8-2012

Identifying Signatures of Plasma Waves and Reconnection Associated with Kelvin-Helmholtz Activity

Thomas Wesley Moore
Emory-Riddle Aeronautical University - Daytona Beach

Follow this and additional works at: https://commons.erau.edu/edt
Part of the Plasma and Beam Physics Commons

Scholarly Commons Citation
https://commons.erau.edu/edt/107

This Thesis - Open Access is brought to you for free and open access by Scholarly Commons. It has been accepted for inclusion in Dissertations and Theses by an authorized administrator of Scholarly Commons. For more information, please contact commons@erau.edu.
IDENTIFYING SIGNATURES OF PLASMA WAVES AND RECONNECTION ASSOCIATED WITH KELVIN-HELMHOLTZ ACTIVITY

By
Thomas Wesley Moore

A Thesis Submitted to the
Physical Sciences Department
In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Engineering Physics

Embry-Riddle Aeronautical University
Daytona Beach, Florida
August 2012
IDENTIFYING SIGNATURES OF PLASMA WAVES AND RECONNECTION ASSOCIATED WITH KELVIN-HELMHOLTZ ACTIVITY

by

Thomas Wesley Moore

This thesis was prepared under the direction of the candidate’s thesis committee chair, Dr. Katarina Nykyri, Department of Physical Sciences, and has been approved by the members of the thesis committee. It was submitted to the Department of Physical Sciences and was accepted in partial fulfillment of the requirements of the Degree of Master of Science in Engineering Physics

THESIS COMMITTEE:

Dr. Katarina Nykyri, Chair

Dr. Anatoly Streltsov, Member

Dr. Anthony Reynolds, Member

Dr. Peter Erdman, MSEP Graduate Program Coordinator

Dr. Robert Fleck, Department Chair, Physical Sciences

Dr. Bob Oxley, Associate V.P. for Academics

Date

9-27-2012
ABSTRACT

Author: Thomas Wesley Moore
Title: Identifying Signatures of Plasma Waves and Reconnection Associated with Kelvin-Helmholtz Activity
Institution: Embry-Riddle Aeronautical University
Degree: Master of Science in Engineering Physics
Year: 2012

The magnetopause marks the boundary between the shocked solar wind and magnetospheric plasma. Understanding the dynamics of the plasma processes at the magnetopause boundary is crucial to the study of plasma transport into the magnetosphere. Previous studies have shown that there exists a temperature asymmetry in the plasma sheet. During northward IMF, the cold component ions are 30-40% hotter in the dawn flank plasma sheet compared to the dusk flank. However, the mechanisms responsible are still not entirely clear. Recent work has shown that reconnection in Kelvin-Helmholtz vortices can transport plasma into the magnetosphere. Previous studies have also shown that mode conversion at the magnetopause can generate kinetic Alfvén wave (KAW) activity. Both magnetic reconnection and plasma wave activity can heat plasma. In this thesis we look for new cases of Kelvin-Helmholtz Instability (KHI) from Cluster spacecraft data and search for signatures of associated magnetic reconnection and plasma wave activity.
ACKNOWLEDGEMENT

This research was funded by the National Science Foundation Career Grant #0847120.

Thanks to the Cluster Instrument teams for providing the data, the Community Coordinated Modeling Center for providing us with global 3D models of our events and Dr. Benoit Lavraud for providing us with ion distribution functions.

I would like to thank my committee members Dr. Katarina Nykyri, Dr. Anthony Reynolds and Dr. Anatoly Streltsov.

Thomas Wesley Moore
August 2012
## Nomenclature

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B_x$</td>
<td>x-component of the magnetic field</td>
</tr>
<tr>
<td>$B_y$</td>
<td>y-component of the magnetic field</td>
</tr>
<tr>
<td>$B_z$</td>
<td>z-component of the magnetic field</td>
</tr>
<tr>
<td>$B_{tot}$</td>
<td>magnitude of the magnetic field</td>
</tr>
<tr>
<td>SW</td>
<td>solar wind</td>
</tr>
<tr>
<td>PS</td>
<td>Parker-Spiral</td>
</tr>
<tr>
<td>OPS</td>
<td>Ortho-Parker-Spiral</td>
</tr>
<tr>
<td>MSH</td>
<td>magnetosheath</td>
</tr>
<tr>
<td>MSP</td>
<td>magnetosphere</td>
</tr>
<tr>
<td>KHI</td>
<td>Kelvin-Helmholtz Instability</td>
</tr>
<tr>
<td>MHD</td>
<td>Magnetohydrodynamic</td>
</tr>
<tr>
<td>KAW</td>
<td>Kinetic Alfvén Wave</td>
</tr>
<tr>
<td>GSM</td>
<td>Geocentric Solar Magnetic</td>
</tr>
<tr>
<td>GSE</td>
<td>Geocentric Solar Ecliptic</td>
</tr>
<tr>
<td>UT</td>
<td>Universal Time</td>
</tr>
</tbody>
</table>
Contents

Abstract iv
Acknowledgement v
List of Figures x

1 Motivation and Goals 1

2 Introduction 5
  2.1 Solar wind interaction with Earth’s magnetosphere 5
  2.1.1 Sun and Solar Wind 5
  2.1.2 Magnetosheath 6
  2.1.3 Magnetosphere 7
  2.1.4 Low Latitude Boundary Layer 9
  2.2 Main Physical Processes at the Magnetopause 9
    2.2.1 Magnetic Reconnection and Reconnection in KH Vortices 9
    2.2.2 Kelvin-Helmholtz Instability 12
    2.2.3 Plasma Wave Modes, e.g Kinetic Alfvén Waves & Compressional Alfvén Waves 13
  2.3 Cluster Spacecraft 15

3 Methodology 17
  3.1 Data-analysis 17
    3.1.1 Wavelet Analysis 17
3.1.2 Boundary Normal Analysis (Minimum and Maximum Variance Analysis and Minimum Faraday Residue) ........................................ 21
3.1.3 Walén and de Hoffman Teller Frame Analysis .......................... 22
3.2 Numerical Simulation Codes ..................................................... 24
  3.2.1 Local MHD Simulations .................................................. 24
  3.2.2 Global MHD Simulations, CCMC Codes ......................... 25

4 Cluster Spacecraft Observations of Periodic Boundary Crossings and Associated Wave Activity During PS IMF Orientation .......... 28
  4.1 Search of Low Latitude Boundary Layer Crossings with Quasi-Periodic Signatures ................................................................. 28
  4.2 Review of Events .................................................................. 29
  4.3 June 6th, 2002 During Parker Spiral Orientation .................... 29
    4.3.1 Solar Wind Conditions and Overview of Event ................. 29
    4.3.2 Single Spacecraft Analysis ....................................... 35
    4.3.3 Large Scale Structure of Magnetosheath and LLBL Using Global MHD Models from the Community Coordinated Modeling Center (CCMC) ................................................... 41
    4.3.4 Local MHD Simulations of KHI for Initial Conditions Picked from Cluster Data ....................................................... 41
    4.3.5 Analysis of Plasma Wave Activity in Association with Boundary Crossings ................................................................. 46
    4.3.6 Search for Reconnection Intervals .................................. 52
    4.3.7 Examination of Ion Spectrograms and Distribution Functions During Reconnection Intervals, Wave Intervals and Reference Intervals .................................................. 53

5 Discussion .............................................................................. 64

6 Conclusions and future work .................................................... 67

A IDL Code .................................................................................. 69
## CONTENTS

A.1 Main Routine ................................................. 69  
A.2 Read Routines ............................................... 72  
A.3 Wavelet Routines ........................................... 74  
A.4 Supporting Routines ....................................... 85  

Bibliography ....................................................... 92
# List of Figures

1.1 Dawn-dusk cold-component temperature profile of the plasma sheet ions [Wing et al. 2005] .......................................................... 3

1.2 Wave power spectral density of the parallel magnetic field (top panel), perpendicular magnetic field (middle panel) and ratio of the perpendicular to total wave power[Johnson et al. 2001]. .......................... 4

2.1 The magnetic field lines bend away from the shock normal on the downstream of the shock (left side) ......................................................... 7

2.2 Cartoon showing the diffusion regions, inflow and outflow regions, and the X-line for magnetic reconnection [Liu and Fujimoto 2011] (top panel). Cartoon showing reconnection inside of a KH vortex [Nykyri and Otto 2001] (bottom panel). ..................... 11

3.1 The left panel depicts the raw magnetic field data from Cluster and the right panel depicts the linearly interpolated data in nanno-Teslas (nT). ................................................................. 19

3.2 Magnetic field parameters in boundary normal coordinates during an interval of KHI at the dawn-flank magnetopause. ................................. 23

3.3 Schematic of simulation plane depicting $\psi$ and $\phi$. ................................. 26

3.4 OpenGGCM simulation grid. ................................................................. 27

4.1 Plasma and magnetic field parameters from Cluster in GSM coordinates - June 6th, 2002. ................................................................. 31
LIST OF FIGURES

4.2 Magnetic field parameters in boundary normal coordinates from Cluster - June 6th, 2002. .................................. 33
4.3 The boundary normals along spacecraft trajectory using MFR method. The black vectors represent the boundary normals, the green vector represents the average boundary that is compiled using data from the entire interval 13:20 - 13:50 UT and the red vectors represent HT frame velocity. .......................................................... 34
4.4 Solar Wind conditions from ACE - June 6th, 2002. .................. 36
4.5 Solar Wind IMF - June 6th, 2002. Image is not to scale. ............ 37
4.6 Tailward plasma flow plotted against plasma density from a 3D simulation of KHI with 0° propagation angle [Hasegawa et al. 2006]. .. 38
4.7 Plasma parameters depicting low density faster than sheath flow during intervals with density gradients (left panel). The red points represent points at which the tailward speed, \( V_x \) is < -290 km/s (where the MSH plasma has a tailward speed of \( \approx 250 \) km/s). Tailward flow plotted against plasma density from November 20th, 2001 (right panel) [Hasegawa et al. 2006]. .......................................................... 39
4.8 Number density (top panel), \( V_x \) (middle panel), and plasma flow vs density using the single spacecraft technique (bottom panel) for June 6th, 2002 between 13:30:00 and 13:50:00 UT. The black, green and blue curves correspond to spacecraft 1, 3 and 4 respectively. .................. 40
4.9 Draping effect due to fast shock at the bow shock depicted by the the OpenGGCM global model of the June 6th, 2002 event. ............ 42
4.10 Local MHD simulation after onset of KHI. The left panel plots the magnetic field vectors with the current density (background color). The right plots the velocity vectors and the plasma density (background color). The sunward direction is located towards the bottom of the Figure - the MSH is on the left side and the MSP is on the right side of each panel. .......................................................... 44
4.11 Plasma and magnetic field parameters produced by a virtual satellite in the simulation plane. .......................................................... 45
4.12 (a) number density. (b) temperature. (c) Red line corresponds to the velocity component parallel to the background magnetic field, the blue curve corresponds to the perpendicular velocity and the black line shows the ratio between the two. (d) $\delta E_\perp$. (e) $\delta B_\perp$. (f) $\delta E_\perp/\delta B_\perp$. (g) Total integrated power, $\delta B_\perp/\delta B_{tot}$. .................................................. 49

4.13 Filtered high resolution magnetic field for all spacecrafts taken from the wave activity interval. The black, red, green and blue curves represent data from satellites 1, 2, 3 and 4 respectively. ................................. 50

4.14 Hodograms from wave activity interval. Eigenvectors $\mathbf{i}$, $\mathbf{k}$ and $\mathbf{j}$ represent the minimum, intermediate and maximum variance directions of $\mathbf{B}$ respectively, where $\mathbf{i}$ corresponds to the direction of propagation of the wave. The angles $\mathbf{iB}$, $\mathbf{kB}$ and $\mathbf{jB}$ represent the angle between the minimum, intermediate and maximum eigenvectors and $\mathbf{B}$. .............................. 51

4.15 Walén relation and deHoffman-Teller frame for spacecraft 1 (top left panel) and 3 (top right panel) and the plasma and magnetic field properties for all spacecraft (bottom panel). ................................. 54

4.16 Ion spectrograms for spacecraft 3 from Cluster Active Archive. ....... 56

4.17 Reference ion distribution functions for intervals when satellite 3 is in MSH-like plasma. ................................................................. 58

4.18 Reference ion distribution functions for intervals when satellite 3 is in MSP-like plasma. ................................................................. 59

4.19 Ion distribution functions around the reconnection interval for spacecraft 1. ................................................................. 61

4.20 Ion distribution functions for intervals during KAW activity. ............ 63
Chapter 1

Motivation and Goals

This research is motivated by observations made by Hasegawa et al. [2003] and Wing et al. [2005] of a Dawn-Dusk Asymmetry in the flanks of the plasma sheet - the cold component ions are hotter by 30-40% at the dawnside plasma sheet compared to the duskside plasma sheet. In Figure 1.1 temperatures of the dawn (top) and dusk (bottom) cold component ions are plotted. There is a region just inside the plasma sheet (with respect to the magnetopause) where the dawn-flank ions are hotter than at the duskside; in this region the dawnside ions are slightly above $5 \times 10^6$ K and the duskside ions are slightly below $4.5 \times 10^6$ K. These temperatures were provided by mapping measurements made by the SSJ4 instrument on-board Defense Meteorological Satellite Program (DMSP). The origin of this asymmetry is not currently understood, however Johnson and Cheng [2001] have shown stochastic ion heating (perpendicular to the magnetic field) via kinetic Alfvén wave (KAW) turbulence. Furthermore Johnson et al. [2001] show that an amplification of perpendicular wave power can be explained by mode conversion of compressional magnetohydrodynamic (MHD) waves into KAWs at the magnetopause. The wave power spectral density (of the magnetic field) from a slow magnetopause crossing is plotted in Figure 1.2. When the spacecraft (WIND) crosses the magnetopause around 19:30 UT the parallel wave power varies very little (top panel), whereas there is a spike in the perpendicular wave power (middle panel); the ratio of the perpendicular wave power to the total wave power (bottom panel) increases to nearly 1 inside of the magnetosphere. It has been
suggested that ultra-low frequency waves (below 0.5 Hz) are associated with mode conversion [Lee et al. 1994; Belmont et al. 1995; De Keyser et al. 1999]. KHI is an ultra-low frequency wave that operates at the magnetopause. Modeling of the magnetosheath properties as a function of upstream solar wind conditions has shown that the dawnside magnetospheric flank is statistically more unstable to Kelvin-Helmholtz Instability (KHI) (Nykyri et al. 2012 (GEM 2012 Presentation)). The fact that the dawnside magnetospheric flank is more KH unstable may lead to more plasma heating at the dawnside flank associated with reconnection in KH vortices, heating via plasma waves associated with KHI or heating via shocks associated with KHI. The motivation of this research is to address this by (1) identifying new unpublished events of KHI at the flanks of the magnetosphere, (2) identifying plasma wave activity and magnetic reconnection associated with KHI at the magnetopause and (3) studying ion distribution functions during intervals of reconnection and wave activity.

**Significance of Work**

This thesis presents for the first time an event of Kelvin-Helmholtz Instability and Mode Conversion at the magnetopause.

Space is a dangerous environment for both lifeforms and instrumentation; understanding our local environment (e.g. the magnetosphere) will help ensure that we are better equipped for space exploration. Furthermore, understanding the physical processes by which our magnetosphere on earth interacts with the solar wind can shed insight to the dynamics and evolution of other planetary systems, which could in turn lead to a higher success rate for interplanetary missions.
Figure 1.1: Dawn-dusk cold-component temperature profile of the plasma sheet ions [Wing et al. 2005]
Figure 1.2: Wave power spectral density of the parallel magnetic field (top panel), perpendicular magnetic field (middle panel) and ratio of the perpendicular to total wave power [Johnson et al. 2001].
Chapter 2

Introduction

2.1 Solar wind interaction with Earth’s magnetosphere

2.1.1 Sun and Solar Wind

Rather than being comprised of separate entities, the solar system exists as a single organism woven together by magnetic and electric fields. At the center of it all lies the Sun, an enormous ball of gas fueled by fusion reactions at its core. The energy generated at the core is carried to the surface through radiative diffusion and convection. Energy leaves the Sun via radiation and solar wind, where the former is emitted from the visible photosphere and the latter from the corona. The Corona is the outermost part of the Sun which is around 1 million K. Since the corona is not in hydrostatic equilibrium with the local interstellar medium (LISM), it is in a state of steady expansion [Parker 1958].

The solar wind is comprised of “hot tenuous plasma” that is ejected from the Sun via “coronal expansion” [Bittencourt 2004] and coronal mass ejections (CMEs), where the former is a continuous process and the latter is intermittent. This plasma moves radially outward from the corona in Archimedean spirals, towards the surrounding solar system. The interplanetary magnetic field (IMF) is comprised of (dominantly) open field lines carried by the solar wind into the solar system. Although the IMF
is stretched out from the Sun in Archimedean spirals, the perception from Earth does not always appear so due to small (relative to the Sun’s scale) perturbations. The differential rotation at the Sun (where the poles rotate faster than the equator) causes the magnetic field lines to become twisted. CMEs also influence the IMF orientation, as they are comprised of plasma “clouds” carrying reconnected field lines. IMF orientation can be divided into five categories: Parker-Spiral (PS), Ortho-Parker-Spiral (OPS) (which is perpendicular to PS at Earth - because of the scale size of the IMF is very large compared to Earth, small variations in the IMF are dramatically perceived at Earth), purely northward and purely southward or radial; the IMF can also be a mixture of the above conditions. This nomenclature is dependent upon Earth’s bow shock in the Geocentric Solar Magnetic (GSM) coordinate system. The bow shock is created by solar wind approaching the magnetopause (Earth’s farthest reaching closed magnetic field line) faster than the local fast magnetosonic speed; because the solar wind is super “fast”, “information” about the solar wind cannot reach the magnetopause fast enough and a standing shock wave is formed. The magnetic field vector in a PS IMF is predominantly parallel to the (bow) shock normal in the x-y plane in Geocentric Solar Magnetic (GSM) coordinate system on the dawn flank and perpendicular to the shock normal on the dusk flank. For a purely northward or southward IMF, the magnetic field lies solely in the z direction. The orientation of IMF is pertinent to the onset conditions for several processes at the magnetopause which will be discussed in subsection 1.2.2.

2.1.2 Magnetosheath

The magnetosheath (MSH) lies between the magnetopause and the bow shock. Plasma in the magnetosheath is cold and dense, with typical ion densities in of about $10^3 \text{ cm}^{-3}$, and temperatures are a few million Kelvin (an order of magnitude colder than plasma in magnetosphere just inside the magnetopause in the equatorial plane).

The magnetic field in this region is supplied by the IMF, but undergoes transformation due to draping at the bow shock. The draping of the field lines is caused by the fast-shock plasma wave mode at the bow shock. Upstream plasma undergoes
CHAPTER 2: INTRODUCTION

7

Figure 2.1: The magnetic field lines bend away from the shock normal on the down-
stream of the shock (left side).

compression and heating at the shock; as a result the downstream plasma has in-
creased density and temperature and continuity dictates that the flow velocity must
decrease. The kinetic energy of the plasma at the shock is converted into magnetic
and thermal energy therefore the total magnetic field strength increases downstream.
The magnetic field carried by the downstream plasma bends away from the shock
normal (see Figure 2.1) [Kivelson and Russell 1995].

2.1.3 Magnetosphere

The magnetosphere (MSP) lies inside the magnetosheath, surrounded by the magne-
topause. The magnetopause is defined as the boundary where Earth’s last magnetic
field lines lie. The Earth’s dipole-like magnetic field reach ends due to a pressure
balance at the magnetopause between the Earth’s field and solar wind. The stand-off
distance is defined by the radial distance from Earth at which the total pressure from
Earth’s magnetosphere (magnetic pressure and plasma pressure) equals the solar wind total pressure, where the dynamic pressure is dominant. This stand-off distance is dynamic and changes under the varying solar wind conditions. The magnetosphere is composed of both open and closed magnetic field lines; the open field lines begin at the magnetic poles where the closed lines meet (the magnetic cusps) and drape across the tail. Plasma inside the MSP is not uniformly distributed; there exist regions of different densities and temperatures (e.g., plasma sheet, ionosphere, and plasma sphere). The scope of this research will cover the plasma sheet, the tailward region of the magnetosphere that lies in the equatorial plane and Low Latitude Boundary Layer (see next section). The plasma in the plasma sheet is hot and tenuous with typical ion densities of about 1 cm$^{-3}$ just inside the magnetopause. Plasma velocities just inside the magnetopause at the flanks are stagnant compared to those in the magnetosheath. The magnetopause is a tangential discontinuity. For a tangential discontinuity, the flow speed remains constant and the magnetic field switches direction, as a result there is a change in magnetic and thermal pressures in order (the total pressure remains constant) for the pressure balance to be maintained \cite{Kivelson1995}.

There are two distinct ion populations in the plasma sheet - hot and cold component ions of magnetospheric origin and magnetosheath origin respectively. It has been shown that during periods of northward IMF, the plasma sheet ions become more cold and dense than when the IMF is southward \cite{Terasawa1997, Fujimoto1998, Stenuit2002, Øieroset2005}. Furthermore, there exists a temperature asymmetry in the magnetosphere - the cold component ions at the dawn flank are hotter than the cold component ions at the dusk flank plasma sheet \cite{Wing2005}. Data from the Defense Meteorological Satellite Program (DMSP) are plotted in Figure 1.1 to show the cold-component temperature profile of the two-component Maxwellian distribution function of the plasma sheet ions \cite{Wing2005}. From Figure 1.1, the cold-component ions on the dawn-flank are hotter tailward of -20 R$_E$ compared to at the dusk-flank.
2.1.4 Low Latitude Boundary Layer

The Low Latitude Boundary Layer (hereby referred to as the LLBL) is a region located at the magnetospheric flanks and dayside magnetosphere where densities and temperatures are mixed between typical magnetosheath and magnetosphere values. The LLBL starts very thin, expands tailward and is most pronounced in the equatorial plane. Formation of LLBL is suggested to be a result of double high-altitude reconnection (e.g., Lavraud et al. [2005] and Li et al. [2005]), reconnection in Kelvin-Helmholtz vortices [Nykyri and Otto 2001], ion diffusion in KH vortices [Fujimoto and Terasawa 1994, 1995] or Kinetic Alfvén Waves [Johnson and Cheng 1997, 2001]. The LLBL acts as a buffer between the magnetopause and the plasma sheet such that the processes that occur at the magnetopause are coupled to the plasma sheet.

*Eastman and Hones* [1979] confirmed the existence of a boundary layer at the dayside of the magnetosphere with observations from the IMP 6 spacecraft. These observations took place at the dawn and dusk flanks of the dayside magnetopause, and show a region where the local plasma parameters (e.g. ion densities, bulk flow and energy) and magnetic field orientation are mixed - between those typical for the magnetosheath and the magnetosphere. Furthermore, this boundary layer was “nearly always present at all latitudes and longitudes” for all IMP 6 crossings of the magnetopause. The boundary layer thickness ranges from about 100 km to greater than 1 Earth radius increasing with tailward distance [*Eastman and Hones* 1979]. This region is relevant for this thesis because observations of KHI and mode conversion occur at the LLBL.

2.2 Main Physical Processes at the Magnetopause

2.2.1 Magnetic Reconnection and Reconnection in KH Vortices

Magnetic reconnection occurs when plasma with anti-parallel field lines converge to electron inertial scales, creating a curl in the magnetic field. This inflow (plasma
CHAPTER 2: INTRODUCTION

carried by convected field lines) creates a diffusion region, where the plasma is de-
coupled from the magnetic field lines. Magnetic field lines are “allowed to reconnect”
when the frozen-in approximation is violated - this process occurs at the X-line (see
Figure 2.2). The plasma is “frozen” back into the magnetic field lines in the outflow
region. The reconnection rate is defined by the amount of magnetic flux that is
reconnected at the X-line per unit time and unit length. Magnetic reconnection is a
rotational discontinuity, providing a means for plasma to pass a boundary (e.g. the
magnetopause) [Liu and Fujimoto 2011, and references therein]. The flow speed and
pressure across a rotational discontinuity are continuous and the plasma is connected
magnetically.

There are two main steady state reconnection models, Sweet-Parker and
Petscheck. The Sweet-Parker model requires that plasma is accelerated in the dif-
fusion region. The rate of reconnection in the Sweet-Parker model is inversely propor-
tional to the magnetic Reynolds number defined by the ratio of the magnitude of the
convective term \( (\nabla \times (v \times B)) \) to the magnitude of the diffusive term \( \frac{1}{\sigma \mu_0} \nabla^2 B \) from
the induction equation (derived from Maxwell’s equations).

\[
\frac{\partial B}{\partial t} = \nabla \times (v \times B) + \frac{1}{\sigma \mu_0} \nabla^2 B \tag{2.1}
\]

Because most space plasmas have a Reynolds number much larger than 1, the
inflow (plasma convected into the diffusion region) is very slow which corresponds to
slow reconnection rates. These slow reconnection rates do not agree with observed
reconnetion events. The Petscheck reconnection model addresses these slow reconnec-
tion rates. This model states that plasma flow into the diffusion region is not
necessary for it to be accelerated, rather standing shock waves (attached to the dif-
fusion region) cause plasma acceleration. The diffusion region in the Petscheck model
is a point centered at the X-line [Kivelson and Russell 1995].

