Excitation of Forced Ion Acoustic Waves,
Large Plasma Sheets, and Magnetic Field Fluctuations
over Gakona, Alaska.

Joel Cohen

June 2009

Plasma Science and Fusion Center
Massachusetts Institute of Technology
Cambridge MA 02139 USA

Excitation of Forced Ion Acoustic Waves, Large Plasma Sheets, and Magnetic Field Fluctuations over Gakona, Alaska

by

Joel Cohen

B.S., Massachusetts Institute of Technology (2007)

Submitted to the Department of Nuclear Science and Engineering
in partial fulfillment of the requirements for the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2009

© Massachusetts Institute of Technology 2009. All rights reserved.

Author . . . .
Department of Nuclear Science and Engineering
February 2009

Certified by . . . .
Prof. Min-Chang Lee
Former Group Head
Plasma Science and Fusion Center
Thesis Supervisor

Certified by . . . .
Prof. Jeffrey P. Freidberg
Associate Director
Plasma Science and Fusion Center
Department of Nuclear Science and Engineering
Thesis Reader

Accepted by . . . .
Prof. Jacquelyn C. Yanch
Department of Nuclear Science and Engineering
Chair, Department Committee on Graduate Theses

Excitation of Forced Ion Acoustic Waves, Large Plasma Sheets, and Magnetic Field Fluctuations over Gakona, Alaska

by

Joel Cohen

Abstract

Two research subjects: (1) excitation of "forced ion acoustic waves", and (2) "simultaneous excitation of plasma density fluctuations and geomagnetic field fluctuations" are reported in my M.S. thesis. The data was acquired in our experiments conducted at Gakona, Alaska from summer 2007 to winter 2008, using DoD/NSF-funded HAARP facilities and our own optical (ASIS) and radio instruments (VLF receiving system of IRIS) aided by GPS satellites as well as AMISR radar at Poker Flat, Alaska. We suggest that "Forced ion acoustic waves" detected by MUIR radar on Oct. 29 during 6:20-6:30 UT arise from keV electron precipitation associated with the occurrence of green aurora. Our work shows, for the first time, that MUIR radar is suitable for probing naturally occurring space plasma processes and not limited to HF heater-induced effects. This would extend the usage of MUIR for the investigation of space weather together with AMISR radar at Poker Flat, to advance our knowledge in space plasma turbulence.

The research on "simultaneous excitation of plasma density fluctuations and geomagnetic field fluctuations" is an extension of my B.S. thesis research on thermal filamentation instability, which started in our summer Gakona experiments in 2005. Large plasma sheets (also known as sheet-like filaments) can be excited by HF O-mode and X-mode heater waves via thermal filamentation instability. The dominant nonlinearity is provided by the differential Joule heating acting on electrons, which subsequently gives rise to a cross-field thermal pressure force, to concomitantly generate spatially varying plasma density fluctuations and geomagnetic field fluctuations. It is interesting to find that the fractional density fluctuations are approximately equal to the fractional magnetic field fluctuations. This gives us the theoretical basis to use ground-based magnetometer measurements to infer the density fluctuations in space plasma turbulence. Such a remote sensing technique for probing the space plasma is much more effective and economic than using a beacon satellite.

Thesis Supervisor: Prof. Min-Chang Lee
Title: Former Group Head
Plasma Science and Fusion Center

Thesis Reader: Prof. Jeffrey P. Freidberg
Title: Associate Director
Plasma Science and Fusion Center
Department of Nuclear Science and Engineering

Acknowledgments

I would like to begin by expressing my gratitude and appreciation for Professor Min-Chang Lee's advising and mentorship since my freshman year of undergraduate studies. Many educational and research experiences have brought me to where I currently am, including numerous tutorial lectures, experiment campaigns, conference proceedings, and journal papers. Professor Lee has a truly remarkable concern for motivating and supporting students in both their research and broader life. I also want to thank my thesis reader Professor Jeffrey Freidberg. I have had the privilege of attending his lectures and am thankful for his excellent teaching and personality. I'm very glad to have him as my thesis reader, and for his interest and encouragement. I am thankful to Prof. Manuel Martinez-Sanchez and Dr. Oleg Batishchev for their interest, encouragement, and collaboration as a team on our research projects. I look forward to having continuing work with Prof. Martinez-Sanchez and Dr. Batishchev for my further research at MIT.

I'd like to thank Professor Brenton Watkins and Dr. Mike Nicolls for recording data for research collaboration, as well as our HAARP sponsors Dr. Paul Kossey, Dr. Ed. Kennedy, and Dr. Craig Selcher for their support under grant: N00014-07-1-0999. I acknowledge DoD/NSF for the use of their HAARP diagnostic instruments for our experiments in Gakona, Alaska.

I'm particularly struck by the ongoing contributions of Rezy Pradipta in our research group, and I'd like to thank him for useful discussions of data analyses and experiment procedures. It is always a pleasure to conduct experiments and pursue research with him.

My family has been very supportive of me from the beginning, and I'm grateful for their encouragement and love. I have been touched by a number of fellow students at MIT and thank them for sharing time, meals, and long conversations. I'm also thankful for my church family which has been a keen source of support and fellowship, especially this past year. Finally, I thank my Lord for keeping me and guiding me in this whole process, for bringing me safely through my undergraduate years and

blessing me with the opportunity to take up graduate study. In the darkest hours He was with me, and in the brighest rays of noonday He is my joy.

Contents

List of Figures

Chapter 1

Motivation

Plasma is a medium that supports a wide variety of normal wave modes, both electrostatic (ES) and electromagnetic (EM). Ion acoustic waves are an engaging area of study, comprising an important branch of plasma waves. They are analogous to sound waves in a neutral medium, but they are mediated by the Coulomb force and can therefore occur in the absence of collisions. The ion-acoustic velocity is given by:

$$v_s = \sqrt{\frac{k_B(T_e + \gamma T_i)}{M}}$$

which depends not only on ion temperature (T_i) but also electron temperature (T_e). These waves can experience significant ion Landau damping, particularly when $T_e \approx T_i$. This is not the case in laser fusion experiments, where $T_e >> T_i$, but it is an important consideration in space plasma where T_e/T_i is, in general, of order unity.

I have been investigating processes such as the thermal filamentation instability [J.A. Cohen, B.S. Thesis, MIT, 2007; Cohen et al., 2007a; Cohen et al., 2007b] and parametric decay instability (PDI) [Cohen et al., 2007c], using the NSF/DoD High Auroral Active Research Project (HAARP) facility in Gakona, Alaska. These instabilities produce plasma turbulence with different length and time scales; PDI in particular involves a three wave-interaction process in which an O-mode pump wave decays into a Langmuir wave and an ion acoustic wave. This Langmuir wave can grow and become a pump wave for another decay into a daughter Langmuir wave

and an ion acoustic wave, yielding a cascading spectrum of Langmuir waves in both resonant and non-resonant processes as described in Kuo and Lee, [2005]. In contrast, PDI-excited ion acoustic waves die out quickly due to Landau damping.

