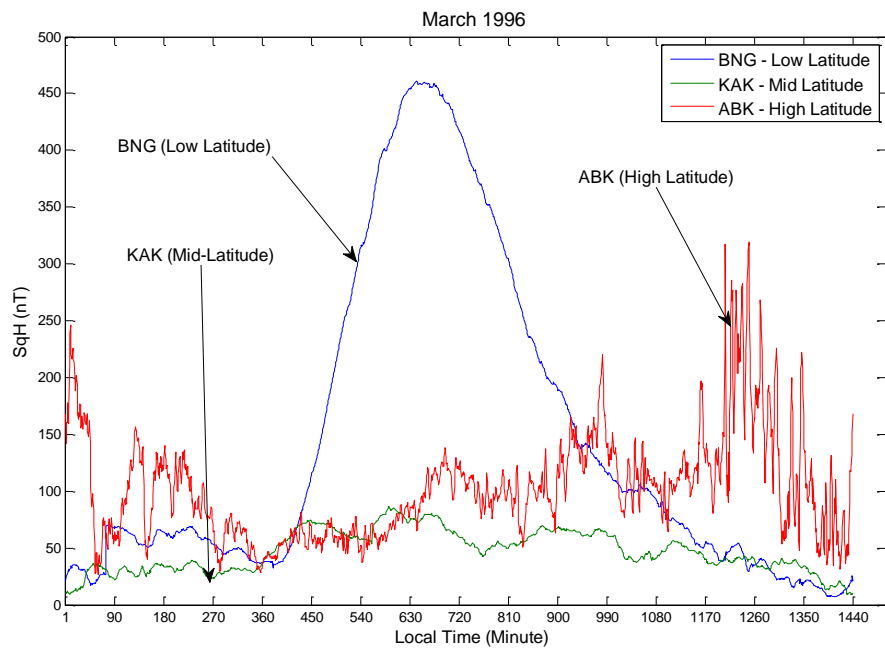


Fig. 4.5: High resolution of SqH across Low-, Mid- and High Latitudes - March

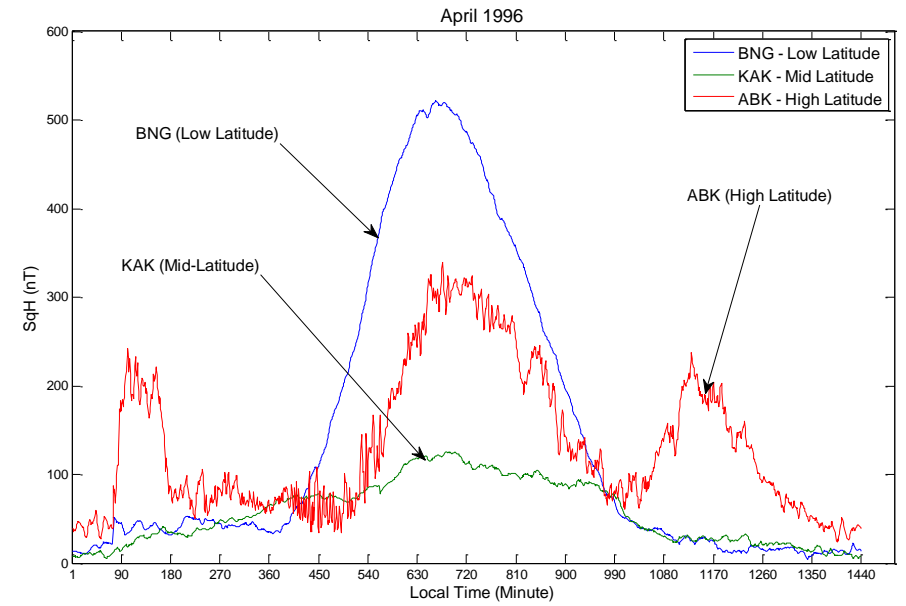


Fig. 4.6: High resolution of SqH across Low-, Mid- and High Latitudes - April

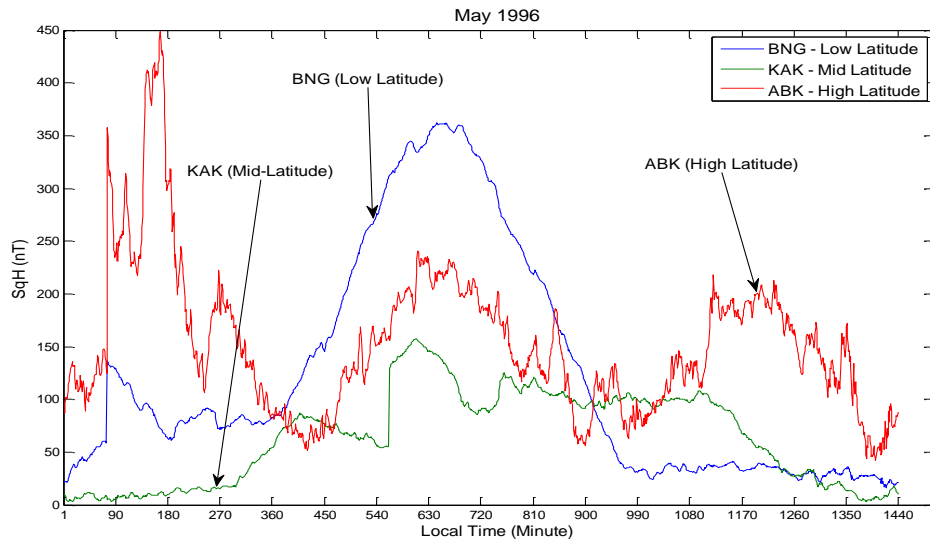


Fig. 4.7: High resolution of SqH across Low-, Mid- and High Latitudes - May

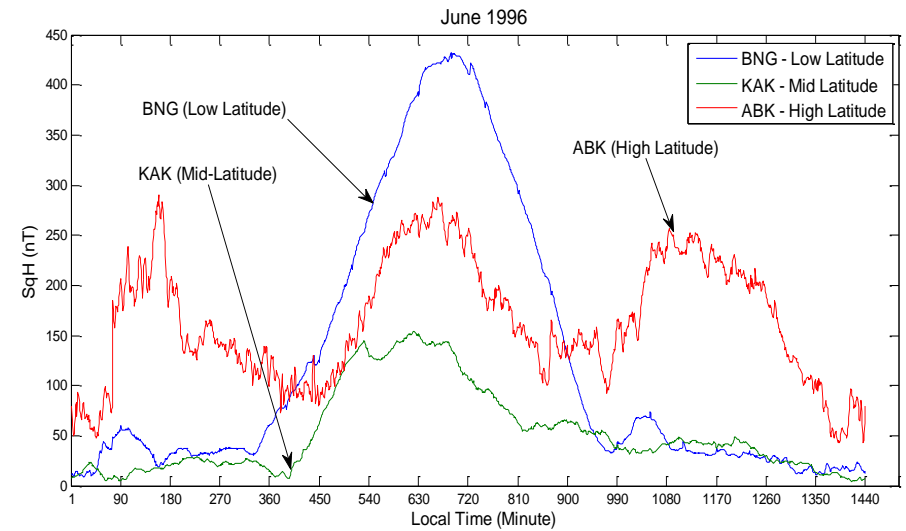


Fig. 4.8: High resolution of SqH across Low-, Mid- and High Latitudes - June

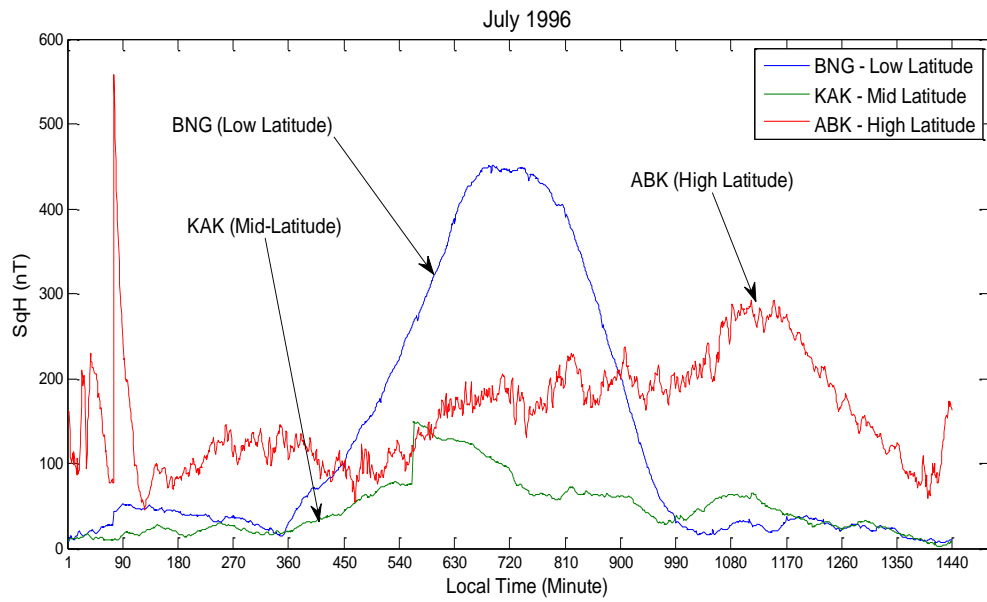


Fig. 4.9: High resolution of SqH across Low-, Mid- and High Latitudes - July

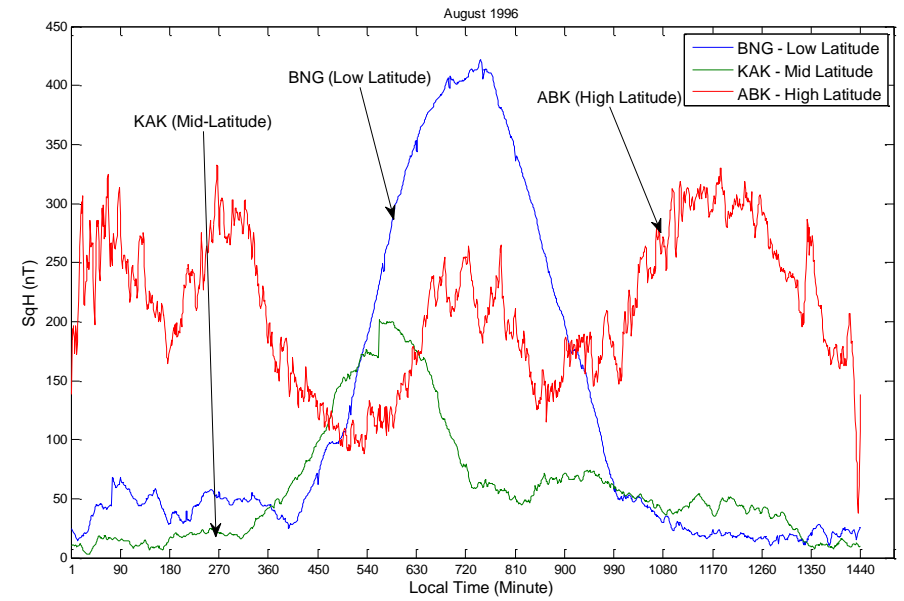


Fig. 4.10: High resolution of SqH across Low-, Mid- and High Latitudes - August

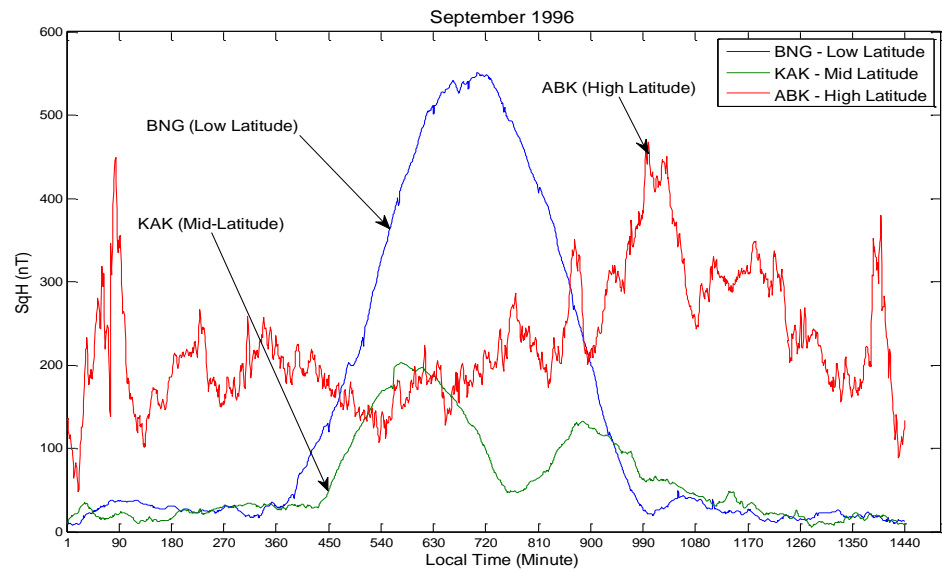


Fig. 4.11: High resolution of SqH across Low-, Mid- and High Latitudes -September

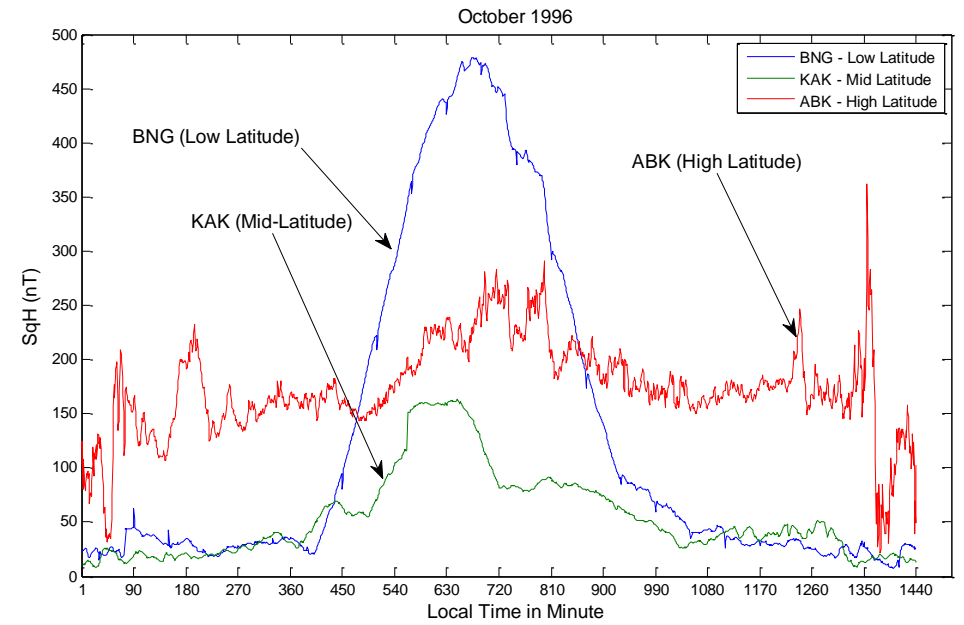


Fig. 4.12: High resolution of SqH across Low-, Mid- and High Latitudes -October

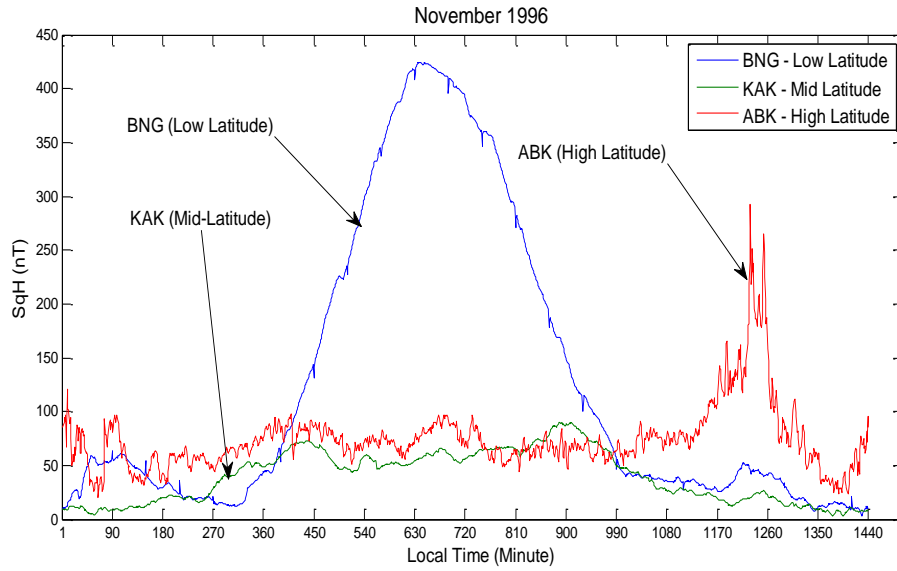


Fig. 4.13: High resolution of SqH across Low-, Mid- and High Latitudes -November

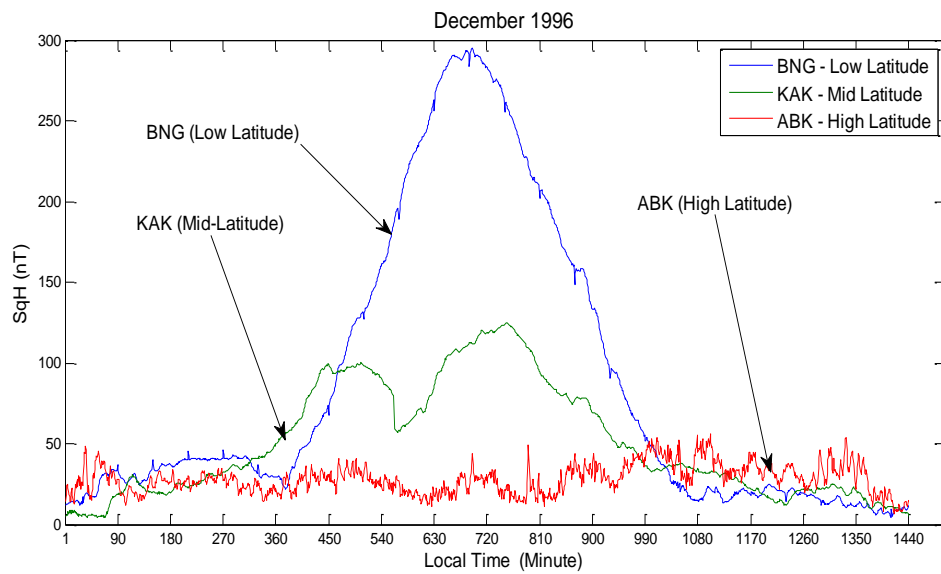


Fig. 4.14: High resolution of SqH across Low-, Mid- and High Latitudes -December

Further details of Sq(H) variations in 2-D form in the ionosphere over various stations enumerated in the study are shown in Figures 4.15 to 4.18. Different representative stations were randomly selected and observed across the low, middle and high latitudes. Figure 4.15 specifically shows the Sq(H) variations in the month of January, February and March over BNG, AMS and ABK stations respectively. It is observed in January that BNG station (4.44°), being close to the equator has more of solar radiation due to intense exposure to Sun. The curve soars higher at noon (720 minutes = 12 local noon) when the Sun is directly overhead at the BNG station, however the Sq(H) curve reduces towards night time when the station loses focus with Sun. At the high latitude, the curve of ABK (Abisko, Sweeden – 68.36°) station, being close to polar region, has geomagnetic spikes and dwindles as a result of limited exposure to Sun especially in January. Station AMS which is located in the southern hemisphere in latitude (-37.83°), has its curve nose-dives at noon (720 min) due to reduced solar activities, and its behavior is similar to KAK when observed in March.

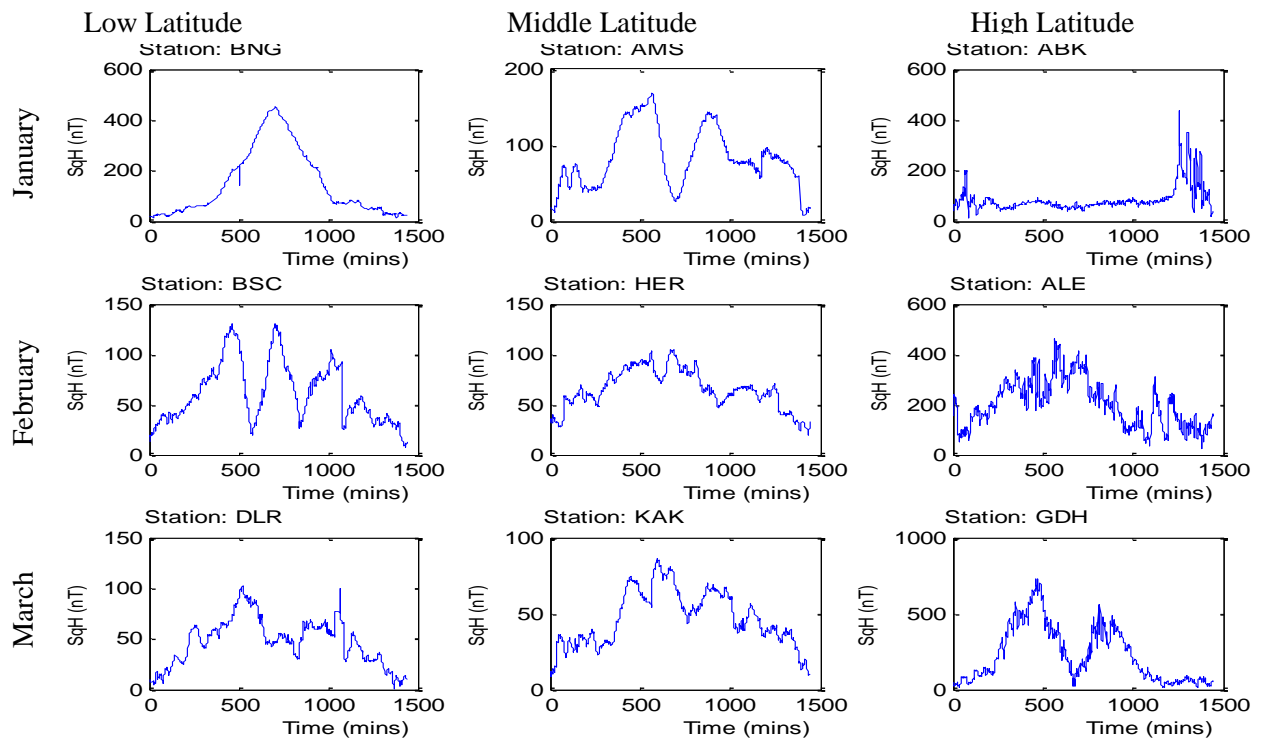


Fig.4.15: Sq(H) variations in January, February and March along the latitudes

Figure 4.16 shows the graphical representations of Sq(H) in 2D for the months of April, May and June along the low, middle, and high latitudes. The results further showed some unique relationship between latitudinal amplitudes and months of the year. At the low latitude, the amplitude of Sq(H) drops from about 400 nT in the first quarter of the year to about 300 nT in second quarter of the year while the values were averagely constant in the high latitude.

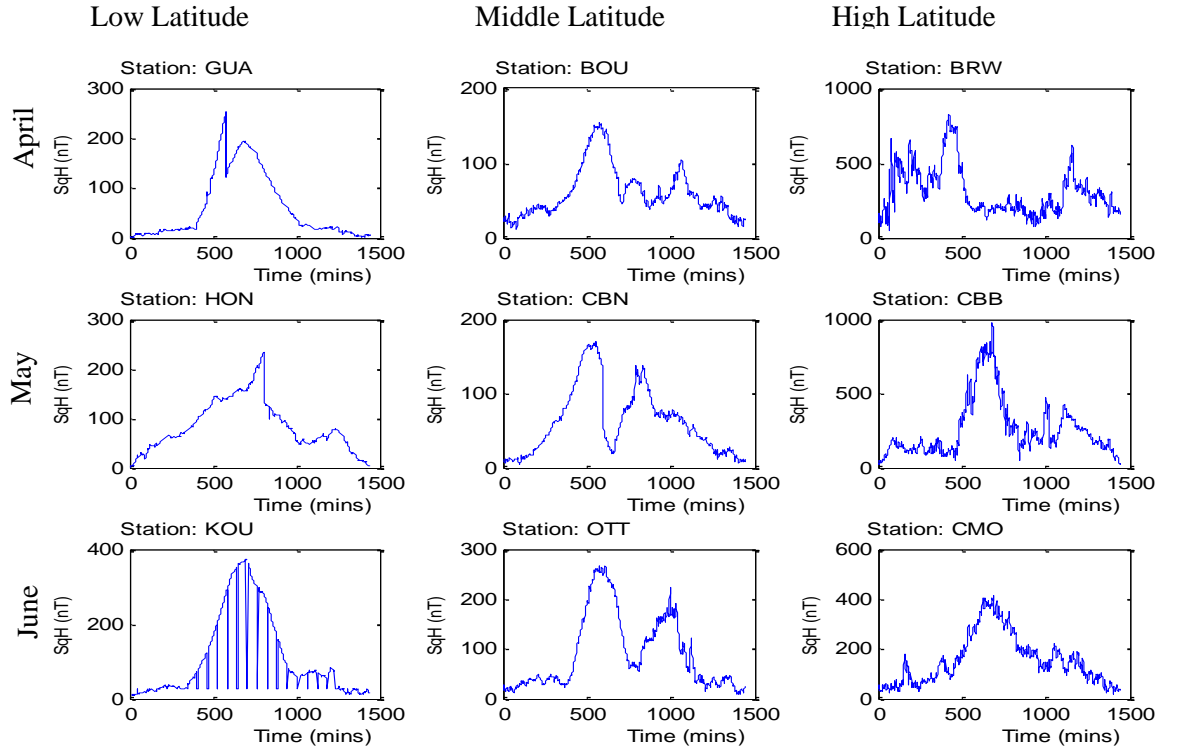


Figure 4.16: Sq(H) variations in April, May and June along the latitudes

It was observed in Figure 4.17 that the Sq(H) amplitude at local time was higher over the stations along the low, mid and high latitude in July, August and September.