It has been shown that reconnection occurs at the dayside magnetopause, in the
magneto-tail, at the cusps and in the LLBL. These anti-parallel fields can also be
generated via KHI from initially parallel configurations [Nykyri and Otto 2001].
Figure 2.2: Cartoon showing the diffusion regions, inflow and outflow regions, and the X-line for magnetic reconnection [Liu and Fujimoto 2011] (top panel). Cartoon showing reconnection inside of a KH vortex [Nykyri and Otto 2001] (bottom panel).
2.2.2 Kelvin-Helmholtz Instability

Kelvin-Helmholtz Instability occurs at the interface of two viscous fluids that have non-zero relative velocity. This phenomenon has been observed via in-situ measurements of magnetopause crossings [Fairfield et al. 2000; Hasegawa et al. 2004; Nykyri et al. 2006; Hwang et al. 2011]. Shear flow occurs between the solar wind plasma in the magnetosheath and magnetospheric plasma at the magnetopause boundary. The onset of the KHI is dependent upon the magnetic field orientation in relation to the shear flow plane \((\mathbf{k} \cdot (\mathbf{V}_1 - \mathbf{V}_2))\), where \(\mathbf{k}\) represents the direction of propagation of the instability, and \(\mathbf{V}_n\) is the shear flow velocity in the \(n^{th}\) region. The condition for the onset of KHI is given by the following relation [Treumann and Baumjohann 1997]:

\[
[k \cdot (V_1 - V_2)]^2 \geq \frac{n_1 + n_2}{4\pi m_0 n_1 n_2} [(k \cdot B_1)^2 + (k \cdot B_2)^2] \tag{2.2}
\]

The onset is maximized when the magnetic field, \(\mathbf{B}\) is perpendicular to \(\mathbf{k}\). In relation to KHI at the magnetopause, the magnetic field is mostly perpendicular to the magnetosheath flow at the “near-tail” LLBL when the IMF is either purely northward or southward. Under PS orientation (when the IMF is parallel to the shock normal at the dawn flank), the dawn flank is more KHI unstable because draping around the magnetosphere creates less tangential magnetic field with respect to the shear flow (related to the dusk flank), where the dusk flank is more stable. For OPS orientation, the IMF is parallel to the shock normal at the dusk flank, and draping around the magnetopause creates less tangential magnetic field with respect to shear flow than at the dawn flank, hence the dusk flank is more KHI unstable. Equation 2.2 can also be expanded in terms of the shear flow velocity and the Alfvén velocity along \(\mathbf{k}\):

\[
k \cdot V_{SF} > k \cdot V_A \tag{2.3}
\]

where the Alfvén wave propagates along the magnetic field line. Larger angles between the shear flow plane and the magnetic field lines can create a KH unstable boundary. It should also be stated that in order for reconnection to occur in the shear
flow plane, the magnetic field that lies in the shear flow plane must be greater than zero.

KHI has been observed under strongly northward IMF solar wind \cite{Fairfield2000, Otto2002, Hasegawa2004} and mass transport across the magnetopause associated with KHI has been quantified and shown to be efficient in generating a cold-dense plasma sheet in the time scale of about 2 hours \cite{Nykyri2001}. KHI has recently been observed for strongly southward IMF \cite{Hwang2011}. The foundation of each of these published events is built upon in-situ satellite observations. \cite{Otto2000} modeled KHI with MHD simulations and compared the results with Geotail satellite data - the simulations were able to reproduce the geotail observations very well. The KHI signatures usually consist of semi-periodic fluctuations of magnetosphere and magnetosheath-like plasma properties such as densities and temperatures, bipolar fluctuations in the magnetic field component normal to the magnetopause, and radical changes in the tailward velocity. These signatures serve as a template for finding new and unpublished events in the archive of Cluster satellite data. Determining whether or not the events in question are in fact signatures of a KH unstable magnetopause is a complex process involving many analytical techniques, including the use of simulations.

\subsection*{2.2.3 Plasma Wave Modes, e.g Kinetic Alfvén Waves \& Compressional Alfvén Waves}

When a plasma is perturbed, it reacts by generating waves. In plasma physics, there is a large spectrum of waves, however this research is primarily concerned with compressional Alfvén waves, Shear Alfvén waves and KAWs. The Compressional Alfvén Wave and Shear Alfvén wave are MHD waves, derived from the MHD equations. The KAW is derived from taking into account kinetic effects and is a dispersive wave - the propagation through a plasma changes with frequency.

A Compressional Alfvén Wave is a longitudinal plasma that perturbs the plasma along the direction of propagation. As they propagate through a magnetized plasma, they compress the magnetic field lines causing fluctuations in the plasma parameters
and the magnitude of the magnetic field. The fast mode magnetosonic wave can travel and transfer energy in any direction to the background magnetic field, whereas the slow mode travels perpendicular to the background magnetic field.

Shear Alfvén waves are low frequency plasma waves on the order of the ion cyclotron frequency. These electromagnetic waves propagate along the magnetic field line, however other wave modes exist that propagate oblique to the magnetic field; the phase velocity is given by $V_A \cos \theta$, where $\theta$ is the angle between the direction of propagation and the magnetic field. For a Shear Alfvén Wave, there is no wave mode that propagates perpendicular to the magnetic field. Shear Alfvén waves are transverse, which have a magnetic and electric field component that fluctuate perpendicular to the direction of propagation. As they travel through a plasma they do not cause oscillations in the plasma pressure, density or magnitude of the background magnetic field. Instead they bend field lines as they travel, creating magnetic tension which acts as a restoring force that accelerates particles.

KAWs are obtained when the MHD Alfvén Wave (Shear Alfvén Wave) develops a large wavenumber transverse to the background magnetic field; in other words, these waves have a dominant perpendicular component to their wave number - $k_\parallel << k_\perp$. This feature translates to a perpendicular wavelength that is on the order of the ion gyroradius (which is the same as the ion inertial scale when $\beta = 1$). The following equation shows the dispersion relation for a KAW derived for a plasma where $m_e/m_i << \beta < 1$.

$$\omega = k_\parallel^2 V_A^2 \left[ 1 + k_\perp^2 \rho_i^2 \left( \frac{3}{4} + \frac{T_e}{T_i} \right) \right]$$ (2.4)

On the right hand side of Equation 2.4, when the perpendicular wavenumber $k_\perp$ is zero, the dispersion relation is the same as for a Shear Alfvén Wave. KAWs are compressive waves, therefore cause perturbations in the plasma parameters and magnitude of the background magnetic field as they propagate through a plasma. The electric field associated with this wave mode can accelerate particles. This acceleration can be in the perpendicular direction (with respect to its propagation) [Hasegawa
1976]. *Johnson and Cheng* [2001] have shown stochastic ion heating (perpendicular to the magnetic field) via Kinetic Alfvén Wave (KAW) turbulence.

## 2.3 Cluster Spacecraft

All data used in this research was taken from the European Space Agency’s Cluster satellites. Cluster is a 12 year mission that is comprised of 4 satellites that orbit the Earth’s magnetosphere in a tetrahedral formation with varying separation. Each spacecraft houses 11 scientific instruments (see Table 2.1 and has a four second spin period (0.25 Hz). In this thesis we use the FGM, EFW and CIS instruments. The Fluxgate Magnetometer (FGM) consists of 2 tri-axial magnetometers at the end of a 5-m boom capable of measuring magnetic field vectors with a resolution of 22.5 Hz. Electric Field and Wave Experiment (EFW) instrument measures electric fields perpendicular to the spin axis. The electric fields are estimated by measuring the potential difference between two probes mounted on 44m wire booms. The third component is estimated based upon the assumption that the electric field along the magnetic field is zero \( \mathbf{E} \cdot \mathbf{B} = 0 \) which provides the full 3D electric field vector. \( \mathbf{E} \cdot \mathbf{B} = 0 \) arises from the ideal MHD equation where \( \mathbf{E} \perp \mathbf{B} \). In regions where there exists a parallel component of the electric field, without a constant generator charges are free to move, canceling the electric field. \( \mathbf{E} \cdot \mathbf{B} = 0 \) is not always a valid assumption however, for example at the reconnection X-line \( \mathbf{E} \cdot \mathbf{B} \neq 0 \). Plasma data including composition, number density and ion distribution functions are gathered from the Cluster Ion Spectrometry Experiment (CIS). CIS contains two separate instruments: the Hot Ion Analyzer (HIA) and the ion Composition and Distribution Function Analyzer (CODIF).
## Chapter 2: Introduction

<table>
<thead>
<tr>
<th>Instrument</th>
<th>Acronym</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Spacecraft Potential Control Experiment</td>
<td>ASPOC</td>
</tr>
<tr>
<td>Cluster Ion Spectroscopy Experiment</td>
<td>CIS</td>
</tr>
<tr>
<td>Digital Wave Processing Instrument</td>
<td>DWP</td>
</tr>
<tr>
<td>Electron Drift Instrument</td>
<td>EDI</td>
</tr>
<tr>
<td>Electric Field and Wave Experiment</td>
<td>EFW</td>
</tr>
<tr>
<td>Fluxgate Magnetometer</td>
<td>FGM</td>
</tr>
<tr>
<td>Plasma Electron and Current Experiment</td>
<td>PEACE</td>
</tr>
<tr>
<td>Research with Adaptive Paricle Imaging Detectors</td>
<td>RAPID</td>
</tr>
<tr>
<td>Spatio-Temporal Analysis of Field Fluctuation Experiment</td>
<td>STAFF</td>
</tr>
<tr>
<td>Wide Band Data Receiver</td>
<td>WBD</td>
</tr>
<tr>
<td>Waves of High Frequency and Sounder for Probing of Density by Relaxation</td>
<td>WHISPER</td>
</tr>
</tbody>
</table>

Table 2.1: Cluster’s Scientific Instruments.
Chapter 3

Methodology

3.1 Data-analysis

3.1.1 Wavelet Analysis

Wavelet spectrum analysis was performed on high resolution magnetic field and electric field data to discover the dynamics of the wave power variation. A Fast Fourier Transform (FFT) assumes that the original time series data is a superposition of sinusoidal waves (smooth curves). An FFT transfers the signal into frequency space and is used to determine how much power is present at each frequency. We can conduct the same analysis using a wavelet transform. The wavelet transform is better suited for time data series with discontinuities and peaks. A wavelet transform allows for different shapes (as opposed to an FFT) called mother wavelets. The most common is the Morlet mother wavelet which is a Gaussian modulated plane wave given by the following equation:

\[
\psi_0(\eta) = \pi^{-1/4}e^{i\omega_0 \eta}e^{-\eta^2 / 2}
\]  

(3.1)

where \( \eta \) is a non-dimensional time parameter. The Morlet wavelet is good for broadband data and more closely resembles the time data series that we will be analyzing. The wavelet transform is defined by Equation 3.2, where \( x'_n \) is the signal (*) indicated the complex conjugate of the daughter wavelet, \( \psi \) is defined as the
daughter wavelet and is calculated by normalizing and scaling the Morlet mother wavelet, $\psi_0$ (see Equation 3.3).

$$W(s) = \sum_{n'=0}^{N-1} x_{n'} \psi^* \left( \frac{(n' - n) \delta t}{s} \right)$$  \hspace{1cm} (3.2)

$$\psi \left( \frac{(n' - n) \delta t}{s} \right) = \left( \frac{\delta t}{s} \right)^{1/2} \psi_0 \left( \frac{(n' - n) \delta t}{s} \right)$$  \hspace{1cm} (3.3)

$x_{n'}$ is a value of the time series data, $n'$ is the time index, $n$ is the “translational” parameter, $\delta t$ is the time interval and $s$ is the “dilation” parameter (changes the scale size) [Torrence and Compo 2012].

For a faster calculation, the wavelet transform is performed in phase space by taking the Fourier transform, then the result is transformed back using the inverse Fourier transform [Soldin 2009, and all references therewithin].

Because the electric and magnetic field data are recorded on board each spacecraft using separate instruments, the time stamps are independent of each other. Because the $\delta E_\perp / \delta B_\perp$ ratio (the ratio of the change in the perpendicular electric wave power to the change in the perpendicular magnetic wave power) calculation requires both electric and magnetic field data simultaneously a common time stamp must be established - this is done though linear interpolation. First an array containing the “master” time stamps is assembled, where the beginning and end times are taken from the maximum and minimum start and end times respectively, where the time step is set to .045 seconds - this time step is chosen because the high resolution data is recorder at 22.5 Hz. Using linear interpolation method, values of the electric and magnetic fields are determined for each time in the master time stamp array. Linear interpolation works by taking a field values adjacent to the desired time (both before and after) and averaging these values. Figure 3.1 shows the magnetic field data before and after linear interpolation; the left panels show the un-interpolated magnetic field data from spacecraft 3. The right panels show the corresponding linearly interpolated magnetic field data from spacecraft 3 - the interpolated data is agreeable with the unfiltered data.
CHAPTER 3: METHODOLOGY

19

Figure 3.1: The left panel depicts the raw magnetic field data from Cluster and the right panel depicts the linearly interpolated data in nanno-Teslas (nT).
We are interested in the parallel and perpendicular wave power of the magnetic field and electric field for our spectral analysis. Let’s first define the wavelet power spectrum for the magnetic field. The power spectrum for each component of the magnetic field is calculated by performing a continuous wavelet transformation on the desired component of the field value, then by squaring the absolute value of the resulting transformation shown in Equation 3.4.

\[ P = |W|^2 \] (3.4)

The parallel magnetic wave power is calculated by taking the wavelet transformation of the magnitude of the magnetic field signal (see Equation 3.5). The total wave power is defined as the sum of the x, y and z-component magnetic wave power; the following Equation 3.5 shows the total wave power for the magnetic field (the same applies for the electric field as well):

\[ P_{B\parallel} = P \frac{1}{\sqrt{B_x^2 + B_y^2 + B_z^2}} \] (3.5)

\[ P_{B_{tot}} = P_{B_x} + P_{B_y} + P_{B_z} \] (3.6)

Since the magnetic wave power can be defined as the sum of the perpendicular and parallel component power, the perpendicular magnetic wave power is defined by the difference between the total magnetic wave power and the parallel magnetic wave power as follows:

\[ P_{B\perp} = P_{B_{tot}} - P_{B\parallel} \] (3.7)

The electric wave power is calculated in a similar manner. The total electric wave power is defined by the sum of the x, y and z-component electric field power. The total electric field power is simply the sum of the parallel and perpendicular wave power, but because the three dimensional electric field is built under the assumption that \( \mathbf{E} \cdot \mathbf{B} = 0 \), the parallel (with respect to the background magnetic field) electric
wave power is zero. Therefore the perpendicular electric wave power is equal to the total electric field power.

\[ P_{E_{tot}} = P_{E_x} + P_{E_y} + P_{E_z} \]  

(3.8)

\[ P_{E_{||}} = 0 \]  

(3.9)

\[ P_{E_{\perp}} = P_{E_{tot}} \]  

(3.10)

We tested that our method worked by reproducing the $\delta E_{\perp}/\delta B_{\perp}$ plot from Sundkvist et al. [2005].

3.1.2 Boundary Normal Analysis (Minimum and Maximum Variance Analysis and Minimum Faraday Residue)

The magnetopause exists at the magnetic field standoff distance - a location defined by the farthest reaching magnetic field lines. The location of the magnetopause changes depending upon perturbations caused by varying solar wind conditions and physical processes which occur at the boundary. Changes in the magnetopause location are observed by spacecraft transversing along or across the boundary. Boundary normal analysis is an important tool used to distinguish between back and forth boundary motion and a boundary perturbed by processes such as KHI of Flux Transfer Events (FTEs) that produce non-zero curvature of the boundary. The vector that lies normal to the magnetopause boundary layer is parallel to the minimum variance of the magnetic field and to the maximum variance of the $-(\mathbf{V} \times \mathbf{B})$ electric field (MVAE) directions. Previous studies have shown that under intervals of non-linear perturbation, there is bipolar variation in the magnetopause normal component of the magnetic field [Sonnerup and Cahill 1967]. Applying minimum and maximum variance analysis to magnetic field data collected from a spacecraft that has traversed the LLBL, the normal to the magnetopause can be calculated [Paschmann and Daly 1998]. Figure 3.2 depicts magnetic field observations from Cluster under a period of
KHI in boundary normal coordinated using MVAE. $B_J$ is the normal component of the magnetic field, notice the bipolar variation.

We also use Minimum Faraday Residue (MFR) technique when computing the boundary normal directions along the spacecraft trajectory. MFR is an analytical technique used to determine the normal component of a current layer (e.g. the magnetopause) and the velocity along that normal. The normal and motion along the normal are derived from Faraday’s Law:

$$\frac{\delta B}{\delta t} = -\nabla \times E$$  \hspace{1cm} (3.11)

where $B$ is the magnetic field and $E$ is the electric field. The electric field is approximated by $-(\mathbf{V} \times \mathbf{B})$ [Khrabrov and Sonnerup 1998].

### 3.1.3 Walén and de Hoffman Teller Frame Analysis

The de Hoffman Teller (HT) frame is a reference frame in which the electric field, often approximated by $-(\mathbf{V} \times \mathbf{B})$, is zero. The technique was first used to analyze jump conditions across MHD shocks [de Hoffmann and Teller 1950]. The existence of such a frame indicates the existence of a quasi-stationary magnetic field and plasma velocity configuration. The HT velocity $\mathbf{V}_{HT}$ is described as the reference frame velocity in which the $-(\mathbf{V} \times \mathbf{B})$ electric field is zero. Further analysis of such a frame can give insight into the physical phenomenon which occur at a boundary. Because the HT frame can be used to test for and track a discontinuity, the technique is ideal for finding intervals of magnetic reconnection – reconnection being a rotational discontinuity, where the plasma across the discontinuity is connected magnetically [Nykyri 2011a]. Rotational discontinuities satisfy the Walén relation, a relationship between the difference in the plasma flow and $\mathbf{V}_{HT}$ component vectors and the Alfvén speed. Also the Walén relation is satisfied for intermediate shocks [Sonnerup et al., 1995].

To summarize, HT analysis searches for a reference frame in which the $-(\mathbf{V} \times \mathbf{B})$ electric field is zero. This reference frame exists when the plasma flow is aligned with the magnetic field. A good correlation exists when the HT frame velocity $\mathbf{V}_{HT}$
Figure 3.2: Magnetic field parameters in boundary normal coordinates during an interval of KHI at the dawn-flank magnetopause.
minimizes \(- (\mathbf{V} \times \mathbf{B})\). Intervals of magnetic reconnection are detectable using HT analysis because they occur at regions where the \(- (\mathbf{V} \times \mathbf{B})\) electric field is zero because plasma is forced to flow parallel to the reconnected field lines away from the x-line. The Walén relation is then used to distinguish between tangential and rotational discontinuities. [Paschmann and Daly 1998].

### 3.2 Numerical Simulation Codes

#### 3.2.1 Local MHD Simulations

Local (simulation plane is set at the magnetospheric flank where the instability occurs) 2D MHD simulations [Nykyri and Otto 2001, 2004; Nykyri et al. 2006] were set up by establishing a shear flow plane from magnetic field and velocity geometries based on Cluster observations. The magnetosheath and magnetosphere magnetic field vectors were constructed from data when Cluster was believed to be in the magnetosheath and magnetosphere respectively. The shear flow plane is assembled by determining the orientation of the magnetic field to the magnetosheath velocity. \(\psi\) is the angle between the shear flow velocity and the magnetic field on one side of the boundary. \(\psi\) is determined by calculating the angle between the tangent component of the magnetic field – the projected magnetic field onto the magnetosheath velocity – and the perpendicular component of the magnetic field – the difference between the squares of the magnitude of the magnetic field and the tangential magnetic field. On both sides of the boundary, the magnetic field is projected onto the magnetosheath velocity, because this is where the dominant shear exists, although the magnitude of the magnetosphere velocity is still taken into account. The shear flow velocity is determined by the difference between the magnitude of the magnetosheath velocity and the projection of the magnetosphere velocity along the magnetosheath velocity. To express the shear flow in normalized simulation units the shear flow velocity is divided by the averaged magnetosonic speeds (from each side of the boundary), where the magnetosonic speed is the addition in quadrature of the Alfvén and sound speeds.
The simulations are done in two steps. Step 1 is to establish a shear flow plane. If the simulation proves to be stable in the shear flow plane, the k-vector (defined as the direction of propagation of the instability) is slightly tilted to check whether the boundary in unstable [Nykyri et al. 2006] - this is done by adjusting the angle $\phi$. This practice is used because we are using a 2D system, whereas the real system is 3D where KHI can propagate along the direction that is most unstable (the direction that minimizes the magnetic field component along the k-vector). See Figure 3.3 for a visual representation of the simulation plane.

The standard practice in forming initial conditions for the local simulations is to use the satellite data from the event in question. Because the onset condition relies heavily on the orientation of the magnetic filed to the shear flow plane (the magnetopause), an accurate depiction of the magnetic field is very important. Therefore when choosing initial conditions for the 2-d MHD simulations, it is ideal to take values from an unperturbed boundary. A perturbed boundary means deformed field lines (especially when the KHI becomes non-linear), which do not give a realistic representation on the initial state of the system. Choosing data too far before the event is a good work around but possibly results in a region that might not be Kelvin-Helmholtz unstable. Initial conditions from the Global MHD code allows for identifying areas of MSP and MSH plasma more distinctly.

3.2.2 Global MHD Simulations, CCMC Codes

The Open Geospace General Circulation Model (OpenGGCM) is a global MHD simulation. For the magnetosphere, the MHD equations are discretized onto a stretched Cartesian grid. The discretization of differential equations is performed by storing values of $f$ on grid points, then approximating the derivatives second order explicit time integration with conservative and flux-limited spatial finite differences (CCM, 2012). OpenGGCM uses an adaptive grid, capable of $\approx 10^5$ to $\approx 10^8$ cells.

The model’s inputs consist of solar wind plasma and magnetic field properties, where the $B_x$ component of the IMF is averaged over time.
Figure 3.3: Schematic of simulation plane depicting $\psi$ and $\phi$. 
The OpenGGCM model is valid for spatial scales much less than the ion inertial length \( L \ll \rho_i \) as well as temporal scales much less than the ion gyro period [Raeder 2001]. OpenGGCM does not include a plasmasphere and the physics breaks down within 5 \( R_E \) from Earth (CCM, 2012).

We use global MHD simulations in this thesis to set an idea of the large scale structure of the flank magnetosphere and magnetosheath to help guide our analysis.
Chapter 4

Cluster Spacecraft Observations of Periodic Boundary Crossings and Associated Wave Activity During PS IMF Orientation

4.1 Search of Low Latitude Boundary Layer Crossings with Quasi-Periodic Signatures

We searched through Cluster satellite data from 2001 to 2005, primarily focused to the months of June and November because during these months Cluster’s polar orbit around the magnetopause intercepts the magnetospheric flanks. Previous studies (e.g., Fairfield et al. [2000], Hasegawa et al. [2004] and Nykyri et al. [2006]) were used as templates to identify signatures in the plasma parameters (e.g. particle densities, tailward velocities and temperatures) and the magnetic fields. A boundary normal analysis was performed to filter out observations that do not show bipolar variation in the normal component of the magnetic field; this kind of signature as stated in § 3.1.2 suggests the boundary has been perturbed by KHI or a FTE. Furthermore, a Walén and de Hoffman Teller frame analysis was utilized to help identify events
CHAPTER 4: CLUSTER SPACECRAFT OBSERVATIONS OF PERIODIC
BOUNDARY CROSSINGS AND ASSOCIATED WAVE ACTIVITY DURING PS
IMF ORIENTATION

with possible reconnection intervals because reconnection has been shown to occur
in KH vortices [Nykyri et al. 2006; Hasegawa et al. 2009]. In total, five events were
found and will be analyzed to determine similarities and differences between solar
wind conditions.

4.2 Review of Events

Table 4.1 displays the summary for each event. One interesting point to note at
first glance is that the solar wind conditions for one event is quite different form the
next. Four of the events have parker PS-oriented IMF (events 1, 3, 4 and 5); of those
four, two have a southward z-component (events 2 and 3). Although events 2 and 3
share similar IMF conditions, the solar wind speed is quite different, as well are the
drapping effects. The magnetic field for event 2 is parallel to the flow tailward flow
speed, whereas event 3’s magnetic field in the MSH is anti-parallel to the tailward
flow. Simulations performed for both event 2 and 3 prove to have a KH unstable
magnetopause.