However, enhanced ion acoustic waves were detected by incoherent scatter radar (ISR) as "ion lines" during ionospheric RF plasma heating experiments. Figure 1 shows a set of ion line and plasma line data recorded at Arecibo Observatory in Puerto Rico [Lee et al., 1994]. As shown in Panel (a), ion lines have a characteristic double-humped structure with upshifted and downshifted frequencies due to Doppler shifts that correspond, respectively, to downgoing and upgoing ion acoustic waves detected by the Arecibo ISR. Doppler spreading due to thermal ion motion causes the broad spectrum of ion lines. In contrast, upshifted and downshifted plasma lines correspond to detected downgoing and upgoing Langmuir waves. Panel (b) shows one example of ion line spectrum with zero-frequency modes, which were occasionally observed, presumably along with Langmuir wave excitation via the oscillating two stream instability (OTSI) [Kuo and Lee, 2005]. Note that enhanced ion lines have an asymmetric double-humped spectrum. I believe that these ion acoustic waves (i.e., ion line enhancements) were not excited directly by PDI. Rather, they were "forced ion acoustic waves" driven non-linearly by the PDI- or OTSI-excited Langmuir wave turbulence. However, forced ion acoustic waves can be excited by naturally occurring ionospheric currents, to be discussed later.

In the next sections I will describe our experiments conducted at Gakona, Alaska during October 24 to 31, 2008. These experiments were aimed at investigating forced ion acoustic waves, large plasma sheets, and magnetic field fluctuations among others, using our All Sky Imaging System (ASIS) and VLF receiving system of IRIS, HAARP HF heater, digisonde, Modular UF Ionospheric Radar (MUIR), SuperDarn HF radar, riometer, GPS satellites, and DMSP satellite.

For my M.S. thesis I will focus on reporting experimental and theoretical research of two subjects (1) excitation of "forced ion acoustic waves", and (2) simultaneous excitation of plasma density fluctuations and geomagnetic field fluctuations. The presentation is organized as follows. Our Gakona experiments are described in Section

(a) Expected ISR Spectrum
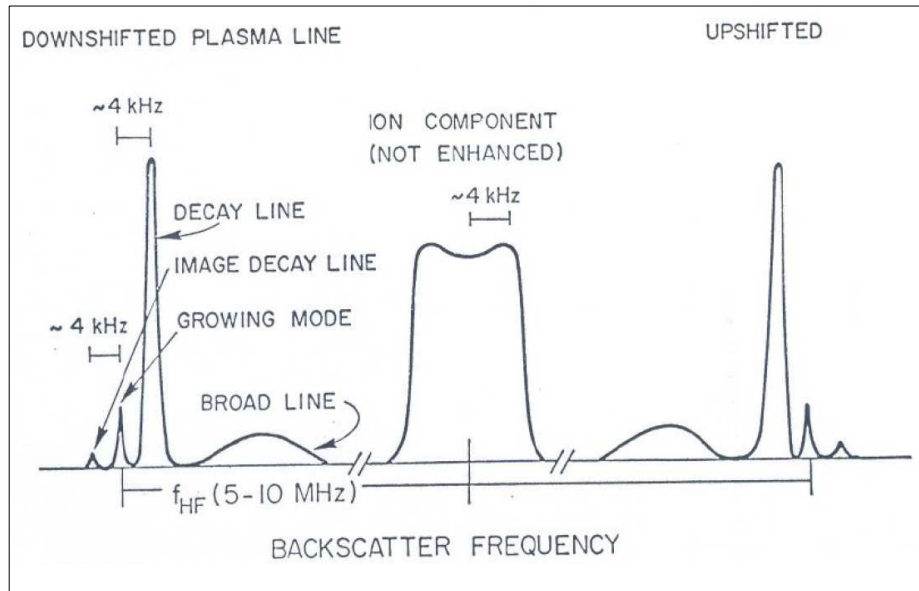
(b) Ion Line Measurement
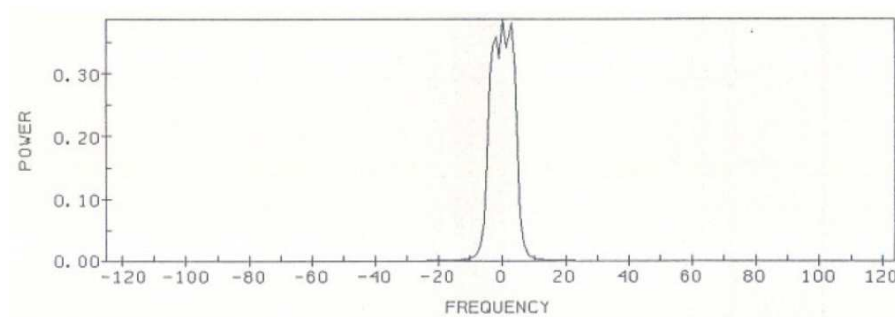
Figure 1-1: (a) ISR measurements can detect a variety of wave modes as ion and plasma lines. (b) Data shows the observation of both zero-frequency modes (at the center) and asymmetric (double-humped) ion enhancement.

2, specifically the detection of "forced ion acoustic waves". The investigation and data analyses are given in Section 3. Discussed in Section 4 are experimental results and analyses for the "simultaneous excitation of plasma density fluctuations and geomagnetic field fluctuations". A summary and conclusions are finally drawn in Section 5.

Chapter 2

Gakona Experiments

We brought materials to build mounting plates to install our All Sky Imaging System (ASIS) in the HAARP Optical Shelter. See photos in Figures 2-1 and 2-2 taken by Prof. Lee, showing the mounted ASIS and the setup of data acquisition system, respectively. Our VLF receiving system was deployed and operated by Rezy Pradipta outside HAARP. This was our first time to run ASIS at HAARP with filters for red, green, and sodium emissions. To be elaborated later, the sodium emissions were particularly useful to identify source mechanism(s) when MUIR detected "forced ion acoustic modes" in the presence of intense electrojet. Electrojet currents flow in the ionospheric E region at high latitudes, when intense energetic particles are precipitated along geomagnetic fields into the lower atmosphere from the magnetosphere. Because of high electric conductivity in the E region, closed current loops form like a circuit. Birkeland currents flowing along the geomagnetic fields are formed by upward moving thermal ionospheric ions, while electrojet currents flow eastward as the Hall currents.

My reported experiments include two parts: (a) forced ion acoustic waves excited by precipitated keV electrons and ions and, possibly, electrojet currents which occur naturally at Gakona, and (b) simultaneous excitation of large plasma sheets and geomagnetic field fluctuations by HAARP HF heater waves. They are presented separately as follows.