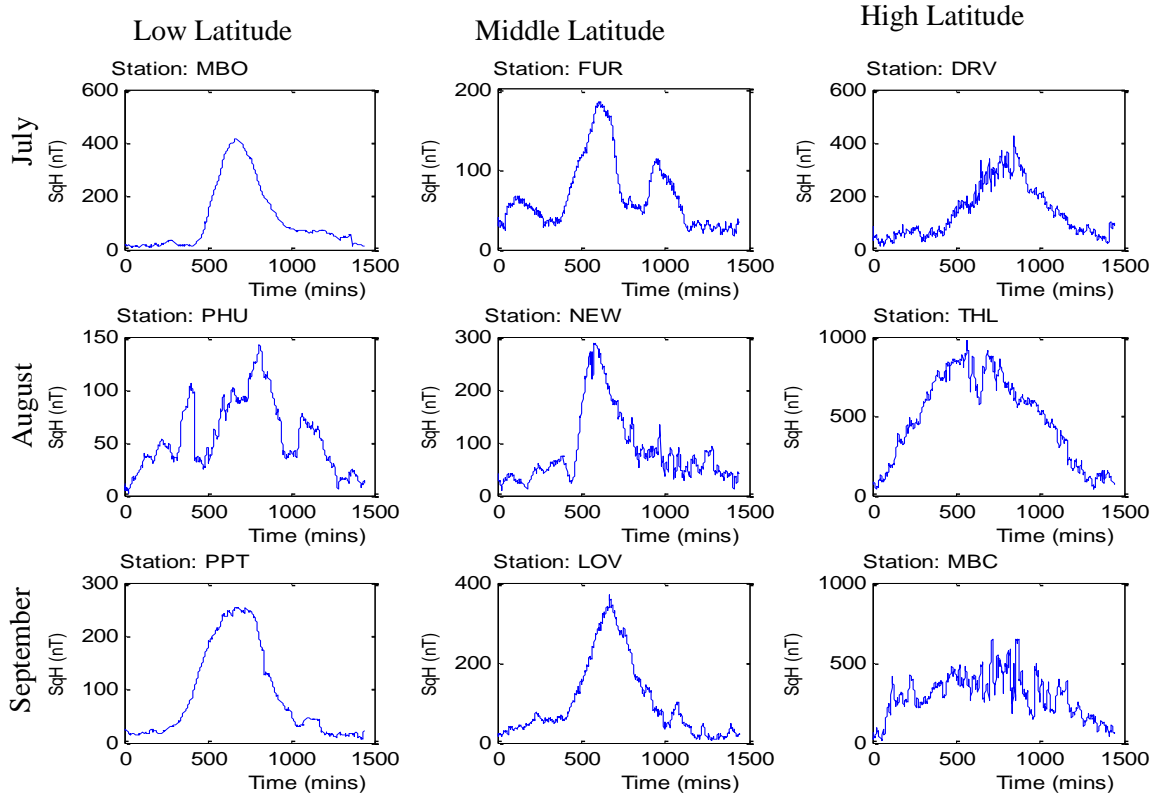


Fig.4.17: Sq(H) variations in July, August and September along the latitudes

Figure 4.18 shows the Sq(H) variations along the low, middle and high latitudes over some selected stations in the last quarter of the year 1996. It was evident that solar activities vary with respect to solar intensities and other heliophysical phenomena such as cloud cover. At the stations local noon time, the sun would always be overhead thereby having more intensity on the area compared with other minutes of the day as well as in the average over months.

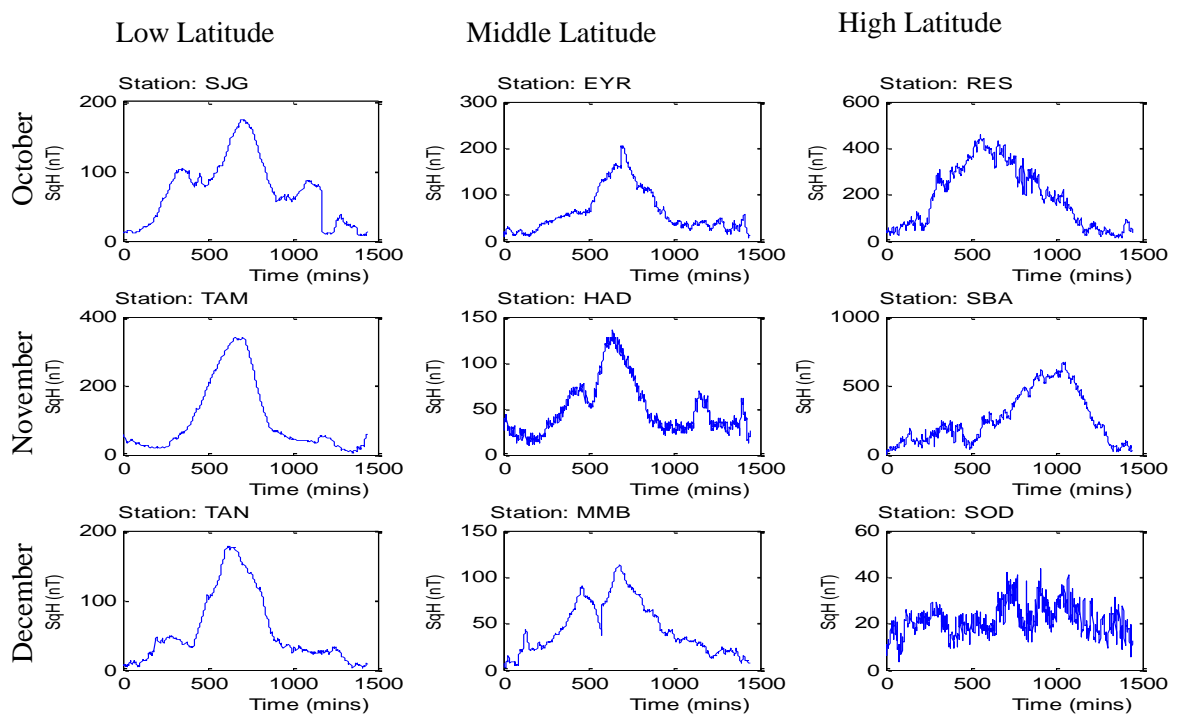


Fig.4.18: Sq(H) variations in October, November and December along the latitudes

5.6 Simulations of Sq Variations in 3-D

Figures 4.19a, 4.19b and 4.19c show results of worldwide Sq(H) 3-D contour in a dynamic variation for the months of January to December. This 3-D format gives global visualization of Sq variations along latitude and longitude with respect to time (minutes). The three dimensional components are longitude, latitude and time.

Four different snapshots of the Sq simulations were taken at midnight (exact 1st minute), 360th minutes (6 am), 720th minute (12noon) and 1080th minute (6pm) local time over each

station. The amplitudes of Sq(H) variations are indicated with different colour codes as shown in the colourbar legend. The deep blue colour shows the lowest, followed by light blue, yellow colour shows average values and deep red shows high values. In the midnight (1st minute) of March, the value of Sq(H) was high in the high latitude as shown in Alert (ALE, Canada), Abisko (ABK, Sweeden) stations among others. These stations are located at the Polar Regions and do have Sunlight even in the night (midnight sun) for almost half of the year. However, the presence of Sq(H) focus near polar region in mid-night could be attributed to non-ionospheric sources. These results have further revealed, through critical observations that the Sq variations along H-component changes in a systematic manner with respect to local time, months and seasons along the latitudes.

Subsequently, the graphical user interface was developed to visualize the dynamic representation of Sq current system (variations) in H, D, Z respectively as shown in Figure 4.20. The GUI platform was designed to be a web application and deployable at the future Space Systems Decision Support Service Centre (SSDSSC).

The developed 3-D SqH, SqD, SqZ codes and several other function calls were compiled in Matlab 2010a environment. The generated Sq application as well as the Users' guide and this thesis in portable document format are included in the attached CD. The Sq application can run outside Matlab environment on any version of Microsoft Windows Operating System.

Mean Monthly variation

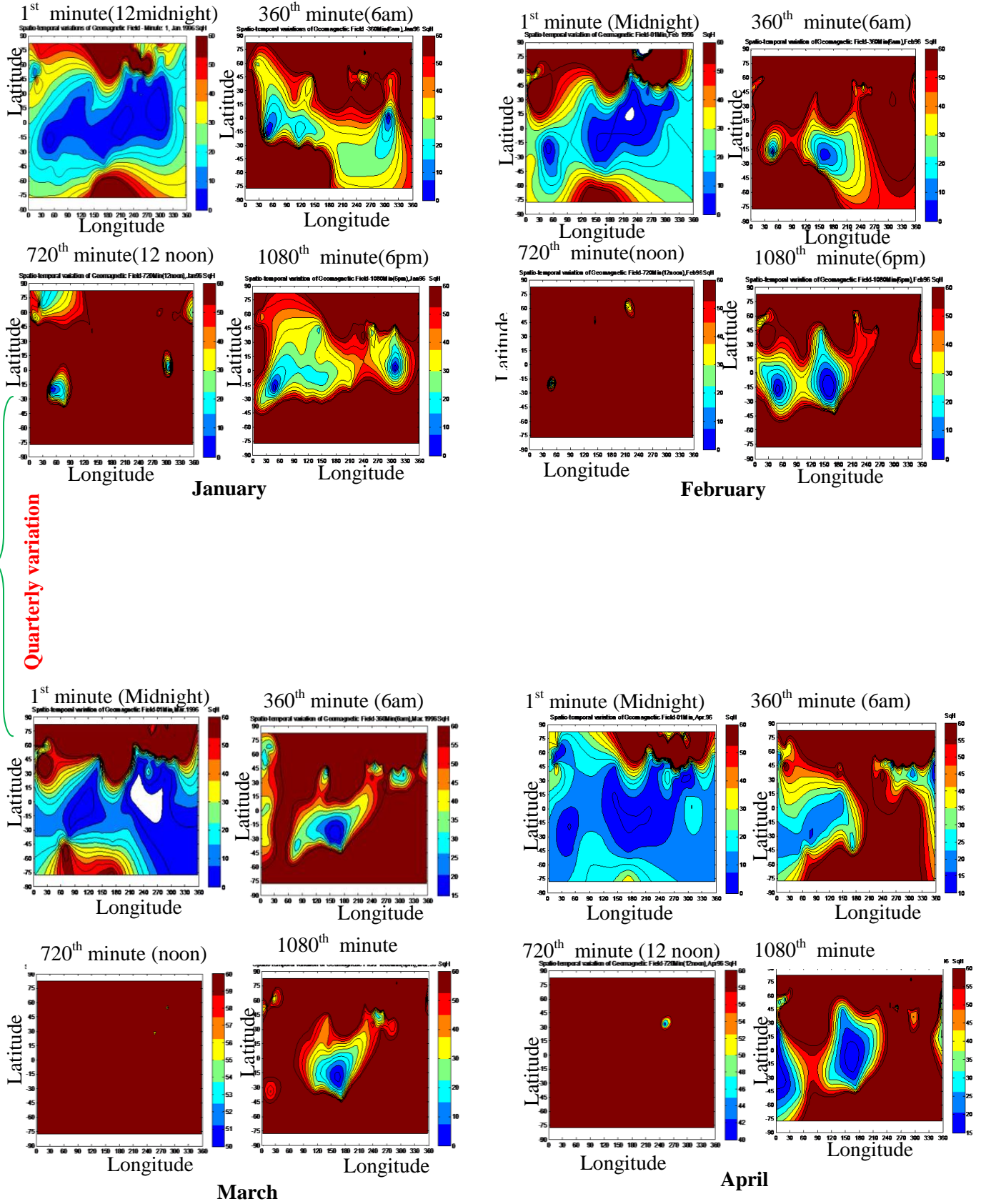


Fig. 4.19a: Dynamic SqH Variations Contour Map (Jan – Apr)

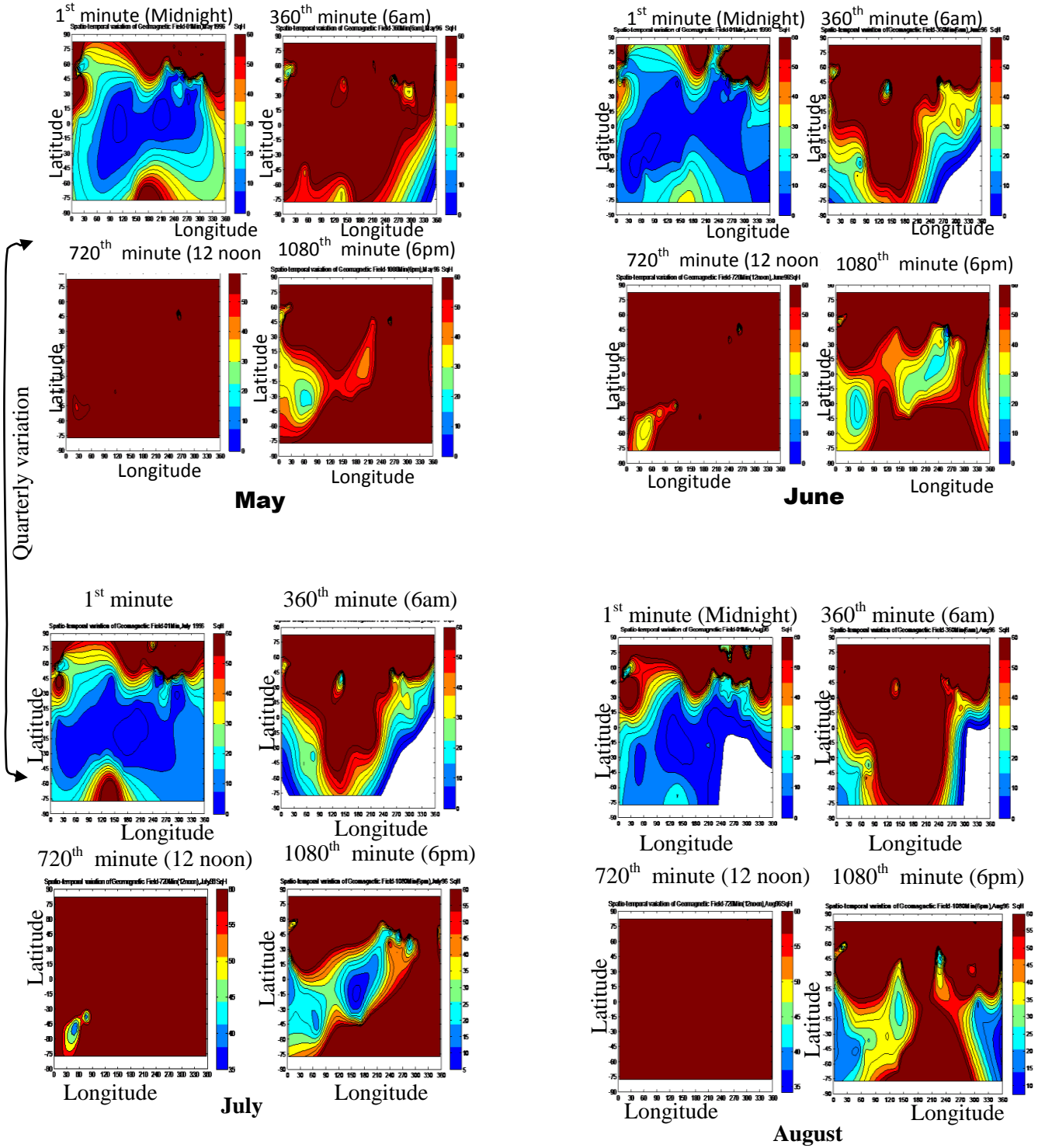


Fig. 4.19b: Dynamic SqH Variations Contour Map (May – Aug)

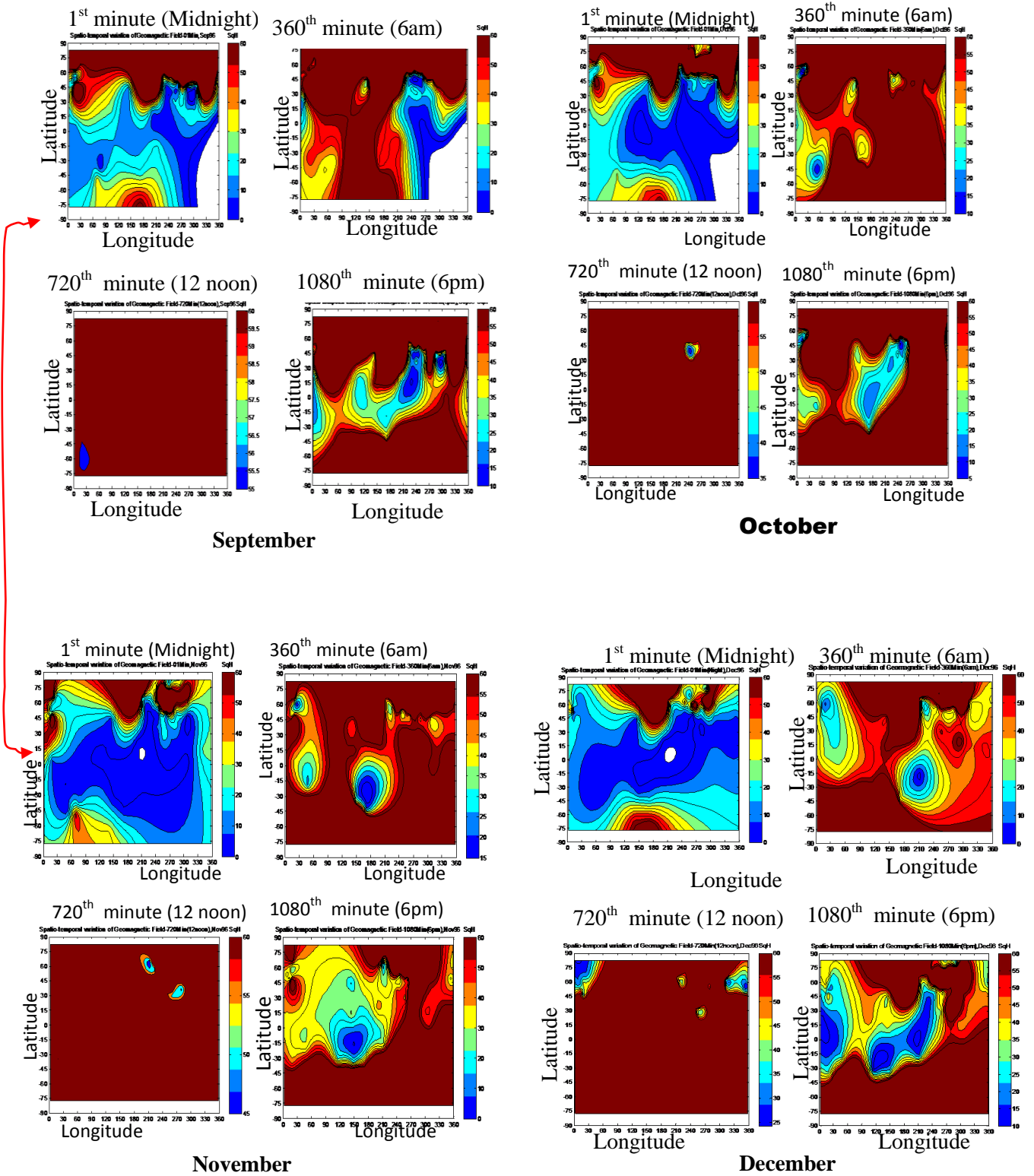


Fig. 4.19c: Dynamic SqH Variations Contour Map (Sep – Dec)

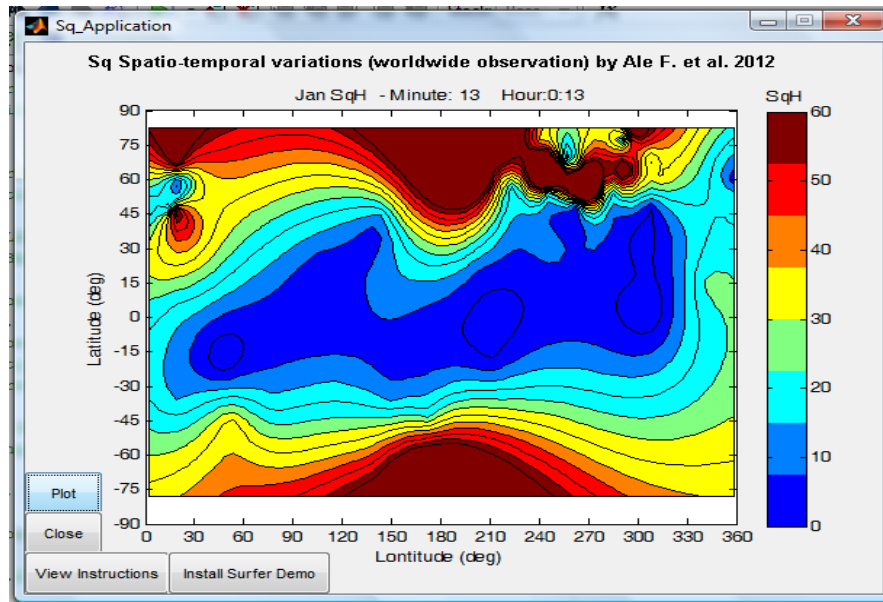


Fig. 4.20: GUI Dynamic Sq Variations Platform

5.7 Performance Analysis and Evaluation of the Sq Algorithm

Table 4.1 shows the details of the execution time and the benchmark metrics of the developed parallel computational platform for solar quiet daily variation of geomagnetic field.

5.7.1 Execution Time, Speedup and Efficiency Metrics

Figure 4.21 shows that the developed sequential Sq program took 18.5 minutes to run on one processor in a uni-processor platform while the parallel version took 13.4 minutes on one processor in a parallel platform, thus T_s was not equal to T_1 . Table 4.1 shows the details of the total runtimes, computation-specific runtimes and overhead time. Figures 4.22, 4.23, 4.24 and 4.25 show graphical representations of the performance metrics with respect to number of processors used to do the job.

Table 4.1: Benchmark Metrics for Performance Analysis of Parallel Sq Simulation

No of CPU (P)	Execution Time(s) (T_p)	Execution Time(min) (T_p)	Computation Time (s)	Overhead Time (t'')	Speedup (S_p)	Efficiency (%) (ϕ)
1	801.549	13.35915	791.347	10.202	1	100
2	409.737	6.82895	402.514	7.223	1.956	97.813
3	302.236	5.037	294.904	7.332	2.652	88.402
4	242.909	4.048	235.39	7.519	3.300	82.495
5	218.074	3.635	210.46	7.614	3.676	73.512
6	208.901	3.482	201.054	7.847	3.837	63.950
7	192.848	3.214	184.768	8.08	4.156	59.377
8	185.517	3.092	177.03	8.487	4.320	54.008

The speedup values of processors 1 – 4 are close to the number of the cores while processors 5-8 have speedup values far away from the number of the cores. This indicates that the use of processors 1-4 were more optimized.

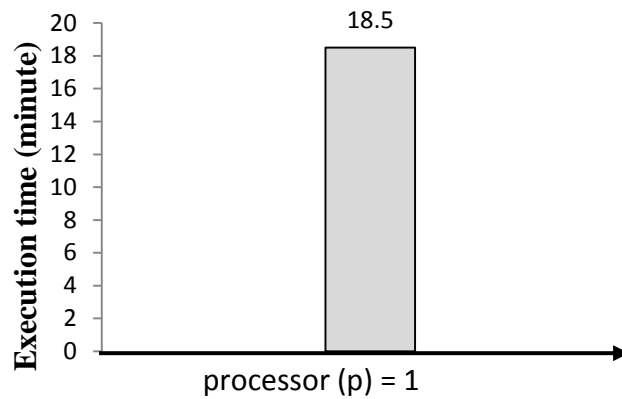


Fig. 4.21: Execution time for Sequential Algorithm on one Processor

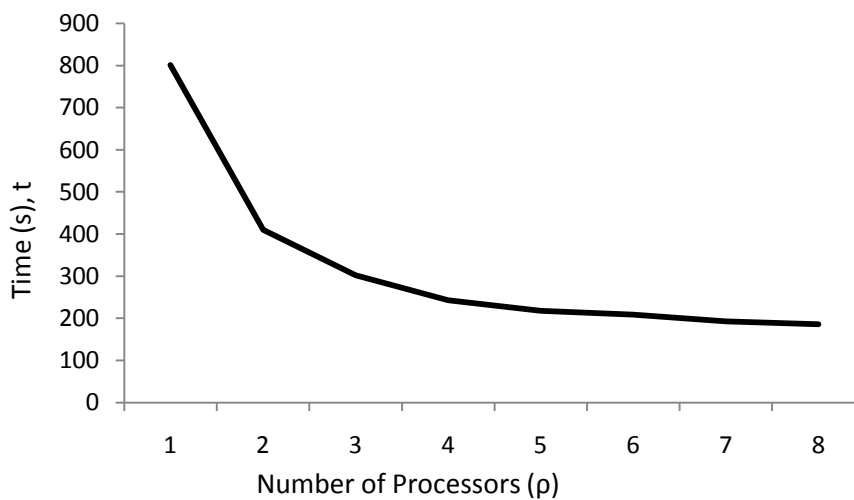


Fig. 4.22: Execution time versus number of Processors

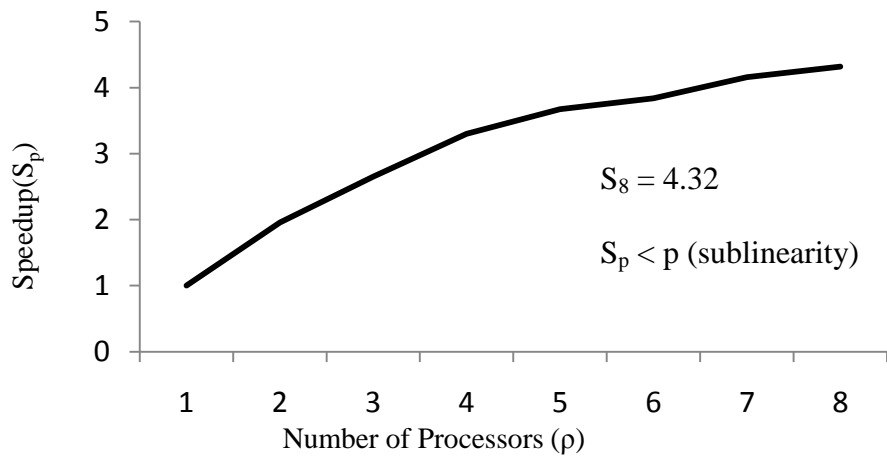


Fig. 4.23: Speedup versus number of processors

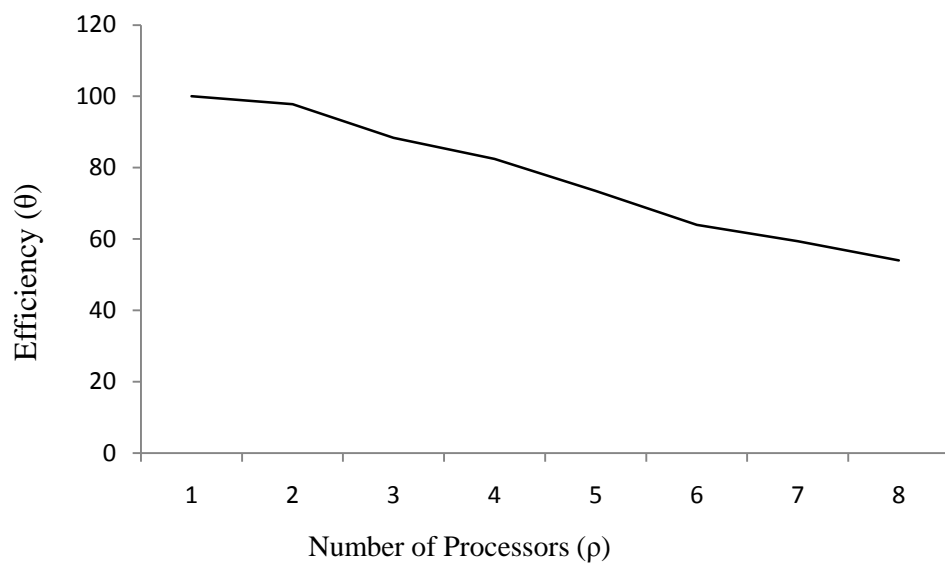


Fig. 4.24: Efficiency of the parallel algorithm

5.7.2 Parallel Overhead

It was observed that the parallel algorithm made use of processor-time that could have been spent solving the Sq computation. This defines the parallel overhead, T_o , as the difference between the total execution time (R_t) and the time spent (R_a) to process the actual job. Their relationship is given as follows:

$$T_o = R_t - R_a$$

This happened because the processors were involved in communication and synchronization with other processors while executing the parallel algorithm, none of which were done by the serial algorithm. The values of T_o are shown in Table 4.1 and represented graphically in Figure 4.25.