Event 5 is the only event with a steady OPS-orientation throughout the entirety
of the event, and is also the only event that has a KH stable magnetopause in the
2D simulations, which is to be expected because of the large tangential field at the
dawn-flank under OPS IMF. The most similar event to 5 is event 4. Like event 5, 4
has a northward z-component to the IMF and a large $V_x$. Also both have anti-parallel
magnetic field orientations compared to the tailward flow.

Event 1 is the only event that has an almost nonexistent z-component to the IMF.

4.3 June 6th, 2002 During Parker Spiral Orientation

4.3.1 Solar Wind Conditions and Overview of Event

Figure 4.1 shows the plasma parameters (in GSM coordinates) as seen by cluster on
June 6th, 2002 from 13:00:00 to 13:40:00 UT. Quasi-periodic fluctuations are observed
### Table 4.1: Summary of events showing quasi-periodic oscillations.

<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
<th>Type</th>
<th>$B_z$ (±/−)</th>
<th>$V_x$ (km/s)</th>
<th>Draping</th>
<th>Wave Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>June 6th, 2002</td>
<td>Event 1</td>
<td>PS</td>
<td>Quiet</td>
<td>-365</td>
<td>Parallel</td>
<td>Unstable</td>
</tr>
<tr>
<td>June 13th, 2002</td>
<td>Event 2</td>
<td>PS</td>
<td>−</td>
<td>-365</td>
<td>Parallel</td>
<td>Unstable</td>
</tr>
<tr>
<td>June 19th, 2004 (i)</td>
<td>Event 3</td>
<td>PS</td>
<td>−</td>
<td>-475</td>
<td>Anti-parallel</td>
<td>Unstable</td>
</tr>
<tr>
<td>June 19th, 2004 (ii)</td>
<td>Event 4</td>
<td>PS</td>
<td>Mostly +</td>
<td>-470</td>
<td>Anti-parallel</td>
<td>Unstable</td>
</tr>
<tr>
<td>June 21st, 2004</td>
<td>Event 5</td>
<td>OPS</td>
<td>+</td>
<td>-417</td>
<td>Anti-parallel</td>
<td>Stable</td>
</tr>
</tbody>
</table>

in the plasma parameters. Cluster intermittently encounters high density, low temperature plasma with high tailward velocities and low density, and high temperature plasma with predominantly low tailward velocities; the density fluctuates between about 0.15 and 6 cm$^{-3}$ and the temperature fluctuates between about $0.1 \times 10^6$ and $56 \times 10^6$ K. Fluctuations in the total pressure range from 0.1 to 0.33 nPa Where the total pressure is defined by the sum of the thermal and magnetic pressures (see Equation 4.1). Dips in the total pressure are well correlated with plateaus in $B_y$.

$$P_T = nkT + \frac{B^2}{2\mu_0}$$  \hspace{1cm} (4.1)

The tailward velocity has a maximum of about -400 km/s and a minimum of 165 km/s. $V_y$ shows variation about 0 km/s with a maximum of about 300 km/s and a minimum of -350 km/s; and $V_z$ is predominantly negative with fluctuations between about -200 and 100 km/s, but spikes to about 325 km/s.

The period of the the density variation (from peak to peak) is about 3 minutes. The period of the magnetic field variation for $B_y$, $B_z$ and $B_T$ is also roughly 180 seconds.

The magnetic field shows bipolar variation in the x and y components, where $B_y$ represents the component most normal to the magnetopause boundary. $B_z$ ranges
Figure 4.1: Plasma and magnetic field parameters from Cluster in GSM coordinates - June 6th, 2002.
from about -8 to 11 nT. There is a clear wave structure observed in $B_y$ roughly between 13:23 and 13:37 UT - quasi-periodic fluctuations with gradual leading edges that plateau followed by peaks (with a maximum peak of about 9 nT) with steep trailing edges (with a minimum trough of about -20 nT). The plateaus in $B_y$ are observed when the total pressure dips to a minimum. There are quasi-periodic fluctuations in $B_z$ with a maximum of about 25 nT and a minimum of about 5 nT; the peaks in $B_z$ occur during when there is a steep decline in $B_y$ just after it peaks. Peaks in the total magnetic field are observed when $B_z$ peaks and there is a steep decline in $B_y$; $B_z$ has a minimum of 5 nT and a maximum of 26 nT. Figure 4.2 depicts variation in the normal component of the magnetic field (normal to the magnetopause boundary) in boundary normal components using maximum variance analysis of the $-(\mathbf{V} \times \mathbf{B})$ electric field. From Figure 4.2 $B_J$ corresponds to the normal component of the magnetic field, whereas $B_K$ and $B_I$ correspond to the tangential components of the magnetic field. Bipolar variation is observed from 13:00:00 to 14:00:00 UT. About mid-way through the hour interval, the normal component of the magnetic field rapidly rises then falls of gradually; this signature has been observed by Nykyri et al. [2006] at the dawn-flank magnetopause under PS orientation. The period of the bi-polar variation in the normal component of the magnetic field is about 180 seconds (which is agreeable with the variation seen in the y-component of the magnetic field in GSM coordinates). Variation in the boundary normal along the spacecraft trajectory for spacecraft 1 from 13:20:00 - 13:40:00 using Minimum Faraday Residue (MFR) method is plotted in Figure 4.3. The MFR method has been shown to perform better at the LLBL [Haaland et al. 2004] than the MVAB and MVAE methods. The black, green and red vectors represent the boundary normal, the average normal and the velocity in the deHoffman Teller (HT) frame respectively. Spacecraft 1 is located at <-3.534,-16.102,-5.625> RE in GSM coordinates and the average normal is <0.511,-0.747,-0.4247> (a unit vector originating from the satellite position). The boundary normal variation across spacecraft 1’s trajectory is bipolar and quasi-periodic. The HT velocity along the spacecraft trajectory is quasi-periodic and bipolar towards the end of the interval.
Figure 4.2: Magnetic field parameters in boundary normal coordinates from Cluster - June 6th, 2002.
Figure 4.3: The boundary normals along spacecraft trajectory using MFR method. The black vectors represent the boundary normals, the green vector represents the average boundary that is compiled using data from the entire interval 13:20 - 13:50 UT and the red vectors represent HT frame velocity.
On June 6, 2002 Cluster periodically encountered the magnetopause boundary at the dawn side LLBL. Spacecraft 3 was located at $<-3.53, -15.79, -6.5> \text{R_E}$ (GSE). The time lagged solar wind data from ACE (Figure 4.4) depicts the plasma parameters. The solar wind plasma density shows variation between about 3.5 to 5.7 cm$^{-3}$. The temperature of the solar wind plasma fluctuates roughly between 15000 to 35000 K. The averaged earthward velocity $V_x$, is about 365 km/s with a maximum of 375 and a minimum of 355 km/s. The $V_y$ and $V_z$ components of the solar wind plasma are somewhat steady and small in magnitude; $V_y$ varies between 0 and -20 km/s and $V_z$ varies from -5 to -19 km/s. The interplanetary magnetic field (IMF) is quite steady from 12:00:00 to 14:00:00 UT (time lagged to Cluster event). $B_x$ varies slightly before 13:00:00 UT, remaining mostly negative, with a maximum of -3.5 nT but shifting slightly positive, with a maximum of about 1.5 nT. After 13:00:00 UT $B_x$ is mostly positive with a maximum of 3.5 nT, but shifting slightly negative (about -0.5 nT). $B_y$ remains at about -5 nT from 13:00:00 to 14:00:00 UT; and $B_z$ begins at about -2 nT till around 12:10:00 and then shifts to about 0 nT till 14:00:00. Drawing the magnetic field orientation in the x-y-plane shows that the IMF is in the PS-orientation, generating the quasi parallel shock on the dawn-side of the magnetospheric flanks (see Figure 4.5).

### 4.3.2 Single Spacecraft Analysis

*Hasegawa et al.* [2006] have shown an analytical technique to detect rolled-up Kelvin-Helmholtz vortices. This single spacecraft technique is valuable when the spacecraft separation is too small to allow for multi-spacecraft techniques *Hasegawa et al.* [2004]. When KHI forms and becomes nonlinear, rolled up vortices form. Mixing of the hot tenuous plasma from the magnetosphere and cold dense plasma from the magnetosheath occur inside of these rolled-up vortices. A unique feature has been observed through simulations where the tailward flow of part of the low-density plasma obtains faster than magnetosheath speeds [*Hasegawa et al.* 2006]. In order for the force balance in the radial direction to be maintained, the hot tenuous plasma must rotate
Figure 4.4: Solar Wind conditions from ACE - June 6th, 2002.
Figure 4.5: Solar Wind IMF - June 6th, 2002. Image is not to scale.
faster than the cold dense plasma [Nakamura et al. 2004]. This feature has been observed in Cluster data at both the dawn and dusk flanks of the LLBL under periods of northward IMF solar wind conditions [Hasegawa et al. 2006]. Observations also show that this low-density faster than sheath flow occurs in regions where there is a significant density gradient [Hasegawa et al. 2006]. This technique is applied by plotting the tailward flow against the density of the plasma from in-situ measurements where the tailward flow is represented by $V_x$ (GSM). Figure 4.7 depicts the low-density faster than sheath feature taken from a 3D simulation at zero degrees from the instability propagation [Hasegawa et al. 2006]. This feature is found in a confirmed KHI event (using multi-spacecraft techniques) [Hasegawa et al. 2006]. Figure 4.7 shows a signature that is consistent with the simulation results [Hasegawa et al. 2006]. Hasegawa shows that during the November 20, 2001 event, the low-density faster than sheath flow occurs in regions where there is a significant density gradient (see Figure 4.7).

Because the satellite separation is so small for this event, we must rely on single spacecraft techniques to assess whether or not the event is KHI. The smallest separation is between spacecrafts 2 and 3 which is $\approx 180$ km whereas the smallest satellite separation for the Hasegawa et al. [2006] event is approximately 1960 km (between satellites 2 and 3). Figure 4.8 depicts the plasma number density (top panel) and the x-component of the flow velocity $V_x$ (center panel) from 13:30:00 - 13:50:00 UT in GSM coordinates for spacecraft 1, 2 and 3. The number density for spacecraft 1 (black) and 3 (green) is strongly correlated, whereas small fluctuations from the basic trend are observed for spacecraft 4 (blue). A similar signature is present in the tailward flow - spacecraft 1 and 3 show strong correlation, with spacecraft 4 fluctuating...
Figure 4.7: Plasma parameters depicting low density faster than sheath flow during intervals with density gradients (left panel). The red points represent points at which the tailward speed, $V_x$ is $< -290$ km/s (where the MSH plasma has a tailward speed of $\approx 250$ km/s). Tailward flow plotted against plasma density from November 20th, 2001 (right panel) [Hasegawa et al. 2006].

Over the main trend, where these fluctuations have the highest amplitude between $\approx 13:30:00$ and $13:32:30$ UT. The average tailward flow for the MSH-like plasma is roughly -200 km/s. The tailward velocity is plotted as a function of the plasma density in the bottom panel. There is a dense population of plasma below cm$^{-3}$ between -150 and 150 km/s. The key feature here in comparison with Hasegawa et al. [2006] is the higher tailward velocity for MSP-like plasma (defined by the low number density). Hasegawa et al. [2006] showed (for Northward IMF cases where KHI was confirmed) faster than sheath flow for MSP-like densities when the KH vortices become rolled-up (see 2.1.6). In contrast to observations made by Hasegawa et al. [2006] on November 20th, 2001 (see Figure 4.7) the signature produced from June 6th, 2002 (Figure 4.8) shows concentrated clusters of low density plasma in the low tailward velocity region. The clustering in the low density region could be due to the fact that Cluster is much closer to the MSP throughout the entire interval for the June 6th, 2002 event than in the Hasegawa et al. [2006] event where the average density is much higher.
Figure 4.8: Number density (top panel), $V_x$ (middle panel), and plasma flow vs density using the single spacecraft technique (bottom panel) for June 6th, 2002 between 13:30:00 and 13:50:00 UT. The black, green and blue curves correspond to spacecraft 1, 3 and 4 respectively.
4.3.3 Large Scale Structure of Magnetosheath and LLBL Using Global MHD Models from the Community Coordinated Modeling Center (CCMC)

A global 3D MHD model was run with solar wind data from ACE from June 6th, 2002. As stated in § 4.3.1 the IMF has was relatively quiet in $B_z$ and $B_x \approx 0$. The solar wind input for the OpenGGCM model is in the Parker-Spiral Orientation. The OpenGGCM model takes an average of the x-component of the solar wind IMF. Figure 4.9 depicts the shocked solar wind plasma draped along the magnetopause. We used this global 3-d MHD model to determine the structure of the magnetic field on the MSH side of the boundary. From the results of the global simulation, we see that $B_x$ is aligned with the MSH flow. This result is consistent with the Cluster observations and was used when setting up the initial conditions for the local 2-d MHD simulation.

4.3.4 Local MHD Simulations of KHI for Initial Conditions Picked from Cluster Data

Table 4.2 shows the initial conditions chosen from the Cluster data; the data shown was gathered slightly before the event. It should be noted that the 2-d simulation takes the tailward velocity in the shear flow plane and divides it over the MSP and MSH sides of the boundary. For example $V_x$ from table 4.2 gives a velocity of 146 km/s in the MSP and -139.42 km/s in the MSH - this corresponds to a flow speed of -292 km/s in the MSH. The plasma $\beta$ for this event (calculated from Cluster observations) has a minimum of about 1.14 (from satellite 4) and a maximum of about 12.1 (from satellite 4). The plasma $\beta$ was moderate in the Johnson et al. [2001] event, about 2 in the MSH decreasing to below 1 when encountering the spacecraft is located in a transition region (density gradient) from the MSH to MSP.

Figure 4.10 shows the progression of the 2D MHD simulation after the vortices are well formed. The left panel plots the magnetic field vector and contour lines; the current density in the z direction is represented by the color bar. The right
Figure 4.9: Draping effect due to fast shock at the bow shock depicted by the openGGCM global model of the June 6th, 2002 event.

<table>
<thead>
<tr>
<th>MSP</th>
<th>MSH</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n (cm^{-3})$</td>
<td>0.5</td>
</tr>
<tr>
<td>$T (10^6 K)$</td>
<td>40</td>
</tr>
<tr>
<td>$B (nT)$</td>
<td>8.8</td>
</tr>
<tr>
<td>$\psi_1$ (degrees)</td>
<td>36.5</td>
</tr>
<tr>
<td>$\psi_2$ (degrees)</td>
<td></td>
</tr>
<tr>
<td>$V_x (km/s)$</td>
<td>-146</td>
</tr>
<tr>
<td>$P_{dyn}$ (nPa)</td>
<td>0.415</td>
</tr>
<tr>
<td>$\beta$</td>
<td>13.5</td>
</tr>
<tr>
<td>$V_A (km/s)$</td>
<td>271</td>
</tr>
<tr>
<td>$V_S (km/s)$</td>
<td>525</td>
</tr>
</tbody>
</table>

Table 4.2: Simulation Initial Conditions (Cluster Data).
CHAPTER 4: CLUSTER SPACECRAFT OBSERVATIONS OF PERIODIC BOUNDARY CROSSINGS AND ASSOCIATED WAVE ACTIVITY DURING PS IMF ORIENTATION

panel plots the plasma velocity where the plasma number density is represented by the color bar. The left and right sides of each panel represent the MSH and MSP respectively. The bottom of each plot is the sunward direction whereas the top is tailward. The condition for KHI (equate 2.3) is satisfied on each side of the boundary initially and the onset time of the instability – when the KH becomes non-linear – occurs at 211.741 seconds into the simulation. Because the satellite separation is so small, we are limited to using single spacecraft measurements when estimating the wavelength. Based on the observed 3 minute period of the instability from Cluster and the flow speed, we can estimate the wavelength \( \lambda = TV_{ph} \). For a very thin boundary the KH mode propagates with a speed of \( V_{KH} = V_{msh} \frac{\rho_{msh}}{\rho_{msh} + \rho_{msp}} \) \cite{Chandrasekhar 1961}, and for the boundary with finite thickness this phase speed becomes smaller. Assuming the \( V_{ph} \approx \frac{1}{2} V_{msh} \) we estimate a wavelength of \( \approx 26000 \) km. We use this estimated wavelength to scale the length of our simulation box along the direction of propagation. \( \phi \) represents the tilt of the shear flow plane, which has been adjusted from 0 because in the real system, the instability does not always grow parallel to the equatorial plane, but instead will grow in the direction where the onset condition is satisfied \cite{Nykyri et al. 2006}.

Figure 4.11 depicts the time series of the local 2D simulation (right panel) alongside of the Cluster observations (left panel). The density and temperature profiles show similar characteristics with the Cluster observations seen in Figure, where Cluster encounters small troughs in the peaks of the density and temperature profiles. In the simulation overview plot, \( B_y \) corresponds to the component of the magnetic field normal to the boundary layer. Observations from the simulation show that \( B_y \) is steady then increases drastically followed by a gradual decline which is consistent with the Cluster observations of the normal component of the magnetic field \( B_J \). The most unique feature is how dips in the total pressure coincides with the plateau in \( B_y \) which is in agreement with the Cluster observations - troughs in the total pressure coincide with plateaus in \( B_y \) (see Figure 4.1).
CHAPTER 4: CLUSTER SPACECRAFT OBSERVATIONS OF PERIODIC BOUNDARY CROSSINGS AND ASSOCIATED WAVE ACTIVITY DURING PS IMF ORIENTATION

Figure 4.10: Local MHD simulation after onset of KHI. The left panel plots the magnetic field vectors with the current density (background color). The right plots the velocity vectors and the plasma density (background color). The sunward direction is located towards the bottom of the Figure - the MSH is on the left side and the MSP is on the right side of each panel.
Figure 4.11: Plasma and magnetic field parameters produced by a virtual satellite in the simulation plane.
4.3.5 Analysis of Plasma Wave Activity in Association with Boundary Crossings

Spectral Analysis

We are interested in finding evidence of mode conversion because of the associated KAW activity. To identify signatures of mode conversion, we will search for an amplification in the perpendicular magnetic wave power; Johnson et al. [2001] showed that there is an amplification in the perpendicular magnetic wave power associated with mode conversion. Because our event takes place over a much shorter interval than that of Johnson et al. [2001] and we are dealing with multiple crossings, it is more beneficial for us to analyze the total integrated power. The total integrated power is calculated by taking the wavelet transform of the high resolution magnetic field, then using the trapezoidal rule integrate over all frequencies \((f_{KH} - f_{cp})\).

From Figure 4.12, the density plotted in panel (a) is a rough guide as to which region spacecraft 3 is located (e.g., whether the plasma is more MSP or MSH-like). Panel (b) depicts the ratio of the perpendicular temperature to the parallel temperature. Peaks in \(T_\perp/T_\parallel\) are mostly correlated to gradients in the number density. The perpendicular velocity, parallel velocity and ratio of the two are plotted in panel (c). \(V_\perp\) stays mostly positive through the entire interval; troughs in \(V_\perp\) are observed during regions of low plasma number density, whereas and peaks \(V_\perp\) is observed during intervals of higher number density. \(V_\parallel\) stays mostly below zero over the course of the entire interval. In general, the troughs in the \(V_\parallel\) signature overlap with the peaks in \(V_\perp\) and the number density.

Various waves have been observed in the auroral and cusp regions, identified by regions of peaks in the wave power spectrum near the local ion cyclotron frequency \((f_{cp})\); these waves have been identified as electrostatic ion-cyclotron waves for spectral peaks above \(f_{cp}\) and electromagnetic ion-cyclotron waves for spectral peaks less than \(f_{cp}\) [Sundkvist et al. 2005, and all references therewithin]. Analysis of the \(\delta E_\perp/\delta B_\perp\) has been used in the identification of electromagnetic wave modes on the order of \(V_A\) [Sundkvist et al. 2005]. Panels (d), (e) and (f) of Figure 4.12 depict the perpendicular electric wave power, perpendicular magnetic wave power and the ratio between the
two $\delta E_\perp/\delta B_\perp$ from 13:00:00 to 14:00:00 UT respectively. The wave powers were calculated using wavelet analysis with a Morlet mother wavelet for the frequency range of 0.005556 Hz ($f_{KH}$) to 2 Hz. The ion cyclotron frequency $f_{cp}$ is fairly steady with slight variation throughout the entire interval at roughly 0.3 Hz. When satellite 3 is in MSP-like plasma for an extended time, the wave power at higher frequencies is relatively low for $\delta E_\perp$ (when compared to the frequent peaks in wave power throughout the entire interval). Peaks in $\delta B_\perp$ are consistently associated with regions of MSH-like plasma. Furthermore, these peaks with the highest wave power occur just below $f_{cp}$, whereas peaks in $dE_\perp$ are slightly above $f_{cp}$. During intervals of higher perpendicular power in the magnetic field near $f_{cp}$ (and MSH densities) $T_\perp/T_\parallel$ peaks above 1. A high $\delta E_\perp/\delta B_\perp$-ratio occurs periodically, concentrated over intervals of low plasma number densities and is observed for frequencies above $\approx 0.1$ Hz. Intervals of observed lower wave power for $\delta E_\perp/\delta B_\perp$ is associated with MSH-like plasma regions. The Alfvén speed (plotted in black over the $dE_\perp/dB_\perp$ spectrum in panel (f)) remains steady until $\approx 13:09:00$ UT, where it decreases dramatically and fluctuates drastically for the remainder of the interval. For frequencies below $f_{cp}$, $\delta E_\perp/\delta B_\perp$ is on the order of the local $V_A$ for the following intervals: 13:16:15 - 13:17:10, 13:36:45 - 13:37:15 and 13:47:15 - 13:48:00. These three intervals are associated with peaks in the perpendicular wave power of the magnetic field less than $f_{cp}$.

Panel (g) of Figure 4.12 depicts the ratio of the perpendicular to the total magnetic integrated power. The peaks occur when there is a gradient in the density, which is consistent with the magnetopause crossing shown by Johnson et al. [2001]. The amplification in perpendicular wave power observed by Johnson et al. [2001] resembles a step function - the wave power is low in the MSH and when the spacecraft encounters the magnetopause there is significant amplification; the wave power then remains high as the spacecraft continues into the MSP. Our event however, does not resemble a step function, because Cluster crosses the magnetopause several times. The most pronounced peak occurs at the beginning of the first interval of wave activity below $f_{cp}$ whose $\delta E_\perp/\delta B_\perp$ ratio is on the order of the $V_A$ (13:16:15 - 13:17:10 UT). Just before this interval at around 13:15:00, spacecraft 3 is in the low density plasma (MSP-like). Roughly one minute later spacecraft 3 encounters a gradient in
the plasma number density - at this time we see a peak in the integrated wave power ratio.

**Wave Polarization**

Using high resolution magnetic field data, hodograms are used to determine the polarization of wave packets. Hodograms track the magnetic field in the direction of propagation of a wave, where the direction of propagation is determined by MVAB. A hodogram of a linearly polarized wave in the maximum variance/intermediate variance plane would resemble a straight line. Analysis of the high resolution magnetic field data was done by visually examining the time series data to identify wave packets. Hodograms are then compiled during these intervals. From the hodograms we can determine the polarization of the wave (either right-hand or left-hand polarization with respect to magnetic field in the spacecraft frame). High resolution magnetic field data is filtered above and below the ion cyclotron frequency by performing a fast Fourier transform, extracting frequencies at the ion cyclotron frequency and performing a reverse transform (see Figure 4.13 for the interval 13:16:15 - 13:17:10 UT). $B_x$ is plotted in panel (a), $B_y$ is plotted in panel (b), $B_z$ is plotted in panel (c) and $B_t$ is plotted in panel (d). The top panel of each component depicts the low pass magnetic field data, the middle panel depicts the high pass magnetic field data and the bottom panel depicts the unfiltered magnetic field data. From the low pass data, small wave packets are made apparent.

Figure 4.14 shows 2 hodograms taken from the wave activity interval (frequencies below $f_{cp}$ whose $\delta E_\perp/\delta B_\perp$ ratio is on the order of the $V_A$) the calculated using the minimum variance analysis of the high resolution magnetic field. Eigenvectors $i$, $k$ and $j$ represent the minimum, intermediate and maximum variance directions of $B$ respectively, where $i$ corresponds to the direction of propagation of the wave. The angles $iB$, $kB$ and $jB$ represent the angle between the minimum, intermediate and maximum eigenvectors and $B$. The angle $iB$ (from the left panel of Figure 4.14) is greater than $90^\circ$ which means that the direction of propagation of the wave is out of the page - the curve in the maximum variance/intermediate variance plane rotates counter-clockwise (when looking in the direction of propagation) which means that
Figure 4.12: (a) number density. (b) temperature. (c) Red line corresponds to the velocity component parallel to the background magnetic field, the blue curve corresponds to the perpendicular velocity and the black line shows the ratio between the two. (d) $\delta E_\perp$. (e) $\delta B_\perp$. (f) $\delta E_\perp/\delta B_\perp$. (g) Total integrated power, $\delta B_\perp/\delta B_{tot}$. 
Figure 4.13: Filtered high resolution magnetic field for all spacecrafts taken from the wave activity interval. The black, red, green and blue curves represent data from satellites 1, 2, 3 and 4 respectively.
Figure 4.14: Hodograms from wave activity interval. Eigenvectors \( i \), \( k \) and \( j \) represent the minimum, intermediate and maximum variance directions of \( \mathbf{B} \) respectively, where \( i \) corresponds to the direction of propagation of the wave. The angles \( \angle iB \), \( \angle kB \) and \( \angle jB \) represent the angle between the minimum, intermediate and maximum eigenvectors and \( \mathbf{B} \).