Figure 2-1: Mounting All Sky Imaging System (ASIS) in HAARP Optical Shelter for experiments

Figure 2-2: Setting up data acquisition system in HAARP Optical Shelter for experiments

2.1 Detections of Forced Ion Acoustic Waves

Although we conducted experiments for a week at Gakona, the optimum conditions for the excitation of forced ion acoustic waves by electrojet currents occurred on October 29, 6:20–6:30 UT. The evidence of electrojet currents flowing over Gakona was seen in the magnetometer data (see Figure 2-3) recorded on the ground at the HAARP site. Large geomagnetic field fluctuations are decoupled into three components denoted by $\delta \mathbf{B}_D$, $\delta \mathbf{B}_H$, and $\delta \mathbf{B}_Z$, measured along east-west, north-south, and vertical directions, respectively. Figure 2-4 shows these three components plotted on the same axis for comparison of their amplitude and temporal evolutions.

Because the vertical component ($\delta \mathbf{B}_Z$) is negative, it implies that the electrojet current flowed eastward. By contrast, the positive north-south component ($\delta \mathbf{B}_H$) indicates the electrojet also tilted to the north. As mentioned earlier, electrojet currents were built up as energetic parcels occurred, producing auroras. This expectation was indeed confirmed by a series of observations presented sequentially in the following. Green aurora began to appear shortly after 5:30 UT on October 29. The attached photo given in Figure 2-5 was taken by Rezy Pradipta, who operated our VLF receiving system outside HAARP to record electrojet current-radiated whistler waves.

Meanwhile, a series of airglows recorded by ASIS with a sodium filter during 6:20–6:30 UT on October 29, 2008, as displayed in Figure 2-6, vividly show the ribbon type of green aurora.

Ion line signals were detected by MUIR radar [displayed in Figure 2-7 in collaboration with Brenton Watkins] beamed along the Earth's magnetic field, when large geomagnetic field fluctuations were recorded in an extensive area, especially to the north of Gakona at that time [see a chain of geomagnetic field fluctuations data in Figure 2-8]. In other words, the HAARP magnetometer chain data showed that intense electrojet currents flowed in an extensive area to the north of Gakona and even extended to the south at Homer, Alaska.

It was noted that during the campaign there was a problem with the HAARP HF
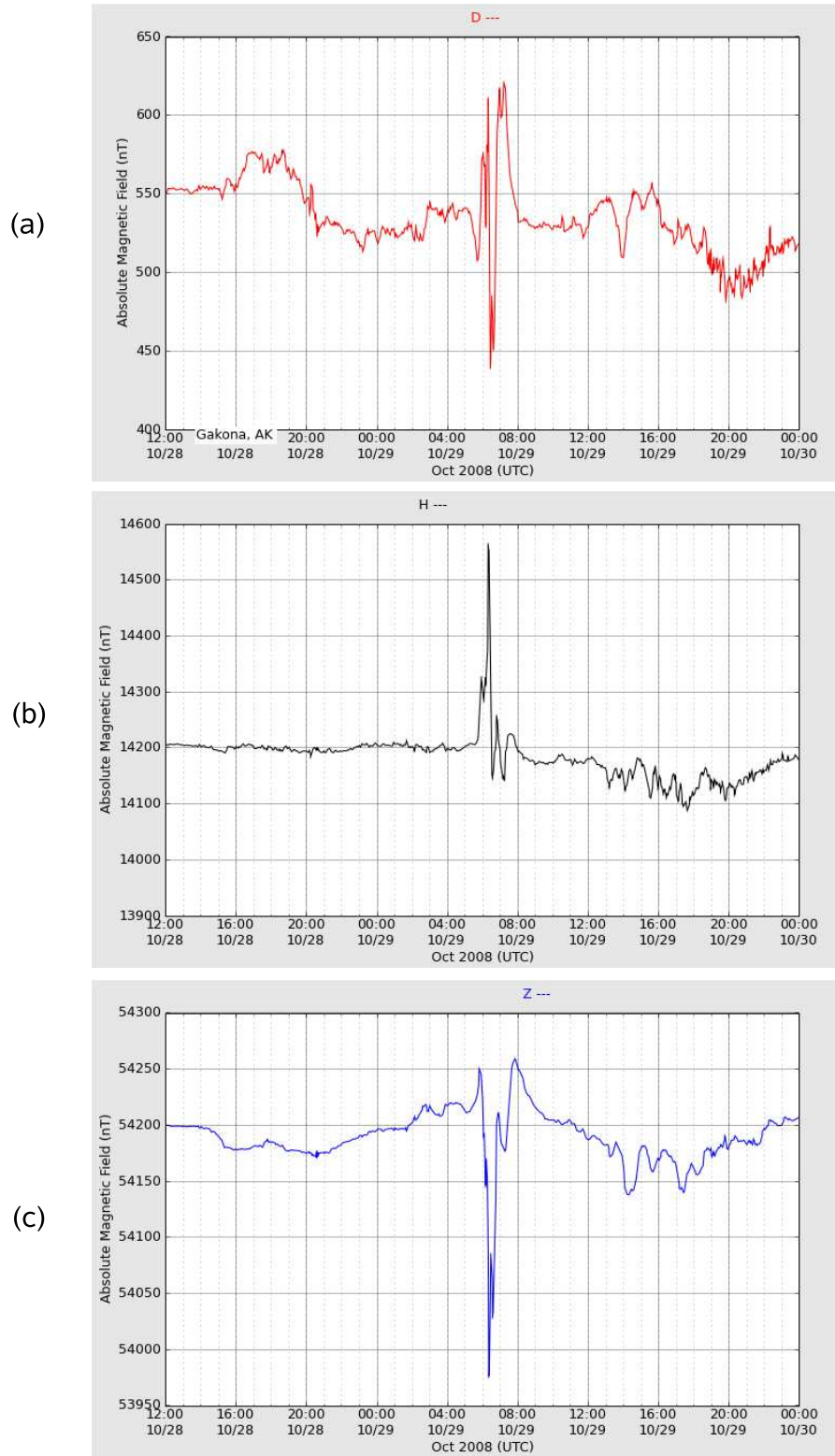
Figure 2-3: Geomagnetic field fluctuations ($\delta\mathbf{B}$) caused by electrojet currents, as seen in (a) East-west component ($\mathbf{B}_D$), (b) north-south component ($\mathbf{B}_H$), and (c) vertical component ($\mathbf{B}_Z$).
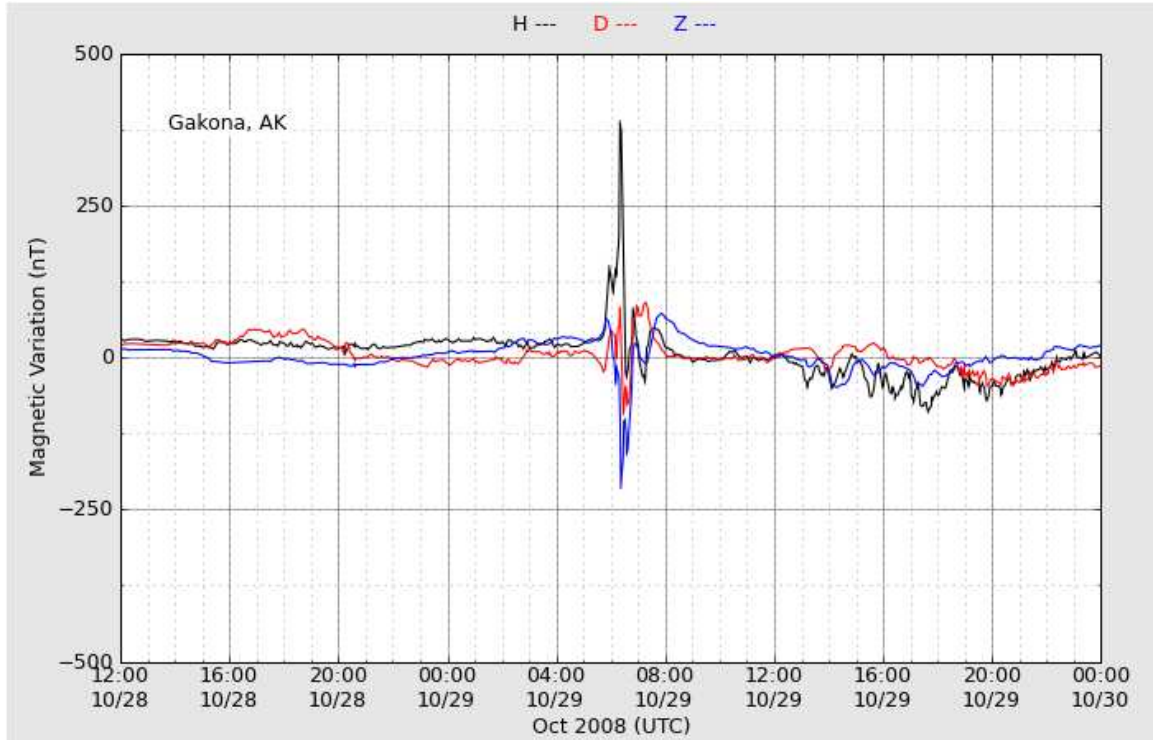
Figure 2-4: Geomagnetic field fluctuations ($\delta\mathbf{B}$) caused by electrojet currents: $\delta\mathbf{B}_D$ (red), $\delta\mathbf{B}_H$ (black), and $\delta\mathbf{B}_Z$ (blue) plotted on the same axis for comparison.