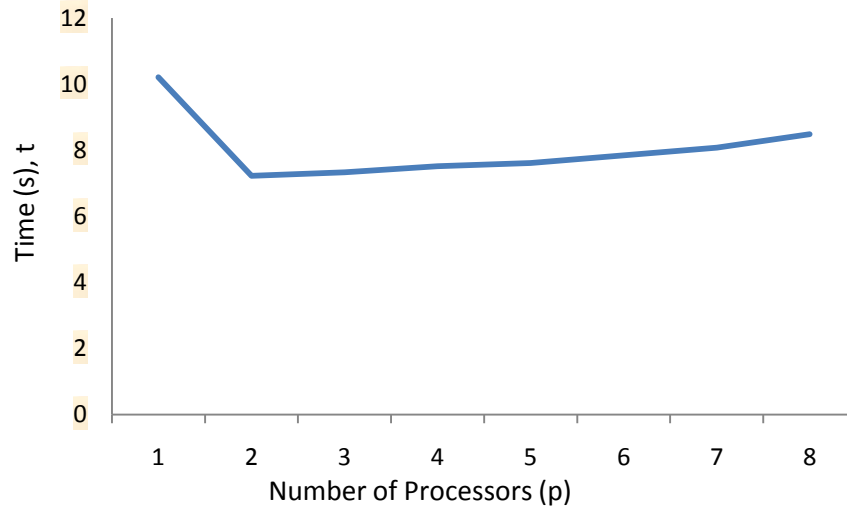


Fig. 4.25: Execution time on parallel overhead

The following factors were observed to be responsible for presence of overhead in the parallel computations:

- i) Time to break the problem to tasks and synchronizations;
- ii) Data communications (message passing);
- iii) Overhead imposed by compilers, libraries, tools and the operating system.
- iv) The total network transfer capacity of the cluster ;
- v) Floating point operations per second (FLOPS) within the cluster
- vi) Latency and bandwidth between CPUs and memory

5.8 Validation of the Parallel Computational Platform using Amdahl's Model

In Table 4.1, the speedup of Sq computation using eight processors ($p=8$) equals 4.32. Therefore, application of the Amdahl's model by substitution shows that the portion of the code that could not be parallelized equals 12%. Thus, the speedup metric of the parallel algorithm exhibits sub-linearity, since the speedup values are less than the number of processors, that is, $S_p < p$. This also indicates scalability and efficiency as the number of resources and size of the problems were varied.

5.9 Improvement on Amdahl's Model for Better Performance

We observed that the execution times of the parallel algorithm decreased as the number of processors increased, so this relationship could be expressed as:

$$T_p \propto \frac{1}{p} \quad \text{and} \quad T_p = f(v + \gamma) \quad (4.1)$$

Where v = workload (actual computation) or size of the problem and γ = parallel overhead

However, it was observed that $T_s \neq T_1$, if we let $\psi = T_s - T_1$.

Therefore, $T_s \rightarrow T_1$ as $\psi \rightarrow 0$. Where ψ = overhead for parallel constructs.

Where ψ is the value that determined how efficiently the algorithm could be parallelized in terms of programming structure, data structure, memory usage, latency among others. The efficiency curve in Figure 4.24 shows a negative gradient as the efficiency reduces with increased number of processors. Therefore, the metric measures the effectiveness of parallel algorithm with respect to the computation time. Thus, the efficiency (ϕ) is a function of load-balance (θ), concurrency (τ), and parallel overhead (γ) parameters as expressed in Equation (4.2). For all these parameters to hold, multiples threads or processes (U) must exist in any program for a parallel computation to be true as represented in Equation 4.2.

$$\phi = f(\theta, \tau, \gamma) \quad \text{and} \quad \tau = \sum_{i=1}^n (U_i) \quad (4.2)$$

Load-balance (θ) was implemented by distributing sub-tasks (n) to n -processors. Table 4.1 shows that the results of efficiency are between 0 and 1 ($0\% \leq \phi \leq 100\%$). Efficiency of the parallel algorithm was observed to be directly proportional to load-balance and concurrency, and at the same time indirectly proportional to parallel overhead. Therefore, Equations (4.3),

(4.4) and (4.5) show the mathematical relationships that exist among the parallel computing performance parameters, as follows:

That is,

$$\varphi \propto \frac{\theta\tau}{\gamma} \quad \text{Then} \quad \varphi = \frac{g\theta\tau}{\gamma} \quad (4.3)$$

$$\text{Therefore} \quad g = \frac{\varphi\gamma}{\theta\tau} \quad (4.4)$$

As a result,

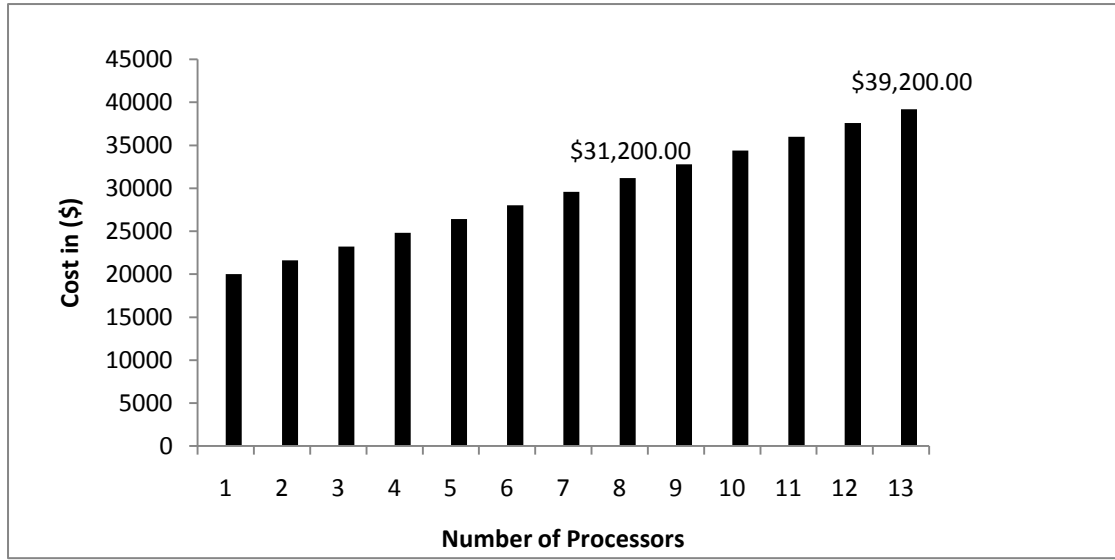
$$S_p = \frac{1}{f + (1-f)/p} + \sum_{i=1}^n \frac{1}{g^i} \quad (4.5)$$

Where n = possible number of processes/threads in the parallel algorithm, g = code tuning factor.

Therefore, Amdahl's equation was modified and improved so as to account for parallel overhead impact. The parameter (g) will therefore improve the overall performance gain and reliability of the parallel computational platform as shown in Equation 4.5. However, more studies are recommended for further empirical validation using other computational-intensive problems.

5.10 Performance Prediction using Least Squares and Pearson Method

In this section, we seek to extrapolate the execution times of the algorithm so as to estimate the optimal number of processors that could solve the problem in fraction of seconds for a near real-time Sq processing and prediction. In addition to challenges of limited expertise in parallel algorithms development, high financial cost of computing hardware especially in terms of processing units in multicore platform are major concern. Figure 4.26 shows the financial cost of Intel 2.33 GHz processors in multi-processor and multi-core technologies. It is therefore reasonable to predict scientifically using the initial empirical and experimental data obtained in this research.



Source: www.dell.com; 2011

Fig. 4.26: Baseline Cost (\$) for Parallel processors

The experimental result of execution times (t) with the number of processors (p) allocation in the parallel computational platform shown in Table 4.1 were further analyzed using regression analysis and least squares method as expressed in Equations (3.45) to (3.55). The graph plot of (p , t) in Figure 4.27 shows a logarithmic curve which modeled a best-fit curved line to display the data values that decrease quickly before leveling out. The mathematical relationship is logarithmic in nature as shown in Equation (4.6).

$$t = \alpha_1 \ln(p) + \alpha_2 \quad (4.6)$$

Where α_1 and α_2 are constants that approximately solve the over-determined non-linear system in form of (p_k , t_k) where $k = 1, 2, 3, \dots, 8$. This generates eight equations with two unknowns. The least squares approach to solving this problem is to minimize the mean squared errors.

(p_k , t_k) = (1, 801.55), (2, 409.74), (3, 302.24), (4, 242.91), (5, 218.07), (6, 208.90), (7, 192.85), (8, 185.52)

Therefore, by substituting p_k and t_k , into equations (3.45-3.55), gives:

$$S(\alpha_1, \alpha_2) = [801.55 - (\alpha_1 \ln(1) + \alpha_2)]^2 + [409.74 - (\alpha_1 \ln(2) + \alpha_2)]^2 + [302.24 - (\alpha_1 \ln(3) + \alpha_2)]^2 + [242.91 - (\alpha_1 \ln(4) + \alpha_2)]^2 + [218.10 - (\alpha_1 \ln(5) + \alpha_2)]^2 + [208.90 - (\alpha_1 \ln(6) + \alpha_2)]^2 + [192.85 - (\alpha_1 \ln(7) + \alpha_2)]^2 + [185.52 - (\alpha_1 \ln(8) + \alpha_2)]^2$$

The minimum is determined by obtaining the partial derivatives of $S(\alpha_1, \alpha_2)$ with respect to α_1 and α_2 and setting them to zero. This results in a normal system of two equations in two unknowns, which results in:

$$\begin{aligned}\alpha_1 &= -276.2, & \alpha_2 &= 686.4 \\ t &= -276.2 \cdot \ln(p) + 686.4\end{aligned}\quad (4.7)$$

The logarithmic function was used to generate best fit approximation in a second order polynomial as shown in Table 4.2 and Figure 4.26.

Table 4.2: Optimization Analysis of the Parallel algorithm

P	Execution Time(t)	Speedup	Efficiency (%)	Logarithmic Model	2 nd order Polynomial (t')
1	801.55	1.00	100.0	686.40	594.32
2	409.74	1.96	97.81	494.95	502.09
3	302.24	2.65	88.40	382.96	417.79
4	242.91	3.30	82.49	303.51	341.40
5	218.07	3.68	73.51	241.87	272.95
6	208.90	3.84	63.95	191.52	212.42
7	192.85	4.16	59.38	148.94	159.82
8	185.52	4.32	54.01	112.06	115.14
9	unobserved			79.53	78.39
10	unobserved			50.43	49.56
11	unobserved			24.10	28.66
12	unobserved			0.07	15.68
13	unobserved			22.04	10.63
14	unobserved			42.51	13.51
15	unobserved			61.56	24.31

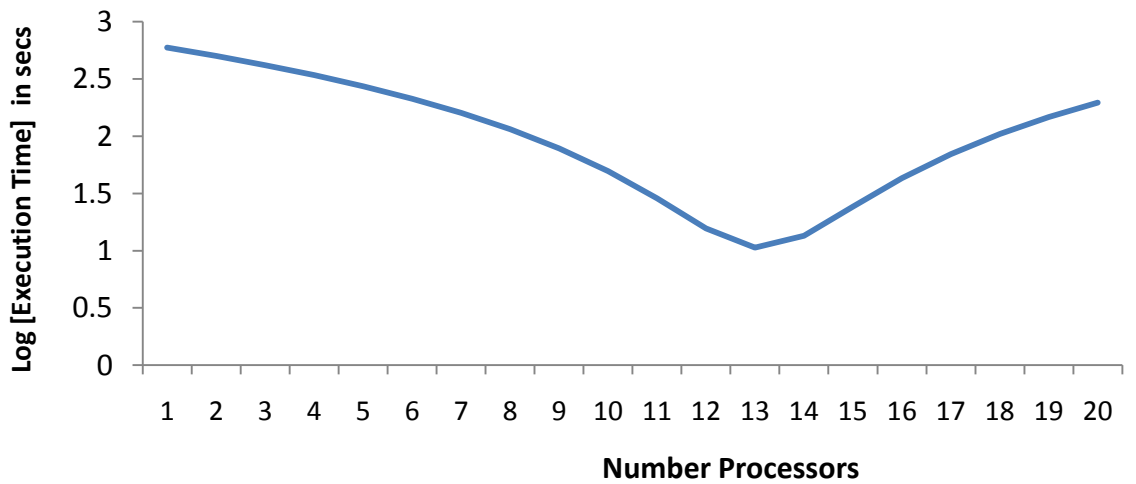


Fig. 4.27: Determination of optimum number of processors for accurate prediction

From table 4.2, t' is the derived execution time (in seconds) from second order polynomial as represented mathematically in equation (4.8).

$$t' = 3.9628p^2 - 104.12p + 694.48 \quad (4.8)$$

To obtain the optimum number of processors (p), we consider the derivative of the absolute values of the model which gives,

$$t' = \frac{d}{dp}(3.9628p^2 - 104.12p + 694.48) \quad (4.9)$$

This simplifies to:

$$7.9256p - 104.12 = 0$$

Thus, the second order polynomial above describes the optimized processors usage and execution time. Hence, the maximum number of processors useful for this computation would be given as the value of (p) when the first derivative of the quadratic vanished. This gives:

$$\frac{dt'}{dp} = 0, p = 13 \quad (4.10)$$

Equation (4.10) shows that the algorithm requires 13 processors to execute this problem on the same computational platform in 10.63 seconds. Thus, parallel slowdown occurs (Figure 4.27) when at values, p , where the first derivative is greater than zero.

$$\frac{dt'}{dp} \geq 0, p \geq 13 \quad (4.11)$$

This will lead to diminishing returns. The new model has coefficient of determination of factor 0.8717 using equation (3.44). This, therefore, indicates a strong correlation.

5.11 Detailed Discussion of the Results

In this work, sequential and parallel algorithms for analysis and simulation of worldwide Sq variations using one-minute sampled geomagnetic dataset were successfully developed. The developed parallel algorithm was implemented on heterogeneous (hybrid) multicore parallel computing platform using eight processors. The 2D results clearly show that SqH field has highest but varying level of amplitude at the low latitude because the sun shines overhead on the zone leading to intense solar radiation. Consequently, this effect generates concentrated ionization trough along the equatorial region also known as equatorial anomaly which significantly affect radio waves especially at the local noon time. The mid-latitude shows lowest but different level of SqH amplitudes because the Sun shines at an angle, so the solar radiation are more spread. Since frequency has a linear relationship with geomagnetic intensity, then ULF waves can be observed more along the mid-latitude. The ULF waves have been confirmed as a precursor to earthquakes which are more pronounced in the mid latitude. It was also observed that SqH has erratic manifestation that is characterized by geomagnetic spikes, amplitude and rapid variations on 60-second basis in the high latitude zone. However, half of the year (April-September) experienced lower but varying amplitude while the SqH increased and maximized averagely at the local noon in the rest half of the year. This is due to irregularities of solar exposure which account for geomagnetic storms and aurora light in the zone.

This solar quiet daily variation (Sq) is a type of quiet daily variations that occur within a period of 24 hours. Generally, Sq is caused by atmospheric tide-dynamo due to electric field conductivity and solar wind in the ionosphere and middle-high altitude atmosphere. In this work, 1-minute dataset of geomagnetic field observed at 64 ground-based observatories during five most quiet days in a month was used to analyze the local time variations of Sq. The geomagnetic field H component of the Sq variations shows a pre-noon to post-noon asymmetrical pattern and maximizes at local noon. This asymmetry is obvious in spring, summer and winter. The ratio between the H component and D component of Sq amplitude in middle and low latitude regions in summer is greater than that in winter and vice versa. While in high latitude region it is smaller than that in the winter. From the pattern of Sq current system, the prenoon-postnoon asymmetry may be caused by the electric field in the high

latitude region. The electric field effect from high latitude to low latitude and its penetration directly to the middle latitude often results to prenoon and postnoon asymmetry of Sq field.

The 3D shows a global visualization of the Sq variations with respect to longitude, latitude and time. It was observed that unexpected Sq focus exist at the Polar Regions due to sources from interplanetary region. Travelling ionospheric disturbance is also obvious and relevant for monitoring some geotechnical phenomenon such as earthquakes that may have inter-regional footprints and occurrences within short time duration. The North Polar Region clearly reveals more SqH amplitude than the South Pole.

On the other hand, the performance results of the parallel computational platform showed that execution time of running serial Sq algorithm on a uniprocessor system took 18.5 minutes while the runtime for corresponding parallel algorithm on eight parallel processors took 2.95 minutes. The model of parallel execution time was observed to be logarithmic. Further analysis by extrapolation into few more points showed that 13 processors would generate execution time in 10.63 seconds, thus providing optimal computing solution in terms of resources and time. Contrary to an expected speedup of eight (8), the overall speedup was by factor of 4.32 due to parallel overhead impact. The algorithm and model developed showed that speedup discrepancy result from excessive parallel overheads. Thus Amdahl's model was improved by adding the derived parallel code tuning factor.

CHAPTER FIVE

5.0 SUMMARY OF FINDINGS, CONTRIBUTIONS TO KNOWLEDGE AND CONCLUSION

5.1 Summary of Findings

A high resolution analysis of spatio-temporal variations of geomagnetic field using parallel computing has been studied. The summary of findings is as follows:

- (i) Time resolution of geomagnetic field processing was successfully scaled down to minute variation.
- (ii) Sq in three directions (i.e. SqH, SqD, and SqZ) were simultaneously computed for all regions.
- (iii) An efficient computational algorithm was developed.
- (iv) Its assessment was carried out using the modified Amdahl's model.
- (v) The 3-D visualization platform was successfully developed with respect to longitude, latitude and time domains.

5.2 Conclusion

A high resolution analysis of spatio-temporal variations of geomagnetic field using parallel computing has been presented. The study has enabled clear observation of sudden impulses and rapid variations in the Sq field at a near real-time simulation. To avoid catastrophic effect of ionospheric current on space-borne and technological systems as well as earthquakes sudden occurrences, a near real-time parallel and scalable computational platform for effective Space weather simulation, monitoring and prediction has been established. The results of this work will be useful for further research on early detection of typhoon, tsunami, travelling ionospheric disturbance and other geophysical phenomena that have inter-regional footprints. Although, one-second data are not readily available for now, our scalable algorithm can serve as a basis for further research on 1-second simulation at the level where memory and processor interplay would be more essential. The modified Amdahl's model will be a reference point for more codes optimization.

Application of parallel computing for simulation of high resolution geomagnetic field spatio-temporal variations would enhance early warning systems and mitigation of pending adverse

space weather phenomena and geophysical activities. Thus, the resultant effects would improve socio-economic development, reliable technological systems, food security and wealth creation at large.

5.3 Contributions to Knowledge

Contributions of this research to knowledge in the field of computational analysis include:

- (i) Improvement on the time resolution for the analysis of geomagnetic field variations to minutes and seconds time frames.
- (ii) Extension of the scope of spatial analysis of geomagnetic field variations from regional to global scale.
- (iii) Generation of high frequency solar quiet daily variations for real-time performance analysis of Space and ground based technological systems;
- (iv) Development of computational platform for acquisition and processing of geomagnetic data
- (v) Improvement on the generalized Amdahl's model for performance gain evaluation of parallel computing systems.

5.4 Recommendations

Suggestions for future work include the under-listed

- a) There is need for establishment of more geomagnetic observatories in the developing countries especially along the equatorial region. This will facilitate further study of equatorial anomalies and other related geophysical phenomena.
- b) Several Space Systems Decision Support Services Centres should be established worldwide that can make use of the results of this study for the benefit of socio-economic development.

REFERENCES

- Akasofu, S.I. & Chapman S. (1972), *Solar - Terrestrial Physics*. Oxford: Oxford University Press.
- Al Geist, A. B., Dongarra J., Weicheng J., Robert M. & Vaidy S. (1994). *PVM: Parallel Virtual Machine - A User's Guide and Tutorial for Networked Parallel Computing*. MIT Press.
- Alberto S., Maria S. P., Jesus M. & Toni C. (April 2010). A high performance suite of data services for grids. *Future Generation Computer systems* , 26 (4), 622-632.
- Afanas'yeva, V. (1961). The corpuscular nature of day-to-day variability of Sq(H). *Planet Space Sci* , 498-507.
- Ale, F. (2005). *Development of GIS-based model for Epidemiological Surveillance*. MSc Thesis, Ile-Ife, Nigeria, Obafemi Awolowo University (unpublished.)
- Alex, S., Kadam, B.D. & Rastogi, R.G. (1992), A new aspect of daily variations of the geomagnetic field at low latitude, *J. of Atmospheric and Terrestrial Physics*, 54(7-8), p.863
- Almasi G.S. & Gottlieb A. (1989). *Highly Parallel Computing*. Redwood: Benjamin-Cummings.
- Antonio R., Nicolas G. & Manuel U. (October 2008). Recognition of circular patterns on GPUs: Performance analysis and contributions. *Parallel and Distributed Computing* , 68 (10), 1329-1338.
- Amdahl, G.M., (1967), Validity of the Single-Processor Approach to Achieving Large-Scale Computing Capabilities,” *Proc. Am. Federation of Information Processing Societies Conf.*, AFIPS Press, pp. 483-485.
- Ananth G., Anshul G., George K. & Vipin K. (2005), *Introduction to parallel Computing*, Boston: Addison Wesley, ISBN:0201-64865-2.
- Anandarao, B.G. & Raghavarao, R. (1987). Structural changes in the currents and fields of the equatorial electrojet due to zonal and meridional winds. *J. geophys. Res* , 92, 2514-2526.
- Aravindan, P. & Iyer K.N. (1990). Day-to-day variability in ionospheric electron content at low latitudes. *Planet Space Sci* , 743-750.
- Ashour, A.A. & Price, A.T. (1965). Night time earth currents associated with the daily magnetic variations. *Geophys. J. R. astr. Soc.* , 1-15.

- Axford, W. (1967). The interaction between the solar wind and the magnetosphere. In S. a. Matsushita, *Physics of Geomagnetic Phenomena* (pp. 1243-1291). New York: Academic Press.
- Bahman J., Jemal H. A. & Mohammad K. A. (November 2008). Performance modeling and analysis of heterogenous meta-computing systems interconnection networks. *Computers and Electrical Engineering* , 34 (6), 488-502.
- Balan, N. & Iyer, K.N. (1983). Equatorial anomaly in ionospheric electron content and its relation to dynamo currents. *geophys. Res.* , 88, 10259.
- Barney, B. (2011). *Lawrence Livermore National Laboratory*. Retrieved June 10, 2011, https://computing.llnl.gov/tutorials/parallel_comp/
- Barbatsi, K.K, (1990), Geomagnetic control of the ionospheric E-region at three different latitudes in the European-African sector, *Advances in Space Research*, 10 (11), p.89
- Behnke, K.A & Hagfors, T. (1974). Evidence for the existence of night-time F-region polarization fields at Arecibo. *Radio Science* , 211-216.
- Bernard, R. (1981). Variability of the semi-diurnal tide in the upper mesosphere. *atmos. Terr. Phys* , 43, 663-674.
- Bhargava, B.N & Yacob, A. (1971). Solar wind associated component in the low latitude magnetic daily variations. *J. Geomagn. Geoelectr.* , 23, 249-258.
- Bossomaier T.R.J. & Green D.G. (2000). *Complex Systems*. Cambridge: University Press.
- Bradford, E. & Mauget, L. (2001). *Linux and Windows interoperability guide*. Pearson Education.
- Brekke, A., Nosawa, S.S. & Sparr, T. (1994). Studies of the E-region neutral wind in the quiet auroral ionosphere. *J. geophys. Res.* , 99, 8801 - 8825.
- Briggs, B. (1984). The variability of ionospheric dynamo currents. *J. atmos. Terr. Phys.* , 419 - 429.
- Brown, G. (1975), Sq variability and aeronomomic structure. *J. atmo. Terr. Phy* , 37, 107-117.
- Brown, G.M. & Williams, W.R., (1969), Some properties of the day-to-day variability of Sq(H) Planetary and Space Science, 17 (3), p.455-470
- Burda, H., Begall, S., Cervený, J; Neef, J & Nemec, P . (2009). Extremely low-frequency electromagnetic fields disrupt magnetic alignment of ruminants. *National Academy of Sciences of the United States of America* . New York: PMID 19299504.
- Butcher E.C. & Brown G.M. (1981). Geophys. J.R. astr. *The variability of sq(H) on normal quiet days* , 64, 527 - 537.