The wave is left-handed. Similarly for the right panel, \( \angle iB \) is less than 90°, which means that the direction of propagation is into the page; the curve rotates counter-clockwise, so the wave is also left-handed. The hodograms suggest that the period is roughly 0.375 s (\( \omega_{sc} = 2.667 \text{ Hz} \)) - the duration of the curve. During this wave interval there is substantial plasma flow (with a magnitude of roughly 300 km/s) which would suggest that the frequency is doppler shifted.

\[
\omega_{sc} = \omega_{plasma} \left( 1 + \frac{V_{plasma}}{V_{ph}} \cos(\theta_{kV}) \right) \tag{4.2}
\]

Equation 4.2 shows the frequency in the spacecraft frame \( \omega_{sc} \) doppler shifted from the plasma frame \( \omega_{plasma} \), where \( V_{plasma} \) is the plasma frame speed, \( V_{ph} \) is the phase speed and \( \theta_{kV} \) is the angle between the flow velocity \( V_{plasma} \) and the direction of propagation \( \mathbf{k} \). For the Alfvén/ion cyclotron mode the phase speed is given by \( V_{ph} = V_A \cos(\theta_{kB}) \), where \( \theta_{kB} \) is the angle between the direction of propagation and the magnetic field (same as \( \angle iB \) from the hodograms). If the quantity inside the parenthesis on the right hand side is negative, it will relate to the wave in the plasma frame to have the opposite polarization than observed in the spacecraft frame.
There is a $180^\circ$ ambiguity in the minimum variance direction of the magnetic field ($\mathbf{k}$) when using MVAB, hence we calculate four possible frequencies in the plasma frame (see Table 4.3). From Table 4.3, each scenario for the frequency in the plasma frame is left-handed; as stated above, there will be a shift in the polarization between the reference frames if the quantity inside the parenthesis is negative. We calculate frequencies on the order of the observed ion cyclotron frequency ($f_{\text{ip}} \approx 0.3$ Hz) for the 1st and 4th case ($\approx 0.3135$ Hz). These results suggest that the observed wave around 13:16:33 UT could be an ion cyclotron wave, however this does not rule out the existence of a KAW. In order to determine KAW activity we will need to reconstruct the $k$-vector using the wave telescope method which is not in the scope of this thesis.

### 4.3.6 Search for Reconnection Intervals

A search for slopes satisfying the Walén relation and HT frame was performed using a sliding window method over 0.22 minute intervals with 5 percent increments for the interval 13:00:00 - 14:00:00 UT. The Walén relation was satisfied for satellites 3 and 4 from 13:49:09 - 13:49:21 UT (top and bottom panel of Figure 4.15 respectively). The reconnection intervals overlap because the satellite separation is relatively small.

For spacecraft 1, $V_{HT} = <-262.17,-203.78,91.84> \text{ km/s}$ and the correlation coefficient in the HT frame is 0.973. The slope for the Walén relation is -0.905 with a correlation coefficient of -0.979. For spacecraft 3, $V_{HT} = <-276.55,-196.23,71.93> \text{ km/s}$.
km/s and the correlation coefficient in the HT frame is 0.977. The slope for the Walén relation is -0.935 with a correlation coefficient of -0.986.

The plasma properties suggest that spacecraft 1, 3 and 4 are traversing MSP-like plasma - the number density is roughly \( \text{cm}^{-3} \) for the reconnection interval. There is a drastic change in the velocity from the start and end of the reconnection interval - from \(<-85.3, 46.8, -2.6> \) to \(<-5.7, -116.9, -21.2> \) km/s for satellite 1. The magnetic field goes from \(<-10.9, -14.7, 4.1> \) to \(<-16.3, -5.4, 7.5> \) nT for satellite 1.

The HT frame velocity \( V_{HT} \) is accelerated from the MSH flow for both spacecraft 1 and 3; the x-component of \( V_{HT} \) is roughly 3 times \( V_x \), the y-component of \( V_{HT} \) is roughly 2 times \( V_y \) and the z-component of \( V_{HT} \) is roughly 3 to 4 times \( V_z \). This acceleration is more pronounced in spacecraft 3. This observed acceleration could be from Cluster traversing through the outflow region of the reconnection region, tailward of the x-line, which would accelerate the plasma tailward. Furthermore, Paschmann et al. [2005] suggest that negative slopes in the Walén relation is an indication of a crossing tailward of the X-line; both satellite 1 and 3 observe negative slopes in the Walén relation.

4.3.7 Examination of Ion Spectrograms and Distribution Functions During Reconnection Intervals, Wave Intervals and Reference Intervals

Next we will show examples of ion spectrograms (energy - time), and ion distribution functions during reference intervals in the MSP and MSH, during reconnection intervals and during wave activity.

Figure 4.16 shows a plot depicting the ion wave activity for satellite 3 produced from the Cluster Active Archive (CAA). The number density from the CIS instrument is plotted in panel (a). Again, the density is shown to act as a rough guide as to what region of plasma the spacecraft is traversing. The ion pitch angle distribution from the HIA instrument is plotted in panel (b) for the high energy ions and panel (c) for the low energy ions. In general, when spacecraft 3 enters MSH-like plasma, large flux is observed for pitch angles for the higher energy ions from 90 to 180 degrees
CHAPTER 4: CLUSTER SPACECRAFT OBSERVATIONS OF PERIODIC BOUNDARY CROSSINGS AND ASSOCIATED WAVE ACTIVITY DURING PS IMF ORIENTATION

Figure 4.15: Walén relation and deHoffman-Teller frame for spacecraft 1 (top left panel) and 3 (top right panel) and the plasma and magnetic field properties for all spacecraft (bottom panel).
(perpendicular to anti-parallel with respect to the background magnetic field). For the low energy ion population, the largest flux (when spacecraft 3 is traversing MSH-like plasma) is associated with pitch angles that are perpendicular to anti-parallel to the background magnetic field, with the exception of $\approx 13:24:15$ UT where the pitch angle is spread from parallel to just about anti-parallel. The Energy spectrograms for the parallel, anti-parallel, perpendicular and omni-directional ions are plotted in panels (d), (e), (f) and (g) respectively. There is an increased flux in the parallel, anti-parallel and perpendicular directions for ion energies ranging from just below 100 eV to just above 1 keV. From panel (g), two distinct ion populations are observed (in the omni-directional direction) - high energy ions (with energies of about 10 keV) in the MSP-like plasma and lower energy ions (of about a few keV) in the MSH-like plasma. These 2 ion populations are clearly separated in energy space, but become indistinguishable later in the interval (see shaded region of Figure 4.16, panel (g)). Ion energy populations become indiscriminant in energy space signifying a region of ion mixing Hasegawa et al. [2003]. Note that the reconnection interval that we reported was during this time period; reconnection can produce plasma mixing.

Ion distribution functions are utilized to analyze the velocity of ions during periods of presumed intervals of magnetic reconnection and wave activity. The ion distribution functions plot the particle (ion) flux in velocity space, where the velocity is split into 3 components - $V_{\text{par}}$, $V_{\text{perp}1}$ and $V_{\text{perp}2}$. The $V_{\text{par}}$ component defines the velocity of the particles parallel to the background magnetic field whereas $V_{\text{perp}1}$ and $V_{\text{perp}2}$ are defined as the particle velocity perpendicular to the background magnetic field where $V_{\text{perp}1}$ is in the direction of the convection flow [Lavraud 2011]. The particle flux is calculated by integrating the first velocity moment from data provided by in-situ measurements from the HIA instrument on board Cluster spacecrafts 1 and 3. All ion distribution functions used in this research were provided by Benoit Lavraud of Centre d'Etude Spatiale des Rayonnements (CESR). We have examined ion distribution functions during reconnection events, wave activity and at reference intervals to put ion distribution functions in context.
Figure 4.16: Ion spectrograms for spacecraft 3 from Cluster Active Archive.
Taylor and Lavraud [2008] analyzed ion distribution functions at the dusk-flank magnetopause during an interval of KH activity. They observed a high energy population of ions with a perpendicular temperature anisotropy (MSP origin) and a low energy population (of MSH origin). The hot population (high energy) with the perpendicular temperature anisotropy is believed to be a product of magnetic drifts [Fujimoto et al. 1998; Hasegawa et al. 2003]. The cold population (low energy) with the perpendicular temperature anisotropy is suggested to be a result of reconnection via KHI [Wilber and Winglee 1995]; it has also been suggested that this temperature anisotropy could be caused by diffusion via KAW [Wing et al. 2006]. A cold population was also observed with a parallel temperature anisotropy which is suggested to be a product of diffusion in KH vortices; reconnection inside KH vortices has also been suggested to produce field aligned ion beams [Otto and Fairfield 2000; Nykyri et al. 2006; Nishino et al. 2007a,b].

**MSH Reference**

Reference ion distribution functions for when satellite 3 is in MSH-like plasma and MSP-like plasma are are shown in Figure 4.17 and Figure 4.18 respectively. There are 3 typical ion distribution functions observed in the MHS-like plasma. From Figure 4.17 we see a shifted Maxwellian distribution weighted in V\textsubscript{perpl} (top panel); the center panel depicts a low energy population that is pancaked and weighted in V\textsubscript{perpl}; in the bottom panel, the general shape is a maxwellian distribution with a high energy population slightly pancaked in V\textsubscript{par}.

**MSP Reference**

When satellite 3 is traversing MSP-like plasma, there are 3 typical types of observed ion distribution functions. From Figure 4.18 we see a distribution slightly pancaked in V\textsubscript{perpl} (top panel); the center panel shows a distribution slightly weighted in V\textsubscript{par}; the bottom panel shows the high energy population is pancaked in V\textsubscript{perpl} and the low energy population is pancaked in V\textsubscript{par}. 
Figure 4.17: Reference ion distribution functions for intervals when satellite 3 is in MSH-like plasma.
Figure 4.18: Reference ion distribution functions for intervals when satellite 3 is in MSP-like plasma.
Distribution Functions During Reconnection

Figure 4.19 shows ion distribution functions from the reconnection interval 13.8191 - 13.8224 UT (13:49:08.76 - 13:49:20.64 UT when the Walén relation was satisfied). During this interval, both satellites (1 and 3) are in the intermediate density plasma (with mixtures of MSP and MSH-like densities); three distinct ion populations are observed. For satellite 1 the high energy ion population (of MSP origin, blue color) is Maxwellian and slightly weighted in \( V_{\text{par}} \) whereas the intermediate energy population (green color) is pancaked in \( V_{\text{par}} \); the low energy population (MSH origin, red color) is pancaked in \( V_{\text{perp}1} \). For satellite 3 the high and intermediate energy population (blue and green color respectively) have a shifted Maxwellian distribution slightly weighted in \( V_{\text{par}} \). The low energy (red color) population is pancaked in \( V_{\text{perp}1} \). The low energy population is concentrated in \( V_{\text{perp}1} \) and slightly pancaked in \( V_{\text{perp}2} \) (from the right plot of the bottom panel). Both satellites 1 and 3 observe a D-shaped low energy population (red color). Phan et al. [2001] observed a D-shaped low energy ion population during reconnection intervals at the dawn-flank magnetopause. D-shaped distributions have also been observed during during reconnection at the high-altitude cusp [Fuselier et al. 2000].

Distribution Functions During Wave Activity

Next we will discuss the ion distribution functions from an interval where wave activity occurs (see § 3.1.3). Figure 4.20 shows ion distribution functions from the interval 13:16:01 - 13:17:37 UT where the \( \delta E_\perp/\delta B_\perp \) ratio is on the order of the \( V_A \) below \( f_c \). At the start of the interval, the high energy ions have are slightly pancaked in \( V_{\text{par}} \) and weighted in \( V_{\text{perp}1} \), whereas the low energy population is weighted in \(-V_{\text{par}}\) and \( V_{\text{perp}1} \); in the \( V_{\text{perp}1} \) and \( V_{\text{perp}2} \) frame, the entire distribution is weighted in \( V_{\text{perp}1} \); as time progresses the low energy ions become pancaked in \( V_{\text{perp}1} \). The ion distribution returns to a Maxwellian distribution after the wave activity interval. Much like the MSH reference distributions, the distribution as a whole is weighted in \( V_{\text{perp}1} \) with two distinct ion populations - the high energy population is weighted in \( V_{\text{perp}1} \) and the low energy population is weighted in positive \( V_{\text{perp}1} \) and negative \( V_{\text{par}} \). All but
CHAPTER 4: CLUSTER SPACECRAFT OBSERVATIONS OF PERIODIC BOUNDARY CROSSINGS AND ASSOCIATED WAVE ACTIVITY DURING PS IMF ORIENTATION

Figure 4.19: Ion distribution functions around the reconnection interval for spacecraft 1.
the first distribution function (13:16:25 UT) during the wave activity (Figure 4.20) show observations where the low energy population is pancaked in $V_{\perp 1}$ and all of the distribution except for the last one in the interval (13:17:13 UT) show observations where the high energy population is slightly pancaked in $V_{\perp 1}$. For this interval the plasma $\beta \approx 1.15$; Stochastic ion heating at the magnetopause via KAW turbulence can heat low energy ions perpendicular to the magnetic field when plasma $\beta \approx 1$ [Johnson and Cheng 2001].
Figure 4.20: Ion distribution functions for intervals during KAW activity.
Chapter 5

Discussion

We have searched for new KHI events by identifying quasi-periodic signatures in the plasma and magnetic field properties from four years of Cluster data (when the satellites’ orbit was in the LLBL in the magnetotail). We limited our results by excluding events which did not show bi-periodic variation in the normal component of the magnetic field (from the MVAE) and that did not also have possible reconnection intervals (from the Walén and HT analysis). We have shown observations of quasi-periodic activity in the plasma and magnetic field parameters when Cluster traversed the LLBL on June 6th, 2002. From these observations, we have conducted minimum and maximum variance analysis verifying that there is bipolar variation in the normal component of the magnetic field and quasi-periodic variation in boundary normals along the satellite trajectory. This variation of boundary normals along the satellite trajectory and bipolar variation of the magnetic field show the existence of a non-zero curvature of the magnetopause boundary which could be produced from KHI or an FTE. Using the single spacecraft technique, we have located an interval where the MSP-like plasma is moving tailward at speeds faster than the MSH-like plasma - Hasegawa et al. [2006] have shown that this signature occurs when the non-linear KH vortices become rolled up. We have run local 2D MHD simulations with initial conditions taken from Cluster observations. We have estimated from observations that $\lambda \approx 26,000$ km, $V_{phase} \approx 133$ km/s and $T \approx 3$ min and set $L_{box} \approx \lambda$ for
the simulation. These simulations verified a KH unstable magnetopause and have reproduced similar key characteristics in the plasma and magnetic field parameters.

At the end of Chapter 3, we showed the general SW conditions for all 5 events, the draping effects along the magnetosphere, the general MSH plasma properties and simulation results. Each event was unique compared to the next. Although there were 3 events with PS-orientation, the z-component of the IMF, draping effects and x-component of the SW velocity varied. Event 1 was the only event that was quiet with respect to the IMF z-component. Event 5 was the only event during OPS-orientation thus generating more tangential magnetic field along the boundary that could stabilize KHI, indeed the local 2D MHD simulations verified that the boundary was stable.

We have shown from the $\delta E_\perp/\delta B_\perp$-ratio, that there exists intermittent wave activity where the ratio is of the order of the local $V_A$ in the low frequency range (below $f_{cp}$). This combined observations of emitted magnetic wave power below $f_{cp}$ suggests that these waves could be Shear Alfvén Waves, KAWs and/or electromagnetic ion-cyclotron waves (left-hand polarized wave with $f_{sc} \approx 0.4245$ Hz and frequency $f_{sc} \approx 0.3125$ Hz due to doppler shift from substantial plasma flow). We have shown that there are peaks in the ratio of the perpendicular and total magnetic integrated power when there is a gradient in the number density which is in agreement with Johnson et al. [2001] who have shown that there is an amplification in the perpendicular wave power associated with mode conversion at the magnetopause from Compressional Alfvén Waves to KAWs.

The low energy ion population that is pancaked across the perpendicular velocity component intervals of wave activity where the $\delta E_\perp/\delta B_\perp$ is on the order of the $V_A$ for the low frequency range (below $f_{cp}$) in conjunction with the above mentioned observations could be due to an energy exchange between waves and the ions. Johnson and Cheng [2001] have suggested Stochastic ion heating perpendicular to the background magnetic field due to transverse KAWs. Observations of the intermediate energy ions pancaked in $V_{par}$ are consistent with the acceleration from the MSH flow to the HT frame velocity - the reconnection interval is observed in the outflow region tailward of the X-line. The low energy ion population is panckaed and D-shaped in
the reconnection interval. This D-shape observation of the low energy ions is consistent with observations by Phan et al. [2001] at the dawn-flank magnetopause during reconnection intervals. Antonius Otto (GEM 2012) showed 2D MHD simulations of magnetic reconnection identifying strong non-adiabatic plasma heating associated with the shocks generated by magnetic reconnection only when plasma beta was low in the MSH. The efficiency of this mechanism for the observed event with plasma $\beta \approx 1$ needs to be further investigated via simulations, but may explain the modest perpendicular heating observed during the reconnection interval.
Chapter 6

Conclusions and future work

Here we summarize the results of our research and discuss briefly the future work.

* Found a new and unpublished KHI event at the dawn-flank magnetopause under PS orientation ($T \approx 180$ s, $V_{ph} \approx 146$ km/s, $\lambda \approx 26,000$ km and $f \approx 0.0056$ Hz).

* Identified electromagnetic plasma wave activity whose $dE_\perp/dB_\perp$ ratio is on the order of $V_A$ for the low frequency range.

* Identified a left-hand polarized wave with an estimated (due to doppler shift) frequency in the plasma frame near the observed ion cyclotron frequency ($f_{cp} \approx 0.4145$ Hz, $f_{plasma} \approx 0.3125$ Hz).

* Identified regions of perpendicular wave amplification during the intermediate boundary crossings.

* Identified a reconnection interval.

* Identified perpendicular heating for the low energy ion population during intervals of wave activity.

* Identified parallel heating of the intermediate energy ion population during a reconnection interval.
* Identified perpendicular heating of the low energy population with a D-shaped distribution during the reconnection interval.

In our future work we plan to carry out a statistical study of the satisfaction of the onset condition of the KHI at the flank magnetosheath for various upstream solar wind conditions by binning and expressing several thousands of Themis data points in average MSH and MSP reference frame. We also plan to express the magnetospheric plasma sheet ion properties in a statistical manner using an average MSP reference frame. Once we have these data products ready to produce the statistical maps of MSH and MSP properties, we can do correlation studies between the two regions to address in a statistical manner whether KH unstable conditions at the flank MSP boundary correlate with a hotter plasma sheet. Also we can verify with the WHAMP code by running it with the observed Cluster parameters and identify more clearly all of the wave modes that could be present. We also plan to take advantage of the small Cluster satellite separation by calculating the k-vector using the wave telescope method in order to determine the dispersion relation and thus be able to pinpoint the exact wave modes of the observed fluctuations.
Appendix A

IDL Code

A.1 Main Routine

wavemode_all_3.pro Main routine for calculating and plotting wavelet power spectra and integrated power.

```idl
PRO wavemode_all_3, FAST=fast
COMMON va, va1, va3, tintva1, tintva3
COMMON vaint, va1int, va3int
COMMON cis, cis1, cis3
COMMON cisint, cis1int, cis3int
COMMON proton, ncis1, timen1, np1, v1, temp1, tpar1, tperp1, $ ncis3, timen3, np3, v3, temp3, tpar3, tperp3, $ ncis4, timen4, np4, v4, temp4, tpar4, tperp4
COMMON interp, b1int, v1int, tintbv1, $ b3int, v3int, tintbv3, $ b4int, v4int, tintbv4
; KEYWORD (N_ELEMENTS(FAST) EQ 0) THEN BEGIN FAST=0 ENDIF
ebstructures, s_ce, s_cb, datestrn
; create structures which house both the E-field and B-field data
Start: ; cluster_wave_* is adopted from Ryan Soldin's code. Wavelet analysis is used on the both time series to determine the perpendicular power IF (FAST EQ 0) THEN BEGIN cluster_wave_b_all, s_cb, datestrn, intpower, powerb_perp, signalb, fb, ion_cyc, menu, bottomb, topb, levelsb ENDIF IF (FAST EQ 1) THEN BEGIN
cluster_wave_b_all, s_cb, datestrn, intpower, powerb_perp, signalb, fb, ion_cyc, menu, bottomb, topb, levelsb ENDIF cluster_wave_e_2, s_ce, datestrn, menu, powere_perp, signal_e, fe, bottome, tope, levelse

levels=menu(3); Contour plot resolution set in cluster_wave_b_2.pro
power=SQRT(powere_perp/powerb_perp); dB_perp/dB_perp ratio=!power=ALOG10(power)
top=max(!power); sets colorbar range bottom=min(!power)
IF (bottom LT -1e8) THEN BEGIN bottom = 0 ENDIF
DEVICE, DECOMPOSED = 0 LOADCT, 39 WINDOW, 6, XSIZE=1252, YSIZE=600, XPOS=0, YPOS=800
plot_window, win, pos, poscb, win=6 win = round(win)
FOR i=1,win DO BEGIN
array1 = strcompress('pos' + string(i) + ' = ' + str(i) + ')'/remove_all) array2 = strcompress('pos' + string(i) + 'cb' + ' = ' + str(i) + ')'/remove_all) aa = EXECUTE(array1) bb = EXECUTE(array2)
ENDFOR
IP.MULTI = [0, 1, win]; the ' +1' is added to compensate for the legend for plot 3, which is placed by plotting an empty array and suppressing the axis
input1 = 'n'
PLOT: ;pos1 = [0.06, 0.72, 0.91, 0.94]; pos2 = [0.06, 0.50, 0.91, 0.72]; pos3 = [0.06, 0.28, 0.91, 0.50]; pos4 = [0.06, 0.06, 0.91, 0.28]; pos1cb = [0.93, 0.72, 0.96, 0.94]; pos2cb = [0.93, 0.50, 0.96, 0.72]; pos3cb = [0.93, 0.28, 0.96, 0.50]

IF (input1 EQ 'y') THEN BEGIN LOADCT, 39 ENDIF

xrng = [min(signalb0), max(signalb0)]
```

69
APPENDIX A: IDL CODE

PLOT, signalb, signalb, YRANGE=[0.1,10], /YLOG, XRANGE=xrng, POSITION=pos1, /NODATA, $ YTITLE='N [1/cc]', XTITLE='T [10^6 K]', xtickname=replicate(' ',60); YTITLE='Wavelet Power Spectrum', xtickname=replicate(' ',60)

IF (menu(8) EQ 1) THEN BEGIN OPLT, cis1(0,*),cis1(1,*); ENDIF
IF (menu(8) EQ 3) THEN BEGIN OPLT, cis3(0,*),cis3(1,*); ENDIF

: Temperature
xrng = [min(signalb(0,*)),max(signalb(0,*))]

IF (menu(8) EQ 1) THEN BEGIN temp = (tpar1+2.*tperp1)/3.; $ T; PLOT,signalb, signalb, YRANGE=[min(temp1),max(temp1)], XRANGE=xrng, POSITION=pos2,/NODATA, $ YTITLE='T [10^6 K]', xtickname=replicate(' ',60); OPLT, timen1, tperp1, $ T_perp : PLOT, signalb, signalb, YRANGE=[min(tperp1),max(tperp1)], XRANGE=xrng, POSITION=pos2,/NODATA, $ YTITLE='TID19' + string(35B) + 'XIN [10^6 K]', xtickname=replicate(' ',60); OPLT, timen1, tperp1, $ T_par : PLOT, signalb, signalb, YRANGE=[min(tpar1),max(tpar1)], XRANGE=xrng, POSITION=pos2,/NODATA, $ YTITLE='TID19' + string(35B) + 'XIN [10^6 K]', xtickname=replicate(' ',60); OPLT, timen1, tperp1