antenna arcing when transmitting certain frequencies (and possibly related to certain pointing directions). There was very broadband radiated power over HF through UHF frequencies, as seen on the HAARP spectrum monitor. It was particularly strong during a short period, but detectable at low levels at other times. Thus, for certain HF frequencies, the HAARP antenna was radiating and this radiation went up into the UHF band and was detected by MUIR [Ed Kennedy and Brenton Watkins, personal communications, 2008]. However, these ion line signals were detected by MUIR in the E region with ground clutter signals. During the ion line data analysis, the background noise level was subtracted. When the antenna arcs, the effect is to raise the overall receiver background noise level. The ion-line or plasma-line spectra (if present) would still be detectable; however the perceived power levels on the spectral components will appear lower because of the enhanced broad-band noise from the arcing antenna [Brenton Watkins, personal communications, 2008].

Furthermore, corresponding periodic occurrence of enhanced ion lines was detected

Figure 2-5: Rezy Pradipta operated our VLF receiving system outside HAARP, taking this photo of green aurora for comparison with ASIS data.
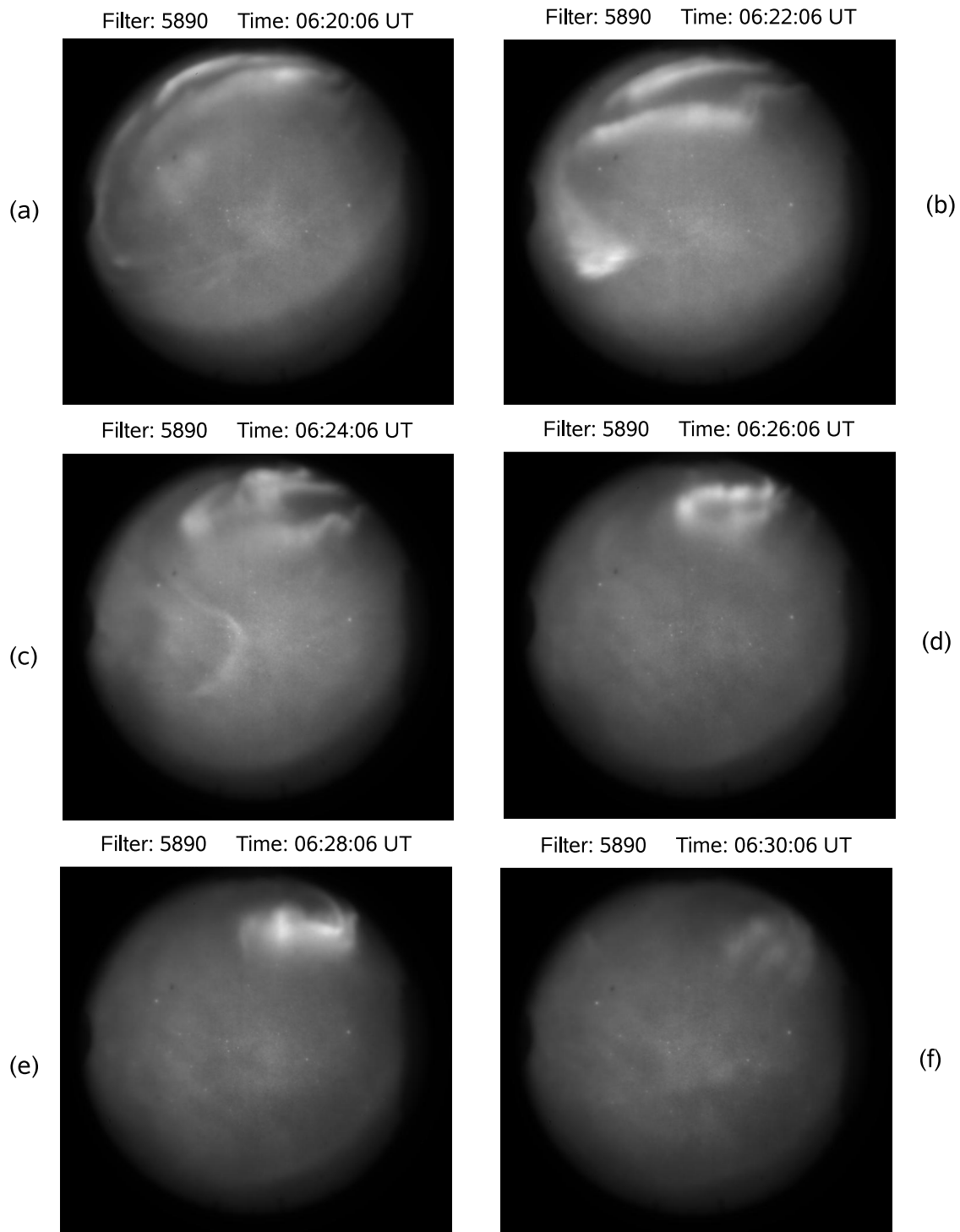
Figure 2-6: A series of airglows recorded by ASIS with a sodium filter during 6:20–6:30 UT on October 29, 2008 vividly show the ribbon type of green aurora brightening and fading gradually.
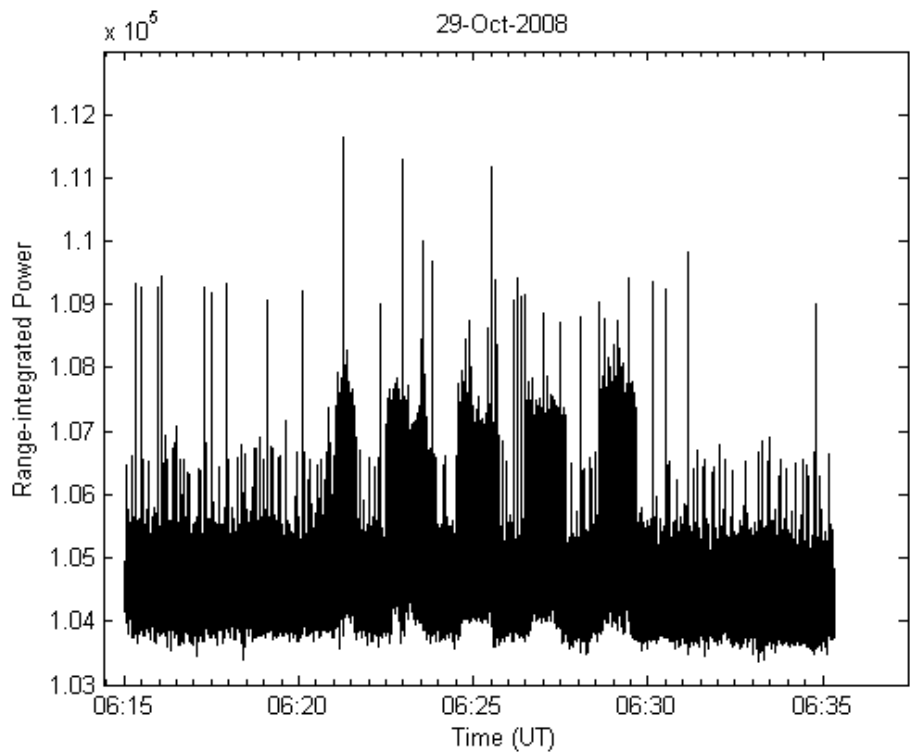
Figure 2-7: Ion line signals were detected by MUIR radar when green aurora was seen and intense airglow was recorded by ASIS with a sodium filter
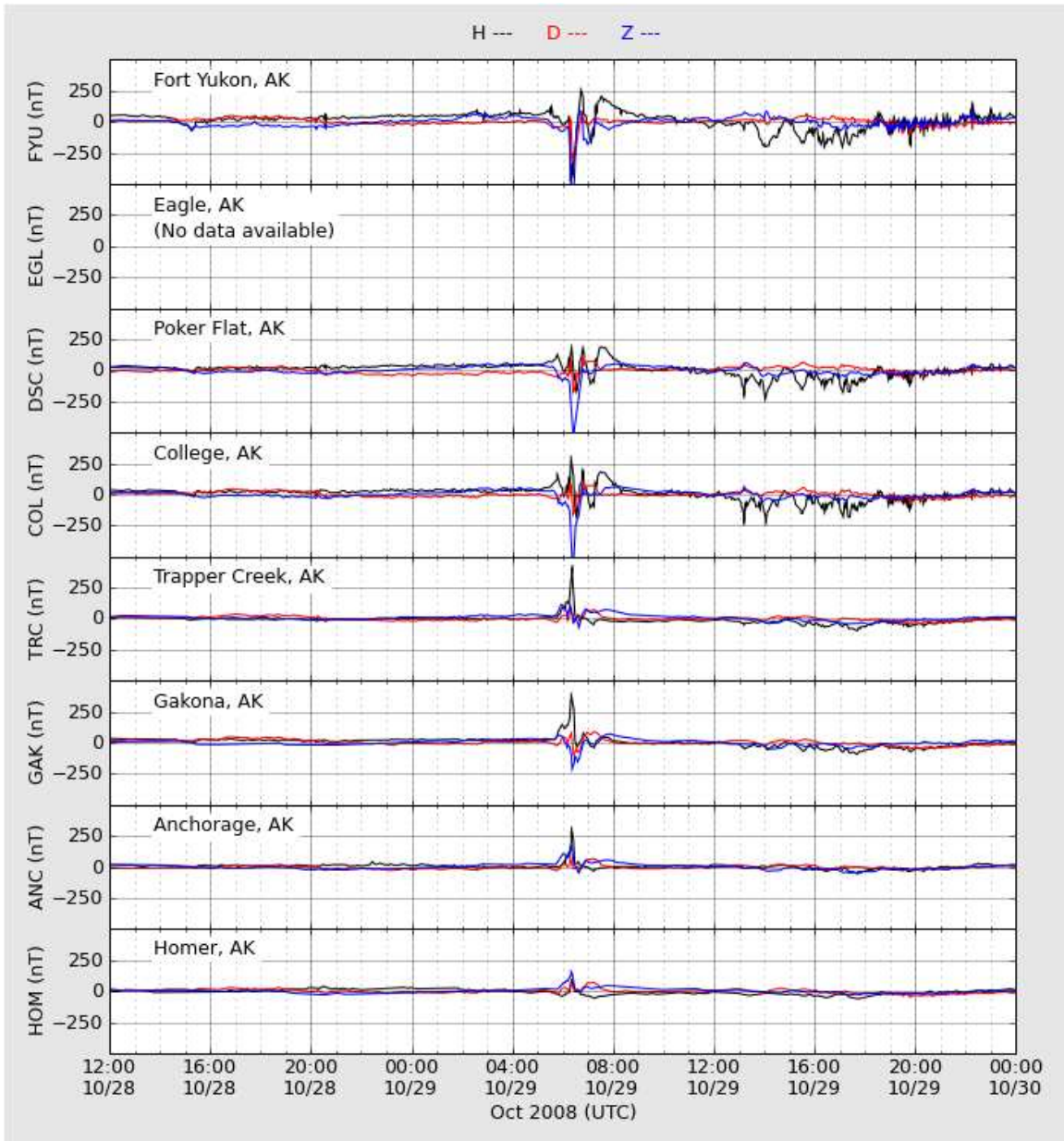
Figure 2-8: Large geomagnetic field fluctuations caused by intense electrojet currents flowing in an extensive area to the north of Gakona and even extending to the south at Homer in Alaska.
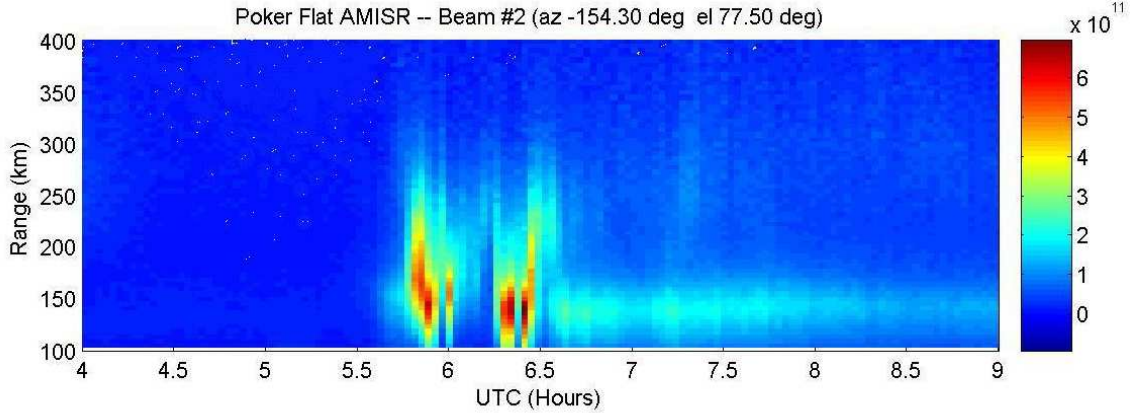
Figure 2-9: RTI plot from AMISR at Poker Flat, showing enhanced ion lines during intense geomagnetic field fluctuations [in collaboration with Mike Nicolls].