- Butcher E. & Schlapp, D.M. (1992). The annual variation of the night-time values of the geomagnetic field. *Geophys. J. Int* , 111, 151-158.
- Butcher, E. (1982). On the latitude of the sq(H) focus in years of sunspot minimum. *Geophys S, J.B. astr.* , 69, 113-120.
- Campbell, W.H. (1997), Introduction to Geomagnetic Fields, Cambridge University Press.
- Campbell, W. (1973). The field levels near midnight at low and equatorial geomagnetic stations. *J. atmos. Terr. Phys.* , 35, 1127 - 1141.
- Campbell, W. (1986). An interpretation of induced electric currents in long pipelines caused by natural geomagnetic sources of the upper atmosphere. *surveys Geophys.* , 8, 239-259.
- Campbell, W. (1982). Annual and semi-annual changes of the quiet daily Variations (sq) in the geomagnetic field at North American location. *J.geophys. Res.* , 785-796.
- Campbell, W. H. (1989). Contribution to geomagnetic Sq-field and equatorial electrojet from the day/night asymmetry of ionospheric current under dawn-to-dusk electric field of magnetospheric origin. Basel: Birkba Vauser Verlag.
- Campbell, W. (1987). Some effects of quiet geomagnetic field changes upon values used for main field modeling. *Phys. Earth Planet Inter.* , 48, 193-199.
- Campbell, W.H. & Schiffmacher, E.R. (1987). Quiet Ionospheric currents and Earth conductivity profile, computed Storm quiet - time geomagnetic field changes in the region of Australia. *Aust. J. Phys.* , 40, 73-87.
- Campbell, W.H. & Schiffmacher, E.R. (1988). Quiet ionospheric currents of the southern hemisphere derived from geomagnetic records. *J. geophys.* , 93, 933-944.
- Carre G., Lanteri S. & Mark L. (November 1999). High performance simulations of compressible flows inside car engine geometries using the N3S-NATUR parallel solver. *Parallel Computing* , 25 (12), 1435-1458.
- Carroll, B. W. & Ostlie, D. A. (1995). *An Introduction to Modern Astrophysics (revised 2nd ed.)*. Benjamin Cummings.
- Chalmers, J. (1962). The first suggestion of the ionosphere. *J. atmos. Terr Phys* , 26, 219-221.
- Chapman, S. (1951). Some phenomena of the upper atmosphere. *Proc. Phys.* , B (64), 833-3
- Chapman S. & Bartels J., (1940), Geomagnetism, Oxford University Press
- Chapman, S. (1967). *Perspectives in Physics of Geomagnetic Phenomenom*. New York: Academic Press.

- Chapman, S. & Stagg, J.M. (1931). On the variability of the quiet day diurnal magnetic variation. *Proc R. Soc. Lond, A*, pp. 668-697. London.
- Chi P.J. & C.T. Russell. (2005). Travel-time magnetoseismology: Magnetospheric sounding by timing the tremors in space. *Geophys. Res. Lett.* , 32.
- Courtillot V. & LeMouél J.L. (2007), The study of Earth's magnetism: a foundation by Pergrinus and subsequent development of geomagnetism and paleomagnetism. *Rev. Geophys* , 1029.
- Courtillot V. & A. Chulliat (eds.), (2008), Observations magnétiques - Magnetic results 2005, Bulletin No 25, Bureau Central de Magnétisme Terrestre, Paris.
- Chulliat, A., J. Savary, K. Telali & X. Lalanne,(2009), Acquisition of 1-second data in IGP magnetic observatories, In: Proceedings of the XIIIth IAGA Workshop on geomagnetic observatory instruments, data acquisition and processing, J. J. Love (ed.), U.S. Geological Survey Open-File Report 2009-1226, pp.271.
- Cressie, N. (1990), The origins of kriging. *Mathematical Geology* 22:239-252.
- Cueto, M., McKnight, D. & Herraiz, M. (2003), Daily geomagnetic variations on the Iberian Peninsula, *Geophysical Journal International*, 152 (1), p.113-123
- Daniel M. S. (2003). Innovation in Maxwell's Electromagnetic Theory: Molecular Vortices, Displacement Current and Light. Cambridge University Press. pp. 96–98. ISBN 0-521-53329-5
- Darema F., George D. A., Norton, V. A. & Pfister G. F. (1988). A single-program-multiple-data computational model for expe fortran. *Parallel Computing* , 7, 11–24.
- David K. (2001), INTERMAGNET: Worldwide Near-Real-Time Geomagnetic Observatory Data, British Geological Survey, Murchison House
- Dongarra J. J., George B., Remi D. & Julien L. (2009), Algorithmic Based Fault Tolerance Applied to High Performance Computing, *Journal of Parallel and Distributed Computing*, Volume 69, pp 410-416
- Dunford, E. (1970). Electric Field and F-region electron densities over Peru. *atmos. Terr. Phys* , 32, 421-432.
- Duncan, R. (1990). A Survey of Parallel Computer Architectures. *IEEE Computer* , 5–16.
- Emilia, D.A. & Last, B.J. (1977). A study of sq variation at Addis Ababa. *J.atmos. terr. Phys* , 39, 375-381.
- Egedal, I. (1947). The magnetic diurnal variation of the horizontal force near the magnetic equator. *Terrest. Magn Atmos., Elec.*, , 51, 447-451.

- Egedal, J. (1948). Daily variation of the horizontal magnetic force at the magnetic equator. *Nature* , 161, 443-444.
- Erich S., J. J. Dongarra, Hans W. M. & Horst D. S. (March-April 2005). Recent trends in the marketplace of high performance computing. *Parallel Computing*, 31 (3-4), 261-273.
- Ethan L. M. & Randy H. K. (June 1997). RAMA: An easy-to-use, high-performance parallel file system. *Parallel Computing* , 23 (4-5), 419-446.
- Faith, R., & Sheshinski, R. (1979). *Misspecification of trend in spatial random-function interpolation with application to oxidant mapping*. Palo Alto, California: SIAM Institute for Mathematics and Society.
- Fleagle, R.G. & Businger, J.A. (1963). *An introduction to Atmospheric Phys.,.* London: Academic Press.
- Flynn, M. (1972). Some Computer Organizations and Their Effectiveness. *IEEE Trans. Comput* , C-21, 948.
- Forbes, J.M. (1981). The equatorial electrojet. *Rev. Geophys Space phys.* , 409-1377.
- Forbush, S.E. & Casaverde, M., (1961). Equatorial electrojet in Peru, CIW. *Publ* , 620.
- Friedman, H. (1960). The sun's ionizing radiations. In J. Ratcliffe, *Physics of the upper Atmosphere* (pp. 133-218). London: Academic Press.
- Geener, J.G. & Schlapp, D.M., (1979). A study of day to day variability of Sq over Europe. *J. atmos. Terr. Phys* , 41, 217-223.
- George B., Remi D., Dongarra J., & Julien L. (April 2009). Algorithm-based fault tolerance applied to high performance computing. 69 (4), 410-416.
- Geosciences Austrilia (2010), http://www.ga.gov.au/oracle/geomag/display_iqd.jsp
- Gilbert, W. (1958). *De Magnete*. Original 1600 book translated P. Fleury Mottelay in 1893, republished by Dover Publications, New York: Dover Publications
- Gilan, M. (June 1994). First-principles calculations on materials using massively parallel computing. 10 (2-3), 213-222.
- Gneryshev, M.N. K .F. Novihora, A.1. OP & Tokarava, N.V., . (1977). Sudden death from cardiovascular diseases and solar activity. In M. & Gnevyshev, *Effects of Solar Activity on the Earth's Atmosphere and Biosphere* (p. 20). Jerusalem.
- Gregorio S. Di, R. Rongo, W. Spataro, G. Spezzano & D. Talia. (April 1997). High performance scientific computing by a parallel cellular environment. *Future Generation Computer Systems* , 12 (5), 357-369.
- Gropp, W., Lusk, E., & Skjellum, A. (1999). *Using MPI: Portable Parallel Programming*

- with the Message Passing Interface*. MIT Press.
- Gropp W, Dinesh K. K., David E. K. & Barry F. S.. (March 2001). High-performance parallel implicit CFD. *Parallel Computing* , 27 (4), 337-362.
- Grivet, C. (1980). *Modeling and analysis of air quality data*. Palo Alto, California: SIAM Institute for Mathematics and Society, Stanford University.
- Gustafson, J.L., (1988), "Reevaluating Amdahl's Law," *Comm. ACM*, May, pp. 532-533.
- Hack, J. J. (May 1989). On the promise of general-purpose parallel computing. *Parallel Computing* , 261-275.
- Hanselman, D. C., & Littlefield, B. (2000). *Mastering Matlab 6, A Comprehensive Tutorial and Reference*. New Jersey: Prentice Hall.
- Haug E., Dubois J., Clinckemaillie J., Vlachoutsis S. & Lonsdale G. (1994). Transport vehicle crash, safety and manufacturing simulation in the perspective of high performance computing and networking). *Future Generation Computer Systems* , 10 (2-3), 173-181.
- Hargreaves, J. K. (1979), *The Upper Atmosphere and Solar Terrestrial Relations*. An introduction to the aerospace environment, Van Nostrand Reinhold Press
- Hargreaves, J. K. (1989). *The Upper Atmosphere and Solar-Terrestrial Relations*. Cambridge: Cambridge University Press.
- Harper, R. (1981). Some results on mean tidal structure and day-to-day variability over Arecibo. *J. atmos terr. Phys.* , 255-262.
- Hasegawa, M. (1960). On the position of the focus of the geomagnetic Sq current system. *J. atmos terr.* , J. geophys. Res.
- Harvey, B. (2007). *Russian planetary exploration: history, development, legacy, prospects*. Moscow: Springer.
- Hayakawa, M., Ito, T., Hattori, K. & Yumoto, K. (2000), ULF electromagnetic precursors for an earthquake at Biak, Indonesia on 17 February 1996, *Geophys. Res. Lett.*, 27, 1531–1534
- Heinz, E. K. (2000). *Magnetic Fields: A comprehensive theoretical treatise for practical use*. Wiley.
- Heppenheimer, T. A. (1998). The Space Shuttle Decision, NASA publication SP-4221.
- Hibberd, F. H., (1981), Day-to-day variability of the Sq geomagnetic field variation, *Australian Journal of Physics*, vol. 34, no. 1, p. 81-90.
- Hibberd, F. H. (1985). The geomagnetic Sq variation-annual, semi-annual and solar cycle

- variations and ring current effects. *J. atmos. Terr. Phys.*, , 341-532.
- Hibberd. F.H . & Davidson, R. E. (1988). Global scale of, the day to day variability of Sq. *Geophys. J* , 315-321.
- Hines, C. (1959). An interpretation of certain ionospheric motions in terms of atmospheric gravity waves. *J. Geophys. Res* , 64, 2210-2211.
- Hipper G. & Tavangarian D. (1998). A concurrent network architecture for cost-efficient parallel computing using workstation clusters, *Simulation Practice and Theory. High Performance Computing and Networking: Simulation Practice* , 6 (2), 197-218.
- Hohn, M. (1988). *Geostatistics and Petroleum Geology*. New York: Van Nostrand Reinhold.
- Hofmann W. B., Lichtenegger B.H. & J. Collins. (1994). *Global Position System (Theory and practice)*. New York: Springer Wien.
- Hopkins, J. (1980). *Glossary of astronomy and astrophysics*. Chicago: Univ of Chicago Press
- Huangfu, G. (1997). The M7.0 Lijiang earthquake in Yunnan on February 3, 1996. *Journal of Seismological Research*, 20-1, 1-8
- Hummel, S. F. (July 1999). Workstation hardware and software support for parallel applications. *Future Generation Computer systems* , 8 (1-3), 49-65.
- Hutton, R. (1967). Sq currents in the American equatorial Zone during the IGY-II. Day to day variability. *J. atmos., Terr. Phys.*, 29, , 26, 1429 - 1442.
- Ivanov V. I & Kostomaroy, D.P. (1963). Computation of electric current Induced in the sea by Sq variations. *Geomag. Aeronomy* , 868-879.
- Jacobs, J. A. (1987). *Geomagnetism Volumes 1, 2 and 3* . New York: Academic Press.
- James A. M., (2007), Solar Storm Threat Analysis, Impact, URL: Bloomfield, Indiana 47424, <http://www.breadandbutter-science.com/SSTA.pdf>
- James K. & Alan D. G. (July 2003). A high-performance communication service for parallel computing on distributed DSP systems. *Parallel Computing* , 29 (7), 851-878.
- Jankowski, J. & C. Sucksdorff. (1996). IAGA Guide for Magnetic Measurements and Observatory Practice. *International Association of Geomagnetism and Aeronomy* .
- Jensen, J.L., Lake, L.W., Corbett, P.W.M. & Goggin, D.J. (1997). *Statistics for Petroleum Engineers and Geoscientists*. New Jersey: Prentice Hall PTR.
- Joseph W. M., David G. K., David L., Subhankar B. & Eric D. (March 2001). Parallel performance of two applications in the Boeing high performance computing benchmark suite. *Parallel Computing* , 27 (4), 457-475.

- Journel, A.G. & Huijbregts, C.J. (1978). *Mining Geostatistics*. London: Academic Press.
- Kane, R. (1971a), Relationship between H ranges at equatorial and middle latitudes. *J. atmos. Terr. Phys.* , 33, 319-327.
- Kane, K. (1975). Day to day variability of ionospheric electron content at mid-latitudes. *J. geophys. Res.* , 80, 3091-3103.
- Kane, R. (1971b). Characteristics of stormtime geomagnetic daily variation. *J. atmos. Terr. Phys.* , 38, 1585 --- 1595.
- Kane, R. (1971c). Effective local time for sq and sd. *Ann. Geophys.* , 27, 237-239.
- Kelley, M. C. & Heelis, R. A., . (1989). *The Earth's Ionosphere: Plasma Physics and Electrodynamics*. New York: Academic Press.
- Kirchoff V.W., J.H. & Carpenter, I. A. (1976). The day to day variability in ionospheric fields and currents. *J. geophys. Res.* , 81, 2737- 2742.
- Kopp, G., Lawrence, G. & Rottman, G. (2005). The Total Irradiance Monitor (TIM): Science Results. *Solar Physics* , 129-139.
- Krige, D.G. (1951), A statistical approach to some mine valuations and allied problems at the Witwatersrand, MSc Thesis, University of Witwatersrand, South Africa
- Lang, K. R. (2000). *The Sun from Space*. Springer .
- Langel, R.A., T.J. Sabaka, R.T. Baldwin & Conrad, J.A. (1996). The near-earth magnetic field from magnetospheric and quiet day ionospheric sources and how it is modeled. *Phys. Earth Planet. Int.*
- Lanzerotti, L. J. (2001), Space weather effects on technologies, in *Space Weather*. Ed. P. Song, H. Singer, & G. Siscoe, American Geophysical Union, Washington, pg. 11
- Lars B., Jorg N. & Wolfgang S., (November 1999). A of P-A High-Performance Middleware Layer for Parallel Distributed Computing. *Parallel and Distributed Computing* , 59 (2), 155-179.
- Last, B. E. (1976). AQD occurrence at Addis Ababa, Trivandrum and Alibag,. *Planet, Space Sci.* , 24, 567-572.
- Li M., D.W. Walker, O.F. Rana, Y. Huang, P.T. Williams & R.C. Ward. (November 2003). Engineering high-performance legacy codes as CORBA components for problem-solving environments. *Parallel and Distributed Computing* , 63 (11), 1152-1163.
- Liebhold, A.M., Rossi, R.E. & Kemp, W.P. (1993). Geostatistics and geographic information system in applied insect ecology. *Annual Review of Entomology* , pp. 303–327.
- Lilley, F. E. (1975). The Analysis of daily Variations records by Magnetometer Arrays.

- Geophys. J. R. Astron. Soc* , 43, 1-16.
- Love, J. J. & Gannon J. L. (2010), Movie maps of low latitude magnetic storm disturbance, *Space Weather*, VOL. 8, S06001, doi:10.1029/2009SW000518.
- Love J. J. & Carol A. F. (2011), The USGS Geomagnetism Program and Its Role in Space Weather Monitoring, *Space Weather*, Vol.9, S07001, doi:10.1029/2011SW000684
- Maed, G. (1964). Deformation of the geomagnetic field by the solar wind,. *J. geophy, Res* , 69, 1181-1195.
- Malin. S.R.C. & Gupta, J. (1977). The Sq current system during the International Geophysical year. *Geophys. J. R. astr. Soc.* , 49, 515-529.
- Mann, R. J. & Schlapp, D.M., (1988). The equatorial electrojet and the day to day variability of Sq. *J. atmos; Terr Phys.* , 50, 57-62.
- Manoj, C., Kuvshinov, A., Maus, S. & Lühr, H. (2006): Ocean circulation generated *magnetic* Signals, *Earth Planets and Space*, 584
- Mark J. C. & Michael J. Q. (October 1997). Automated performance prediction for scalable parallel computing. *Parallel Computing* , 23 (10), 1405-1420.
- Masahiko T. (1999), Time variation of global geomagnetic Sq field in 1964 and 1980
Journal of Atmospheric and Solar-Terrestrial Physics, Volume 61, Issue 17, 15
November 1999, Pages 1321-1322
- Matheron, G., (1963), Principles of Geostatistics, *Economic Geology*, 58, pp 1246–1266.
- Mattson T.G., Sanders B.A. & Massingill B.L. (2005). *Patterns for Parallel Programming*.
Boston: Addison-Wesley, ISBN 0-321-22811-1.
- Matsushita, S. & Tarpley, J.D., (1970). Effects of dynamo region electrical fields on the magnetosphere. *J. geophys. Res.* , 75, 5433-5443.
- Matsushita, S. (1960). Seasonal and day to day, changes of the central position of the Sq overhead current system. *J. geophys. Res.* , 65, 3825 - 3839.
- Matsushita, S. (1967). Solar quiet and lunar daily variation fields . In S. .. Matsushita, *Physics of Geomagnetic Phenomena* (pp. 302-425). New York: Academic Press.
- Matsushita, S. (1969). Dynamo currents, winds and electric fields. *Radio sci.* , 4, 271 – 778.
- Matsushita, S. & Maeda, H.,. (1965b). On the geomagnetic solar quiet daily variation field during the IGY. *J. geophys. Res.* , 70, 2535 - 2558.
- Matsushita, S. & Maeda, H. (1965a). On the geomagnetic lunar daily variation field. *J. geophys. Res.* , 70, 2559 - 2577.
- Matsushita, S. (1975). IMF polarity on the Sq current focus location. *Geophys. Res.* , 80,

- 4751 - 4762.
- Matthias A. B., Graham E. F. & Michael M. R. (October 1999). Message-passing environments for metacomputing. *Future Generation Computer Systems* , 15 (5-6), 699-712.
- Matveyenkov, I. (1983). Possible effects of distant field sources on Sq variations,. *Geomagnetic & Aeronomy* , 23, 276 -- 277.
- Maugh, T. (1982). Magnetic navigation an attractive possibility. *Science* , 215, 1492-1493.
- Mayaud, P. (1976). Magnetospheric and night- time induced effects in the regular daily variation SR. *Planet Space Sci* , 34, 1049-1057.
- Maynard N. (1995), Space weather prediction, Review of Geophysics, 547-557
- Merrill, R. T., M. W. McElhinny & P. L. McFadden. (1996). *The Magnetic Field of the Earth*. New York: Academic Press.
- Mazauder, C. (1993). On the relationship between ionospheric phenomena & the day to day variability of the Earth magnetic field regular variation 'SR': Advances made during the last two decades with the incoherent scatter sounder network results. *Geophys. Res.* , 397-420.
- Melissa C. S. & Gregory D. P. (May 2005). Parallel application performance on shared high performance reconfigurable computing resources. *Performance Evaluation* , 60 (1-4).
- Mendillo, M., Lynch, F.X. & Klobucher, J.A.,. (1981). Geomagnetic activity control of ionospheric variability in solar-Terrestrial prediction. *space environment laboratory*. Boulder: Boulder, CO.
- Meyer-Vernet, N. (2007). *Basics of the Solar Winds*. Cambridge: Cambridge University Press.
- McDonald, S. E., (2007), Day to day and longitudinal variability of the nighttime low latitude terrestrial ionosphere, George Mason University Press, NY
- Michael J. Quinn, (1994), *Parallel Computing Theory and Practice*, McGraw-Hill, Inc., NY, USA
- Mitsuhsa S., Hiroshi T., Atsushi H., Yutaka I., Satoshi S. Hidemoto N., Satoshi M. & Umpei N.. (March 1998). NinF and PM: Communication libraries for global computing and high-performance cluster computing. *Future Generation Computer Systems* , 13 (4-5), 349-359.
- NASA. (1993). Perspective from Space. *Poster Series Users Guide* .
- Needham, J. (1962). *Science and Civilisation in China*. Cambridge: Cambridge University Press.

- Nosé, M., Ohtani S., Takahashi K., Lui A. T. Y., McEntire R. W., Williams D. J., Christon S. P. & Yumoto K. (2001), Ion composition of the near-Earth plasma sheet in storm and quiet intervals: Geotail/EPIC measurements, *J. Geophys. Res.*, 106, 8391-8403
- Ogbuchi, P.O., Onwumechili, C.A. & Ifedili, S.O., (1967). The equatorial electrojet and the worldwide Sq currents. , *J. atmos. terr. phys.*, , 29, 149-160.
- Ogunade, S. O. (1986). Induced electromagnetic fields in oil pipelines under electrojet current sources,. *Phys. Earth Planet. Inter.* , 43, 307 - 315.
- Okeke, F. (1996). The equatorial electrojet and day to day variability of sq. *Nig J. phys*,41-43
- Okeke, F. N., C. A. Onwumechili, and B. A. Rabi (1998), Day-to-day variability of geomagnetic hourly amplitudes at low latitudes,*Geophys.J.Int.*,134,484–500
- Oko, S.O., Onwumechili, C.A. & Ezema, P.O. (1996), Geomagnetically quiet day ionospheric currents over the Indian sector—II. Equatorial electrojet currents, *Journal of Atmospheric and Terrestrial Physics*, 58 (5), p.555-564, Apr 1996
- Olaf S., Klaus G., Wolfgang F., Andreas S.. (September 2001). PARDISO: A high-performance serial and parallel sparse linear solver in semiconductor device simulation. *Future Generation Computer Systems* (1), 6978.
- Oldehoeft, R. (2000). Taming complexity in high-performance computing. *Mathematics and Computers in Simulation* , 54 (4-5), 341-357.
- Oliver M. A. & Webster R. (1990), Kriging: a method of interpolation for geographical information system, *INT. J. Geographical Information Systems*, VOL. 4, No.3, 313-2
- Olsen, Nils (2007), On induction effects of geomagnetic daily variations from equatorial electrojet and solar quiet sources at low and middle latitudes, *Journal of Geophysical Research* (ISSN: 0148-0227); vol: 112, issue: B 10
- Olson, W. (1974). A model of the distributed magnetospheric currents. *geophys. Res* , 3731–8
- Olson, W. P. (1989). The contribution of magnetospheric current to Sq. *Pure Appl. Geophys* , 447 - 462.
- Olson, W. (1970), Variations in the Earth's surface magnetic field from the magnetopause current. *Planet. Space Sci.* , 18, 1471 – 1484.
- Onwumechilli C.A. & Alexander N.S. (1959), Variations in the geomagnetic field at Ibadan, Nigeria—I Solar variations, *Journal of Atmospheric and Terrestrial Physics*, Volume 16, Issues 1–2, October 1959, Pages 106–114
- Onwumechili, C.A. & Ogbuchi, P.O. (1962). *J. atmos. terr.phys. Fluctuations in the geomagnetic horizontal field* , 24, 173 - 190.

- Onwumechili, C. (1985). Satellite measurements of the equatorial electrojet,. *J. Geomag. Geoelectr.* , 37, 11-36.
- Onwumechili, C. (1992a). A study of rocket measurements of ionospheric currents -I.
General setting and night-time ionospheric currents. *Geophys. J. Int.* , 108, 633 - 640.
- Onwumechili, C. (1992b). A study of rocket measurements of ionospheric currents – II.
Ionospheric current outside the dip equatorial zone. *Geophys. J. int.* , 108, 633 -640.
- Onwumechili, C. (1992c). A study of rocket measurements of ionospheric currents - III.
Ionospheric currents at the magnetic dip equator. *Geophys. J. Int.* , 108,, 647 -- 659.
- Onwumechili, C. (1992d). A study of rocket measurements of ionospheric currents - IV.
Ionospheric currents in the transition zone and the overview of the study. *Geophys. J. Int.* , 108, 660 - 672.
- Onwumechili, C. (1992f). Study of the return current of the equatorial electrojet. *J. Geomag., Geoelectr.* , 44, 1 - 42.
- Onwumechili, C.A., Agu, C.E. & Ozoemena, P.C.,. (1989). Effect of the equatorial electrojet intensity on its landmark distances. *J. Geomag. Geoelectr.* , 41, 461 - 457.
- Onwumechili, C.A. & Agu, C.E. (1981). The relationship between the current and the width of the equatorial electrojet. *J. atmos. terr. phys.* , 43, 573-578.
- Onwumechili, C.A. (1967). Geomagnetic variations in the equatorial zone, in *Physics of Geomagnetic Phenomena*, Vol. 1, pp.425-507, eds Matsushita, S. & Campbell, W.H., Academic Press, New York
- Onwumechili, C. ,A. & Ezema, P.O.,. (1977). On course of the geomagnetic daily variation in low latitude. *J. atmos. terr. phys.* , 39, 1079 - 1086.
- Oslen, N. (1996). Magnetospheric contributions to geomagnetic daily variations. *Ann. Geophysicae* , 14, 538 - 544.
- Osborne, D. (1966). Correlations between quiet day magnetic ranges. *J. atmos. Terr, phys.* , 28, 45 - 51.
- Osborne, D. (1964). Daily and seasonal changes of equatorial electrojet in Peru. *J. atmos. terr. phys.* , 26, 1097-1105.
- Osborne, D. (1962). Position and movement of the equatorial electrojet over Ghana. *J. atmos. terr. phys.* , 24, 491 - 502.
- Osborne, D. (1968). Some further correlations in magnetic variation. *J. atmos. terr. phys.* , 30, 1479 – 1486.
- Patterson David A. & John L. Hennessy . (1998). *Computer Organization and Design*.