IF (menu(8) EQ 3) THEN BEGIN temp = (tpar3+2.*tperp3)/3.; $ T; PLOT,signalb, signalb, YRANGE=[min(temp3),max(temp3)], XRANGE=xrng, POSITION=pos2,/NODATA, $ YTITLE='T [10^6 K]', xtickname=replicate(' ',60); OPLT, timen3, tperp3, $ T_perp : PLOT, signalb, signalb, YRANGE=[min(tperp3),max(tperp3)], XRANGE=xrng, POSITION=pos2,/NODATA, $ YTITLE='TID19' + string(35B) + 'XIN [10^6 K]', xtickname=replicate(' ',60); OPLT, timen3, tperp3, $ T_par : PLOT, signalb, signalb, YRANGE=[min(tpar3),max(tpar3)], XRANGE=xrng, POSITION=pos2,/NODATA, $ YTITLE='TID19' + string(35B) + 'XIN [10^6 K]', xtickname=replicate(' ',60); OPLT, timen3, tperp3

IF (menu(8) EQ 4) THEN BEGIN temp = (tpar4+2.*tperp4)/3.; $ T; PLOT,signalb, signalb, YRANGE=[min(temp4),max(temp4)], XRANGE=xrng, POSITION=pos2,/NODATA, $ YTITLE='T [10^6 K]', xtickname=replicate(' ',60); OPLT, timen4, tperp4, $ T_perp : PLOT, signalb, signalb, YRANGE=[min(tperp4),max(tperp4)], XRANGE=xrng, POSITION=pos2,/NODATA, $ YTITLE='TID19' + string(35B) + 'XIN [10^6 K]', xtickname=replicate(' ',60); OPLT, timen4, tperp4

: E-field power spectrum
xrng = [min(signalb(0,*)),max(signalb(0,*))]
yrng = [menu(5),menu(6)]; $ yrng = [menu(5),max(fe)]
userlevels = [DOUBLE(topb-bottomb)/(levelse -1)] + INDGEN(levelse) + bottomb colors = ROUND((254d/(levelse -1)) + INDGEN(levelse))
CONTOUR, ALOG10(power_perp), signalb(0,*),fe, $ XRANGE=xrng, XSTYLE=1, YRANGE=xrng, YSTYLE=1, $ YTITLE='14' + string(100B) + 'XEXID19' + string(128B)+'XIN [10^6 K]', xtickname=replicate(' ',60), $ LEVELS=userLevels, /FILL, /COLORS(colors), POSITION=pos3, $ TITLE='logID10'(10M)/E21/NHz), $ B-field power spectrum
xrng = [min(signalb(0,*)),max(signalb(0,*))]
yrng = [menu(5),menu(6)]; $ yrng = [menu(5),max(fb)]
userlevels = [DOUBLE(topb-bottomb)/(levelse -1)] + INDGEN(levelse) + bottomb colors = ROUND((254d/(levelse -1)) + INDGEN(levelse))
APPENDIX A: IDL CODE

```
CONTOUR, ALOG10(powerb_perps), signalb(0, *), fb, $  
XRANGE=xrng , XSTYLE=1, $  
YRANGE=yrng , YSTYLE=1, YLOG, $  
XTICKS=1.0 , XCHARSIZE=1, XTICKFORMAT= '(F3.2) ' , $  
xtitlename=replicate(' ', 60), $  
YTITLE='Power (log !D10!N (1000 km/s)) ', $  
LEVELS=userLevels , /FILL , C_COLORS=colors , POSITION=pos5 , $  
AXIS, $  
XAXIS=0,YAXIS=0 , /NODATA , /SAVE , $  
OPLOT, tintva3 , va3 , COLOR=255 ENDIF $  

IF (menu(8) EQ 1) THEN BEGIN A X I S , Y A X I S=1 , Y T I C K S=10 , Y R A N G E=[min (va3int), max (va3int)], YSTYLE=9 , COLOR=255 , /NODATA , /SAVE , xtitlename=replicate(' ', 60) $  
ENDIF  

IF (input1 EQ 'n') THEN BEGIN  
AXIS , YAXIS=1 , YTITLE='[F10] , YRANGE=[min (va3int), max (va3int)] , YSTYLE=9 , COLOR=255 , /NODATA , /SAVE , xtitlename=replicate(' ', 60) $  
ENDIF
```

CONTOUR, ALOG10(powerb_perps), signalb(0, *), fb, $  
XRANGE=xrng , XSTYLE=1, $  
YRANGE=yrng , YSTYLE=1, YLOG, $  
XTICKS=1.0 , XCHARSIZE=1, XTICKFORMAT= '(F3.2) ' , $  
xtitlename=replicate(' ', 60), $  
YTITLE='Power (log !D10!N (1000 km/s)) ', $  
LEVELS=userLevels , /FILL , C_COLORS=colors , POSITION=pos5 , $  
AXIS, $  
XAXIS=0,YAXIS=0 , /NODATA , /SAVE , $  
OPLOT, tintva3 , va3 , COLOR=255 ENDIF $  

IF (input1 EQ 'y') THEN BEGIN  
AXIS , YAXIS=1 , YTITLE='[F10] , YRANGE=[min (va3int), max (va3int)] , YSTYLE=9 , COLOR=255 , /NODATA , /SAVE , xtitlename=replicate(' ', 60) $  
ENDIF
```

CONTOUR, ALOG10(powerb_perps), signalb(0, *), fb, $  
XRANGE=xrng , XSTYLE=1, $  
YRANGE=yrng , YSTYLE=1, YLOG, $  
XTICKS=1.0 , XCHARSIZE=1, XTICKFORMAT= '(F3.2) ' , $  
xtitlename=replicate(' ', 60), $  
YTITLE='Power (log !D10!N (1000 km/s)) ', $  
LEVELS=userLevels , /FILL , C_COLORS=colors , POSITION=pos5 , $  
AXIS, $  
XAXIS=0,YAXIS=0 , /NODATA , /SAVE , $  
OPLOT, tintva3 , va3 , COLOR=255 ENDIF $  

IF (input1 EQ 'n') THEN BEGIN  
AXIS , YAXIS=1 , YTITLE='[F10] , YRANGE=[min (va3int), max (va3int)] , YSTYLE=9 , COLOR=255 , /NODATA , /SAVE , xtitlename=replicate(' ', 60) $  
ENDIF
```

CONTOUR, ALOG10(powerb_perps), signalb(0, *), fb, $  
XRANGE=xrng , XSTYLE=1, $  
YRANGE=yrng , YSTYLE=1, YLOG, $  
XTICKS=1.0 , XCHARSIZE=1, XTICKFORMAT= '(F3.2) ' , $  
xtitlename=replicate(' ', 60), $  
YTITLE='Power (log !D10!N (1000 km/s)) ', $  
LEVELS=userLevels , /FILL , C_COLORS=colors , POSITION=pos5 , $  
AXIS, $  
XAXIS=0,YAXIS=0 , /NODATA , /SAVE , $  
OPLOT, tintva3 , va3 , COLOR=255 ENDIF $  

IF (input1 EQ 'y') THEN BEGIN  
AXIS , YAXIS=1 , YTITLE='[F10] , YRANGE=[min (va3int), max (va3int)] , YSTYLE=9 , COLOR=255 , /NODATA , /SAVE , xtitlename=replicate(' ', 60) $  
ENDIF
```

CONTOUR, ALOG10(powerb_perps), signalb(0, *), fb, $  
XRANGE=xrng , XSTYLE=1, $  
YRANGE=yrng , YSTYLE=1, YLOG, $  
XTICKS=1.0 , XCHARSIZE=1, XTICKFORMAT= '(F3.2) ' , $  
xtitlename=replicate(' ', 60), $  
YTITLE='Power (log !D10!N (1000 km/s)) ', $  
LEVELS=userLevels , /FILL , C_COLORS=colors , POSITION=pos5 , $  
AXIS, $  
XAXIS=0,YAXIS=0 , /NODATA , /SAVE , $  
OPLOT, tintva3 , va3 , COLOR=255 ENDIF $  

IF (input1 EQ 'n') THEN BEGIN  
AXIS , YAXIS=1 , YTITLE='[F10] , YRANGE=[min (va3int), max (va3int)] , YSTYLE=9 , COLOR=255 , /NODATA , /SAVE , xtitlename=replicate(' ', 60) $  
ENDIF
```

LOADCT, 39 $  

Integrated Power $  
xrng = [min (signalb(0,*)), max(signalb(0,*)))] $  
tmin = min(signalb) tmax = max(signalb) $  

PLOT, signalb , YRANGE=[0,1] , XRANGE=xrng , POSITION=pos6 , /NODATA , YTITLE='P !D19 + string(120B)+'!XIN'/4 + string(100B)+'!XBIN' $  
XSTYLE=9 $  
XTICKFORMAT= '(F3.2) ' , $  
XTICKNAME='Time [UT]' $  
LEVELS=userLevels , /FILL , C_COLORS=colors , POSITION=pos5 , $  
AXIS, $  
XAXIS=0,YAXIS=0 , /NODATA , /SAVE , $  
OPLOT, signalb , YRANGE=[0,1] , XRANGE=xrng , POSITI
A.2 Read Routines

readclusterallwav.pro Adapted from code courtesy of Katarina Nykyri. Reads and sorts data into useable arrays.

```idl
PRO readclusterallwav, nameefw, namefgm, namecis, datestrn
COMMON hefield, nefw1, timehe1, he1x, he1y, he1z, nefw2, timehe2, he2x, he2y, he2z, nefw3, timehe3, he3x, he3y, he3z, nefw4, timehe4, he4x, he4y, he4z, hbold, hpar, hperp, he1, he2, he3, he4
COMMON hbfield, nfgm1, timehb1, hb1x, hb1y, hb1z, nfgm2, timehb2, hb2x, hb2y, hb2z, nfgm3, timehb3, hb3x, hb3y, hb3z, nfgm4, timehb4, hb4x, hb4y, hb4z, hbold, hpar, hperp, hb1, hb2, hb3, hb4
COMMON scraft, sc1x, sc1y, sc1z, sc2x, sc2y, sc2z, sc3x, sc3y, sc3z, sc4x, sc4y, sc4z, re, scdr1, scdr2, scdr3, scdr4
COMMON proton, ncis1, timen1, np1, v1, temp1, tpar1, tperp1, ncis3, timen3, np3, v3, temp3, tpar3, tperp3, ncis4, timen4, np4, v4, temp4, tpar4, tperp4

; Pick date to use date=0 date=string(date) print, 'What date do you want to use? (ex: 20050206)', date read, date
datestrn = date+'_'+'GSE'
EFW="C*_CP_EFW_L2_E3D_GSE__"+date+'*.CEF' FGM="C*_CP_FGM_FULL__"+date+'*.CEF' CIS="C*_CP_CIS-HIA_ONBOARD_MOMENTS__"+date+'*.CEF'

nameefw=FILE_SEARCH(EFW) ; electric field data namefgm=FILE_SEARCH(FGM) ; magnetic field data namecis=FILE_SEARCH(CIS) ; proton density data

; READ DATA ;; this will assign names to each .cef file based on spacecraft number
namee1=nameefw(0) hreadefw1, namee1, nefw1, timehe1, he1x, he1y, he1z, dEz1, Ebit1, Equal1 namee2=nameefw(1) hreadefw2, namee2, nefw2, timehe2, he2x, he2y, he2z, dEz2, Ebit2, Equal2 namee3=nameefw(2) hreadefw3, namee3, nefw3, timehe3, he3x, he3y, he3z, dEz3, Ebit3, Equal3 namee4=nameefw(3) hreadefw4, namee4, nefw4, timehe4, he4x, he4y, he4z, dEz4, Ebit4, Equal4

; Splits the electric field vector arrays into single column arrays ;; by their i, j, k (x, y, z) component he1x=e1[0,*] he1y=e1[1,*] he1z=e1[2,*]
he2x=e2[0,*] he2y=e2[1,*] he2z=e2[2,*]
he3x=e3[0,*] he3y=e3[1,*] he3z=e3[2,*]
he4x=e4[0,*] he4y=e4[1,*] he4z=e4[2,*]

; READ DATA FGM namef1=namefgm(0) hreadfgm_csf1, namef1, nfgm1, timehb1, scdr1, b1, b1
amef2=namefgm(1) hreadfgm_csf2, namef2, nfgm2, timehb2, scdr2, b2, b2
amef3=namefgm(2) hreadfgm_csf3, namef3, nfgm3, timehb3, scdr3, b3, b3
amef4=namefgm(3) hreadfgm_csf4, namef4, nfgm4, timehb4, scdr4, b4, b4
```

APPENDIX A: IDL CODE
APPENDIX A: IDL CODE

```
;; Splits the magnetic field vector and spacecraft position vector ;; arrays into single column arrays by their i_th (x,y,z) component
sc1x = scdr1(0,*)
scl2 = scdr1(1,*)
scl3 = scdr1(2,*)
scl4 = scdr4(0,*)
scl4y = scdr4(1,*)
scl4z = scdr4(2,*)
hb1x = b1(0,*)
hb1y = b1(1,*)
hb1z = b1(2,*)
hb2x = b2(0,*)
hb2y = b2(1,*)
hb2z = b2(2,*)
hb3x = b3(0,*)
hb3y = b3(1,*)
hb3z = b3(2,*)
hb4x = b4(0,*)
hb4y = b4(1,*)
hb4z = b4(2,*)
hb1 = [b1, DBLARR(1, N_ELEMENTS(hb1x))]
hb2 = [b2, DBLARR(1, N_ELEMENTS(hb2x))]
hb3 = [b3, DBLARR(1, N_ELEMENTS(hb3x))]
hb4 = [b4, DBLARR(1, N_ELEMENTS(hb4x))]
```

```
RETURN END
```

```
hreadefw.pro Adapted from code courtesy of Katariina Nykyri. Reads the cef format high resolution EFW data.

PRO hreadefw , namee, nefw, timee, e, dEz, Ebit, Equal
  idefw = CEF_READ(namee) ; reads .cef file
  ;; VARIABLES ; nefw - dimension ; timee - time, UT (YYYY-MM-DDTHH:MM:SS.sssZ) ; e - electric field data (Ex, Ey, Ez) ; dEz - Error of electric field (ISR2) ; Ebit - Electric field measurement quality bitmask ; Equal - Electric field measurement quality flag (4=best)
  timee = (*idefw(0)).DATA
  nefw = N_ELEMENTS(timee)
  e = (*idefw(1)).DATA
  dEz = (*idefw(2)).DATA
  Ebit = (*idefw(3)).DATA
  Equal = (*idefw(4)).DATA
END
```

```
hreadfgm_cef.pro Adapted from code courtesy of Katariina Nykyri. Reads the cef format high resolution FGM data.

PRO hreadfgm_cef , namef, nfmg, timem, sc, b, bt
  idfgm = CEF_READ(namef) ; reads .cef file
  ;; VARIABLE ; nfmg - dimension ; timem - time ; sc - spacecraft position ; b - magnetic field vector ; bt - |B|
  timem = (*idfgm(0)).DATA
  nfmg = N_ELEMENTS(timem)
  sc = (*idfgm(4)).DATA
  b = (*idfgm(2)).DATA
  bt = (*idfgm(3)).DATA
END
```

```
hreadcis_cef.pro Adapted from code courtesy of Katariina Nykyri. Reads the cef format high resolution CIS data.

PRO hreadcis_cef , namen, ncis, timen, np, v, temp, tperp, tpar
  COMMON proton, ncis1, timen1, np1, v1, temp1, tperpl, tpar1
  $ ncis3, timen3, np3, v3, temp3, tperp3, tpar3
  $ ncis4, timen4, np4, v4, temp4, tperp4, tpar4
  idcis = CEF_READ(namen) ; reads .cef file
  ;; VARIABLE ; ncis - dimension ; timen - time ; np - proton density ; v - velocity (GSE)
  timen = (*idcis(0)).DATA
  ncis = N_ELEMENTS(timen)
  np = (*idcis(4)).DATA
  v = (*idcis(7)).DATA
  temp = (*idcis(8)).DATA
  tperp = (*idcis(5)).DATA
  tpar = (*idcis(9)).DATA
END
```
A.3 Wavelet Routines

cluster_wave_b_all.pro Adapted from Ryan Soldin’s code - calculates perpendicular wavelet power spectrum for the magnetic field using a wavelet technique (see wavelet.pro). This program also calculates the total integrated wave power.

```
PRO cluster_wave_b_all, s_c, datestrn, intpower, powerb_perp, signalb, fb, ion_cyc, menu,
    bottom, top, levels

COMMON proton, ncis1, timen1, np1, v1, temp1, tpar1, tperp1, $
    ncis3, timen3, np3, v3, temp3, tpar3, tperp3, $
    ncis4, timen4, np4, v4, temp4, tpar4,
    tperp4
COMMON vs, va, va3, tintva1, tintva3
COMMON cis, cis1, cis3
COMMON cisint, cis1int, cis3int
;
MENU INPUTS FOR E-FIELD POWER SPECTRUM menu = FLTARR(9,1); satellite#(0), tmin(1),
    tmax(2), #f’s(3), colorbar_resolution(4), fmin(5), fmax(6), detrend(7), np_sat#(8) menu
    (8) = 1

***************************************************************
;*****LOAD DEFAULT VALUES - MAIN MENU***************************
***************************************************************

input = [0,1, $ ;Main Menu, Satellite Number
    0, N_ELEMENTS(s_c.(0)(0,*))-1, $ ;Time index Cluster 1
    0, N_ELEMENTS(s_c.(1)(0,*))-1, $ ;Time index Cluster 2
    0, N_ELEMENTS(s_c.(2)(0,*))-1, $ ;Time index Cluster 3
    0, N_ELEMENTS(s_c.(3)(0,*))-1, $ ;Time index Cluster 4
    MIN(s_c.(0)(0,*)), MAX(s_c.(0)(0,*)), $ ;Time
4,0,1] ;Component, Parts

Detrended In, Alpha input_dscrptn = ['Main Menu','Satellite Number','Min Time
Cluster 1','Max Time Cluster 1', $ 'Min Time Cluster 2','Max Time Cluster 2', 'Min Time Cluster 3','Max Time Cluster 3', $ 'Min Time Cluster 4','Max Time Cluster 4', 'Min Time','Max Time', 'Component', $
'Parts Detrended In','Alpha'] signal = s_c.(0) ;Cluster 1
dtsignal = signal(1:4,*) ;Not detrended color = 'FFFFFF'XL ;White time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(2):input(3)))),2) + 't to ' + STRTRIM(MAX(s_c.(input(1)-1)(0,input(2):input(3)))),2) component = 'Bt' ;Total Magnetic Field

Detrended = 'Not detrended' window = 'Not windowed' sub = 0

***************************************************************
;*****LOAD DEFAULT VALUES - WAVELET SUBMENU********************
***************************************************************

sst = transpose(dtsignal((input(12)-1),*)) winput = [0,1,1,0,0,0.025,265,1,6,0,1,10,20] ; SubMenu(0), normalize(1), pad(2), s0(3), dj(4), J
    (5), mother(6), param(7), fmin(8), fmax(9), #f’s(10) norm = 'y' pad = 'y' mother = 'Morlet' parameter = 'w0 = ' ;SCALE INPUTS FOR WAVELET TRANSFORMS, COMPUTED FROM fmin,fmax,and #f's sto f = ($
    winput(7)*SQR(2*(winput(7)**2)/ (4*1PI)) winput(3) = sto/f winput(9)$
    s0 winput(5) = winput(10) - 1 ; smax = sto/f winput(8) winput(4) = (1/
    winput(5))*(ALOG10(smax/winput(3)))/(ALOG10(2)) ; dj$
    ;FREQUENCY BASED ON DEFAULTS m = winput(5)+1 scale = DINDGEN(m)*winput(4) scale = 240*(scale)*winput(3) f = sto/scale display = 10 ;default from original program is 10

***************************************************************
;*****PLOT ORIGINAL DATA - ALL SATELLITES & DEFAULT SATELLITE***
***************************************************************

PLODATA: CLUSTERPLOT4_b,s_c.(0),s_c.(1),s_c.(2),s_c.(3),datestrn
    ;window 0 CLUSTERPLOT1, signal, color,
    input(1), datestrn
    CLUSTERPLOTDETRENDED, signal, dtsignal, input(12), component, input(1), color,
    detrended, window, datestrn ;window 2

IF (detrend EQ 'y') THEN BEGIN GOTO,option4 ENDIF
GOTO,option1
```

APPENDIX A: IDL CODE

;*************************************************************** ;****1. CHOOSE WHICH SATELLITE TO ANALYZE***********************
;***************************************************************

option1: CLS

PRINT, (''Cluster 1 - White, enter 1'') PRINT, (''Cluster 2 - Red, enter 2'') PRINT,
(''Cluster 3 - Yellow, enter 3'') PRINT, (''Cluster 4 - Blue, enter 4'') PRINT,
(''READ, input2, PROMPT = ('ENTER THE SATELLITE #: ') & input(1) = input2

IF (input(1) EQ 1) THEN BEGIN signal = s_c.(0)(*,input(2):input(3)) color = 'FF0000'XL blue input(1) = 1 input(2) = MIN(ABS(s_c.(0)(0,*)-s_c.(0)(0,input(2))),index) & input(2) = index input(3) = MIN(ABS(s_c.(0)(0,*)-s_c.(0)(0,input(3))),index) & input(3) = index time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(2):input(3))),2)+ ' to ' +STRTRIM(MAX(s_c.(input(1)-1)(0,input(2):input(3))),2) GOTO,option2 ENDIF IF (input(1) EQ 2) THEN BEGIN signal = s_c.(1)(*,input(4):input(5)) color = '0000FF'XL red input(1) = 2 input(4) = MIN(ABS(s_c.(1)(0,*)-s_c.(1)(0,input(4))),index) & input(4) = index input(5) = MIN(ABS(s_c.(1)(0,*)-s_c.(1)(0,input(5))),index) & input(5) = index time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(4):input(5))),2)+ ' to ' +STRTRIM(MAX(s_c.(input(1)-1)(0,input(4):input(5))),2) GOTO,option2 ELSE BEGIN error='' PRINT,('') PRINT,('!!!!!!!!!!!!!!!!!!!!!!!!!!!') PRINT,('!!NOT A VALID SATELLITE #!!') PRINT,('!!!!!!!!!!!!!!!!!!!!!!!!!!!') PRINT,('') READ, error , PROMPT=('Push enter to return to menu.') GOTO,option1 ENDELSE

GOTO,option2

;*************************************************************** ;*****2. DEFINE TIME INTERVAL***********************************
;***************************************************************

PRINT,('') PRINT,(' Change time interval to be examined ') PRINT,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') PRINT,('The full time interval available is '+STRTRIM(MIN(s_c.(input(1) -1)(0,*)),2)+' hrs to '+STRTRIM(MAX(s_c.(input(1) -1)(0,*)),2)+' hrs') PRINT,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') PRINT,('') PRINT,('Enter desired tmin ') READ, input3 ,PROMPT=('tmin = ') & input(10) = input3 If (input(10) EQ 0) THEN BEGIN GOTO,DT ENDIF PRINT,('Enter desired tmax ')' READ, input4 ,PROMPT=('tmax = ') & input(11) = input4 IF (input(11) LT input(10)) THEN BEGIN PRINT,('ERROR') GOTO,option2 ENDIF datestrn = strcompress(datestrn+'_'+string(input3,format='(f4.1)')+'_'+string(input4,format='(f4.1)'))

menu(0) = input2 menu(1) = input3 menu(2) = input4

;Indices to match desired time interval for all 4 satellites input(2) = MIN(ABS(s_c.(0)(0,*)-s_c .(0)(0,input(2))),index) & input(2) = index input(3) = MIN(ABS(s_c.(0)(0,*)-input(11)),index) & input(3) = index input(4) = MIN(ABS(s_c.(1)(0,*)-input(10)),index) & input(4) = index input(5) = MIN(ABS(s_c.(1)(0,*)-input(11)),index) & input(5) = index input(6) = MIN(ABS(s_c.(2)(0,*)-input(10)),index) & input(6) = index input(7) = MIN(ABS(s_c.(2)(0,*)-input(11)),index) & input(7) = index input(8) = MIN(ABS(s_c.(3)(0,*)-input(10)),index) & input(8) = index input(9) = MIN(ABS(s_c.(3)(0,*)-input(11)),index) & input(9) = index cis1_start = MIN(ABS(timen1(0,*)-input(10)),index) & cis1_start = index cis1_end = MIN(ABS(timen1(0,*)-input(11)),index) & cis1_end = index cis3_start = MIN(ABS(timen3(0,*)-input(10)),index) & cis3_start = index cis3_end = MIN(ABS(timen3(0,*)-input(11)),index) & cis3_end = index