by AMISR (Advanced Modular Incoherent Scatter Radar) at Poker Flat (located to the north of Gakona) during about the same period of time (see the RTI plots given in Figure 2-9, in collaboration with Mike Nicolls). Note that the intensity of the RTI (range-time-intensity) plot is basically the integrated ion line spectra, although the unit for the color bar is m^{-3}. These integrated ion lines are associated with "forced ion acoustic modes" giving rise to large radar cross-sections. They clearly do not represent the real electron density profile, because this RTI plot was acquired in late evening when the background plasma density was rather low. Four (4) beams were operated by AMISR. We pick the RTI plot produced by the radar beam pointing along the Earth's magnetic field (called the "up-B direction") for comparison with the data (see Figure 2-7) which was recorded by MUIR, also beamed along the Earth's magnetic field.

It is thus unlikely, as further elaborated in Chapter 3, that they were coincidentally created by the MUIR interactions with HAARP arcing emissions. Rather, the occurrence of aurora, under favorable geoplasma conditions, led to intense ion density fluctuations.

24

Chapter 3

Excitation of Forced Ion Acoustic Waves

Based on the series of experiment data recorded by HAARP and our own diagnostic instruments, we present a plausible scenario and theory to suggest the excitation of forced ion acoustic waves during 6:20–6:30 UT on October 29, 2008 at Gakona as follows.

One can see from the MUIR (and AMISR) data, presented in Figures 2-7 and 2-9, that ion line signals were detected in several cycles of periods with durations of a few minutes from 6:20 to 6:30 (5:45 to 6:30) UT. This indicates that energetic particles were precipitated into the lower ionosphere along the geomagnetic field lines. This indication was supported by the appearance of concomitantly occurring green aurora, which is typically associated with precipitation of keV electrons. Furthermore, the process of keV electron and proton precipitation is confirmed by their detection using the satellite-borne particle detector on the DoD's DMSP satellite (see Figure 3-1).