- Morgan Kaufmann .
- Phillips, A. & Briggs, B.H. (1991). The day to day variability of upper atmosphere tidal winds and dynamo currents. *J. atmos, Terr phys* , 53, 39 - 47.
- Plaza, A. J. (January 2008). Parallel techniques for information extraction from hyperspectral imagery using heterogeneous networks of workstations. *Parallel and Distributed Computing* , 68 (1), 93-111.
- Price A.T. (1963), The Non-cyclic variation during quiet days, Journal of Geophysical Research (US), vol.68, pp. 6383-9
- Price. A.T. & Wilkins, G.A. (1963). New methods for the analysis of geomagnetic fields and their application to the Sq field of 1932-3. *Phil. Trans. R. Soc* , A256, 31- 98.
- Rabiu, A. B. (2000), *Geomagnetic Field Variations at Middle Latitudes*, Ph.D. University of Nigeria, Nsukka, Nigeria: Thesis.
- Rajaram, M. (1983). Determination of the latitude of Sq focus and its relation to the electrojet variations. *J. atmos. terr. Phys.* , 45, 573-578.
- Rajaram, M. & Mitra, S.,. (1981). Correlations between convulsive seizure and geomagnetic activity. *Neurosic Lett* , 24, 187- 191.
- Randall, W. & Randall, S. (1991). The solar wind hallucinations - a possible relation to magnetic disturbances, Bioelectromagnetics. *J. atmos. terr. Phys* , 67 -70.
- Rastogi, R.G ., Chandra, H. & Chakravarty S.C.,. (1971). The disappearance of Es and the reversal of electrojet current. *Proc. Ind. Acad Sci.*, A74, 6267.
- Rastogi, R. (1996). Solar flare effects on zonal and meridional currents at the equatorial electrojet station, Annamalainagar. *J. atmos. Terr. phys.* , 58, 14130 - 1420.
- Reddy. C.A. & Devasia, C.V. (1981). Height and latitudinal structures of electric fields and currents due to local east-west winds in the equatorial electrojet. *J. , geophy, Res* , 86, 5751-5767.
- Rieger M. & Leitingner R. (2002). Assessment of TID activity from GPS phase data collected in a dense network of GPS receivers. *Acta Geodet., Geophys. Hung* .
- Richard, C. C. (1859, September 1). Description of a Singular Appearance seen in the Sun on September 1, 1859. *Monthly Notices of the Royal Astronomical Society* , pp. 13-18.
- Riktake, T. (1961). The effect of the ocean on rapid geomagnetic changes. *Geophys J. Roy. Astro. Soc* , 5, 1-15.
- Rishbeth, H. (1971). The F-layer dynamo. *Planet Space Sci.*, , 21, 1897-1910.

- Robert, H. (1885). The History of the Kew Observatory. *Royal Society of London* (pp. 37-86). London: Academic Press.
- Rolf H. & David W. W. (May 1999). The emergence of the MPI message passing standard for parallel computing. *Computer Standards and Interfaces* , 21 (1), 51-62.
- Salen, D. (1968). Solar flare effects and simultaneous magnetic daily variation. *J. atmos. Terr. Phys* , 30, 1641- 1648.
- Sarabhai, V. and Nair, K.N. (1969). Daily variation of the geomagnetic field and the deformation of the magnetosphere. *Nature* , 223, 603-604.
- Sastri, J. (1981). Occurrence of abnormal Sq(H) phase in the equatorial electrojet region. *Planet Space Sci.* , 29, 74 1-745.
- Satellite News Digest (2011), Satellite Outages and Failures, <http://sat-nd.com/failures/>
- Schlapp, D. (1976). Day-to-day variability of the latitudes of the Sq foci. *J. atoms, terr. Phys.*, 38, 573-577.
- Schlapp, D. M. (1968). Worldwide morphology of day to day variability of Sq. *J. atmos Terr. Phys* , 30, 1761 - 1776.
- Schlapp, D.M. & Mann, R.J. (1983). The spatial scale of correlation of the day-To-day variability of Sq. *Geophys. J.R. astro. Soc.* , 73, 671-673.
- Schlapp D.M., Mann R.J. & Butcher E.C., (1998). On the relation between days of small amplitude of Sq(H) and abnormal quiet days. *Geophys. J.R. astro Soc.* , 95, 633-639.
- Schmucker, U. (1999a). A Spherical Harmonic Analysis of Solar Daily Variations in the Years 1964–1965 : Response Estimates and Source Fields for Global Induction—I. Method. *Geophys. J. Int* , 136, 439-454.
- Schmucker, U. (1999b). A Spherical Harmonic Analysis of Solar Daily Variations in the Years 1964–1965: Response Estimates and Source Fields for Global Induction—II. Results. *Geophys. J. Int.* , 136, 455-476.
- Schuster, A. (1989). The diurnal variation of terrestrial magnetism. *Phil. Trans. Roy. Soc.* , 467-518.
- Sizova, L. Z. (2002). The field-aligned currents effect on equatorial geomagnetic field variations. *Adva. Sp. Res.* , 2247 – 2252.
- Sillanpaa A, H. Luhr, A. Viljanen & P. Ritter (2004), Quiet-time magnetic variations at high latitude observatories, *Earth Planets Space*, 56, 47–65
- Smirnova, N. A., Hayakawa, M. & Ito, T. (1999). Structure of the ULF geomagnetic noise

- in a seismoactive zone and its relation to the earthquake, in: Noise in Physical Systems and 1/f Fluctuations (Proc. 15th Int. Conf. ICNF'99, Hong Kong, 23–26 August), (Ed) Surya, Ch., World Scientific, 471–474, 1999.
- Soicher, H. & Gorman, F.J. (1981). Variability of total electron constant at temperate and high latitudes in Scientific Engineering Uses of Satellite Radio Beacons. *COSPAR/URSI Symposium* (pp. 99-105). Warszawa, Poland: Polish Scientific Publishers.
- Stening, R. (1973). The electrojet field in the ionosphere. *Planet Space Sci.* , 21, 1897-1910.
- Stening, R, Reztsova, T. & Huy-Minh, L. (2007), Variation of Sq focus latitudes in the Australian/Pacific region during a quiet sun year, *J. of Atmospheric and Solar-Terrestrial Physics*, 69 (6), p.734-740
- Stern, D. P. (2002). A millennium of geomagnetism. *Rev. Geophys* , 40(3), 1007, doi:10.1029/2000RG000097.
- Suzuki, A. (1978). Geomagnetic Sq field at successive universal times. *J. atmos, Terr. Phys.* , 40, 449-463.
- Suzuki, A. (1979). UT and day-to-day variations in equivalent current systems for world geomagnetic variation. *J., Geomag, Electr.* , 31, 21-46.
- Takeda, M. (1984). Day to day variation of equivalent Sq current system during March 11-26 1976. *J. geomag. Geoelectr.* , 36, 215-228.
- Takeda, M. (1985). U.T variation of internal Sq currents and the oceanic effects during 1980 March 1 - 18. *Geophys. ,J. R astro. Soc* , 80, 649-659.
- Thakur, N. K., & Rao, D. R. (2003). On Sq(Z) variation in the Indian region. *Journal of Atmospheric and Terrestrial Physics* , 507-512.
- Taylor, E. G. (1951). The South-Pointing Needle. *Imago Mundi* , 8, 1-7.
- Tarpley, J. (1970). Ionospheric wind dynamo II, solar tides. *Planet Space Sci.* , 18, 1091-3.
- Uozumi, T., Yumoto, K., Kawano, H., Yoshikawa, A., Olson, J.V., Solov'yev, S.I. & Vershinin, E.F. (2000). Characteristics of energy transfer of Pi 2 magnetic pulsations: Latitudinal dependence. *Geophysical Research Letters* 27(11): doi: 10.1029/1999GL010767
- Vestine, E. H. (1967). Main geomagnetic field. In S. & Matshushita, *Physics of Geomagnetic Phenomena* (pp. 181-234). New York: Academy Press.
- Walker, A.D. M. Pinnock, M., Baker, K. B., Dudency, J. R. & Rash, J. P. S., (1998). Strong flow bursts in nightside ionosphere during extremely quiet solar wind conditions. *Geophys, Res, Letters* , 25, 881-884.

- Wu, C. C., Liou, K., Lepping, R. P. & Meng, C. I.,. (2004). Identification of substorms within storms. *Jour. Atmos. Solar-Terr. Phys.* , 66 (2), 125-132.
- Yacob, A.R. & Rao, D.R. (1996). Solar and Annual variations of Sq(H) at Alibag. *J. atmos Terr.phys* , 28, 351-360.
- Yifeng Z. & Hong J. (February 2006). CEFT: A cost-effective, fault-tolerant parallel virtual file system. *Parallel and Distributed Computing* , 66 (2), 291-306.
- Yumoto, K. (1996). The STEP 210 magnetic meridian network project. *Geomag Geoelectr* , 1297-1310.
- Xu, W.Y. (1989), Polar region Sq, in *Quiet Daily Geomagnetic Fields*, edited by W. H. Campbell, pp. 317–393, Birkhauser Verlag, Basel, 1989
- Zenios, S. A. (December 1999). High-performance computing in finance: The last 10 years and the next. *Parallel Computing* , 25 (13-14), 2149-2175.

APPENDIX A Detailed Distributions of INTERMAGNET Observatories

Table A: The distribution of the 64 worldwide observatories with their IAGA code (station ID)

S/N	StationID	Locality	Country	Orientation	Long (deg)	*Lat (deg)
1	ABK	Abisko	Sweden	HDZF	18.823	68.358
2	ALE	Alert	Canada	XYZF	297.65	82.5
3	AMS	MartinDe Vivis	France	XYZF	77.574	37.796
4	BDV	Budkov	Czech Republic	XYZF	14.015	49.08
5	BEL	Belsk	Poland	XYZF	20.79	51.84
6	BFE	Brofelde	Denmark	XYZF	11.67	55.63
7	BLC	Baker Lake	Canada	XYZF	263.97	64.33
8	BNG	Bangui	Central African Republic	XYZF	18.566	4.333
9	BOU	Boulder	USA	HDZF	254.76	40.14
10	BRW	Point Barrow	USA	HDZF	203.38	71.38
11	BSC	Bay St. Loius	USA	HDZF	270.6	30.4
12	CBB	Cambridge Bay	Canada	XYZF	254.969	69.123
13	CLF	France	Chambon la Foret	XYZF	2.266	48.017
14	CMO	College	USA	HDZF	212.16	64.86
15	CNB	Canberra	Australia	XYZF	149.36	-35.31
16	CZT	Port Alfred (CROZET)	France	XYZF	51.86	46.431
17	DLR	Del Rio	USA	HDZF	259.08	29.49
18	DRV	Dumont d'Urville	Fr ance	XYZF	140.007	66.665
19	ESK	Eskdalemuir	UK	HDZF	356.8	55.3
20	EYR	Eyrewell	New Zealand	HDZF	172.35	-43.42
21	FCC	Fort Churchill	Canada	XYZF	265.912	58.759
22	FRD	Fredericksburg	USA	HDZF	282.63	38.2
23	FRN	Fresno	USA	HDZF	240.28	37.09
24	FUR	Furstenfeldbruck	Germany	XYZF	11.28	48.16
25	GDH	Godhavn	Denmark	XYZF	306.47	69.25
26	GLN	Glenlea	Canada	XYZF	262.88	49.645
27	GNA	Gnangara	Australia	XYZF	116	-31.8
28	GUA	Guaguan	USA	HDZF	144.87	13.58
29	HAD	Hartland	UK	HDZF	355.5	51
30	HER	Hermanus	South Africa	HDZF	19.225	34.425
31	HON	Honolulu	USA	HDZF	202	21.32
32	IQA	Lqaluit	Canada	XYZF	291.482	63.75
33	KAK	Kakioka	Japan	HDZF	140.18	36.23
34	KOU	Kourou	France	XYZF	307.269	2.21
35	LER	Lerwick	UK	HDZF	358.8	60.1
36	LOV	Lovo	Sweedeen	XYZF	17.824	59.344
37	MBC	Mould Bay	Canada	XYZF	240.638	76.315
38	MBO	Mbour	Senegal	XYZF	343.033	14.384
39	MEA	Meanook	Canada	XYZF	246.653	54.616
40	MMB	Memambetsu	Japan	HDZF	144.2	43.9

41	NAQ	Narsarsuaq	Denmark	XYZF	314.56	61.16
42	NCK	Nagycenk	Hungary	HDZF	16.717	47.633
43	NEW	Newport	USA	HDZF	242.88	48.26
44	NGK	Niemegk	Germany	HDZF	12.68	52.07
45	NUR	Nurmijarvi	Finland	XYZF	24.66	60.51
46	OTT	Ottawa	Canada	XYZF	284.448	45.403
47	PAF	Port-aux-Francais	France	XYZF	70.262	49.353
48	PBQ	Poste-de-la-Baleine	Canada	XYZF	282.255	55.277
49	PHU	Phu Thuy	Vietnam	XYZF	105.951	21.028
50	PPT	Pamatai	France	XYZF	210.416	17.566
51	RES	Resolute Bay	Canada	XYZF	265.105	74.69
52	SBA	Scott Base	New Zealand	HDZF	172.78	-77.85
53	SIT	Sitka	USA	HDZF	224.67	57.06
54	SJG	San Juan	USA	HDZF	293.85	18.11
55	SOD	Sodankyla	Finland	XYZF	26.63	67.37
56	STJ	St. John's	Canada	XYZF	307.323	47.595
57	TAM	Tamanrasset	Algeria	XYZF	5.527	22.792
58	TAN	Antananarivo	Madagascar	XYZF	47.525	18.917
59	THL	Qaanaaq	Denmark	XYZF	290.77	77.47
60	THY	Tihany	Hungary	XYZF	17.54	46.9
61	TUC	Tucson	USA	HDZF	249.17	32.25
62	VIC	Victoria	Canada	XYZF	236.58	48.52
63	WNG	Wingst	Germany	XYZF	9.073	53.743
64	YKC	Yelloknife	Canada	XYZF	245.518	62.482

* Latitude values are in the Northern hemisphere are positive while they are negative in southern hemisphere

APPENDIX B: MATLAB PROGRAMS

B1: Matlab (sequential) program for Analysis Sq Daily Variations of Geomagnetic Field

% Temporal variations of geomagnetic fields by Ale F. under supervision of Prof. Ibidapo-Obe; Dr.
%T.A. Fashanu & Dr. O.A.Agboola

% Aim: To study the spatial distribution of the ionospheric current responsible for geomagnetic field
% variation using parallel computing

% data needed are geomagnetic field data of 64 stations, longlat1996.dat, longlat1996a.dat,
% IQDDays1996.dat

clear; clc;
echo off;

store_time = zeros(3, 6);

store_time(1,:)=clock; % [year month day hour min sec]

profile on;

HRESALL1996Jan=[];
HRESALL1996Feb=[];
HRESALL1996Mar=[];
HRESALL1996Apr=[];
HRESALL1996May=[];
HRESALL1996Jun=[];
HRESALL1996Jul=[];
HRESALL1996Aug=[];
HRESALL1996Sep=[];
HRESALL1996Oct=[];
HRESALL1996Nov=[];
HRESALL1996Dec=[];

DRESALL1996Jan=[];
DRESALL1996Feb=[];
DRESALL1996Mar=[];
DRESALL1996Apr=[];
DRESALL1996May=[];
DRESALL1996Jun= [];
DRESALL1996Jul= [];
DRESALL1996Aug=[];
DRESALL1996Sep=[];
DRESALL1996Oct=[];
DRESALL1996Nov=[];
DRESALL1996Dec=[];

ZRESALL1996Jan=[];
ZRESALL1996Feb=[];
ZRESALL1996Mar=[];
ZRESALL1996Apr=[];
ZRESALL1996May=[];

```
ZRESALL1996Jun=[];
ZRESALL1996Jul=[];
ZRESALL1996Aug=[];
ZRESALL1996Sep=[];
ZRESALL1996Oct=[];
ZRESALL1996Nov=[];
ZRESALL1996Dec=[];
```

```
obslist =
['ABK','ALE','AMS','BDV','BEL','BFE','BLC','BNG','BOU','BRW','BSL','CBB','CLF','CMO','CNB','
CZT','DLR','DRV','ESK','EYR','FCC','FRD','FRN','FUR','GDH','GLN','GNA','GUA','HAD','HER','H
ON','IQA','KAK','KOU','LER','LOV','MBC','MBO','MEA','MMB','NAQ','NCK','NEW','NGK','NUR','
OTT','PAF','PBQ','PHU','PPT','RES','SBA','SIT','SJG','SOD','STJ','TAM','TAN','THL','THY','TUC','
VIC','WNG','YKC'] ;
```

```
% selecting stations by X, Y, Z or H, D, Z components(64x3 = 192)
for n=1:64
```

```
    obs=obslist(n,:);
```

```
    Xobs=cat(2,'X',obs,'.xls');
    Yobs=cat(2,'Y',obs,'.xls');
    Hobs=cat(2,'H',obs,'.xls');
    Dobs=cat(2,'D',obs,'.xls');
    Zobs=cat(2,'Z',obs,'.xls');
```

```
    if exist(Xobs, 'file');
```

```
        % calling function filterChar to remove headerlines and reshape
        Xfile=readingExcel(Xobs);
        Yfile=readingExcel(Yobs);
        Zfile=readingExcel(Zobs);
```

```
% getting the universal time (UT) in seconds
```

```
    UT = 1:1440; %
```

```
% magnetic field values
```

```
    fileX = Xfile(1:366,1:1440);
    fileY = Yfile(1:366,1:1440);
    fileZ = Zfile(1:366,1:1440);
```

```
% converting X and Y components to H and D respectively
```

```
    fileH = sqrt(fileX.^2 + fileY.^2);
    fileD = (180/pi)*600*atan(fileY./fileX);
```

```
else
```

```
    %calling function filterChar to remove headerlines and reshape
    Hfile=readingExcel(Hobs);
    Dfile=readingExcel(Dobs);
    Zfile=readingExcel(Zobs);
```

```

% getting the universal time (UT)in seconds
UT = 1:1440; % this section config. the universal time for referencing

% magnetic field values
fileH = Hfile(1:366,1:1440);
fileD = Dfile(1:366,1:1440);
fileZ = Zfile(1:366,1:1440);

end

load longlat1996.dat % Longitude and latitudes of the 64 stations

longlat1996R=longlat1996; % use longlat1996R (64x2) in the plotting

longlat1996R(1,:)=[];

Long(n) = longlat1996(n+1:n+1,1);
yt = UT + round ((Long(n)/15)*60);

% converting the LT to the time that must strictly be between 1 and 1440 minute

for i=1:1440;

    if yt(i) <= 1440;
        LT(i) = yt(i);
    elseif yt(i) > 1440
        LT(i) = yt(i)-1440;
    end;
end;

load IQDDays1996.dat % load in data of quiet and disturbed days in the year 1996
IDDs= IQDDays1996(2:61,1);

for II = IDDs
    AH = fileH(II,UT);
end

[m,i]=max(LT);

if i < 1440

    BH= [AH(1:60,i+1:1440) AH(1:60,1:i)];

elseif i == 1440

    BH= [AH(1:60,i:1440) AH(1:60,1:i)];
end;

BH =BH(1:60,1:1440) ;

% baseline values average of the 240 minutes flanking the local midnight
I=1:60;

```

```

baseHQ= (mean([BH(I,1:120) BH(I,1321:1440)]))'
for ii=1:1440
    matrBLHQ(:,ii)=baseHQ;
end
% minute departure

MinuteDpHQ= BH - matrBLHQ; % minute departure in H Q
% correcting non-cyclic variation Cyclic= (H1440-H1)/1439
% SH = MinuteDp - (n-1)Cyclic Minute departure

cyclicHQ = (MinuteDpHQ(I,1440:1440)-MinuteDpHQ(I,1:1))/1439;

for jj=1:1440
    mat24cyHQ(:,jj)=cyclicHQ;
end

% construction of the appropriate matrix for the Minute cyclic correction factor.
% that is (n-1)*cyclic correction factor mat24cy

t=1:1440;
p= (t-1);
for kk=1:60
    trp(kk,:)=p;
end
appmatcycHQ=trp.*mat24cyHQ;

SH=abs(MinuteDpHQ-appmatcycHQ);
% estimating the disturbance minute-based variation

janHQmin=SH(1:5,:);
janHQminMe=mean(janHQmin);

febHQmin=SH(6:10,:);
febHQminMe=mean(febHQmin);

%-----

marHQmin=SH(11:15,:);
marHQminMe=mean(marHQmin);

%-----

aprHQmin=SH(16:20,:);
aprHQminMe=mean(aprHQmin);

%-----

mayHQmin=SH(21:25,:);
mayHQminMe=mean(mayHQmin);

%-----

junHQmin=SH(26:30,:);

```



```

junHQminMe=mean(junHQmin);
%-----

julHQmin=SH(31:35,:);

julHQminMe=mean(julHQmin);
%-----
% August
augHQmin=SH(36:40,:);
augHQminMe=mean(augHQmin);
%-----

sepHQmin=SH(41:45,:);
sepHQminMe=mean(sepHQmin);
%-----

octHQmin=SH(46:50,:);
octHQminMe=mean(octHQmin);

% -----

novHQmin=SH(51:55,:);
novHQminMe=mean(novHQmin);

decHQmin=SH(56:60,:);
decHQminMe=mean(decHQmin);

%-----

HRESALL1996Jan=[HRESALL1996Jan;janHQminMe]
HRESALL1996Feb=[HRESALL1996Feb;febHQminMe]
HRESALL1996Mar=[HRESALL1996Mar;marHQminMe]
HRESALL1996Apr=[HRESALL1996Apr;aprHQminMe]
HRESALL1996May=[HRESALL1996May;mayHQminMe]
HRESALL1996Jun=[HRESALL1996Jun;junHQminMe]
HRESALL1996Jul=[HRESALL1996Jul;julHQminMe]
HRESALL1996Aug=[HRESALL1996Aug;augHQminMe]
HRESALL1996Sep=[HRESALL1996Sep;sepHQminMe]
HRESALL1996Oct=[HRESALL1996Oct;octHQminMe]
HRESALL1996Nov=[HRESALL1996Nov;novHQminMe]
HRESALL1996Dec=[HRESALL1996Dec;decHQminMe]

```

```

% this section computes the Sq, SD and whatever has to do with component D
%-----
for II = IDD's
    AD = fileD(II,UT);
    %end
end

[m,i]=max(LT);

    if i < 1440
        BD= [AD(1:60,i+1:1440) AD(1:60,1:i)];
    elseif i == 1440
        BD= [AD(1:60,i:1440) AD(1:60,1:i)];
    end;

    BD =BD(1:60,1:1440);

% the Minute departure needs to be calculated

% baseline values average of the 4 hrs flanking the local midnight

I=1:60;

baseDQ= (mean([BD(I,1:120) BD(I,1321:1440)]))'

for ii=1:1440
    matrBLDQ(:,ii)=baseDQ;
end

% minute departure
    MinuteDpDQ= BD - matrBLDQ; % Minute departure in H Q
%     MinuteDpZQ= rawHrlyZQ - matrBLZQ; % Minute departure in Z Q

% correction for non-cyclic variation
% SH = MinuteDp - (n-1)Cyclic
% cyclicDQ = (MinuteDpDQ(I,24:24)-MinuteDpDQ(I,1:1))/23;

cyclicDQ = (MinuteDpDQ(I,1440:1440)-MinuteDpDQ(I,1:1))/1439;

% cyclicZQ = (HourlyDpZQ(I,24:24)-MinuteDpZQ(I,1:1))/23;

% Construction of the matrix for 1440 values of cyclic consMEAt   mat24cy
for jj=1:1440
    mat24cyDQ(:,jj)=cyclicDQ;
end

% construction of the appropriate matrix for the minute-based cyclic correction factor.
% that is (n-1)*cyclic correction factor mat24cy
t=1:1440;
p= (t-1);
for kk=1:60
    trp(kk,:)=p;
end

```

```

appmatcycDQ=trp.*mat24cyDQ;
% appmatcycZQ=trp.*mat24cyZQ;
% tMEAefore the corrected solar daily variation will be SH
SD=abs(MinuteDpDQ- appmatcycDQ);
%
janDQmin=SD(1:5,:);
janDQminMe=mean(janDQmin);
%-----% February
febDQmin=SD(6:10,:);
febDQminMe=mean(febDQmin);
%-----
% March
marDQmin=SD(11:15,:);
marDQminMe=mean(marDQmin);
%-----% April

aprDQmin=SD(16:20,:);
aprDQminMe=mean(aprDQmin);
%-----
% May
mayDQmin=SD(21:25,:);
mayDQminMe=mean(mayDQmin);
%-----
% June
junDQmin=SD(26:30,:);
junDQminMe=mean(junDQmin);
%-----
% July
julDQmin=SD(31:35,:);
julDQminMe=mean(julDQmin);
% August
augDQmin=SD(36:40,:);
augDQminMe=mean(augDQmin);
%-----
% September

sepDQmin=SD(41:45,:);
sepDQminMe=mean(sepDQmin);
%-----
% October
octDQmin=SD(46:50,:);
octDQminMe=mean(octDQmin);
% -----
% November
novDQmin=SD(51:55,:);
novDQminMe=mean(novDQmin);
%-----
% December
decDQmin=SD(56:60,:);
decDQminMe=mean(decDQmin);

% -----