GOTO,option1 ENDFIELD

PRINT,('') PRINT,('Change time interval to be examined') PRINT,('') PRINT,('The full time interval available is '+STRTRIM(MIN(s_c.(input(1)-1)(0,*)),2)+' hrs to '+STRTRIM(MAX(s_c.(input(1)-1)(0,*)),2)+' hrs') PRINT,('') PRINT,('Enter desired tmin ') PRINT,('') PRINT,('Enter desired tmax') PRINT,('') PRINT,('NOT A VALID SATELLITE #!!') PRINT,('') PRINT,('Push enter to return to menu.') GOTO,option1
va1_start = MIN(ABS(tintva1(0,*))-input(10)),index) & va1_start = index
va1_end = MIN(ABS(tintva1(0,*))-input(11)),index) & va1_end = index
va3_start = MIN(ABS(tintva3(0,*))-input(10)),index) & va3_start = index
va3_end = MIN(ABS(tintva3(0,*))-input(11)),index) & va3_end = index

IF (input(1) EQ 1) THEN BEGIN
signal = s_c.(0)(*,input(2):input(3))
cis1int = [transpose(timen1(cis1_start:cis1_end)),transpose(np1(cis1_start:cis1_end))]
valint = [transpose(valva1(0,*)],tvalva1(0,*))]
valint = [transpose(valva1(0,*)],tvalva1(0,*))]
cis1 = [timen1,np1] time_str = STRTRIM(MIN(s_c.(0)(input(1)-1)(0,input(2):input(3))),2) + ' to ' + STRTRIM(MAX(s_c.(0)(input(1)-1)(0,input(2):input(3))),2)
ENDIF

IF (input(1) EQ 2) THEN BEGIN
signal = s_c.(1)(*,input(4):input(5))
time_str = STRTRIM(MIN(s_c.(1)(input(1)-1)(0,input(4):input(5))),2) + ' to ' + STRTRIM(MAX(s_c.(1)(input(1)-1)(0,input(4):input(5))),2)
ENDIF

IF (input(1) EQ 3) THEN BEGIN
signal = s_c.(2)(*,input(6):input(7))
cis3int = [transpose(timen3(cis3_start:cis3_end)),transpose(np3(cis3_start:cis3_end))]
va3int = [transpose(va3(va3_start:va3_end))]
cis3 = [timen3,np3] time_str = STRTRIM(MIN(s_c.(2)(input(1)-1)(0,input(6):input(7))),2) + ' to ' + STRTRIM(MAX(s_c.(2)(input(1)-1)(0,input(6):input(7))),2)
ENDIF

IF (input(1) EQ 4) THEN BEGIN
signal = s_c.(3)(*,input(8):input(9))
time_str = STRTRIM(MIN(s_c.(3)(input(1)-1)(0,input(8):input(9))),2) + ' to ' + STRTRIM(MAX(s_c.(3)(input(1)-1)(0,input(8):input(9))),2)
ENDIF

detrend='y'
GOTO,DT

;***************************************************************
;*****4. DETREND THE SIGNAL TO BE ANALYZED**********************
;***************************************************************
PRINT,'(Remove linearity & Apply Hanning Window )'
PRINT,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ ')
PRINT,('1. Detrend data. ('+detrended+')')
PRINT,('2. Apply Hanning/Hamming window. ('+window+')')
PRINT,('3. Continue without applying window or detrending .')
PRINT,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~')
READ,input6,PROMPT=('Enter option #:') sub = 1
IF (input6 EQ 1) THEN BEGIN
PRINT,('')
READ,input7,PROMPT=('Enter # of time intervals for detrending: ') & input(13) = input7
IF (input(13) EQ 0) THEN BEGIN menu(7) = 0 GOTO DT ENDIF IF (input(13) EQ 1) THEN BEGIN GOTO,DT ENDIF breaks = FLTARR(input(13)+1);last part i changed , i think its working again, not 100%
FOR b=1,(input(13) -1) DO BEGIN
PRINT,('')
READ,input18,PROMPT=('Enter the '+STRTRIM(b,2)+' break time: ') breaks(b) = input18
ENDFOR breaks(0)=0 & breaks(input(13)+1)=(N_ELEMENTS(signal(0,*))-1)

IF (input(13) EQ 0) THEN BEGIN
detrended = 'Not detrended ' Label for menu
IF (input(1) EQ 1) THEN BEGIN
dtsignal = s_c.(0)(1:4,input(2):input(3))
ENDIF IF (input(1) EQ 2) THEN BEGIN
dtsignal = s_c.(1)(1:4,input(4):input(5))
ENDIF IF (input(1) EQ 3) THEN BEGIN
dtsignal = s_c.(2)(1:4,input(6):input(7))
ENDIF IF (input(1) EQ 4) THEN BEGIN
dtsignal = s_c.(3)(1:4,input(8):input(9))
ENDIF
han = HANNING(N_ELEMENTS(dtsignal(0,*)),alpha=input(14))
dttemp1 = dtsignal(0,*)*han
dttemp2 = dtsignal(1,*)*han
dttemp3 = dtsignal(2,*)*han
dttemp4 = dtsignal(3,*)*han
dtsignal = [dttemp1,dttemp2,dttemp3,dttemp4]

CLUSTERPLOTDETRENDED ,signal,dtsignal,input(12),component,input(1),color,
detrended,window,datestrn;window 2 ENDIF
APPENDIX A: IDL CODE

77

IF (input(13) EQ 1) THEN BEGIN detrended = 'Detrended in 1 part' DETRENDB2, TRANSPOSE(signal(0,*)),signal(1:4,*),0,(N_ELEMENTS(signal(0,*))-1),dtsignal
han = HANNING(N_ELEMENTS(dtsignal(0,*)),alpha=input(14)) dttemp1 = dtsignal(0,*)*han dttemp2 = dtsignal(1,*)*han dttemp3 = dtsignal(2,*)*han dttemp4 = dtsignal(3,*)*han dtsignal = [dttemp1,dttemp2,dttemp3,dttemp4]
CLUSTERPLOTDETRENDED,signal,dtsignal,input(12),component,input(1),color,detrended
window,datenstrn;window 2 ENDIF

IF (input(13) GT 1) THEN BEGIN detrended = 'Detrended in '+strtrim(fix(input(13)),2)+' parts'
FOR i=1,(N_ELEMENTS(breaks)-1) DO BEGIN DETRENDB2, TRANSPOSE(signal(0,breaks(i-1):breaks(i))),signal(1:4,*),breaks(i-1),breaks(i),out
endfor han = HANNING(N_ELEMENTS(dtsignal(0,*)),alpha=input(14)) dttemp1 = dtsignal(0,*)*han dttemp2 = dtsignal(1,*)*han dttemp3 = dtsignal(2,*)*han dttemp4 = dtsignal(3,*)*han dtsignal = [dttemp1,dttemp2,dttemp3,dttemp4]
CLUSTERPLOTDETRENDED,signal,dtsignal,input(12),component,input(1),color,detrended,window,datenstrn;window 2 ENDIF

IF (sub EQ 1) THEN GOTO,option4 ENDIF
IF (sub EQ 0) THEN GOTO,option4 ENDIF

;*************************************************************** ;*****3. CHOOSE MAGNETIC FIELD COMPONENT TO BE ANALYZED********* ;*************************************************************** option3: CLS ; PRINT,(' Choose the signal to be analyzed ') ;PRINT,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') ;PRINT,('Bx - Enter 1') ;PRINT,('By - Enter 2') ;PRINT,('Bz - Enter 3') ;PRINT,('Bt - Enter 4') ;PRINT,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') ;PRINT,('') ;READ, input5 , PROMPT = ('Choose component: ') & input(12) = input5
;ENDIF

;*************************************************************** ;*****5. COMPUTE AND DISPLAY WAVELET POWER SPECTRUM************* ;*************************************************************** option5: sst = transpose(dtsignal((input(12) -1),*))
n = N_ELEMENTS(signal(0,*)) sampling_rate = FLTARR(n-1) FOR k=11,n-1 DO BEGIN
dt = TOTAL(sampling_rate)/N_ELEMENTS(sampling_rate)*3600
GOTO,optionw6
;SUB MENU FOR CHANGING DEFAULT WAVELET PARAMETERS startwave: CLS PRINT,('') PRINT,('Wavelet Transform') PRINT,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') PRINT,('1. Normalize to standard deviation. (**norm**)') PRINT,('2. Pad signal with zeros. (**pad,2**)') PRINT,('3. Define frequency range. (**strtrim(string(MAX(f),FORMAT='(F8.1)'),2)** to **strtrim(string(MAX(f),FORMAT='(F8.4)'),2)**)') PRINT,('4. Define mother wavelet. (**mother**, **parameter**, **strtrim(string(input(7),FORMAT='(f5.2)'),2)**)') PRINT,('5. Change display properties. (**label** **strtrim(string(MAX(f),FORMAT='(F8.1)'),2)**)') PRINT,('6. Compute transform. (**label** **strtrim(string(MAX(f),FORMAT='(F8.4)'),2)**)') PRINT,('7. Return to main menu. (**label** **strtrim(string(MAX(f),FORMAT='(F8.1)'),2)**)') READ, input10, PROMPT = ('Enter # to change desired parameter: ') & winput(0) = input10

IF (winput(0) EQ 1) THEN BEGIN PRINT,('') PRINT,('Normalize signal to standard deviation') PRINT,('') READ, norm, PROMPT = ('Enter (y/n): ') ;Something looks funny here?????!? i think its fixed IF (norm EQ 'y') THEN BEGIN
isse = 1 ENDIF
IF (norm EQ 'n') THEN BEGIN
isse = 1 ENDIF

IF (winput(0) EQ 2) THEN BEGIN PRINT,('') PRINT,('Pad signal with zeros') PRINT,('') READ, pad, PROMPT = ('Enter (y/n): ') ;Something looks funny here?????!? i think its fixed IF (pad EQ 'y') THEN BEGIN
isse = 1 ENDIF
IF (pad EQ 'n') THEN BEGIN
isse = 1 ENDIF

IF (winput(0) EQ 3) THEN BEGIN PRINT,('') PRINT,('Define frequency range') PRINT,('') READ, finput(5), PROMPT = ('Enter (y/n): ') ;Something looks funny here?????!? i think its fixed IF (finput(5) EQ 'y') THEN BEGIN
isse = 1 ENDIF
IF (finput(5) EQ 'n') THEN BEGIN
isse = 1 ENDIF

IF (winput(0) EQ 4) THEN BEGIN PRINT,('') PRINT,('Define mother wavelet') PRINT,('') READ, winput, PROMPT = ('Enter (y/n): ') ;Something looks funny here?????!? i think its fixed IF (winput EQ 'y') THEN BEGIN
isse = 1 ENDIF
IF (winput EQ 'n') THEN BEGIN
isse = 1 ENDIF

IF (winput(0) EQ 5) THEN BEGIN PRINT,('') PRINT,('Define display properties') PRINT,('') READ, label, PROMPT = ('Enter (y/n): ') ;Something looks funny here?????!? i think its fixed IF (label EQ 'y') THEN BEGIN
isse = 1 ENDIF
IF (label EQ 'n') THEN BEGIN
isse = 1 ENDIF

IF (winput(0) EQ 6) THEN BEGIN PRINT,('') PRINT,('Compute transform') PRINT,('') READ, label, PROMPT = ('Enter (y/n): ') ;Something looks funny here?????!? i think its fixed IF (label EQ 'y') THEN BEGIN
isse = 1 ENDIF
IF (label EQ 'n') THEN BEGIN
isse = 1 ENDIF

IF (winput(0) EQ 7) THEN BEGIN PRINT,('') PRINT,('Return to main menu') PRINT,('') READ, norm, PROMPT = ('Enter (y/n): ') ;Something looks funny here?????!? i think its fixed IF (norm EQ 'y') THEN BEGIN
isse = 1 ENDIF
IF (norm EQ 'n') THEN BEGIN
isse = 1 ENDIF
APPENDIX A: IDL CODE

78

IF (pad EQ 'n') THEN BEGIN
  winput(1) = 0 i think this was wrong
  winput(2) = 0 ENDIF
GOTO,startwave ENDIF

IF (winput(0) EQ 3) THEN BEGIN GOTO,optionw3 ENDIF
IF (winput(0) EQ 4) THEN BEGIN GOTO,optionw4 ENDIF
IF (winput(0) EQ 5) THEN BEGIN GOTO,optionw5 ENDIF
IF (winput(0) EQ 6) THEN BEGIN GOTO,optionw6 ENDIF
IF (winput(0) EQ 7) THEN BEGIN GOTO,optionEND ENDIF

;*********************************************************************** ;****W3. DEFINE FREQUENCY**************************************************
;***********************************************************************
PRINT,('Define Frequency')
PRINT,('The frequency range is currently set to '+STRTRIM(MIN(f),2)+' Hz to '+STRTRIM(MAX(f),2)+' Hz. ') PRINT,('There are currently '+STRTRIM(ROUND(winput(5)+1),2)+' resolvable frequencies. ')
PRINT,('Note: Frequency will be plotted as powers of 10. ') PRINT,('Frequency steps are computed as fractional powers of 2. ')
PRINT,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~')
PRINT,( ' ') PRINT,( 'Enter desired fmin ') READ,input11,PROMPT=('fmin = ') & winput(8) = input11
PRINT,( ' ') PRINT,( 'Enter desired fmax ') READ,input12,PROMPT=('fmax = ') & winput(9) = input12
PRINT,( ' ') PRINT,( 'Enter number of frequencies to resolve:') READ,input13,PROMPT=(' # f 's = ') & winput(5) = input13

win13 = input13 menu(3) = input13 menu(5) = input11 menu(6) = input12 f_recal: ;MORLET IF (winput(6) EQ 1) THEN BEGIN stof = (winp(7)+SQRT(2+(winp(7))^2))/(4*PI)
  winput(3) = stof/winput(9);s0 smax = stof/winput(8) winput(4) = ALOG10(smax/winput(3))/(winput(5)*ALOG10(2))
  n = winp(5)+1 scale = DINDGEN(n)*winput(4) scale = 2d0^(scale)
  winput(3) f = stof/scale ENDIF
;

PAUL IF (winput(6) EQ 2) THEN BEGIN stof = (2*winp(7)+1)/(4*PI)
  winput(3) = stof/winput(9);s0 smax = stof/winput(8) smax winp(4) = ALOG10(smax/winput(3))/(winp(5)*ALOG10(2))
  n = winp(5)+1 scale = DINDGEN(n)*winput(4) scale = 2d0^(-scale) winp(3) f = stof/scale ENDIF
;

DOG IF (winput(6) EQ 3) THEN BEGIN stof = SQRT((winp(7)+1)/2) / (2*PI)
  winput(3) = stof/winput(9);s0 smax = stof/winput(8) smax winp(4) = ALOG10(smax/winput(3))/(winp(5)*ALOG10(2))
  n = winp(5)+1 scale = DINDGEN(n)*winput(4) scale = 2d0^(-scale) winp(3) f = stof/scale ENDIF

GOTO, optionw5

;*********************************************************************** ;****W4. DEFINE MOTHER WAVELET******************************************
;***********************************************************************
PRINT,('Define Mother Wavelet')
PRINT,('1. Mother Wavelet: ( +mother+ ) PRINT,('2. Wavelet Parameter: (+ parameter+)+strtrim(string(winp(7),format=('(f5.2) '),2))+') PRINT,('3. Return to wavelet menu. ') PRINT,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~')
PRINT,( ' ') READ,input14,PROMPT=('Enter option #: ') IF (input EQ 1) THEN BEGIN PRINT,(' ') PRINT,( 'Choose Mother Wavelet: ') & winput(6) = input14
IF (winput(6) EQ 1) THEN BEGIN mother = 'Morlet' winp(7) = 6 ENDIF
IF (winput(6) EQ 2) THEN BEGIN mother = 'Paul' winp(7) = 4 ENDIF
IF (winput(6) EQ 3) THEN BEGIN mother = 'DOG' winp(7) = 6 ENDIF
GOTO,optionw4 ENDIF
APPENDIX A: IDL CODE

IF (minput EQ 2) THEN BEGIN PRINT,( ' ') PRINT,( ' ') READ,input19,PROMPT=('Enter Wavelet Parameter : ') & winput(7) = input19 GOTO, optionw4 ENDIF
IF (minput EQ 3) THEN BEGIN GOTO, f_recalc ENDIF

BODY: DISPLAY PROPERTIES

optionw5 : ; IF (display EQ 0) THEN BEGIN bottom = 'min' top = 'max power calculated ' ; ENDIF
CLS PRINT, ( ' Wavelet Display Properties ' ) PRINT,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') PRINT,( 'The number of contour levels is set to ' + STRTRIM(levels ,2)+'.') PRINT,( 'The range of power displayed is set to '+ STRTRIM(bottom ,2)+' and '+ STRTRIM(top ,2)+'.') PRINT,( 'Note: If these are changed after the powerspectrum is created , you ' ) PRINT,( 'need to recompute the transform. ' ) PRINT,
('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') PRINT,

WAVELET( sst , dt , S0=winput (3) ,DJ =winput (4) ,J=winput (5) ,MOTHER=mother , $
PARAM=winput (7) , PAD=winput (2) , PERIOD=period ) ; stop power = ( ABS( wave) )^2 ; log_power = ALOG10(power)

powerbx=power GOTO, option3b ENDIF IF (INPUT(12) EQ 1) THEN BEGIN powerby=power GOTO, option3c ENDIF IF (INPUT( 1 2 ) EQ 3) THEN BEGIN powerbz=power GOTO, option3d ENDIF powerbt=power signalb=signal (0 ,
fb=f powerb_perp = ABS((powerbx + powerby + powerbz) ) log_power = ALOG10( powerb_perp ) ;
intpoweri_perp = 0 intpoweri_par = 0 intpoweri_tot = 0 X T I T L E = 'Time [UT]' ,YTITLE='P_perp/P_tot ' ,TITLE='Integrated Power ' ; device , /CLOSE

plot , signal(0,*),intpower(0,*),XRANGE=[13.3,13.8] ,XTITLE='Time [UT]' ,YTITLE='P_perp/ P_tot', _TITLE='Integrated Power ' ;
SET_PLOT, 'PS' ; DEVICE, FILENAME='int_power_ratio.ps',/COLOR, BITS=8 ;
DEVICE, /landscape , /times , /bold ; DEVICE, /inches , xsize=10. , ysize =6. ; scale_factor=0.95 ; plot , signal(0,*),intpower(0,*),XRANGE=[13.3,13.8] ,XTITLE =''Time [UT]' ,YTITLE='P_perp/P_tot', _TITLE='Integrated Power ' ; device , /CLOSE

W6. COMPUTE AND DISPLAY WAVELET POWER SPECTRUM

WAVELET( sst , dt , S0=winput (3) ,DJ =winput (4) ,J=winput (5) ,MOTHER=mother , $
PARAM=winput (7) , PAD=winput (2) , PERIOD=period ) ; stop power = ( ABS( wave) )^2 ; log_power = ALOG10(power)

optionw6 : ; IF (INPUT(12) EQ 1) THEN BEGIN powerbx=power GOTO, option3b ENDIF IF (INPUT(12) EQ 2) THEN BEGIN powerby=power GOTO, option3c ENDIF IF (INPUT(12) EQ 3) THEN BEGIN powerbz=power GOTO, option3d ENDIF

intpoweri_perp = 0 intpoweri_par = 0 intpoweri_tot = 0 X T I T L E = 'Time [UT]' ,YTITLE='P_perp/P_tot ' ,TITLE='Integrated Power ' ; device , /CLOSE

WAVELET( sst , dt , S0=winput (3) ,DJ =winput (4) ,J=winput (5) ,MOTHER=mother , $
PARAM=winput (7) , PAD=winput (2) , PERIOD=period ) ; stop power = ( ABS( wave) )^2 ; log_power = ALOG10(power)

optionw6 : ; IF (INPUT(12) EQ 1) THEN BEGIN powerbx=power GOTO, option3b ENDIF IF (INPUT(12) EQ 2) THEN BEGIN powerby=power GOTO, option3c ENDIF IF (INPUT(12) EQ 3) THEN BEGIN powerbz=power GOTO, option3d ENDIF

powerbt=power signalb=signal (0 ,
fb=f powerb_perp = ABS((powerbx + powerby + powerbz) ) log_power = ALOG10( powerb_perp ) ;
intpoweri_perp = 0 intpoweri_par = 0 intpoweri_tot = 0 X T I T L E = 'Time [UT]' ,YTITLE='P_perp/P_tot ' ,TITLE='Integrated Power ' ; device , /CLOSE

WAVELET( sst , dt , S0=winput (3) ,DJ =winput (4) ,J=winput (5) ,MOTHER=mother , $
PARAM=winput (7) , PAD=winput (2) , PERIOD=period ) ; stop power = ( ABS( wave) )^2 ; log_power = ALOG10(power)

optionw6 : ; IF (INPUT(12) EQ 1) THEN BEGIN powerbx=power GOTO, option3b ENDIF IF (INPUT(12) EQ 2) THEN BEGIN powerby=power GOTO, option3c ENDIF IF (INPUT(12) EQ 3) THEN BEGIN powerbz=power GOTO, option3d ENDIF

powerbt=power signalb=signal (0 ,
fb=f powerb_perp = ABS((powerbx + powerby + powerbz) ) log_power = ALOG10( powerb_perp ) ;
intpoweri_perp = 0 intpoweri_par = 0 intpoweri_tot = 0 X T I T L E = 'Time [UT]' ,YTITLE='P_perp/P_tot ' ,TITLE='Integrated Power ' ; device , /CLOSE

WAVELET( sst , dt , S0=winput (3) ,DJ =winput (4) ,J=winput (5) ,MOTHER=mother , $
PARAM=winput (7) , PAD=winput (2) , PERIOD=period ) ; stop power = ( ABS( wave) )^2 ; log_power = ALOG10(power)

optionw6 : ; IF (INPUT(12) EQ 1) THEN BEGIN powerbx=power GOTO, option3b ENDIF IF (INPUT(12) EQ 2) THEN BEGIN powerby=power GOTO, option3c ENDIF IF (INPUT(12) EQ 3) THEN BEGIN powerbz=power GOTO, option3d ENDIF

powerbt=power signalb=signal (0 ,
fb=f powerb_perp = ABS((powerbx + powerby + powerbz) ) log_power = ALOG10( powerb_perp ) ;
intpoweri_perp = 0 intpoweri_par = 0 intpoweri_tot = 0 X T I T L E = 'Time [UT]' ,YTITLE='P_perp/P_tot ' ,TITLE='Integrated Power ' ; device , /CLOSE

WAVELET( sst , dt , S0=winput (3) ,DJ =winput (4) ,J=winput (5) ,MOTHER=mother , $
PARAM=winput (7) , PAD=winput (2) , PERIOD=period ) ; stop power = ( ABS( wave) )^2 ; log_power = ALOG10(power)

optionw6 : ; IF (INPUT(12) EQ 1) THEN BEGIN powerbx=power GOTO, option3b ENDIF IF (INPUT(12) EQ 2) THEN BEGIN powerby=power GOTO, option3c ENDIF IF (INPUT(12) EQ 3) THEN BEGIN powerbz=power GOTO, option3d ENDIF

powerbt=power signalb=signal (0 ,
fb=f powerb_perp = ABS((powerbx + powerby + powerbz) ) log_power = ALOG10( powerb_perp ) ;
intpoweri_perp = 0 intpoweri_par = 0 intpoweri_tot = 0 X T I T L E = 'Time [UT]' ,YTITLE='P_perp/P_tot ' ,TITLE='Integrated Power ' ; device , /CLOSE

WAVELET( sst , dt , S0=winput (3) ,DJ =winput (4) ,J=winput (5) ,MOTHER=mother , $
PARAM=winput (7) , PAD=winput (2) , PERIOD=period ) ; stop power = ( ABS( wave) )^2 ; log_power = ALOG10(power)

optionw6 : ; IF (INPUT(12) EQ 1) THEN BEGIN powerbx=power GOTO, option3b ENDIF IF (INPUT(12) EQ 2) THEN BEGIN powerby=power GOTO, option3c ENDIF IF (INPUT(12) EQ 3) THEN BEGIN powerbz=power GOTO, option3d ENDIF

powerbt=power signalb=signal (0 ,
fb=f powerb_perp = ABS((powerbx + powerby + powerbz) ) log_power = ALOG10( powerb_perp ) ;
intpoweri_perp = 0 intpoweri_par = 0 intpoweri_tot = 0 X T I T L E = 'Time [UT]' ,YTITLE='P_perp/P_tot ' ,TITLE='Integrated Power ' ; device , /CLOSE

WAVELET( sst , dt , S0=winput (3) ,DJ =winput (4) ,J=winput (5) ,MOTHER=mother , $
PARAM=winput (7) , PAD=winput (2) , PERIOD=period ) ; stop power = ( ABS( wave) )^2 ; log_power = ALOG10(power)
APPENDIX A: IDL CODE

; SET_PLOT, 'PS'; DEVICE, FILENAME='conversion_ratio.ps', /COLOR, BITTS=8 ; DEVICE, /landscape, /times, /bold ; DEVICE, /inches, xsize=10., ysize =6., scale_factor=0.95 ;LOADCT, 39 ;pos1 = [0.05,0.05,0.88,0.95] ; pos2 = [0.90,0.05,0.95,0.95] ; xrng = [min(signal(0,*)),max(signal(0,*))] ; yrng = [0, max(f)]; IF (display EQ 0) THEN BEGIN ; top = 1;MAX(logp); bottom = 0; ENDIF ; stop ; userlevels = (DOUBLE(top-bottom)/(levels-1))*INDGEN(levels)-bottom ; colors = ROUND((254d/(levels-1))*INDGEN(levels)); ; CONTOUR, logp, signal(0,*), f, $ ; X R A N G E = x r n g, X S T Y L E = 1, Y R A N G E = y r n g 1, X T I T L E = 'U T [Hrs]', $ ; YTITLE='Frequency [Hz]', TITLE='Wavelet Power Spectrum (P!Dperp!N/P!Dtot!N)' ; LEVELS=userLevels, /FILL, /COLORS=colors, /POSITION=pos1 ; OPLOT, signal(0,*), $ ; COLORBAR, /VERTICAL, /RIGHT, DIVISION=2, RANGE=[bottom,top]; POSITION=pos2, /TITLE='P!Dperp!N/P!Dtot!N'; device, /CLOSE ; SET_PLOT, 'x' ;stop GOTO, option END ;*****************************************************; POLARIZATION;*****************************************************

POLARIZATION

(cluster_wave_e_2.pro Adapted from Ryan Soldin's code - calculates perpendicular wavelet power spectrum for the electric field using a wavelet technique (wavelet.pro).)