DMSP F16 and F17 crossed the evening/afternoon MLT sectors of the northern auroral oval near 6:44 and 5:55 UT, respectively, when large magnetic field fluctuations were seen over Gakona (Poker Flat) during 6:20–6:30 (5:45–6:30) UT. These are the closest (not very ideal) measurements in space/time that DMSP can give. Intense particle precipitation was indeed detected by satellite-borne instruments. As a matter of fact, the anomalous ionization caused by energetic particle precipitation
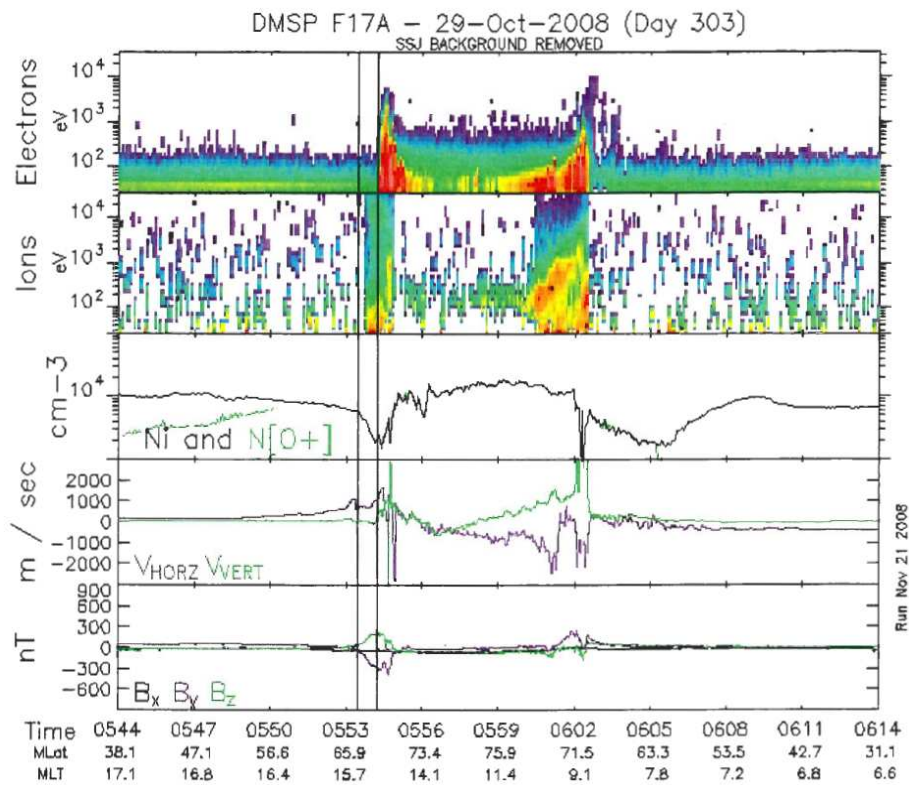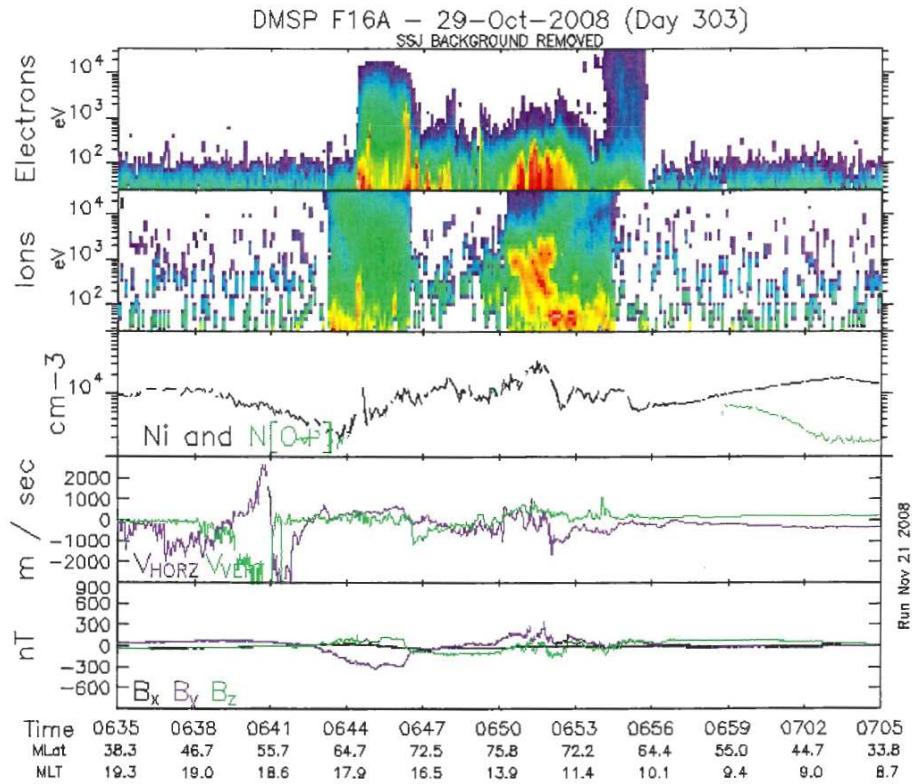
Figure 3-1: Detection of intense particle precipitation by DMSP F16 and F17 crossing the evening/afternoon MLT sectors of the northern auroral oval near 6:44 and 5:55 UT, respectively.

26

would be intense enough to cause significant ion density fluctuations for MUIR to detect.

However, in our prior Arecibo experiments, we found that precipitated keV electrons may generate plasma waves in the ionospheric F region [Labno et al., 2007] and E region [Pradipta et al., 2007]. I thus also examine plasma waves generated by streaming energetic electrons as an important source mechanism producing "forced ion acoustic waves," as the the consequence of keV electron and proton precipitation, along the lines of my M.S. thesis prospectus.

Although intense currents may excite ion acoustic waves [e.g., Kindel and Kennel, 1971] in the topside ionosphere, I will not consider the Birkeland currents (flowing downward along the geomagnetic field) and electrojet currents (i.e., Hall currents flowing eastward), subsequently induced by precipitated keV electrons stopped in the E region. The reasons are as follows. Birkeland currents are associated with ionospheric thermal particles; thus, they are rather weak. Although electrojet currents can be intense in the presence of keV electron and proton precipitation, they flow in the E region. It therefore requires rather large thresholds to excite ion acoustic modes in the E region in comparison to the conditions in the topside ionosphere. Moreover, electrojet currents flow eastward in the direction perpendicular to the geomagnetic field. Thus, the excited ion acoustic waves cannot be detected by MUIR radar beamed along the geomagnetic field.

3.1 Forced Ion Acoustic Waves Excited by Plasma Waves

This process is illustrated in Figure 3-2, showing that forced ion acoustic waves are produced as the "low-frequency beat waves" of streaming electron-excited electron plasma waves. The shaded areas in the wave-vector (k) space represent the distribution of the electron plasma waves. These electron plasma waves are expected to have small propagation angles initially. They may, however, be scattered by E re-
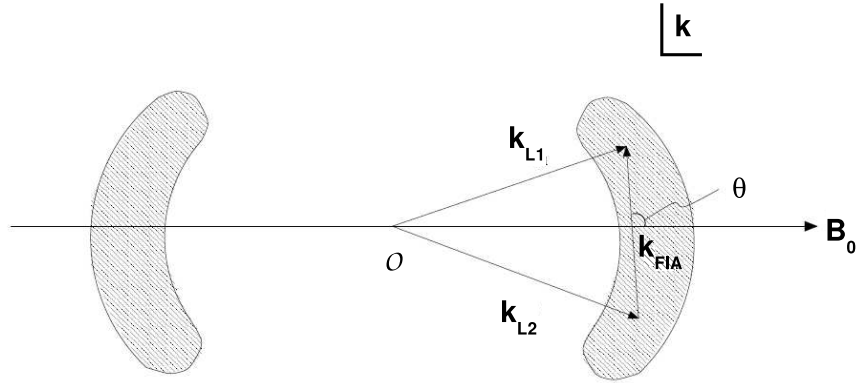
Figure 3-2: **k**-space diagram showing the generation of a "forced ion acoustic beat wave" from two electron plasma waves.

gion plasma density fluctuations (irregularities) associated with sporadic E layers or anomalous ionizations created by the precipitated keV electrons. Hence, they may propagate at larger angles across the geomagnetic field. As shown in Figure 3-2, two electron plasma waves L1 and L2 with wave vectors $\mathbf{k}_{L1}$ and $\mathbf{k}_{L2}$, respectively, can beat together to produce a low-frequency product FIA (forced ion acoustic) wave with a wave vector $\mathbf{k}_{FIA}$. The MUIR radar detects those FIA waves that have a scale length $(= 2\pi/\mathbf{k}_{FIA})$ equal to half the radar wavelength.

While this process can potentially generate "electron plasma modes and forced ion acoustic modes" for radar detection, MUIR radar unfortunately cannot detect them at HAARP in our experiments for two reasons. (1) MUIR was operated for detection of ion lines rather than plasma lines at night, when ionospheric plasma densities were rather low. (2) As mentioned previously, MUIR radar beam was injected along the geomagnetic field for the optimum detection of ion lines. "Forced ion acoustic waves" produced by electron plasma waves in this process will have wave vectors with pretty large angles with respect to the geomagnetic field, as delineated in Figure 3-2.