```

```

DRESALL1996Jan=[DRESALL1996Jan;janDQminMe]
DRESALL1996Feb=[DRESALL1996Feb;febDQminMe]
DRESALL1996Mar=[DRESALL1996Mar;marDQminMe]
DRESALL1996Apr=[DRESALL1996Apr;aprDQminMe]
DRESALL1996May=[DRESALL1996May;mayDQminMe]
DRESALL1996Jun=[DRESALL1996Jun;junDQminMe]
DRESALL1996Jul=[DRESALL1996Jul;julDQminMe]
DRESALL1996Aug=[DRESALL1996Aug;augDQminMe]
DRESALL1996Sep=[DRESALL1996Sep;sepDQminMe]
DRESALL1996Oct=[DRESALL1996Oct;octDQminMe]
DRESALL1996Nov=[DRESALL1996Nov;novDQminMe]
DRESALL1996Dec=[DRESALL1996Dec;decDQminMe]

%-----
% this section computes the Sq, SD and whatever has to do with component Z
for II = IDDs
    AZ = fileZ(II,UT);
    %end
end

[m,i]=max(LT);

    if i < 1440
        BZ= [AZ(1:60,i+1:1440) AZ(1:60,1:i)];
    elseif i == 1440
        BZ= [AZ(1:60,i:1440) AZ(1:60,1:i)];
    end;

    BZ =BZ(1:60,1:1440) ;

% calculation of the minute departure needs to be calculated

% % baseline values average of the 4 hrs flanking the local midnight
I=1:60;

% baseZQ= ((BZ(I,1:1)+BZ(I,2:2)+BZ(I,23:23)+BZ(I,24:24))/4);

baseZQ= (mean([BZ(I,1:120) BZ(I,1321:1440)]))'

% baseline values average of the 240minutes (4hrs) flanking the local midnight
for ii=1:1440

```

```

    matrBLZQ(:,ii)=baseZQ;
end

% Minute departure in minutes
    MinuteDpZQ= BZ - matrBLZQ; % minute-departure in H Q

% correction for non-cyclic variation

% cyclicZQ = (MinuteDpZQ(I,24:24)-MinuteDpZQ(I,1:1))/23;

cyclicZQ = (MinuteDpZQ(I,1440:1440)-MinuteDpZQ(I,1:1))/1439;

for jj=1:1440
    mat24cyZQ(:,jj)=cyclicZQ;
end

% matrix for the minute-based cyclic correction factor.

t=1:1440;
p= (t-1);
for kk=1:60
    trp(kk,:)=p;
end

    appmatcycZQ=trp.*mat24cyZQ;
% appmatcycZQ=trp.*mat24cyZQ;
% tMEAefore the corrected solar daily variation will be SH
SZ=abs(MinuteDpZQ- appmatcycZQ);
% SZ=MinuteDpZQ- appmatcycZQ;

% January
janZQmin=SZ(1:5,:);
janZQminMe=mean(janZQmin);
%-----
% feb
febZQmin=SZ(6:10,:);
febZQminMe=mean(febZQmin);

marZQmin=SZ(11:15,:);
marZQminMe=mean(marZQmin);
%-----

aprZQmin=SZ(16:20,:);
aprZQminMe=mean(aprZQmin);
%-----
% may
mayZQmin=SZ(21:25,:);
mayZQminMe=mean(mayZQmin);
%-----
% june
junZQmin=SZ(26:30,:);
junZQminMe=mean(junZQmin);
%-----

```

```

% july
julZQmin=SZ(31:35,:);
julZQminMe=mean(julZQmin);
%-----
augZQmin=SZ(36:40,:);
augZQminMe=mean(augZQmin);
%-----
sepZQmin=SZ(41:45,:);
sepZQminMe=mean(sepZQmin);
%-----

octZQmin=SZ(46:50,:);
octZQminMe=mean(octZQmin);
% -----

novZQmin=SZ(51:55,:);
novZQminMe=mean(novZQmin);
%-----
decZQmin=SZ(56:60,:);
decZQminMe=mean(decZQmin);
%-----
ZRESALL1996Jan=[ZRESALL1996Jan;janZQminMe]

ZRESALL1996Feb=[ZRESALL1996Feb;febZQminMe]

ZRESALL1996Mar=[ZRESALL1996Mar;marZQminMe]

ZRESALL1996Apr=[ZRESALL1996Apr;aprZQminMe]

ZRESALL1996May=[ZRESALL1996May;mayZQminMe]

ZRESALL1996Jun=[ZRESALL1996Jun;junZQminMe]

ZRESALL1996Jul=[ZRESALL1996Jul;julZQminMe]

ZRESALL1996Aug=[ZRESALL1996Aug;augZQminMe]

ZRESALL1996Sep=[ZRESALL1996Sep;sepZQminMe]

ZRESALL1996Oct=[ZRESALL1996Oct;octZQminMe]

ZRESALL1996Nov=[ZRESALL1996Nov;novZQminMe]

ZRESALL1996Dec=[ZRESALL1996Dec;decZQminMe]

end % n=64

% ----- writing H component result in text format -----

dlmwrite('HRESALL1996Jan.txt',HRESALL1996Jan);
dlmwrite('HRESALL1996Feb.txt',HRESALL1996Feb);
dlmwrite('HRESALL1996Mar.txt',HRESALL1996Mar);
dlmwrite('HRESALL1996Apr.txt',HRESALL1996Apr);

```

```

dlmwrite('HRESALL1996May.txt',HRESALL1996May);
dlmwrite('HRESALL1996Jun.txt',HRESALL1996Jun);
dlmwrite('HRESALL1996Jul.txt',HRESALL1996Jul);
dlmwrite('HRESALL1996Aug.txt',HRESALL1996Aug);

dlmwrite('HRESALL1996Sep.txt',HRESALL1996Sep);
dlmwrite('HRESALL1996Oct.txt',HRESALL1996Oct);
dlmwrite('HRESALL1996Nov.txt',HRESALL1996Nov);
dlmwrite('HRESALL1996Dec.txt',HRESALL1996Dec);

% ----writing D component result in excel format -----

dlmwrite('DRESALL1996Jan.txt',DRESALL1996Jan);
dlmwrite('DRESALL1996Feb.txt',DRESALL1996Feb);
dlmwrite('DRESALL1996Mar.txt',DRESALL1996Mar);
dlmwrite('DRESALL1996Apr.txt',DRESALL1996Apr);

dlmwrite('DRESALL1996May.txt',DRESALL1996May);
dlmwrite('DRESALL1996Jun.txt',DRESALL1996Jun);
dlmwrite('DRESALL1996Jul.txt',DRESALL1996Jul);
dlmwrite('DRESALL1996Aug.txt',DRESALL1996Aug);

dlmwrite('DRESALL1996Sep.txt',DRESALL1996Sep);
dlmwrite('DRESALL1996Oct.txt',DRESALL1996Oct);
dlmwrite('DRESALL1996Nov.txt',DRESALL1996Nov);
dlmwrite('DRESALL1996Dec.txt',DRESALL1996Dec);

% ----- writing Z component result in excel format -----

dlmwrite('ZRESALL1996Jan.txt',ZRESALL1996Jan);
dlmwrite('ZRESALL1996Feb.txt',ZRESALL1996Feb);
dlmwrite('ZRESALL1996Mar.txt',ZRESALL1996Mar);
dlmwrite('ZRESALL1996Apr.txt',ZRESALL1996Apr);

dlmwrite('ZRESALL1996May.txt',ZRESALL1996May);
dlmwrite('ZRESALL1996Jun.txt',ZRESALL1996Jun);
dlmwrite('ZRESALL1996Jul.txt',ZRESALL1996Jul);
dlmwrite('ZRESALL1996Aug.txt',ZRESALL1996Aug);

dlmwrite('ZRESALL1996Sep.txt',ZRESALL1996Sep);
dlmwrite('ZRESALL1996Oct.txt',ZRESALL1996Oct);
dlmwrite('ZRESALL1996Nov.txt',ZRESALL1996Nov);
dlmwrite('ZRESALL1996Dec.txt',ZRESALL1996Dec);

store_time(2, :) = clock; % [year month day hour minute seconds]

sqResTime=(store_time(2,4:6)-store_time(1,4:6))*[3600;60;1];

fprintf('\n\nThe time spent to process the Sq results: %f minutes\n',sqResTime/60);

store_time(3, :) = clock; % [year month day hour minute seconds]
% ----- analysis of execution time spent by code segments, functions and subroutines
profile viewer;

```

B2: Matlab (parallel) program for Analysis Sq Daily Variations of Geomagnetic Field

Sq MATLAB PROGRAM LISTING - PARALLEL SOURCE CODE

```
-----
% Temporal variations of geomagnetic fields by Ale F. under supervision of Prof.
%Ibidapo-Obe; Dr. T.A. % Fashanu & Dr. % O.A.Agboola

% To study the spatial distribution of the ionospheric current responsible
% for geomagnetic field variation using parallel computing

% readingExcel function called for filtering and reshaping
% functions called and used are: readingExcel, txtFilter
% sqpicker.m funtion calls sqcontour.m

% data needed are geomagnetic field data of 64 stations,
longlat1996.dat, longlat1996a.dat, IQDDays1996.dat
echo off;
profile on;
diary on;
store_time = zeros( 3, 6);

store_time(1,:)=clock; % [year month day hour min sec]

load longlat1996.dat % Longitude and latitudes of the 64 stations

longlat1996R=longlat1996; % use longlat1996R (64x2) in the plotting

longlat1996R(1,:)=[];

load IQDDays1996.dat % load in data of quiet days in the year 1996
IDDs= IQDDays1996(2:61,1);

%*****
if (matlabpool('size') > 0)==1 % checking if matlabpool is open
    matlabpool close
end

pro=input('Please enter the number of processors you wish to use: ');

for n_labs = 1:pro

disp(['n_labs = ' int2str(n_labs)]);

store_time(1, :) = clock; % [year month day hour minute seconds]

% if n_labs > 0 ;
matlabpool('open', n_labs);

% matlabpool open local 8

store_time(2, :) = clock;

HRESALL1996Jan=[];
```



```

HRESALL1996Feb=[];
HRESALL1996Mar=[];
HRESALL1996Apr=[];
HRESALL1996May=[];
HRESALL1996Jun=[];
HRESALL1996Jul=[];
HRESALL1996Aug=[];
HRESALL1996Sep=[];
HRESALL1996Oct=[];
HRESALL1996Nov=[];
HRESALL1996Dec=[];

```

```

DRESALL1996Jan=[];
DRESALL1996Feb=[];
DRESALL1996Mar=[];
DRESALL1996Apr=[];
DRESALL1996May=[];
DRESALL1996Jun=[];
DRESALL1996Jul=[];
DRESALL1996Aug=[];
DRESALL1996Sep=[];
DRESALL1996Oct=[];
DRESALL1996Nov=[];
DRESALL1996Dec=[];

```

```

ZRESALL1996Jan=[];
ZRESALL1996Feb=[];
ZRESALL1996Mar=[];
ZRESALL1996Apr=[];
ZRESALL1996May=[];
ZRESALL1996Jun=[];
ZRESALL1996Jul=[];
ZRESALL1996Aug=[];
ZRESALL1996Sep=[];
ZRESALL1996Oct=[];
ZRESALL1996Nov=[];
ZRESALL1996Dec=[];

```

```

obslist =
['ABK'; 'ALE'; 'AMS'; 'BDV'; 'BEL'; 'BFE'; 'BLC'; 'BNG'; 'BOU'; 'BRW'; 'BSL'; 'CBB'; 'C
LF'; 'CMO'; 'CNB'; 'CZT'; 'DLR'; 'DRV'; 'ESK'; 'EYR'; 'FCC'; 'FRD'; 'FRN'; 'FUR'; 'GDH'
; 'GLN'; 'GNA'; 'GUA'; 'HAD'; 'HER'; 'HON'; 'IQA'; 'KAK'; 'KOU'; 'LER'; 'LOV'; 'MBC'; 'M
BO'; 'MEA'; 'MMB'; 'NAQ'; 'NCK'; 'NEW'; 'NGK'; 'NUR'; 'OTT'; 'PAF'; 'PBQ'; 'PHU'; 'PPT'
; 'RES'; 'SBA'; 'SIT'; 'SJG'; 'SOD'; 'STJ'; 'TAM'; 'TAN'; 'THL'; 'THY'; 'TUC'; 'VIC'; 'W
NG'; 'YKC'] ;

```

```

% selecting stations by X, Y, Z or H, D, Z components (64x3 = 192)

```

```

parfor n=1:64

```

```

    obs=obslist(n,:);

```

```

    Xobs=cat(2,'X',obs, '.xls');
    Yobs=cat(2,'Y',obs, '.xls');
    Hobs=cat(2,'H',obs, '.xls');
    Dobs=cat(2,'D',obs, '.xls');
    Zobs=cat(2,'Z',obs, '.xls');

```

```

        if exist(Xobs, 'file');

% calling function filterChar to remove headerlines and reshape
    Xfile=readingExcel(Xobs);
    Yfile=readingExcel(Yobs);
    Zfile=readingExcel(Zobs);

% getting the universal time (UT) in seconds

    UT = 1:1440; %

% magnetic field values

fileX = Xfile(1:366,1:1440);
fileY = Yfile(1:366,1:1440);
fileZ = Zfile(1:366,1:1440);

% converting X and Y components to H and D respectively

fileH = sqrt(fileX.^2 + fileY.^2);
fileD = (180/pi)*600*atan(fileY./fileX);

else

%calling function filterChar to remove headerlines and reshape
    Hfile=readingExcel(Hobs);
    Dfile=readingExcel(Dobs);
    Zfile=readingExcel(Zobs);

    UT = 1:1440; % this section config. the universal time for referencing

        % magnetic field values
        fileH = Hfile(1:366,1:1440);
        fileD = Dfile(1:366,1:1440);
        fileZ = Zfile(1:366,1:1440);

    end

Long(n) = longlat1996(n+1:n+1,1);
yt = UT + round ((Long(n)/15)*60);

% converting the LT to the time that must strictly be between 1
% and 1440 minutes

% for i=1:1440
%
%         if yt(i) <= 1440;
%             LT(i) = yt(i);
%         elseif yt(i) > 1440
%             LT(i) = yt(i)-1440;
%         end;
%     end;

% applying logical indexing for LT

```

```

LT=yt;
LT(LT>1440)=LT(LT>1440)-1440;

for II = IDDs
    AH = fileH(II,UT);
end

[m,i]=max(LT);

if i < 1440

    BH= [AH(1:60,i+1:1440) AH(1:60,1:i)];

elseif i == 1440

    BH= [AH(1:60,i:1440) AH(1:60,1:i)];
end;

BH =BH(1:60,1:1440) ;

% baseline values average of the 240 minutes flanking the local midnight
I=1:60;

baseHQ = (mean([BH(I,1:120) BH(I,1321:1440)]'))';
% parallel conversion
matrBLHQ=baseHQ*ones(1,1440);

% minute departure

MinuteDpHQ= BH - matrBLHQ; % minute departure in H Q

% correcting non-cyclic variation Cyclic= (H1440-H1)/1439
% SH = MinuteDp - (n-1)Cyclic Minute departure

cyclicHQ = (MinuteDpHQ(I,1440:1440)-MinuteDpHQ(I,1:1))/1439;

mat24cyHQ=cyclicHQ*ones(1,1440); %// for parallel computation
% construction of the appropriate matrix for the Minute cyclic correction
factor

t=1:1440;
p= (t-1);
%
trp=(p'*ones(1,60))';
appmatcycHQ=trp.*mat24cyHQ;

SH=abs(MinuteDpHQ-appmatcycHQ);

% estimating the disturbance minute-based variation

janHQmin=SH(1:5,:);
janHQminMe=mean(janHQmin);

febHQmin=SH(6:10,:);
febHQminMe=mean(febHQmin);

```

```

%-----

marHQmin=SH(11:15,:);
marHQminMe=mean(marHQmin);

%-----

aprHQmin=SH(16:20,:);
aprHQminMe=mean(aprHQmin);

%-----

mayHQmin=SH(21:25,:);
mayHQminMe=mean(mayHQmin);
%-----
junHQmin=SH(26:30,:);
junHQminMe=mean(junHQmin);
%-----
julHQmin=SH(31:35,:);
julHQminMe=mean(julHQmin);
%-----
%August
augHQmin=SH(36:40,:);
augHQminMe=mean(augHQmin);
%-----

sepHQmin=SH(41:45,:);
sepHQminMe=mean(sepHQmin);
%-----
octHQmin=SH(46:50,:);
octHQminMe=mean(octHQmin);

%-----

novHQmin=SH(51:55,:);
novHQminMe=mean(novHQmin);

decHQmin=SH(56:60,:);
decHQminMe=mean(decHQmin);

%-----
HRESALL1996Jan=[HRESALL1996Jan;janHQminMe]

HRESALL1996Feb=[HRESALL1996Feb;febHQminMe]

HRESALL1996Mar=[HRESALL1996Mar;marHQminMe]

HRESALL1996Apr=[HRESALL1996Apr;aprHQminMe]

HRESALL1996May=[HRESALL1996May;mayHQminMe]

HRESALL1996Jun=[HRESALL1996Jun;junHQminMe]

HRESALL1996Jul=[HRESALL1996Jul;julHQminMe]

HRESALL1996Aug=[HRESALL1996Aug;augHQminMe]

HRESALL1996Sep=[HRESALL1996Sep;sepHQminMe]

HRESALL1996Oct=[HRESALL1996Oct;octHQminMe]

```

```

HRESALL1996Nov=[HRESALL1996Nov;novHQminMe]

HRESALL1996Dec=[HRESALL1996Dec;decHQminMe]

% this section computes the Sq, SD and whatever has to do with component D
%
%-----
for II = IDD,
    AD = fileD(II,UT);
    %end
end

[m,i]=max(LT);

    if i < 1440
        BD= [AD(1:60,i+1:1440) AD(1:60,1:i)];
    elseif i == 1440
        BD= [AD(1:60,i:1440) AD(1:60,1:i)];
    end;

    BD =BD(1:60,1:1440) ;

% the Minute departure needs to be calculated
% daily Minute values of the elements starts from row 2

I=1:60;

baseDQ= (mean([BD(I,1:120) BD(I,1321:1440)]'))'

matrBLDQ=baseHQ*ones(1,1440);

% minute departure
    MinuteDpDQ= BD - matrBLDQ; % Minute departure in H Q
%
    MinuteDpZQ= rawHrlyZQ - matrBLZQ; % Minute departure in Z Q

% %
% effort to correct for non-cyclic variation
cyclicDQ = (MinuteDpDQ(I,1440:1440)-MinuteDpDQ(I,1:1))/1439;

% % corrected Solar daily variation will be SH, SqH fpr quiet and Sd for

mat24cyDQ=cyclicDQ*ones(1,1440);

% construction of the appropriate matrix for the minute-based cyclic
correction factor.
% that is (n-1)*cyclic correction factor mat24cy
t=1:1440;
p= (t-1);
for kk=1:60
    trp(kk,:)=p;
end

appmatcycDQ=trp.*mat24cyDQ;
% appmatcycZQ=trp.*mat24cyZQ;

```

```

% tMEAefore the corrected solar daily variation will be SH
SD=abs(MinuteDpDQ- appmatcycDQ);
% SZ=MinuteDpZQ- appmatcycZQ;
%
% %mean monMEAy Minute values of S is mean(SH)

janDQmin=SD(1:5,:);
janDQminMe=mean(janDQmin);
%-----
% February
febDQmin=SD(6:10,:);
febDQminMe=mean(febDQmin);
%-----
% March
marDQmin=SD(11:15,:);
marDQminMe=mean(marDQmin);
%-----
% April

aprDQmin=SD(16:20,:);
aprDQminMe=mean(aprDQmin);
%-----
% May
mayDQmin=SD(21:25,:);
mayDQminMe=mean(mayDQmin);
%-----
% June
junDQmin=SD(26:30,:);
junDQminMe=mean(junDQmin);
%-----
% July
julDQmin=SD(31:35,:);
julDQminMe=mean(julDQmin);
%-----
% August
augDQmin=SD(36:40,:);
augDQminMe=mean(augDQmin);
%-----
%September

sepDQmin=SD(41:45,:);
sepDQminMe=mean(sepDQmin);
%-----
%October
octDQmin=SD(46:50,:);
octDQminMe=mean(octDQmin);
%-----
%November
novDQmin=SD(51:55,:);
novDQminMe=mean(novDQmin);
%-----
%December
decDQmin=SD(56:60,:);
decDQminMe=mean(decDQmin);
%-----

DRESALL1996Jan=[DRESALL1996Jan;janDQminMe]

DRESALL1996Feb=[DRESALL1996Feb;febDQminMe]

```

```

DRESALL1996Mar=[DRESALL1996Mar;marDQminMe]

DRESALL1996Apr=[DRESALL1996Apr;aprDQminMe]

DRESALL1996May=[DRESALL1996May;mayDQminMe]

DRESALL1996Jun=[DRESALL1996Jun;junDQminMe]

DRESALL1996Jul=[DRESALL1996Jul;julDQminMe]

DRESALL1996Aug=[DRESALL1996Aug;augDQminMe]

DRESALL1996Sep=[DRESALL1996Sep;sepDQminMe]

DRESALL1996Oct=[DRESALL1996Oct;octDQminMe]

DRESALL1996Nov=[DRESALL1996Nov;novDQminMe]

DRESALL1996Dec=[DRESALL1996Dec;decDQminMe]

%-----
% this section computes the Sq, SD and whatever has to do with component Z
%-----

for II = IDDs,
    AZ = fileZ(II,UT);
    %end
end

[m,i]=max(LT);

    if i < 1440
        BZ= [AZ(1:60,i+1:1440) AZ(1:60,1:i)];
    elseif i == 1440
        BZ= [AZ(1:60,i:1440) AZ(1:60,1:i)];
    end;

    BZ =BZ(1:60,1:1440) ;

% baseline values average of the 240 minutes flanking the local midnight
I=1:60;

baseZQ= (mean([BH(I,1:120) BH(I,1321:1440)]'))'

% baseline values average of the 240minutes (4hrs) flanking the local
midnight
matrBLZQ=baseZQ*ones(1,1440);

% minute departure in minutes
MinuteDpZQ= BZ - matrBLZQ; % minute-departure in H Q

% effort to correct for non-cyclic variation
cyclicZQ = (MinuteDpZQ(I,1440:1440)-MinuteDpZQ(I,1:1))/1439;

% corrected Solar daily variation will be SH, SqH fpr quiet and Sd for

```

```

mat24cyZQ=cyclicZQ*ones(1,1440);

% matrix for the minute-based cyclic correction factor

t=1:1440;
p= (t-1);
for kk=1:60
    trp(kk,:)=p;
end

appmatcycZQ=trp.*mat24cyZQ;
% tMEAefore the corrected solar daily variation will be SH
SZ=abs(HourlyDpZQ- appmatcycZQ);

%January
janZQmin=SZ(1:5,:);
janZQminMe=mean(janZQmin);
%-----
% feb
febZQmin=SZ(6:10,:);
febZQminMe=mean(febZQmin);
%-----
marZQmin=SZ(11:15,:);
marZQminMe=mean(marZQmin);
%-----
aprZQmin=SZ(16:20,:);
aprZQminMe=mean(aprZQmin);
%-----
% may
mayZQmin=SZ(21:25,:);
mayZQminMe=mean(mayZQmin);
%-----
% june
junZQmin=SZ(26:30,:);
junZQminMe=mean(junZQmin);
%-----
% july
julZQmin=SZ(31:35,:);
julZQminMe=mean(julZQmin);
%-----
% augZQmin=SZ(36:40,:);
augZQminMe=mean(augZQmin);
%-----
% sepZQmin=SZ(41:45,:);
sepZQminMe=mean(sepZQmin);
%-----

octZQmin=SZ(46:50,:);
octZQminMe=mean(octZQmin);
% -----
novZQmin=SZ(51:55,:);
novZQminMe=mean(novZQmin);
%-----

decZQmin=SZ(56:60,:);
decZQminMe=mean(decZQmin);
%-----

```



```

ZRESALL1996Jan=[ZRESALL1996Jan;janZQminMe]

ZRESALL1996Feb=[ZRESALL1996Feb;febZQminMe]

ZRESALL1996Mar=[ZRESALL1996Mar;marZQminMe]

ZRESALL1996Apr=[ZRESALL1996Apr;aprZQminMe]

ZRESALL1996May=[ZRESALL1996May;mayZQminMe]

ZRESALL1996Jun=[ZRESALL1996Jun;junZQminMe]

ZRESALL1996Jul=[ZRESALL1996Jul;julZQminMe]

ZRESALL1996Aug=[ZRESALL1996Aug;augZQminMe]

ZRESALL1996Sep=[ZRESALL1996Sep;sepZQminMe]

ZRESALL1996Oct=[ZRESALL1996Oct;octZQminMe]

ZRESALL1996Nov=[ZRESALL1996Nov;novZQminMe]

ZRESALL1996Dec=[ZRESALL1996Dec;decZQminMe]

end % n=64 - end of parfor loop

% ----- writing H component result in text format
dlmwrite('HRESALL1996Jan.txt',HRESALL1996Jan);
dlmwrite('HRESALL1996Feb.txt',HRESALL1996Feb);
dlmwrite('HRESALL1996Mar.txt',HRESALL1996Mar);
dlmwrite('HRESALL1996Apr.txt',HRESALL1996Apr);
dlmwrite('HRESALL1996May.txt',HRESALL1996May);
dlmwrite('HRESALL1996Jun.txt',HRESALL1996Jun);
dlmwrite('HRESALL1996Jul.txt',HRESALL1996Jul);
dlmwrite('HRESALL1996Aug.txt',HRESALL1996Aug);

dlmwrite('HRESALL1996Sep.txt',HRESALL1996Sep);
dlmwrite('HRESALL1996Oct.txt',HRESALL1996Oct);
dlmwrite('HRESALL1996Nov.txt',HRESALL1996Nov);
dlmwrite('HRESALL1996Dec.txt',HRESALL1996Dec);

% ----writing D component results
dlmwrite('DRESALL1996Jan.txt',DRESALL1996Jan);
dlmwrite('DRESALL1996Feb.txt',DRESALL1996Feb);
dlmwrite('DRESALL1996Mar.txt',DRESALL1996Mar);
dlmwrite('DRESALL1996Apr.txt',DRESALL1996Apr);

dlmwrite('DRESALL1996May.txt',DRESALL1996May);
dlmwrite('DRESALL1996Jun.txt',DRESALL1996Jun);
dlmwrite('DRESALL1996Jul.txt',DRESALL1996Jul);
dlmwrite('DRESALL1996Aug.txt',DRESALL1996Aug);

dlmwrite('DRESALL1996Sep.txt',DRESALL1996Sep);
dlmwrite('DRESALL1996Oct.txt',DRESALL1996Oct);
dlmwrite('DRESALL1996Nov.txt',DRESALL1996Nov);
dlmwrite('DRESALL1996Dec.txt',DRESALL1996Dec);