PRO cluster_wave_e_2 , s_c,datestrn ,menu,powere_perp ,signale ,fe,bottom ,top,levels ;*************************************************************** ;*****LOAD DEFAULT VALUES - MAIN MENU*************************** ;*************************************************************** input = [0,1, $ ; Main Menu ,Satellite Number 0,N_ELEMENTS(s_c.(0)(0,*))-1, $ ; Time index Cluster 1 0,N_ELEMENTS(s_c.(1)(0,*))-1, $ ; Time index Cluster 2 0,N_ELEMENTS(s_c.(2)(0,*))-1, $ ; Time index Cluster 3 0,N_ELEMENTS(s_c.(3)(0,*))-1, $ ; Time index Cluster 4 MIN(s_c.(0)(0,*)),MAX(s_c.(0)(0,*)), $ ; Time 4,0,1] ; Component, Parts detrended In, Alpha input_dscrptn = ['Main Menu','Satellite Number','Min Time Cluster 1','Max Time Cluster 1','Min Time Cluster 2','Max Time Cluster 2','Min Time Cluster 3','Max Time Cluster 3','Min Time Cluster 4','Max Time Cluster 4','Min Time','Max Time','Component', $ 'Parts Detrended In','Alpha'] signal = s_c.(0) ;Cluster 1 dtsignal = signal(1:4,*) ;Not detrended color = 'F6F6F6' ;White time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(2):input(3)))) + ' to ' + STRTRIM(MAX(s_c.(input(1)-1)(0,input(2):input(3)))) component = 'Et' ;Total Electric Field Strength detrended = 'Not detrended ' window = 'Not windowed ' sub = 0 ;*************************************************************** ;*****LOAD DEFAULT VALUES - WAVELET SUBMENU******************** ;*************************************************************** st = transpose(dtsignal((input(12)-1),*)) winput = [0,1,1,0,0,0.528,265,i,6,0,0.01,10,20] ; SubMenu(0), normalize(1), pad(2),s0(3),dj(4),J (5), mother(6), param(7), fmin(8), fmax(9), f's(10) norm = 'y' pad = 'y', mother = 'Morlet' parameter = 'w0 = ' ; SCALE INPUTS FOR WAVELET TRANSFORMS, COMPUTED FROM fmin,fmax, and f's stof = ( wininput(7)+SQRT(2*(wininput(7)-1)) ) / (4*pi1) wininput(3) = stof/wininput(9) ;s0 wininput(5) = wininput(10) - 1 ;J smax = stof/wininput(8) wininput(4) = (1/;J smax = stof/wininput(8) wininput(4) = (1/ J smax) *(ALOG10(smam/wininput(3)))/(ALOG10(2)); dj ;FREQUENCY BASED ON DEFAULTS n = wininput(5)+1 scale = DINDGEN(n)*wininput(4) scale = 2d0*(scale)*wininput(3) f = stof/scale display = 0 levels = 10 ;default from original program is 10 ;*************************************************************** DATA - ALL SATELLITES & DEFAULT SATELLITE*** ;*************************************************************** detrend = 'n'
APPENDIX A: IDL CODE

PLOTDATA: CLUSTERPLOT4_e, s_c.(0), s_c.(1), s_c.(2), s_c.(3), datestrn
; window 0 CLUSTERPLOT1_e, signal, color
; window 1 CLUSTERPLOT_DETRENDED_e, signal, dt_signal, input(12), component, input(1), color
; window 2 detrended, window, datestrn; window 3

IF (detrend EQ 'y') THEN BEGIN GOTO, DT ENDIF
GOTO, option1

;***************************************************************
;****1. CHOOSE WHICH SATELLITE TO ANALYZE**************************
;***************************************************************

option1: CLS
input2 = '' PRINT ,('') PRINT ,(' Choose which satellite to use') PRINT ,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') PRINT ,('Cluster 1 - White, enter 1') PRINT ,('Cluster 2 - Red, enter 2') PRINT ,('Cluster 3 - Yellow, enter 3') PRINT ,('Cluster 4 - Blue, enter 4') PRINT ,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') PRINT ,('')
READ , input2 , PROMPT = ('ENTER THE SATELLITE #:') & input(1) = input2
input(1) = menu(0)
IF (input(1) EQ 1) THEN BEGIN signal = s_c.(0)(*,input(2):input(3)) color = 'FFFFFF' XL; white input(1) = 1 input(2) = MIN(ABS(s_c.(0)(0,*)-s_c.(0)(0,input(2))),index) & input(2) = index input(3) = MIN(ABS(s_c.(0)(0,*)-s_c.(0)(0,input(3))),index) & input(3) = index time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(2):input(3))),2)+ ' to ' + STRTRIM(MAX(s_c.(input(1)-1)(0,input(2):input(3))),2) GOTO, option2 ENDIF
IF (input(1) EQ 2) THEN BEGIN signal = s_c.(1)(*,input(4):input(5)) color = '0000FF' XL; red input(1) = 2 input(4) = MIN(ABS(s_c.(1)(0,*)-s_c.(1)(0,input(4))),index) & input(4) = index input(5) = MIN(ABS(s_c.(1)(0,*)-s_c.(1)(0,input(5))),index) & input(5) = index time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(4):input(5))),2)+ ' to ' + STRTRIM(MAX(s_c.(input(1)-1)(0,input(4):input(5))),2) GOTO, option2 ENDIF
IF (input(1) EQ 3) THEN BEGIN signal = s_c.(2)(*,input(6):input(7)) color = '00FF00' XL; yellow input(1) = 3 input(6) = MIN(ABS(s_c.(2)(0,*)-s_c.(2)(0,input(6))),index) & input(6) = index input(7) = MIN(ABS(s_c.(2)(0,*)-s_c.(2)(0,input(7))),index) & input(7) = index time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(6):input(7))),2)+ ' to ' + STRTRIM(MAX(s_c.(input(1)-1)(0,input(6):input(7))),2) GOTO, option2 ENDIF
IF (input(1) EQ 4) THEN BEGIN signal = s_c.(3)(*,input(8):input(9)) color = 'FFFF00' XL; blue input(1) = 4 input(8) = MIN(ABS(s_c.(3)(0,*)-s_c.(3)(0,input(8))),index) & input(8) = index input(9) = MIN(ABS(s_c.(3)(0,*)-s_c.(3)(0,input(9))),index) & input(9) = index time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(8):input(9))),2)+ ' to ' + STRTRIM(MAX(s_c.(input(1)-1)(0,input(8):input(9))),2) GOTO, option2 ENDIF
ELSE BEGIN error = '' PRINT ,('') PRINT ,('!!!!!!!!!!!!!!!!!!!!!!!!!!!') PRINT ,('!!NOT A VALID SATELLITE#!') PRINT ,('!!!!!!!!!!!!!!!!!!!!!!!!!!!') PRINT ,('') READ, error , PROMPT=('Push enter to return to menu.') GOTO, option1 ENDIF
GOTO, option2

;***************************************************************
;*****2. DEFINE TIME INTERVAL***********************************
;***************************************************************

option2: CLS
PRINT ,('') PRINT ,(' Change time interval to be examined ') PRINT ,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') PRINT ,('The full time interval available is '+ STRTRIM(MIN(s_c.(input(1)-1)(0,*)),2)+' hrs to '+ STRTRIM(MAX(s_c.(input(1)-1)(0,*)),2)+' hrs') PRINT ,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') PRINT ,('') PRINT ,('Enter desired tmin ') ; READ ,input3 ,PROMPT=('tmin = ') & input(10) = input3
input(10) = menu(1)
IF (input(10) EQ 0) THEN BEGIN GOTO, DT ENDIF
PRINT ,('Enter desired tmax') PRINT , menu(2) input(11) = menu(2)
READ , input4 , PROMPT=('tmax = ') & input(11) = input4
IF (input(11) LT input(10)) THEN BEGIN PRINT ,('ERROR') GOTO, option2 ENDIF
indices to match desired time interval for all 4 satellites input(2) = MIN(ABS(s_c.(0)(0,*)-input(10)),index) & input(2) = index input(3) = MIN(ABS(s_c.(0)(0,*)-input(11)),index) & input(3) = index input(4) = MIN(ABS(s_c.(1)(0,*)-input(10)),index) & input(4) = index input(5) = MIN(ABS(s_c.(1)(0,*)-input(11)),index) & input(5) = index input(6) = MIN(ABS(s_c.(2)(0,*)-input(10)),index) & input(6) = index input(7) = MIN(ABS(s_c.(2)(0,*)-input(11)),index) & input(7) = index input(8) = MIN(ABS(s_c.(3)(0,*)-input(10)),index) & input(8) = index input(9) = MIN(ABS(s_c.(3)(0,*)-input(11)),index) & input(9) = index
APPENDIX A: IDL CODE

IF (input(1) EQ 1) THEN BEGIN
  signal = s_c.(0)(*,input(2):input(3))
  time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(2):input(3))),2) + ' to ' + STRTRIM(MAX(s_c.(input(1)-1)(0,input(2):input(3))),2)
ENDIF

IF (input(1) EQ 2) THEN BEGIN
  signal = s_c.(1)(*,input(4):input(5))
  time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(4):input(5))),2) + ' to ' + STRTRIM(MAX(s_c.(input(1)-1)(0,input(4):input(5))),2)
ENDIF

IF (input(1) EQ 3) THEN BEGIN
  signal = s_c.(2)(*,input(6):input(7))
  time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(6):input(7))),2) + ' to ' + STRTRIM(MAX(s_c.(input(1)-1)(0,input(6):input(7))),2)
ENDIF

IF (input(1) EQ 4) THEN BEGIN
  signal = s_c.(3)(*,input(8):input(9))
  time_str = STRTRIM(MIN(s_c.(input(1)-1)(0,input(8):input(9))),2) + ' to ' + STRTRIM(MAX(s_c.(input(1)-1)(0,input(8):input(9))),2)
ENDIF

detrend='y'
GOTO,PLOTDATA

DT: IF (menu(7) NE 0) THEN BEGIN
  READ,input7 ,PROMPT=('Enter # of time intervals for detrending: ') & input(13) = input7
ENDIF
IF (input(13) EQ 0) THEN BEGIN
  GOTO,DTa
ENDIF
IF (input(13) EQ 1) THEN BEGIN
  FOR b=1,(input(13) -1) DO BEGIN
    PRINT ,('')
    READ,input18 ,PROMPT=('Enter the '+STRTRIM(b,2)+' break time: ')
    breaks(b) = input18
  ENDFOR
  FOR c=1,N_ELEMENTS(breaks) DO BEGIN
    break_index = MIN(ABS(signal(0,*)-breaks(c-1)),index)
    breaks(c-1) = index
  ENDFOR
  breaks(0)=0 & breaks(input(13)+1)= (N_ELEMENTS(signal(0,*))-1)
ENDIF

DTa: IF (input(13) EQ 0) THEN BEGIN
  detrended = 'Not detrended'
ENDIF
IF (input(13) EQ 1) THEN BEGIN
  detrended = 'Detrended in 1 part'
ENDIF
IF (input(13) GT 1) THEN BEGIN
  detrended = 'Detrended in '+strtrim(fix(input(13)),2)+' parts'
ENDIF

GOTO,optionw3

option3: CLS ;
PRINT ,(' Choose the signal to be analyzed ') ;PRINT ,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') ;PRINT ,('Bx - Enter 1') ;PRINT ,('By - Enter 2') ;PRINT ,('Bz - Enter 3') ;PRINT ,('Bt - Enter 4') ;PRINT ,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~') ;PRINT ;READ , input5 , PROMPT = ('Choose component: ') & input(12) = input5
IF (input(12) EQ 1) THEN BEGIN
  input(12) = 1 component = 'Ex' GOTO, option5 ;ENDIF
IF (input(12) EQ 2) THEN BEGIN
  input(12) = 2 component = 'Ey' GOTO, option5 ;ENDIF
IF (input(12) EQ 3) THEN BEGIN
  input(12) = 3 component = 'Ez' GOTO, option5 ;ENDIF
IF (input(12) EQ 4) THEN BEGIN
  input(12) = 4 component = 'Et' GOTO, option5 ;ENDIF
APPENDIX A: IDL CODE

83

```idl
; IF ((input(12) NE 1) AND (input(12) NE 2) AND (input(12) NE 3) AND (input(12) NE 4)) 
$ ; THEN BEGIN ; PRINT(('')); PRINT(('! NOT A VALID INPUT FOR CHOICE OF COMPONENT!!')); PRINT(('')); PRINT(('!!!!!')); ENDIF ; READ, error 
PRMT = ('Enter 0 to return to menu: '); GOTO, option3 ;ENDIF
GOTO, DT
GOTO, option5;
;***************************************************************
;*****5. COMPUTE AND DISPLAY WAVELET POWER SPECTRUM*************
;***************************************************************
option5: 

n = N_ELEMENTS(signal[0,*]) sampling_rate = FLTARR(n-1) FOR k=1,n-1 DO BEGIN 
sampling_rate(k-1) = signal[0,k] - signal[0,(k-1)] ENDFOR dt = TOTAL(sampling_rate)/N_ELEMENTS(sampling_rate)*3600
GOTO, optionw6
; SUB MENU FOR CHANGING DEFAULT WAVELET PARAMETERS startwave:
CLS PRINT(('')) PRINT(('Wavelet Transform')) PRINT(('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~')) PRINT(('1. Normalize to standard deviation. (norm)')) PRINT(('2. Pad signal with zeros. (pad)')) PRINT(('3. Define frequency range. (fmin, fmax)')) PRINT(('4. Define mother wavelet. (mother)')) PRINT(('5. Change display properties. (')) PRINT(('6. Compute transform. (ST)')) PRINT(('7. Return to main menu.')) PRINT(('')) READ, input10, PROMPT=('Enter # to change desired parameter: ') & winput(0) = input10 IF (winput(0) EQ 1) THEN BEGIN PRINT(('')) PRINT(('Normalize signal to standard deviation')) PRINT(('')) READ,norm, PROMPT=('Enter (y/n): ') IF (norm EQ 'y') THEN BEGIN sst = sst / STDDEV(sst) ; winput(0) = 1 ENDIF IF (norm EQ 'n') THEN BEGIN sst = transpose(dtsignal((input(12) -1),*)) ; winput(0) = 0 ENDIF GOTO, startwave ENDIF IF (winput(0) EQ 2) THEN BEGIN PRINT(('')) PRINT(('Pad signal with zeros')) PRINT(('')) READ, pad, PROMPT=('Enter (y/n): ') IF (pad EQ 'y') THEN BEGIN winput(2) = 1 IF (pad EQ 'n') THEN BEGIN winput(2) = 0 ENDIF GOTO, startwave ENDIF IF (winput(0) EQ 3) THEN BEGIN PRINT(('')) PRINT(('Define Frequency')) PRINT(('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~')) PRINT(('Enter desired fmin')) PRINT(('fmin = ')) PRINT(('Enter desired fmax')) PRINT(('fmax = ')) PRINT(('Enter number of frequencies')) PRINT(('f_recalc:')) PRINT(('')) ; winput(6) = menu(3) winput(7) = menu(4) f_recalc: ; MORLET IF (winput(6) EQ 1) THEN BEGIN stof = (winput(7)+SQRT(2+(winput(7))^2)) / (4*!PI)
```

APPENDIX A: IDL CODE

```idl
winput(3) = stof/winput(9);s0 smax = stof/winput(8);max winput(4) = ALOG10(smax/winput(3))/(winput(5)+ALOG10(2))
   n = winput(5)+1; scale = DINDGEN(n)*winput(4)

n = winput(5)+1; scale = DINDGEN(n)*winput(4)
   scale = 2d0^scale*winput(3)
   f = stof/scale
GOTO,optionw5

;******************************************************************************
;*****W4. DEFINE MOTHER WAVELET******************************************************************************
;******************************************************************************
optionw4: CLS
PRINT ,(' Define Mother Wavelet ')
PRINT ,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~')
PRINT ,('1. Mother Wavelet. ( 'mother'+')')
PRINT ,('2. Wavelet Parameter. ( '+strtrim(string(winput(7),format='(f5.2)'),2)+')')
PRINT ,('3. Return to wavelet menu.' ) PRINT ,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~')
PRINT ,('')
READ ,minput ,PROMPT=('Enter option #: ') & winput(6) = input14
   IF (winput(6) EQ 1) THEN BEGIN
      mother = 'Morlet' winput(7) = 6
      parameter = 'w0 = '
   ENDIF
   IF (winput(6) EQ 2) THEN BEGIN
      mother = 'Paul' winput(7) = 4
      parameter = 'm = '
   ENDIF
   IF (winput(6) EQ 3) THEN BEGIN
      mother = 'DOG' winput(7) = 6
      parameter = 'm = '
   ENDIF
   GOTO,optionw4

GOTO,optionw4 ENDIF

;******************************************************************************
;*****W5. DISPLAY PROPERTIES******************************************************************************
;******************************************************************************
optionw5: ;IF (display EQ 0) THEN BEGIN bottom = 'min' top = 'max power calculated'
   ;ENDIF
CLS PRINT ,(' Wavelet Display Properties ')
PRINT ,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~')
PRINT ,('The number of contour levels is set to '+STRTRIM(levels ,2)+'.')
PRINT ,('The range of power displayed is set to '+STRTRIM(bottom ,2)+' and '+STRTRIM(top,2)+'.')
PRINT ,('')
PRINT ,('Note: If these are changed after the power spectrum is created, you')
PRINT ,('need to recompute the transform.') PRINT ,('')
PRINT ,('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~')
PRINT ,('')
READ ,input15 ,PROMPT=('Enter # of levels: ') & levels = input15 levels = menu(4)
PRINT ,('')
READ ,input16,PROMPT=('Min power: ') & bottom = input16 PRINT ,('')
READ ,input17,PROMPT=('Max power: ') & top = input17
GOTO,option3

;******************************************************************************
;*****W6. COMPUTE AND DISPLAY WAVELET POWER SPECTRUM******************************************************************************
;******************************************************************************

wave = WAVELET(est,dt,SO-winput(3),DJ-winput(4),J-winput(5),MOTHER=mother, $
         PARAM=winput(7),PAD=winput(2),PERIOD=period)
   power = (ABS(wave))~2 ; log_power = ALOG10(power)
```

APPENDIX A: IDL CODE

85

***************************************************************
*****OUTPUTS
***************************************************************
IF (INPUT(12) EQ 1) THEN BEGIN powerex=power GOTO,option3b ENDIF IF (INPUT(12) EQ 2) THEN BEGIN powerey=power GOTO,option3c ENDIF IF (INPUT(12) EQ 3) THEN BEGIN powerez=power GOTO,option3d ENDIF

poweret=power
signale=signal(0,*)
fe=f
powere_perp = (powerex + powerey + powerez) - poweret
log_power = ALOG10(powere_perp)

C = (1.60217646e-19)/(2*!PI*1.67262158e-27)
ion_cyc = (C*signal(4,*)*1e-9)

DEVICE, DECOMPOSED = 0
LOADCT, 39
WINDOW,1,XSIZE=1252,YSIZE=600,XPOS=0,YPOS=0
pos1 = [0.05,0.05,0.88,0.95]
pos2 = [0.90,0.05,0.95,0.95]
xrng = [min(signal(0,*)),max(signal(0,*))]

IF (display EQ 0) THEN BEGIN
top = MAX(log_power)
bottom = MIN(log_power)
ENDIF;
userlevels = (DOUBLE(top-bottom)/(levels -1))*INDGEN(levels)
colors = ROUND((254d/(levels -1))*INDGEN(levels))
CONTOUR,log_power,signal(0,*),f,XRANGE=xrng,XSTYLE=1,YRANGE=yrng1,YTITLE='UT [Hrs]',TITLE='Wavelet Power Spectrum (E_perp)',LEVELS=userLevels,FILL,C_COLORS=colors,POSITION=pos1

COLORBAR,/VERTICAL,/RIGHT,RANGE=[bottom,top],POSITION=pos2,TITLE='Power'
GOTO,optionEND

***************************************************************
*****POLARIZATION**********************************************
***************************************************************

1. FIND MINIMUM VARIANCE MATRIX
n=N_ELEMENTS(signal(0,*))
varmat ,n,transpose(signal(1,*)),transpose(signal(2,*)),transpose(signal(3,*)),bavg,bmat

2. FIND THE EIGENVALUES AND VECTORS FOR THE VARIANCE MATRIX

wavelet.pro A wavelet technique provided courtesy of Christopher Torrence and Gilbert P. Compo of University of Colorado, Atmospheric and Oceanic Sciences (http://paos.colorado.edu/research/wavelets). This program also calculates the total integrated wave power.

A.4 Supporting Routines

ebstructures.pro Essembles linearly interpolated data into structures to be used with cluster_wave_5_all.pro and cluster_wave_e_2.pro.