```

```

% ----- writing Z component results

dlmwrite('ZRESALL1996Jan.txt',ZRESALL1996Jan);
dlmwrite('ZRESALL1996Feb.txt',ZRESALL1996Feb);
dlmwrite('ZRESALL1996Mar.txt',ZRESALL1996Mar);
dlmwrite('ZRESALL1996Apr.txt',ZRESALL1996Apr);

dlmwrite('ZRESALL1996May.txt',ZRESALL1996May);
dlmwrite('ZRESALL1996Jun.txt',ZRESALL1996Jun);
dlmwrite('ZRESALL1996Jul.txt',ZRESALL1996Jul);
dlmwrite('ZRESALL1996Aug.txt',ZRESALL1996Aug);
dlmwrite('ZRESALL1996Sep.txt',ZRESALL1996Sep);
dlmwrite('ZRESALL1996Oct.txt',ZRESALL1996Oct);
dlmwrite('ZRESALL1996Nov.txt',ZRESALL1996Nov);
dlmwrite('ZRESALL1996Dec.txt',ZRESALL1996Dec);

%end of spmd
store_time(3, :) = clock; % [year month day hour minute seconds]

matlabpool close;

store_time(4, :) = clock;
conTime1(n_labs)=(store_time(2,4:6)-store_time(1,4:6))*[3600;60;1]
conTime2(n_labs)=(store_time(4,4:6)-store_time(3,4:6))*[3600;60;1]

conTime(n_labs)=conTime1(n_labs)+conTime2(n_labs);
sqTime(n_labs)=(store_time(3,4:6)-store_time(2,4:6))*[3600;60;1]

allTime(n_labs)=(store_time(4,4:6)-store_time(1,4:6))*[3600;60;1]
% fprintf('\n\nThe time spent to process the 3D contour display:
%f\n',contourTime);
xx(n_labs)=n_labs;

%computing speedup and efficiency
sp(n_labs)=sqTime(1)/allTime(n_labs); % speedup
eff(n_labs)=(sp(n_labs)/n_labs)*100; % efficiency

end % n_labs

plot(xx,sqTime);

ylabel('Time (s) ');
xlabel('Number of processors');

title(['Parallel processing using ',int2str(pro), ' processors']);
grid on;
fprintf('\n\nThe time spent to open connections: %f minutes\n',conTime/60);

fprintf('\n\nThe time spent to execute Sq: %f minutes\n',sqTime/60);

fprintf('\n\nTotal time spent: %f minutes\n',allTime/60);
disp('Tabulating Processing Timing Result using Sq block')
disp('-----');

fprintf('\n\n');

```

```

disp('|No of processor |Execution Time(sqTime)| Speedup    | Efficiency
(%)');
disp('-----');
for b=1:pro

    fprintf(['|
\t',num2str(b),'\t\t|\t\t',num2str(sqTime(b)),num2str(sp(b)),'\t\t',num2str
(eff(b)),'\t\t|\n']);

end
fprintf('\n\n');

disp('Computing the Speedup and Efficiency of the parallel processing');

disp('Speedup = serial algorithm time/parallel algorithm time');

sp=sqTime(1)/sqTime(b) %speedup

eff=(sp/b)*100;
disp('*****');
fprintf('\nSpeedup = %2.4f \n',sp);
fprintf('\nEfficiency = %2.2f% \n',eff);

fprintf('\n\n');
% tabulating the timing results using allTime
disp('Tabulating Processing Timing Result using total time')
disp('-----');

fprintf('\n\n');

disp('| No of processor    | Execution Time(allTime)|');
disp('-----');

for b1=1:pro

    fprintf(['|
\t\t',num2str(b1),'\t\t\t|\t\t',num2str(allTime(b1)),'\t\t|\n']);
end

sp1=allTime(1)/allTime(b1); %speedup

eff1=(sp1/b1)*100;
disp('****computing speedup and efficiency using total time****');
fprintf('\nSpeedup = %2.4f \n ',sp1);
fprintf('\nEfficiency = %2.2f% \n',eff1);
save parSqVars.mat;
diary('sqParResVar.txt');
diary off;
hgsave('parSqFig');
profile off
profsave(profile('info'),'Sqprofile_results')

```

B3: Matlab program for Sq Graphical Users' Interface

```
function varargout = Sq_Application(varargin)
% Sq_Application M-file for Sq_Application.fig
% Sq_Application, by itself, creates a new Sq_Application or raises the
existing singleton*.
%
% H = Sq_Application returns the handle to a new Sq_Application or the
%handle to the existing singleton*.
%
% Sq_Application('CALLBACK',hObject,eventData,handles,...) calls the local
% function named CALLBACK in Sq_Application.M with the given input
arguments.
%
% Sq_Application('Property','Value',...) creates a new Sq_Application or
%raises the
% existing singleton*. Starting from the left, property value pairs are
% applied to the GUI before Sq_Application_OpeningFcn gets called. An
% unrecognized property name or invalid value makes property application
% stop. All inputs are passed to Sq_Application_OpeningFcn via varargin.
%
% *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
% instance to run (singleton)".
%
% Sq_Application Program v.2 was developed by ALE Felix last modified 14-
%Jun-2012 12:20:49

% Begin initialization code
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @Sq_Application_OpeningFcn, ...
                  'gui_OutputFcn',  @Sq_Application_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code

% --- Executes just before Sq_Application is made visible.
function Sq_Application_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)
% varargin    command line arguments to Sq_Application (see VARARGIN)

% Choose default command line output for Sq_Application
handles.output = hObject;

% Update handles structure
```

```

guidata(hObject, handles);

% UIWAIT makes Sq_Application wait for user response (see UIRESUME)
% uiwait(handles.figure1);
clc
movegui('center')
set(handles.pbClose, 'Enable', 'off');
set(handles.pbPlot, 'Enable', 'off');
set(handles.pbSurfer, 'Enable', 'off');

% variations (worldwide observation) by Ale et al. 2012', 'MenuBar', 'None');
% variations (worldwide observation) by Ale et al. 2012', 'MenuBar', 'None');
% --- Outputs from this function are returned to the command line.
function varargout = Sq_Application_OutputFcn(hObject, eventdata, handles)
% varargout cell array for returning output args (see VARARGOUT);
% hObject handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pbPlot.
function pbPlot_Callback(hObject, eventdata, handles)
% hObject handle to pbPlot (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

load 'longlat1996.dat';
longlat1996R=longlat1996;
longlat1996R(1,:)=[];

sqname='HRESALL1996'; % sq parent name (sqH, sqD, sqZ)

% months list
monlist=['Jan'; 'Feb'; 'Mar'; 'Apr'; 'May'; 'Jun'; 'Jul'; 'Aug'; 'Sep'; 'Oct'; 'Nov';
'Dec'];

for m=1:12 % iteration of months from jan - dec

mon=monlist(m,:);

sqval=cat(2,sqname,mon, '.txt'); % sq values

    if exist(sqval, 'file');
        %load 'sqval';

        sqdata=load(sqval);
        [a f]=size(sqdata);

for t=1:f

    Z = sqdata(:,t);

```

```

h1=rem(t,60);h11=num2str(h1);% min

h2=fix(t/60);h22=num2str(h2); %hrs

X = longlat1996R(:,1); % longitude of the 64 stations
Y = longlat1996R(:,2); % latitude of the 64 stations

[Xi,Yi]=meshgrid(X,Y);

[Xi,Yi,Zi]=kriginggriddata(X,Y,Z,Xi,Yi, 'kriging');

grid(handles.axes1,'on');
    conts=0:5:60;
    contourf(handles.axes1,Xi,Yi,Zi,conts,'k'); % colour
fill
    % surf(handles.axes1,peaks(35));
    % contour(Xi,Yi,Zi,conts,'k'); % no colour

    set(gca,'XTick',(0:30:360));
    set(gca,'YTick',(-90:15:90));

    xlim([0 360]);
    ylim([-90 90]);

    colormap(jet(8));
h=colorbar('peer',handles.axes1);

title(h, 'SqH'); % labelling the colorbar

    xlabel('Longitude (deg)');
    ylabel('Latitude (deg)');

title ([mon,' ', 'SqH - Minute: ', num2str(t), ' Hour:',h22,':',h11]);

pause(0.01);

end % end t

    end %end if

end %end m

% --- Executes on button press in pbClose.
function pbClose_Callback(hObject, eventdata, handles)
% hObject handle to pbClose (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
button = questdlg('Ready to quit?', ...
    'Exit Sq Application','Yes','No','No');
switch button
    case 'Yes',
        disp('Exiting MATLAB');
        exit;
    case 'No',
        quit cancel;
end

```

```

% --- Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pbInstructions.
function pbInstructions_Callback(hObject, eventdata, handles)
% hObject    handle to pbInstructions (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

a=msgbox(sprintf('1. This movie display is the result of parallel
computation of Sq(H);\n2. Geomagnetic data was obtained from 64
observatories in 20 countries worldwide, courtesy of INTERMAGNET; \n3. One-
minute (60-second) time resolution data was used; \n4. Kriging
interpolation algorithm used provided data gridding for the unobserved
values; \n5. Surfer 8 (evaluation copy) installation was used to provide
the kriging method as an interface application to Matlab codes; \n6. Click
the OK button and install Surfer and press PLOT button
afterward.'),'Installation Guide for Sq(H) movie','modal');
uiwait(a)
set(handles.pbClose,'Enable','on');
set(handles.pbPlot,'Enable','on');
set(handles.pbSurfer,'Enable','on');

% --- Executes on button press in pbSurfer.
function pbSurfer_Callback(hObject, eventdata, handles)
% hObject    handle to pbSurfer (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% add this segment to access surfer-kriging method via the installed
% application
button2 = questdlg('Do you want to Install Surfer-kriging?', ...
    'Installation of Surfer (Evaluation copy)','Yes','No','No');
switch button2
    case 'Yes',
        disp('Installing Surfer Application');
        open('surferDemo.exe'); % evaluation copy of Surfer 8
from Golden Software

%winopen('C:\Users\user\Documents\MATLAB\finalPackage060112\data\surferDemo
.exe');
    case 'No',
        quit cancel;
end

```

APPENDIX C: High Resolution Sq(D) Variations in 2-D

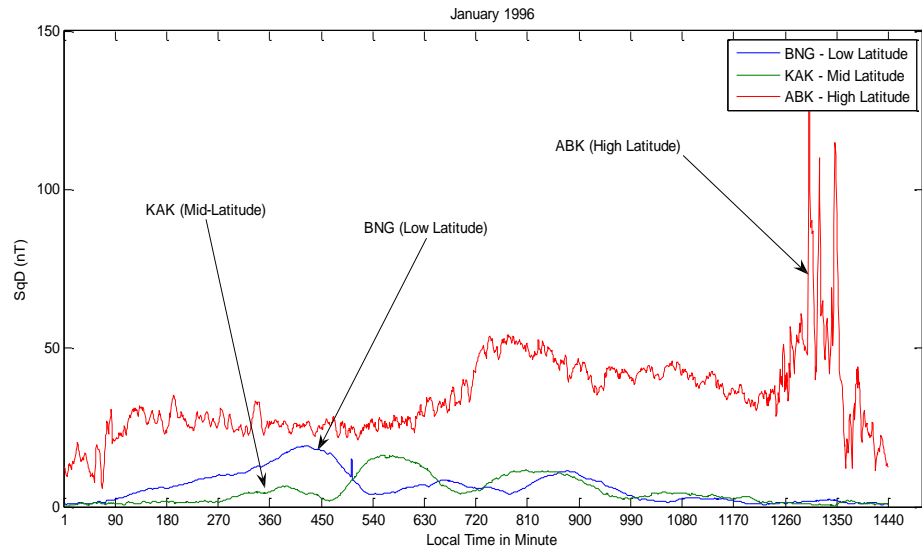


Fig. C1: High resolution of SqD across Low-, Mid- and High Latitudes – January

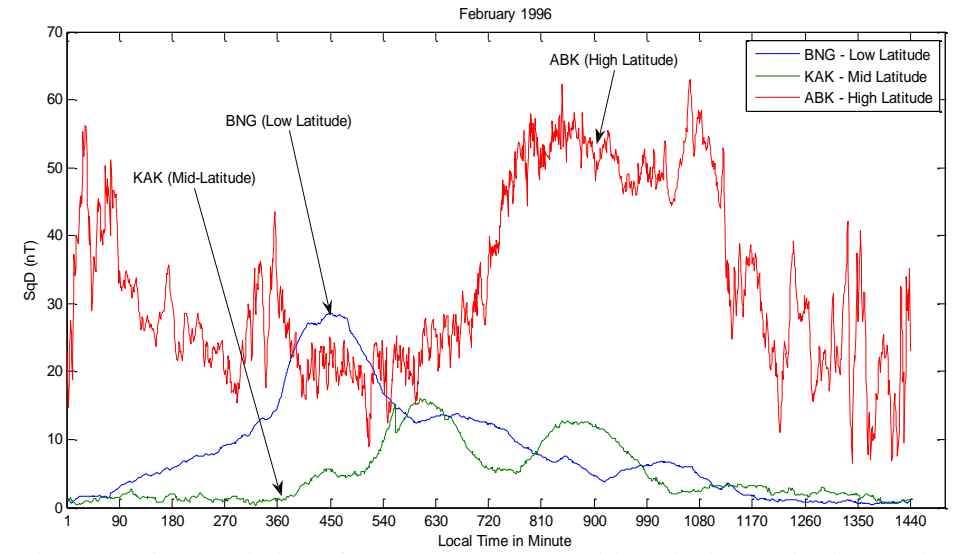


Fig. C2: High resolution of SqD across Low-, Mid- and High Latitudes – February

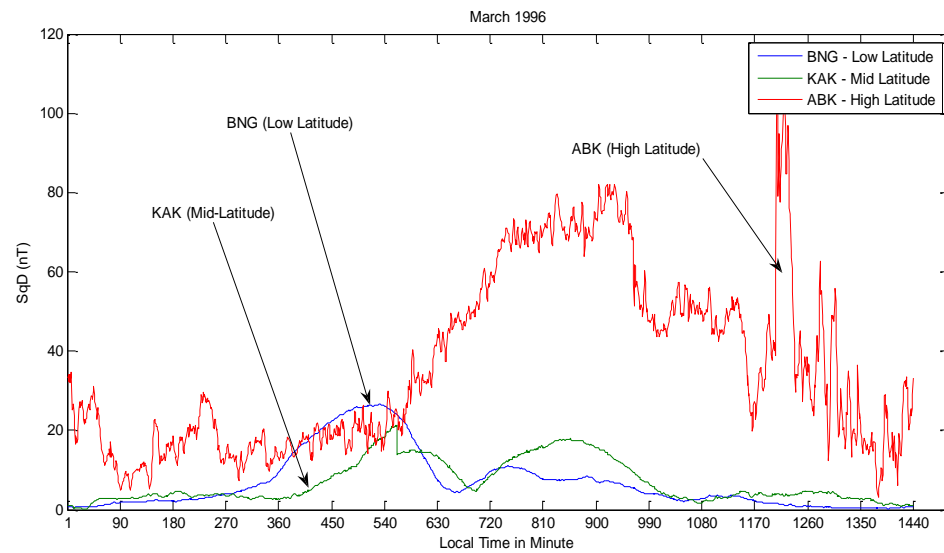


Fig. C3: High resolution of SqD across Low-, Mid- and High Latitudes – March

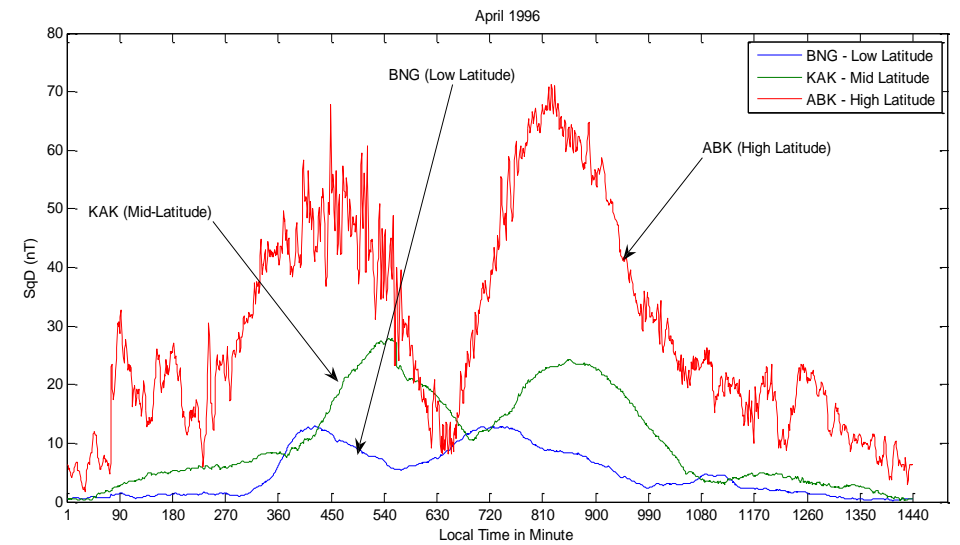


Fig. C4: High resolution of SqD across Low-, Mid- and High Latitudes – April

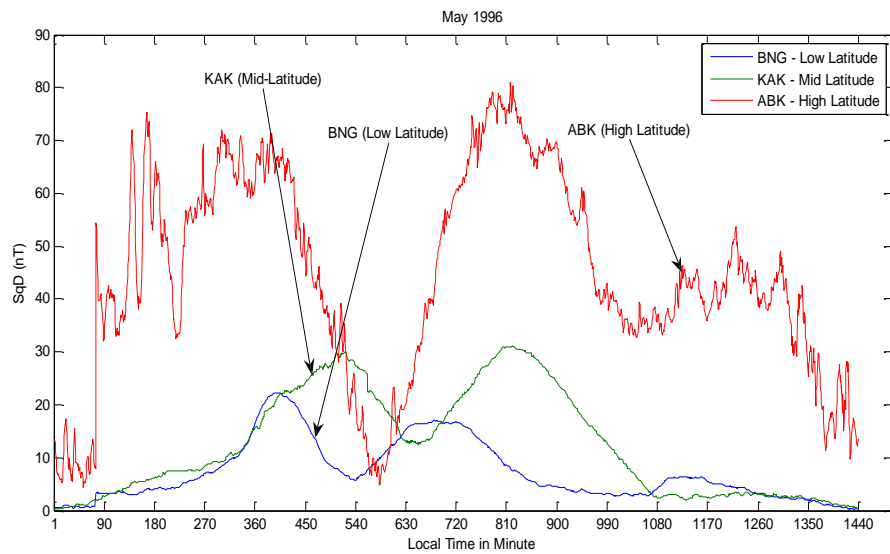


Fig. C5: High resolution of SqD across Low-, Mid- and High Latitudes – May

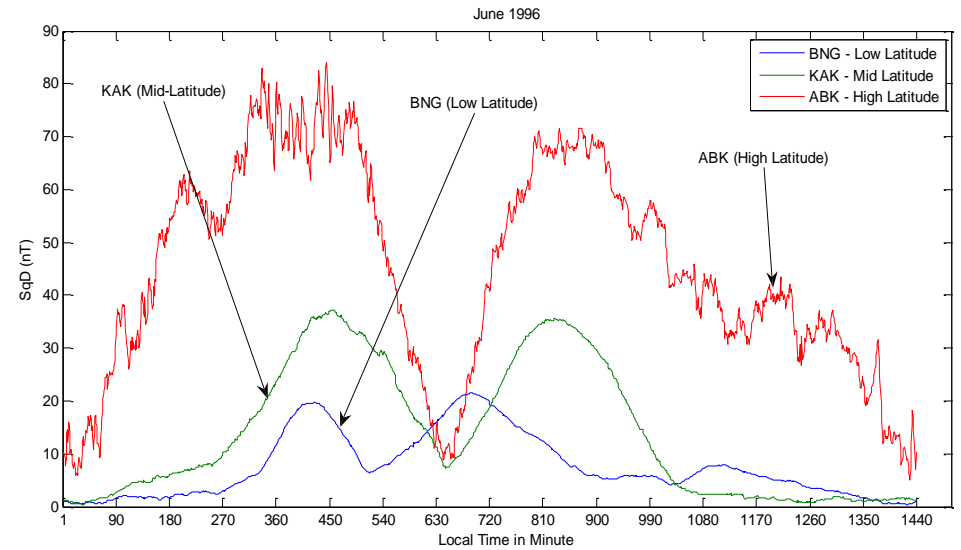


Fig. C6: High resolution of SqD across Low-, Mid- and High Latitudes –June

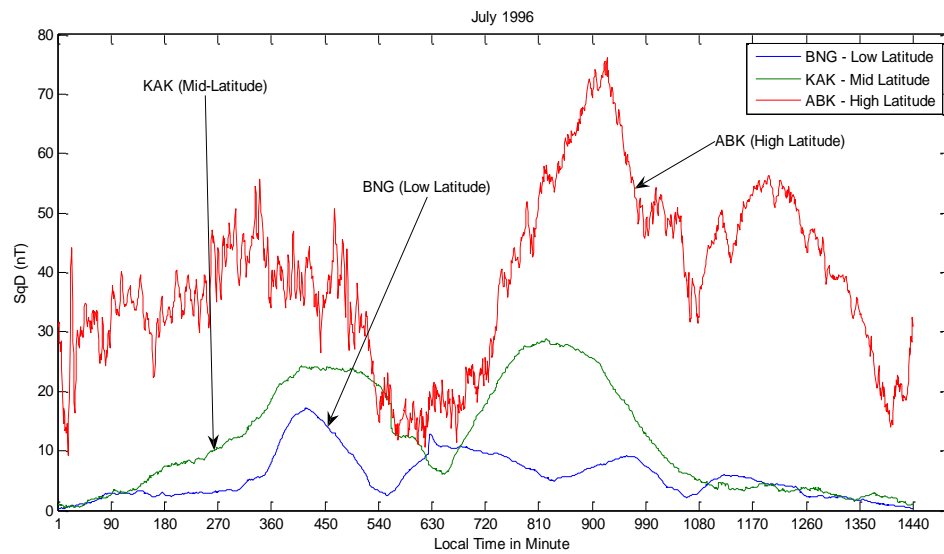


Fig. C7: High resolution of SqD across Low-, Mid- and High Latitudes – July

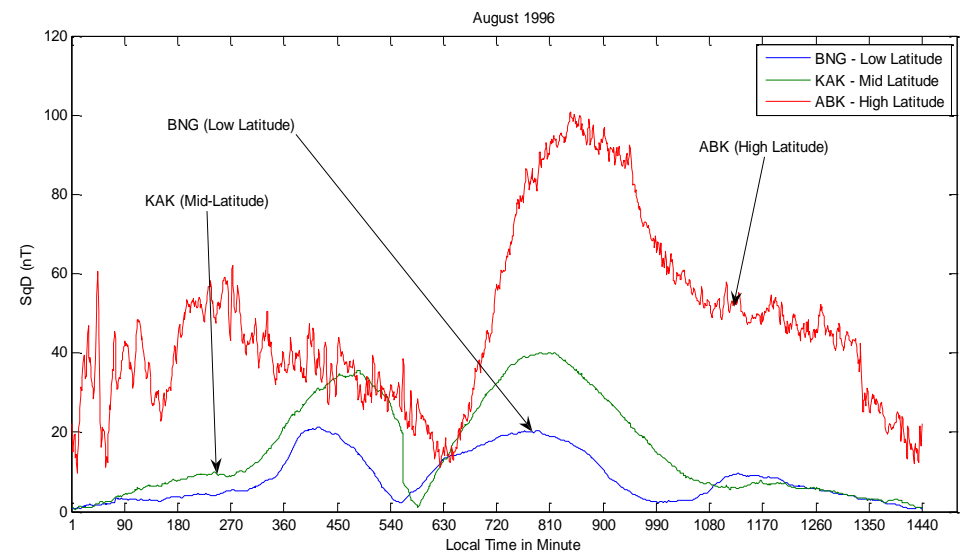


Fig. C8: High resolution of SqD across Low-, Mid- and High Latitudes –August

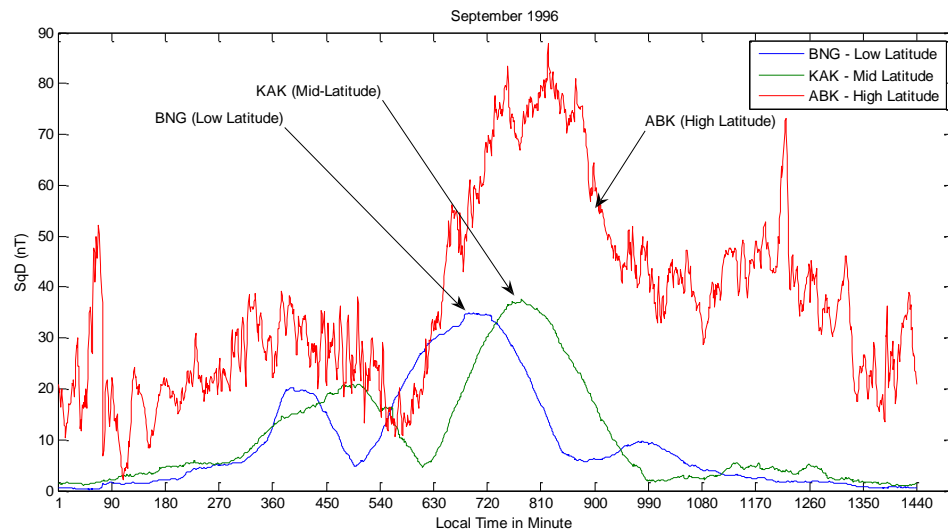


Fig. C9: High resolution of SqD across Low-, Mid- and High Latitudes –September

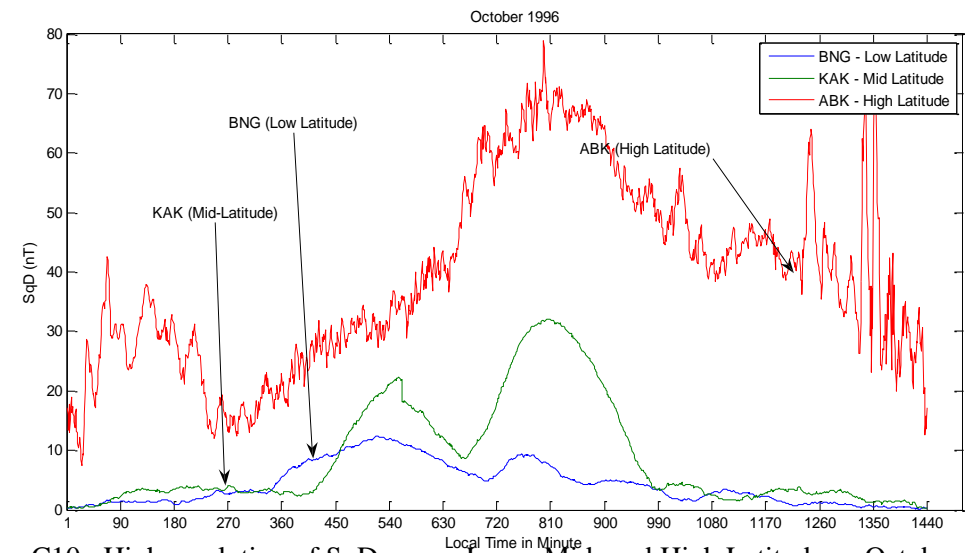


Fig. C10: High resolution of SqD across Low-, Mid- and High Latitudes – October

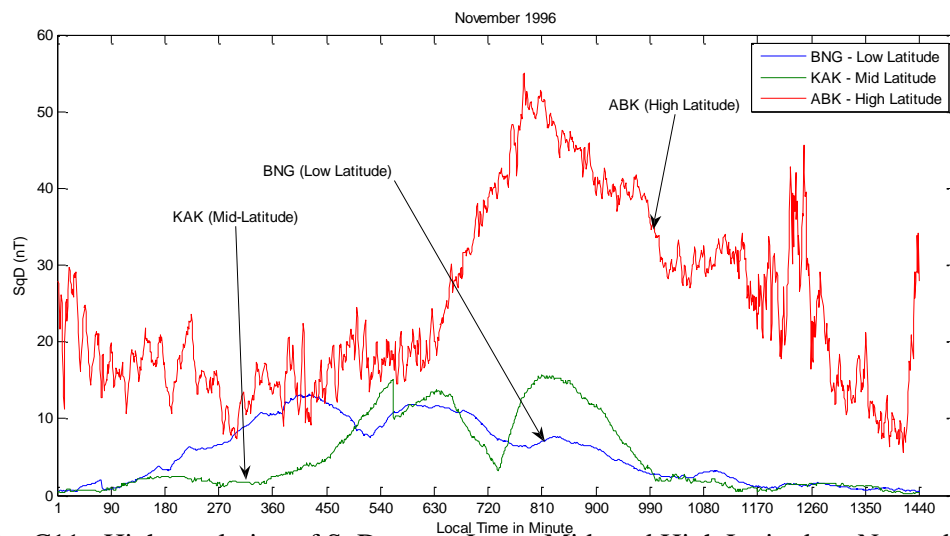


Fig. C11: High resolution of SqD across Low-, Mid- and High Latitudes –November

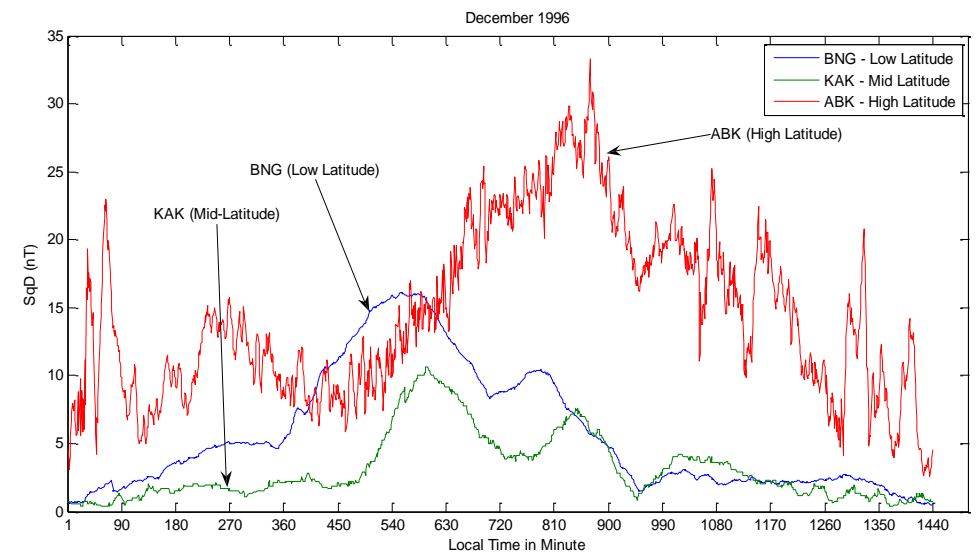


Fig. C12: High resolution of SqD across Low-, Mid- and High Latitudes –December

APPENDIX D: High Resolution Sq(Z) Variations in 2-D

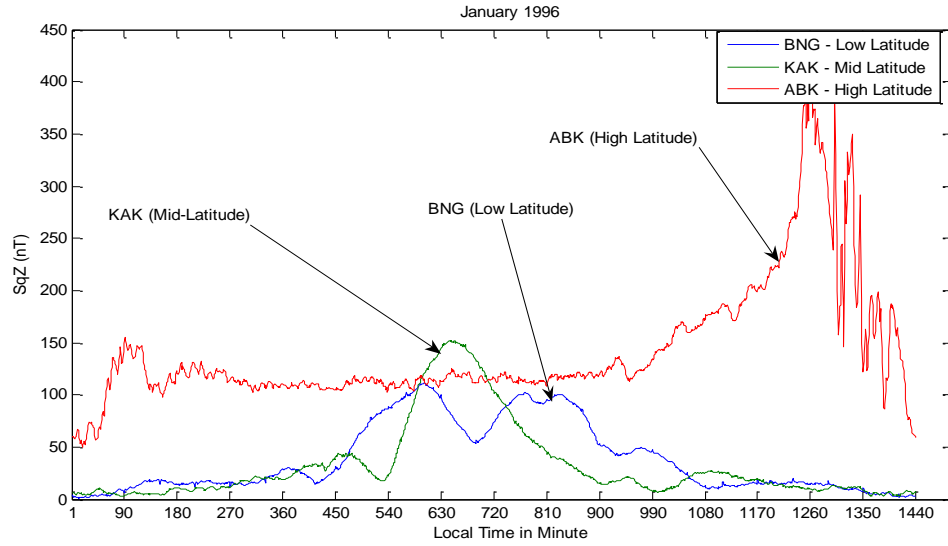


Fig. D1: High resolution of SqZ across Low-, Mid- and High Latitudes – January

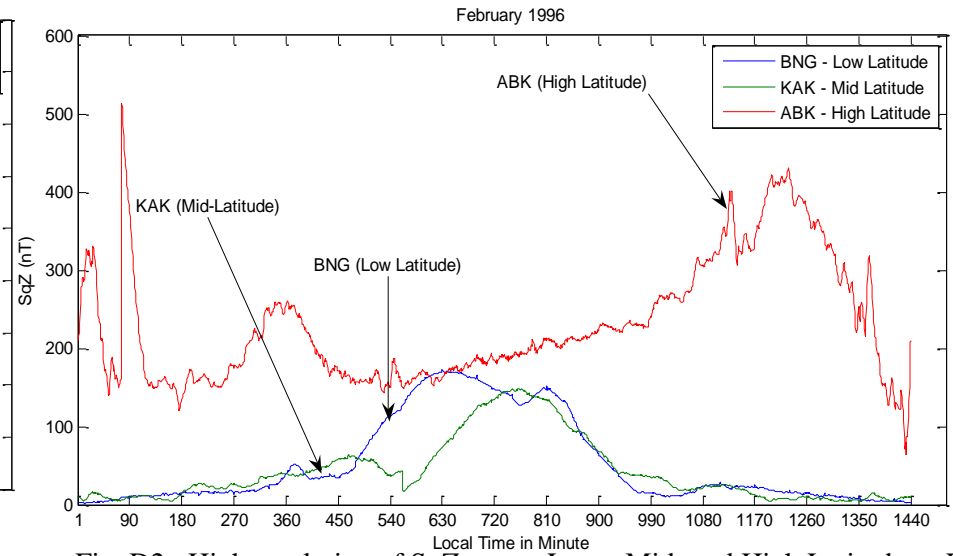


Fig. D2: High resolution of SqZ across Low-, Mid- and High Latitudes – February

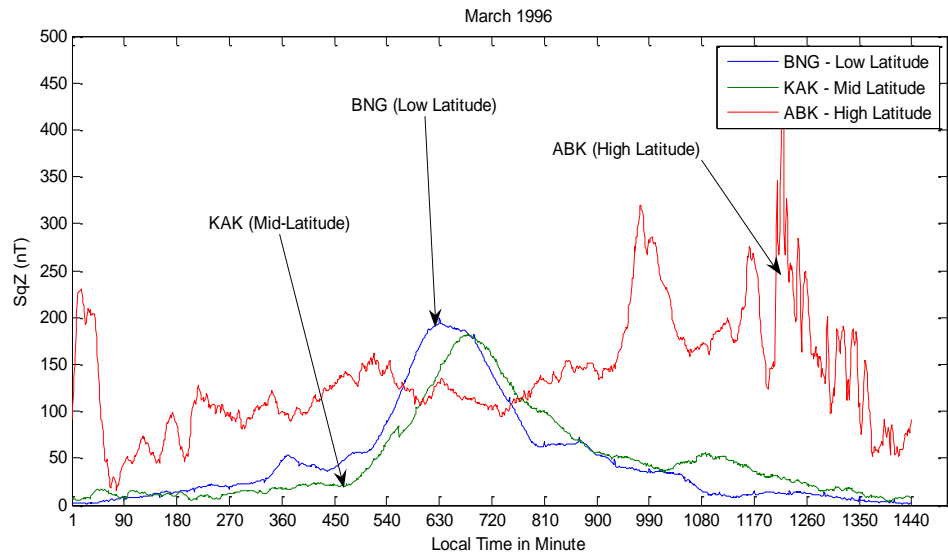


Fig. D3: High resolution of SqZ across Low-, Mid- and High Latitudes – March

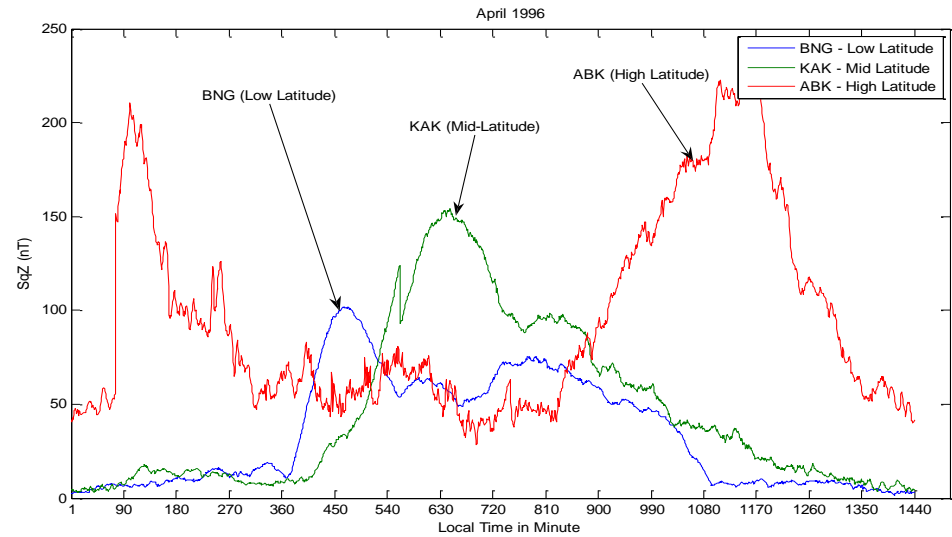


Fig. D4: High resolution of SqZ across Low-, Mid- and High Latitudes – April

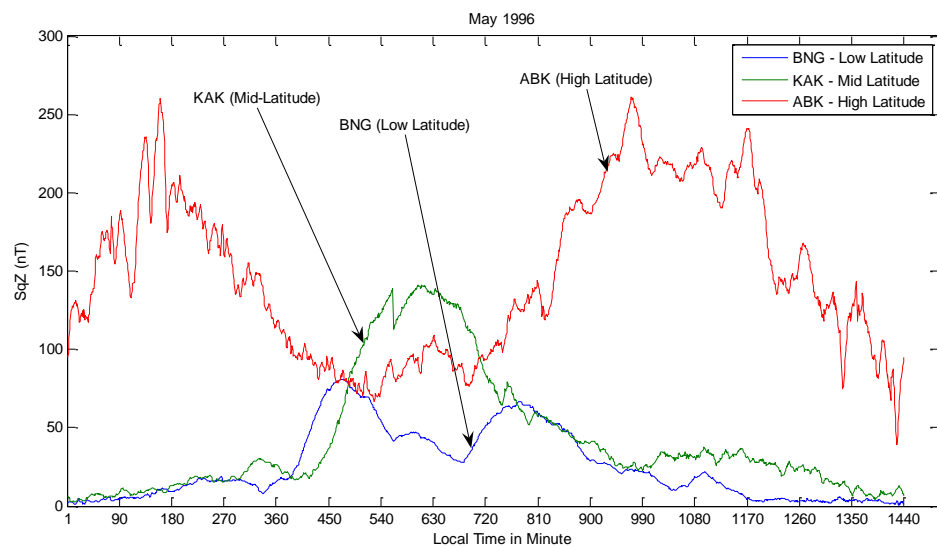


Fig. D5: High resolution of SqZ across Low-, Mid- and High Latitudes – May

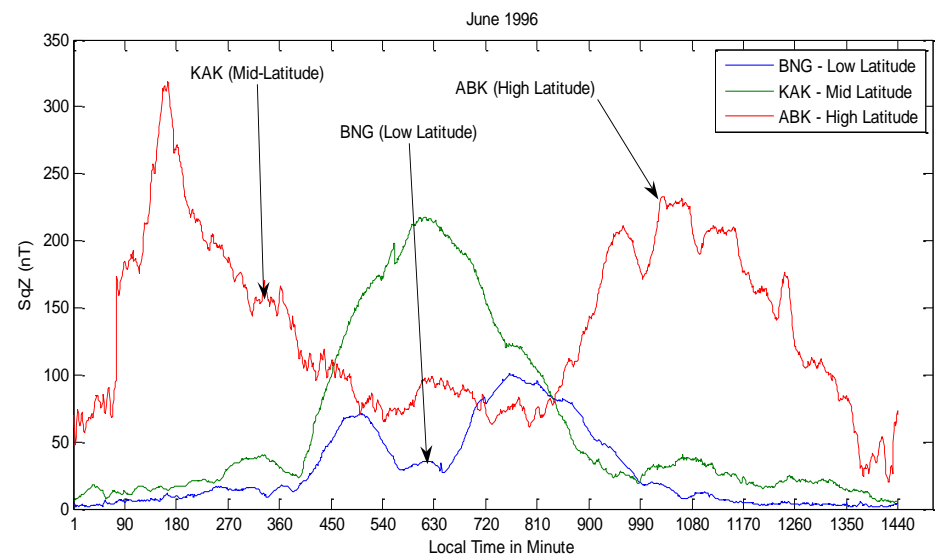


Fig. D6: High resolution of SqZ across Low-, Mid- and High Latitudes – June

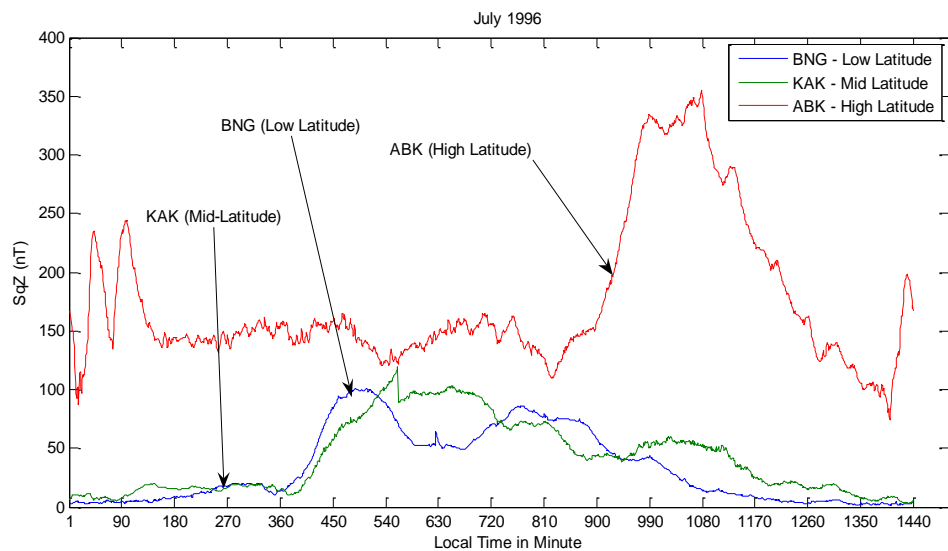


Fig. D7: High resolution of SqZ across Low-, Mid- and High Latitudes – July

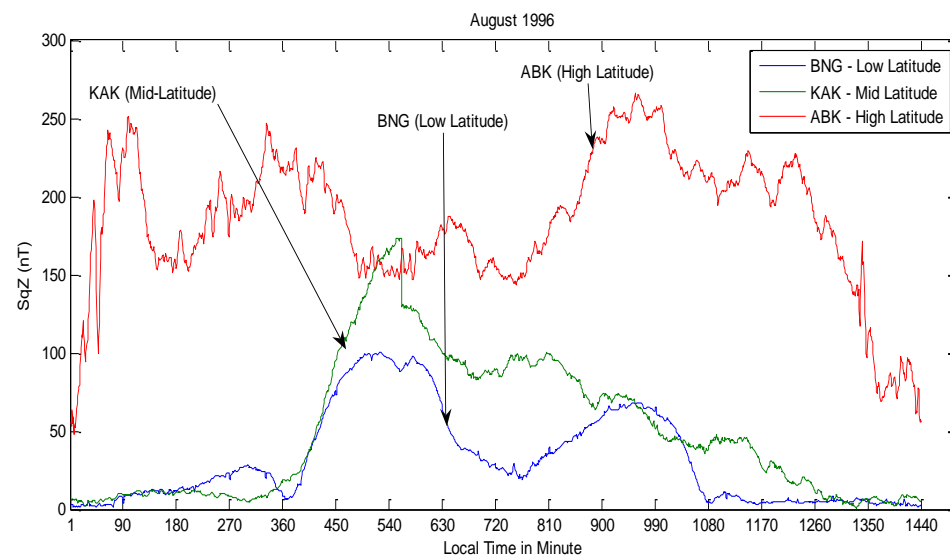


Fig. D8: High resolution of SqZ across Low-, Mid- and High Latitudes – August

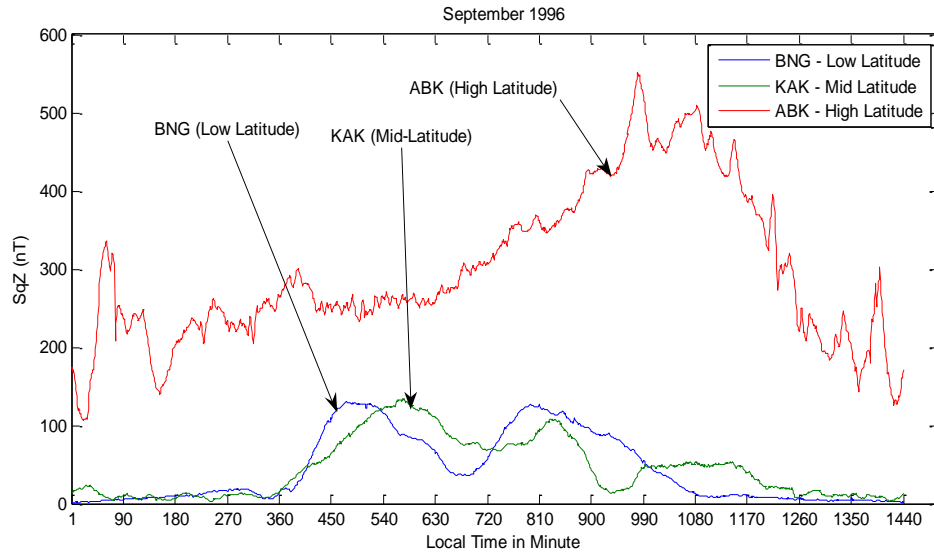


Fig. D9: High resolution of SqZ across Low-, Mid- and High Latitudes – September

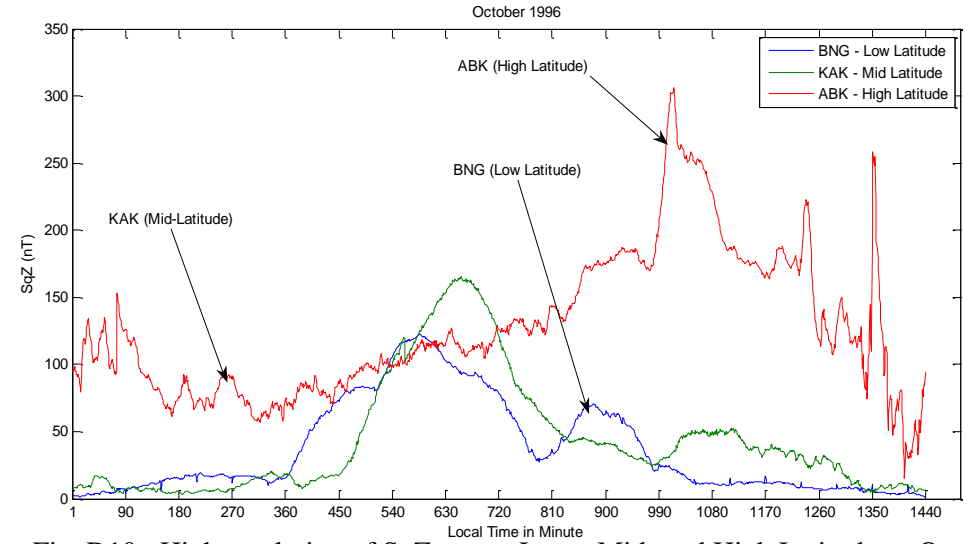


Fig. D10: High resolution of SqZ across Low-, Mid- and High Latitudes – October

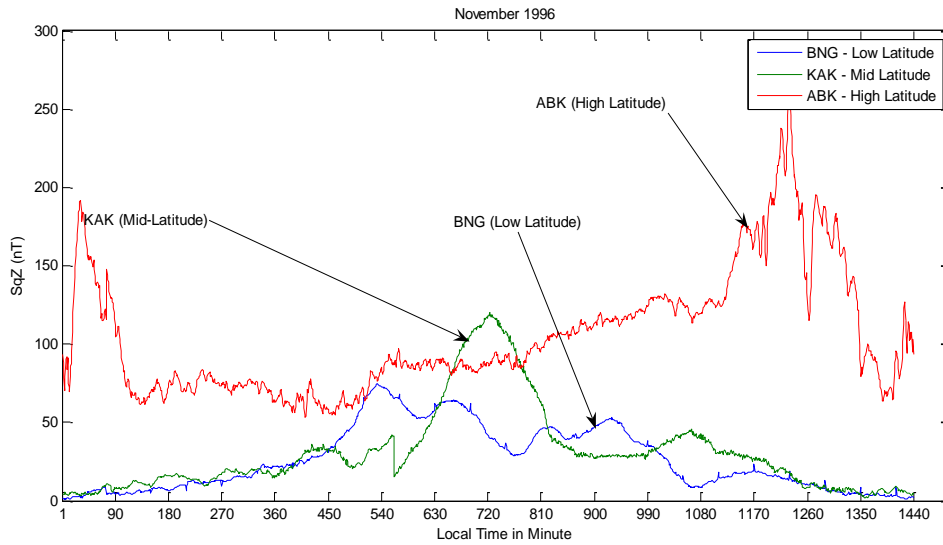


Fig. D11: High resolution of SqZ across Low-, Mid- and High Latitudes – November

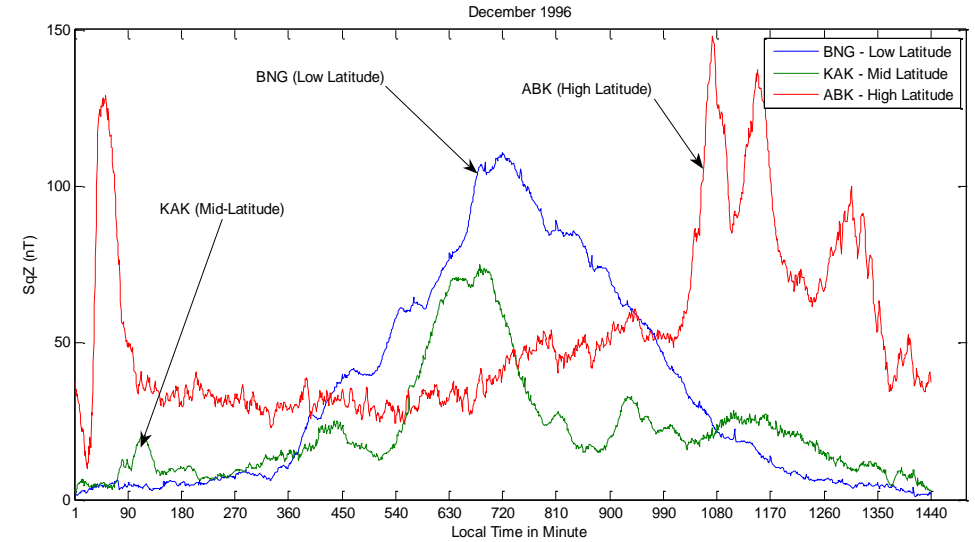


Fig. D12: High resolution of SqZ across Low-, Mid- and High Latitudes – December