PRO ebstructures, s_ce,scb,DATESTRN
COMMON hefield, nefv1,timehel,he1x,he1y,he1z, he2x,he2y,he2z, he3x,he3y,he3z, nefv1,timehe3,he3x,he3y,he3z, he4x,he4y,he4z,he4x,he4y,he4z
COMMON hefield_filtered,nef1,timehel,he1x,he1y,he1z, he2x,he2y,he2y,he2z,he2y,he2z, he3x,he3y,he3z,he3x,he3y,he3z,he3x,he3y,he3z
COMMON hbf,b1x,b1y,b1z, b2x,b2y,b2z, b3x,b3y,b3z, b4x,b4y,b4z
COMMON imeni,npci,ni1,v1,tempi1,td1,td2,td3,td4,npci,ni1,ni2,ni3,ni4

#define cluster_wave_all
#define cluster_wave_e

PRO interp, biint,viint,tintbv1, $

; Load Cluster data (EFW and FSM) and extracts time, position, magnetic field and electric field vectors readclusterallwav, nameefw, nameefg, namecis, datestrn
tvarmin=0.0 tvarmax=0.0
APPENDIX A: IDL CODE

; DATA STRINGS AND INTERVALS
hclint_hr, timehb1, timehb1 hclint_hr, timehb2, timehb2 hclint_hr, timehb3, timehb3 hclint_hr, timehb4, timehb4 hclint_hr, timehe1, timehe1 hclint_hr, timehe2, timehe2 hclint_hr, timehe3, timehe3 hclint_hr, timehe4, timehe4 hclint_hr, timen1, timen1 hclint_hr, timen3, timen3

; Converts time from minutes to seconds
timehb1 = timehb1*60 timehb2 = timehb2*60 timehb3 = timehb3*60 timehb4 = timehb4*60 timehe1 = timehe1*60 timehe2 = timehe2*60 timehe3 = timehe3*60 timehe4 = timehe4*60 timen1 = timeen1*60 timen3 = timeen3*60

; Converts time from minutes to seconds
timehb1 = timehb1*60 timehb2 = timehb2*60 timehb3 = timehb3*60 timehb4 = timehb4*60
timen1 = timen1*60 timen3 = timen3*60

; Converts time from minutes to seconds
timehb1 = timehb1*60 timehb2 = timehb2*60 timehe1 = timehe1*60 timehe2 = timehe2*60 timehe3 = timehe3*60 timehe4 = timehe4*60

; Converts time from minutes to seconds
timehb1 = timehb1*60 timehb2 = timehb2*60 timehb3 = timehb3*60 timehb4 = timehb4*60
timen1 = timen1*60 timen3 = timen3*60

; Converts time from minutes to seconds
timen1 = timen1*60 timen3 = timen3*60

; Converts time from minutes to seconds
timen1 = timen1*60 timen3 = timen3*60

; Converts time from minutes to seconds
timen1 = timen1*60 timen3 = timen3*60

; Converts time from minutes to seconds
timen1 = timen1*60 timen3 = timen3*60

; Converts time from minutes to seconds
timen1 = timen1*60 timen3 = timen3*60

; Converts time from minutes to seconds
timen1 = timen1*60 timen3 = timen3*60

; Converts time from minutes to seconds
timen1 = timen1*60 timen3 = timen3*60

; Converts time from minutes to seconds
timen1 = timen1*60 timen3 = timen3*60

; Calculates max and min times for each spacecraft
tmin1 = max((timehb1(0), timehe1f(0)))
tmin2 = max((timehb2(0), timehe2f(0)))
tmin3 = max((timehb3(0), timehe3f(0)))
tmin4 = max((timehb4(0), timehe4f(0)))
tmin1b = max((timeen1(0), timen1(0)))
tmin3b = max((timeen3(0), timen3(0)))
tmax1 = max((timehb1(nfgm1-1), timehe1f(nefw1f-1)))
tmax2 = max((timehb2(nfgm2-1), timehe2f(nefw2f-1)))
tmax3 = max((timehb3(nfgm3-1), timehe3f(nefw3f-1)))
tmax4 = max((timehb4(nfgm4-1), timehe4f(nefw4f-1)))
tmax1b = max((timeen1(nfgm1-1), timen1(ncis1-1)))
tmax3b = max((timeen3(nfgm3-1), timen3(ncis3-1)))
tmaxva1 = max((timeen1(nfgm1-1), timen1(ncis1-1)))
tmaxva3 = max((timeen3(nfgm3-1), timen3(ncis3-1)))

; Linear interpolation of data per spacecraft --> creates identical time tags for E and B fields per spacecraft
lin_interpolation, timen1, timen2, timen3, timen4, timen5, timen6, timen7, timen8, timeen1, timeen2, timeen3, timeen4, timeen5, timeen6, timeen7, timeen8

; Converts time from minutes to hours
timen1 = timen1/3600 timen3 = timen3/3600
timen1 = timen1/3600 timen3 = timen3/3600
timen1 = timen1/3600 timen3 = timen3/3600
timen1 = timen1/3600 timen3 = timen3/3600
timen1 = timen1/3600 timen3 = timen3/3600
timen1 = timen1/3600 timen3 = timen3/3600
timen1 = timen1/3600 timen3 = timen3/3600

; Calculates total magnetic field strength
hbint3(3,*) = SQRT(hbint3(0,*) + hbint3(1,*) + hbint3(2,*) + hbint3(3,*)

; Calculates total electric field strength
heint3(3,*) = SQRT(heint3(0,*) + heint3(1,*) + heint3(2,*) + heint3(3,*)

; Alfvén Speed
va1 = (1.0e-3)*(1.0e-9)*sqrt(b1int(0,*)^2 + b1int(1,*)^2 + b1int(2,*)^2 + b1int(3,*)^2)

; Calculates total magnetic field strength
hbint3(3,*) = SQRT(hbint3(0,*) + hbint3(1,*) + hbint3(2,*) + hbint3(3,*)

; Calculates total electric field strength
heint3(3,*) = SQRT(heint3(0,*) + heint3(1,*) + heint3(2,*) + heint3(3,*)

; Alfvén Speed
va1 = (1.0e-3)*(1.0e-9)*sqrt(b1int(0,*)^2 + b1int(1,*)^2 + b1int(2,*)^2 + b1int(3,*)^2)
APPENDIX A: IDL CODE

87

MAGNETIC POWER SPECTRA

Put data into convenient arrays and converts time from seconds to hours.

\[
\text{time}_{cl1} = \text{tint1}/3600 \quad \text{time}_{cl2} = \text{tint2}/3600 \quad \text{time}_{cl3} = \text{tint3}/3600 \quad \text{time}_{cl4} = \text{tint4}/3600
\]

\[
\text{time}_{clb} = \left[\text{time}_{cl1}, \text{hb1int}\right] \quad \text{time}_{cl2b} = \left[\text{time}_{cl2}, \text{hb2int}\right] \quad \text{time}_{cl3b} = \left[\text{time}_{cl3}, \text{hb3int}\right] \quad \text{time}_{cl4b} = \left[\text{time}_{cl4}, \text{hb4int}\right]
\]

\[
\text{time}_{cl1b} = \left[\text{time}_{cl1}, \text{hb1int}\right] \quad \text{time}_{cl2b} = \left[\text{time}_{cl2}, \text{hb2int}\right] \quad \text{time}_{cl3b} = \left[\text{time}_{cl3}, \text{hb3int}\right] \quad \text{time}_{cl4b} = \left[\text{time}_{cl4}, \text{hb4int}\right]
\]

\[
\text{structure containing all satellite}
\]

\[
\text{rawvslin}, \text{timehb3}, \text{hb3x}, \text{hb3y}, \text{hb3z}, \text{time}_{cl3}, \text{hb3int}
\]

RETURN

END

lin_interp.pro Adapted from code courtesy of Katariina Nykyri. Linearly interpolates magnetic field and velocity data for uniform time stamps.

PRO lin_interp, tmin1, tmin2, tmin3, tmin4, tmax1, tmax2, tmax3, tmax4, hb1int, hb2int, hb3int, hb4int, he1int, he2int, he3int, he4int, tint1, tint2, tint3, tint4

COMMON hefield_filtered, nefw1f, timehe1f, he1xf, he1yf, he1zf, nefw2f, timehe2f, he2xf, he2yf, he2zf, nefw3f, timehe3f, he3xf, he3yf, he3zf, nefw4f, timehe4f, he4xf, he4yf, he4zf, he1f, he2f, he3f, he4f

COMMON hbfield, nfgm1, timehb1, hb1x, hb1y, hb1z, nfgm2, timehb2, hb2x, hb2y, hb2z, nfgm3, timehb3, hb3x, hb3y, hb3z, nfgm4, timehb4, hb4x, hb4y, hb4z, hb1, hb2, hb3, hb4

s_hb = {s_hb1:hb1, s_hb2:hb2, s_hb3:hb3, s_hb4:hb4} ; structure containing all satellite

s_he = {s_he1:he1, s_he2:he2, s_he3:he3, s_he4:he4} ;structure containing all satellite

dtime = 0.0445

ntime1 = (tmax1-tmin1)/dtime ntime1 = round(ntime1) + 1

ntime2 = (tmax2-tmin2)/dtime ntime2 = round(ntime2) + 1

ntime3 = (tmax3-tmin3)/dtime ntime3 = round(ntime3) + 1

ntime4 = (tmax4-tmin4)/dtime ntime4 = round(ntime4) + 1

hb1int = DBLARR(3,ntime1) hb2int = DBLARR(3,ntime2) hb3int = DBLARR(3,ntime3) hb4int = DBLARR(3,ntime4)

he1int = DBLARR(3,ntime1) he2int = DBLARR(3,ntime2) he3int = DBLARR(3,ntime3) he4int = DBLARR(3,ntime4)

tint1 = DBLARR(1,ntime1) tint2 = DBLARR(1,ntime2) tint3 = DBLARR(1,ntime3) tint4 = DBLARR(1,ntime4)

i1 = 0L eitos1 = 0L eitos2 = 0L eitos3 = 0L eitos4 = 0L

for i1=1L,ntime1-2 do begin

  if (((i1 mod 1000) eq 0) then begin ; print, 'i=', i1 ; endif

  ttmin1 = tmin1 + i1*dtime

  its1=(LINDGEN(1))(0) while (((timehb1(its1) lt ttmin1)) do its1 = its1+1

  eits1=(LINDGEN(1))(0) while (((timehe1f(eits1) lt ttmin1)) do eits1 = eits1+1

  itbs1 = timehb1(its1-1) idtimehb1 = ttmin1 - timehb1(its1-1) idtimehb1f = timehe1f(ite1) - timehe1f(ite1-1)

  hb1int(*,i1) = hb1(*,ite1-1) + idtimehb1/idtimehb1f(*,ite1-1) - he1f(*,ite1-1) + idtimehe1f/idtimehe1f(*,ite1-1)

  ttmin1 = ttmin1 + dtime

for i2=1L,ntime2-2 do begin

  if (((i2 mod 1000) eq 0) then begin ; print, 'i2=', i2 ; endif

  ttmin2 = tmin2 + i2*dtime

  its2=(LINDGEN(1))(0) while (((timehb2(its2) lt ttmin2)) do its2 = its2+1

  eits2=(LINDGEN(1))(0) while (((timehe2f(eits2) lt ttmin2)) do eits2 = eits2+1

  itbs2 = timehb2(its2-1) idtimehb2 = ttmin2 - timehb2(its2-1) idtimehb2f = timehe2f(ite2) - timehe2f(ite2-1)

  ttmin2 = ttmin2 + dtime

for i3=1L,ntime3-2 do begin

  if (((i3 mod 1000) eq 0) then begin ; print, 'i3=', i3 ; endif

  ttmin3 = tmin3 + i3*dtime

  its3=(LINDGEN(1))(0) while (((timehb3(its3) lt ttmin3)) do its3 = its3+1

  eits3=(LINDGEN(1))(0) while (((timehe3f(eits3) lt ttmin3)) do eits3 = eits3+1

  itbs3 = timehb3(its3-1) idtimehb3 = ttmin3 - timehb3(its3-1) idtimehb3f = timehe3f(ite3) - timehe3f(ite3-1)

  hb3int(*,i3) = hb3(*,ite3-1) + idtimehb3/idtimehb3f(*,ite3-1) - he3f(*,ite3-1) + idtimehe3f/idtimehe3f(*,ite3-1)

  ttmin3 = ttmin3 + dtime

for i4=1L,ntime4-2 do begin

  if (((i4 mod 1000) eq 0) then begin ; print, 'i4=', i4 ; endif

  ttmin4 = tmin4 + i4*dtime

  its4=(LINDGEN(1))(0) while (((timehb4(its4) lt ttmin4)) do its4 = its4+1

  eits4=(LINDGEN(1))(0) while (((timehe4f(eits4) lt ttmin4)) do eits4 = eits4+1

  itbs4 = timehb4(its4-1) idtimehb4 = ttmin4 - timehb4(its4-1) idtimehb4f = timehe4f(ite4) - timehe4f(ite4-1)

  hb4int(*,i4) = hb4(*,ite4-1) + idtimehb4/idtimehb4f(*,ite4-1) - he4f(*,ite4-1) + idtimehe4f/idtimehe4f(*,ite4-1)

  ttmin4 = ttmin4 + dtime
hb2int(*,i2) = hb2(*,itbs2-1) + idtimehb2f*(hb2(*,itbs2 -1) - he2f(*,ites2-1))
he2int(*,i2) = he2f(*,ites2-1) + idtimehe2f*(he2f(*,ites2) - he2f(*,ites2-1))
tint2(i2) = ttmin2
if (i2 eq ntime2-2) then begin
for i3=1L,ntime3 -2 do begin
if ((i1 mod 1000) eq 0) then begin ; print, 'i=', i3 ; endif
ttmin3 = ttmin3 + i3*dttime
; its3=(LINDGEN(1))(0) while ((timehb3(itbs3) lt ttmin3)) do itbs3 = itbs3+1
; eits3=(LINDGEN(1))(0) while ((timehe3f(ites3) lt ttmin3)) do eits3 = eits3+1
idtimehb3i = ttmin3 - timehb3(itbs3 -1) idtimehb3f = timehb3(itbs3) - timehb3(itbs3 -1)
idtimehe3i = ttmin3 - timehe3f(ites3 -1) idtimehe3f = timehe3f(ites3) - timehe3f(ites3 -1)
hb3int(*,i3) = hb3(*,itbs3 -1) + idtimehb3f*(hb3(*,itbs3) - hb3(*,itbs3 -1))
he3int(*,i3) = he3f(*,ites3 -1) + idtimehe3f*(he3f(*,ites3) - he3f(*,ites3 -1))
tint3(i3) = ttmin3
if (i3 eq ntime3-2) then begin
for i4=1L,ntime4 -2 do begin
if ((i1 mod 1000) eq 0) then begin ; print, 'i=', i4 ; endif
ttmin4 = ttmin4 + i4*dttime
; its4=(LINDGEN(1))(0) while ((timehb4(itbs4) lt ttmin4)) do itbs4 = itbs4+1
; eits4=(LINDGEN(1))(0) while ((timehe4f(eits4) lt ttmin4)) do eits4 = eits4+1
idtimehb4i = ttmin4 - timehb4(itbs4 -1) idtimehb4f = timehb4(itbs4) - timehb4(itbs4 -1)
idtimehe4i = ttmin4 - timehe4f(ites4 -1) idtimehe4f = timehe4f(ites4) - timehe4f(ites4 -1)
hb4int(*,i4) = hb4(*,itbs4 -1) + idtimehb4f*(hb4(*,itbs4) - hb4(*,itbs4 -1))
he4int(*,i4) = he4f(*,ites4 -1) + idtimehe4f*(he4f(*,ites4) - he4f(*,ites4 -1))
tint4(i4) = ttmin4
endfor ; end of i4 loop
endif
endfor ; end of i3 loop
endif
endfor ; end of i2 loop
endif
endfor ; end of i1 loop
hb1int = hb1int(*,1:ntime1-2) hb2int = hb2int(*,1:ntime2-2) hb3int = hb3int(*,1:ntime3-2)
hb4int = hb4int(*,1:ntime4-2)
he1int = he1int(*,1:ntime1-2) he2int = he2int(*,1:ntime2-2) he3int = he3int(*,1:ntime3-2)
he4int = he4int(*,1:ntime4-2)
tint1 = tint1(0,1:ntime1-2) tint2 = tint2(0,1:ntime2-2) tint3 = tint3(0,1:ntime3-2)
tint4 = tint4(0,1:ntime4-2)
RETURN

lin_interb_bv.pro Adapted from code courtesy of Katariina Nykyri. Linearly interpolates magnetic and electric field data for uniform time stamps.
PRO lin_interp_bv , tmin1,tmin3,tmax1,tmax3 ; PRO lin_interp_bv , tmin1,tmin3, ,tmin4, tmax1,tmax3,tmax4
COMMON hbfield, nfgm1,timehb1,hbx1,hby1,hb1z, $ nfgm2,timehb2,hbx2, $ nfgm3,timehb3,hbx3, $ nfgm4,timehb4,hbx4,hby4,hbx4,hb1,hb2,hb3,hb4
COMMON proton, ncis1,timen1,np1,v1,temp1,tpar1,tperp1, $ ncis3,timen3, np3,v3,temp3,tpar3,tperp3, $ ncis4,timen4,np4,v4,temp4,tpar4, $ tperp4
COMMON interp, b1int,v1int,tintbv1, $ b2int,v2int,tintbv2, $ b3int,v3int,tintbv3, $ b4int,v4int,tintbv4
;s_hb = {s_hb1:hbx1,s_hb2:hbx2,s_hb3:hbx3,s_hb4:hbx4} ;s_he = {s_he1:he1,s_he2:he2,s_he3:he3,s_he4:he4}
dtime = 4.02
ntime1 = (tmax1-tmin1)/dtime nttime1 = round(nitime1) + 1 ;ntime2 = (tmax2-tmin2)/dtime
;nttime2 = round(nitime2) + 1 ;ntime3 = (tmax3-tmin3)/dtime ;ntime3 = round(nitime3) + 1 ;ntime4 = (tmax4-tmin4)/dtime ;ntime4 = round(nitime4) + 1
b1int = DLARR(3,ntime1) ;b2int = DLARR(3,ntime2) b3int = DLARR(3,ntime3) ;b4int = DLARR(3,ntime4)
v1int = DBLARR(3,ntime1) ; v2int = DBLARR(3,ntime2) v3int = DBLARR(3,ntime3) ; v4int = DBLARR(3,ntime4)
tintb1 = DBLARR(1,ntime1) ; tintb2 = DBLARR(1,ntime2) tintb3 = DBLARR(1,ntime3) ; tintb4 = DBLARR(1,ntime4)
its1 = OL eits1 = OL ; its2 = OL eits2 = OL its3 = OL eits3 = OL ; its4 = OL eits4 = OL
print, 'Please wait while data is interpolated'
for i1=1L,ntime1 - 2 do begin
if ((i1 mod 1000) eq 0) then begin ; print, 'i=', i1 ; endif
ttmin1 = tmin1 + i1*dtime
; its1=(LINDGEN(1))(0) while ((timehb1(its1) lt ttmin1)) do its1 = its1+1
itbsh1 = ttmin1 - timehb1(its1-1) idtimeb1i = ttmin1 - timehb1(its1 -1)
; idtimeb1f = ttmin1 - timehb1(its1-1)
b1int(*,i1) = hb1(*,its1-1) + idtimeb1i/idtimeb1f*(hb1(*,its1) - hb1(*,its1-1))
v1int(*,i1) = v1(*,its1-1) + idtimev1i/idtimev1f*(v1(*,its1) - v1(*,its1-1))
tintbv1(i1) = ttmin1
if (i1 eq ntime1 - 2) then begin
for i3=1L,ntime3 - 2 do begin
if ((i3 mod 1000) eq 0) then begin ; print, 'i=', i3 ; endif
ttmin3 = tmin3 + i3*dtime
; its3=(LINDGEN(1))(0) while ((timehb3(its3) lt ttmin3)) do its3 = its3+1
itbsh3 = ttmin3 - timehb3(its3-1) idtimeb3i = ttmin3 - timehb3(its3-1)
; idtimeb3f = ttmin3 - timehb3(its3-1)
b3int(*,i3) = hb3(*,its3-1) + idtimeb3i/idtimeb3f*(hb3(*,its3) - hb3(*,its3-1))
v3int(*,i3) = v3(*,its3-1) + idtimev3i/idtimev3f(v3(*,its3) - v3(*,its3-1))
tintbv3(i3) = ttmin3
if (i3 eq ntime3 - 2) then begin
; for i4=1L,ntime4 - 2 do begin
; ttmin4 = tmin4 + i4*dtime
; while ((timehb4(its4) lt ttmin4)) do its4 = its4+1 ; itbsh4 = its4
; while ((timehb4(its4) lt ttmin4)) do its4 = its4+1 ; itbsh4 = its4
; idtimeb4i = timehb4(its4-1) ; idtimeb4f = timehb4(its4-1) ; idtimev4i = timehb4(its4-1) ; idtimev4f = timehb4(its4-1)
; b4int(*,i4) = b4(*,its4-1) + idtimeb4i/idtimeb4f*(hb4(*,its4) - hb4(*,its4-1))
v4int(*,i4) = v4(*,its4-1) + idtimev4i/idtimev4f(v4(*,its4) - v4(*,its4-1))
; tintbv4(i4) = ttmin4
;endfor ; end of i4 loop
endif
endfor ; end of i3 loop
endif
endfor ; end of i1 loop
b1int = b1int(*,1:ntime1 - 2) ; b2int = b2int(*,1:ntime2 - 2) b3int = b3int(*,1:ntime3 - 2)
; b4int = b4int(*,1:ntime4 - 2)
v1int = v1int(*,1:ntime1 - 2) ; v2int = v2int(*,1:ntime2 - 2) v3int = v3int(*,1:ntime3 - 2)
; v4int = v4int(*,1:ntime4 - 2)
tintb1 = tintb1(0,1:ntime1 - 2) ; tintb2 = tintb2(0,1:ntime2 - 2) tintb3 = tintb3(0,1:ntime3 - 2)
; tintb4 = tintb4(0,1:ntime4 - 2)
RETURN END

lin_interp_va.pro Adapted from code courtesy of Katariina Nykyri. Linearly interpolates density, velocity and magnetic field data for uniform time stamps.

PRO lin_interp_va, tminva1, tminva3, tmaxva1, tmaxva3, b1int, b3int, np1int, np3int, tintva1, tintva3
COMMON hbfield, nfgm1, timehb1, hb1x, hb1y, hb1z, $ nfgm2, timehb2, hb2x, hb2y, hb2z, $ nfgm3, timehb3, hb3x, hb3y, hb3z, $ nfgm4, timehb4, hb4x, hb4y, hb4z, hb1, hb2, hb3, hb4

APPENDIX A: IDL CODE
COMMON proton, ncis1, timen1, np1, v1, temp1, tpar1, tperp1, $ncis3, timen3, np3, v3, temp3, tpar3, tperp3, $ncis4, timen4, np4, v4, temp4, tpar4, tperp4 dt time = 4.05 ntime1 = (tmaxva1 - tminva1)/dt time ntime1 = round(ntime1) + 1 ntime3 = (tmaxva3 - tminva3)/dt time ntime3 = round(ntime3) + 1 b1int = DBLARR(3, ntime1) b3int = DBLARR(3, ntime3) np1int = DBLARR(3, ntime1) np3int = DBLARR(3, ntime3) tintva1 = DBLARR(1, ntime1) tintva3 = DBLARR(1, ntime3) its1 = 0L eits1 = 0L its3 = 0L eits3 = 0L print, 'Please wait while data is interpolated' for i1 = 1L, ntime1 - 2 do begin; if ((i1 mod 1000) eq 0) then begin ; print, 'i=', i1 ; endif ttmin1 = tminva1 + i1*dt time ; ; its1 = (LINDGEN(1))(0) while ((timehb1(its1) lt ttmin1)) do its1 = its1 + 1 itbs1 = its1 ; ; eits1 = (LINDGEN(1))(0) while ((timen1(eits1) lt ttmin1)) do eits1 = eits1 + 1 idtimehb1 = ttmin1 - timehb1(itbs1-1) idtimehb1f = timehb1(itbs1) - timehb1(itbs1-1) idtimen1 = ttmin1 - timen1(ites1-1) idtimen1f = timen1(ites1) - timen1(ites1-1) b1int(*, i1) = hb1(*, itbs1-1) + idtimehb1f/idtimehb1f*(hb1(*, itbs1) - hb1(*, itbs1-1)) np1int(*, i1) = np1(*, ites1-1) + idtimen1f/idtimen1f*(np1(*, ites1) - np1(*, ites1-1)) tintva1(i1) = ttmin1; if (i1 eq ntime1-2) then begin for i3 = 1L, ntime3 - 2 do begin; if ((i3 mod 1000) eq 0) then begin ; print, 'i=', i3 ; endif ttmin3 = tminva3 + i3*dt time ; ; its3 = (LINDGEN(1))(0) while ((timehb3(its3) lt ttmin3)) do its3 = its3 + 1 itbs3 = its3 ; ; eits3 = (LINDGEN(1))(0) while ((timen3(eits3) lt ttmin3)) do eits3 = eits3 + 1 idtimehb3 = ttmin3 - timehb3(itbs3-1) idtimehb3f = timehb3(itbs3) - timehb3(itbs3-1) idtimen3 = ttmin3 - timen3(ites3-1) idtimen3f = timen3(ites3) - timen3(ites3-1) b3int(*, i3) = hb3(*, itbs3-1) + idtimehb3f/idtimehb3f*(hb3(*, itbs3) - hb3(*, itbs3-1)) np3int(*, i3) = np3(*, ites3-1) + idtimen3f/idtimen3f*(np3(*, ites3) - np3(*, ites3-1)) tintva3(i3) = ttmin3; endif; end of i3 loop endif; end of i1 loop b1int = b1int(*, 1: ntime1-2) b3int = b3int(*, 1: ntime3-2) np1int = np1int(*, 1: ntime1-2) np3int = np3int(*, 1: ntime3-2) tintva1 = tintva1(0, 1: ntime1-2)/3600 tintva3 = tintva3(0, 1: ntime3-2)/3600 RETURN END hclint_hr.pro Adapted from code courtesy of Katarina Nykyri. Translates time stamps. PRO hclint_hr, time, epoch ; SUBTRACT days, months and years --> time is in minutes epoch=iso2cdfepoch_hr(time) end bogusdatafilter_common.pro Filters out non-physical electric field data. PRO bogusdatafilter_common, data1, data2, data3, time, datane1, datane2, datane3, timenew ; COMMON hefield_filitered, nefu1f, timehef1f, he1zf, he1yf, he1zf, $ he2zf, he2zf, he2zf, he2zf, $ he3zf, he3zf, he3zf, $ he4zf, he4zf, he4zf, $ he5zf, he5zf, $ good_data=where(data1 gt -1e+8) n=n_elements(good_data) datane1=DLBLARR(1,n) datane2=DLBLARR(1,n) datane3=DLBLARR(1,n) timenew=DLBLARR(1,n) for i=0L, n-1 do begin datane1(i) = data1(good_data(i)) datane2(i) = data2(good_data(i)) datane3(i) = data3(good_data(i)) timenew(i) = time(good_data(i))
endfor
RETURN END
Bibliography


Lavraud, B., Email, 2011.


Nykyri, K., Data analysis techniques, University Lecture, 2011a.
Nykyri, K., Bow shock, magnetosheath and magnetosphere, University Lecture, 2011b.

Nykyri, K., Local and global simulations, University Lecture, 2011c.