Thus, it is unlikely for MUIR radar to detect these "forced ion acoustic modes" as the beat products of streaming electron-induced electron plasma waves. Among the several processes enumerated, I conclude that the most likely one generating ion density fluctuations is keV electron precipitation, during the occurrence of green

aurora, leading to the detection of enhanced ion lines by MUIR (AMISR) at Gakona (Poker Flat) during 6:20–6:30 (5:45–6:30) UT on Oct. 29.

Chapter 4

Simultaneous Excitation of Large Plasma Sheets and Geomagnetic Field Fluctuations

4.1 Further Investigation of Thermal Filamentation

I have also conducted experiments at HAARP to investigate the generation of large plasma sheets by HF heater waves via thermal filamentation instabilities in the past 3 years for my B.S. thesis [2007], and a portion of my M.S. thesis research. These experiments are aimed at understanding the characteristic features of HF heater-induced large-scale field-aligned plasma structures. The generation of these large plasma sheets by HF heater has different configurations, depending upon the polarizations (i.e., O- or X-mode) of the heater waves as illustrated in Figure 4-1. These experiments were motivated by our earlier Arecibo experiments, observing heater-generated large plasma sheets [Lee et al., GRL, 1998]. These large plasma sheets (also known as artificial ionospheric ducts or waveguides) successfully supported NAU-launched 28.5 kHz whistler waves to propagate between Arecibo, Puerto Rico and Trelew, Argentina along the L = 1.35 magnetic flux tube [Mike Starks, Ph.D. Thesis, MIT, 1999;
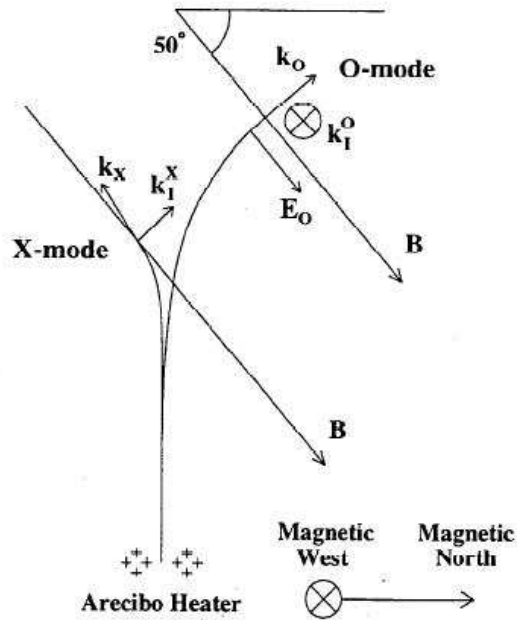
Figure 4-1: Ray paths of O- and X-mode heater waves in generating plasma sheets within and orthogonal to meridional plane, respectively.

Starks and Lee, 2000; Starks et al., 2001]. This work has important applications for radiation belt diagnostics and global VLF communications, using ground-based radio facilities.

The sheet-like nature of these thermal filamentation irregularities was seen as slanted stripes in the range-time-intensity (RTI) plots of Arecibo Incoherent Scatter Radar [see Figure 4-2]. By contrast, this feature can be investigated by different diagnostic instruments such as digisonde, magnetometers, All Sky Imager, and scintillation measurements together with SuperDARN radars at HAARP.

The generation of large plasma sheets at HAARP has been inferred in our previous HAARP experiments. The geometries of these O- and X-mode heater wave-excited plasma sheets are specifically shown in Figures 4-3 and 4-4 for the illustration of how they were diagnosed at HAARP. Ionosonde signals had no problem to be totally reflected and appear in ionograms in the presence of O-mode generated plasma sheets, which are parallel to the meridional plane [see Figure 4-3]. In contrast, because X-mode heater waves induce orthogonal plasma sheets, ionosonde signals transmitted
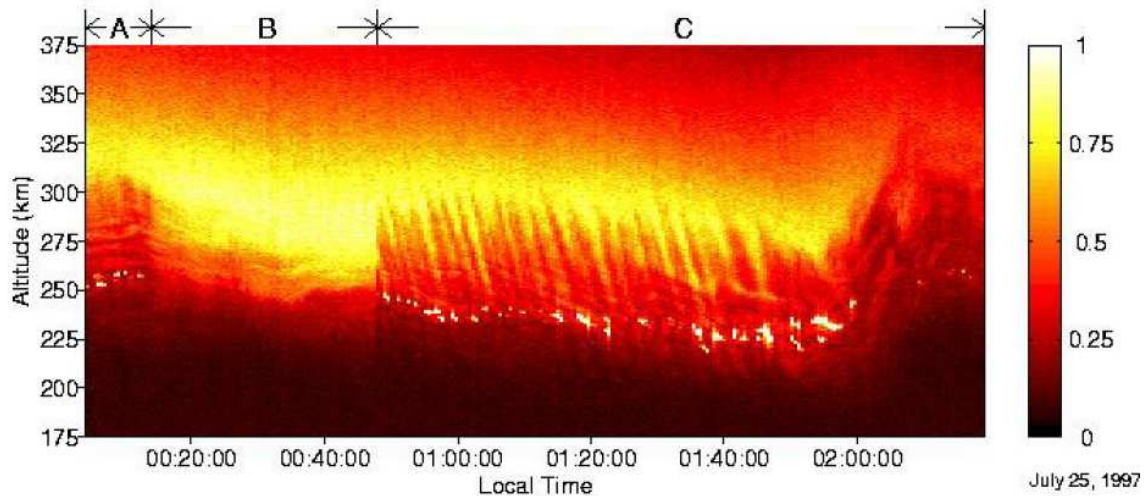
Figure 4-2: Arecibo RTI plot of HF heater-created large plasma sheets [Lee et al., 1998].

near the zenith will be guided to propagate away. They thus cannot be reflected to appear in the ionograms. However, when plasma blobs are present, the ionosonde signals transmitted at a large angle may still be bounced back from the remote plasma blobs [see Figure 4-4]. Preliminary results were published in a URSI GA Proceedings paper by Cohen et al. [2008].

4.2 Striking Feature of Thermal Filamentation Process

One striking feature of thermal filamentation instabilities, which I focus on to investigate in our HAARP experiments this past summer and fall, is the simultaneous excitation of sheet-like plasma density fluctuations (termed ionospheric density irregularities) and geomagnetic field fluctuations. This physical process is briefly described as follows with the aid of Figures 4-5 and 4-6 [adapted from Lee and Kuo, 1985].

HF heater waves can excite sheet-like filaments (see Figures 4-1, 4-3, and 4-4) near their reflection heights. They start from small perturbations in ionospheric density, resulting in a modulation of the plasma dielectric constant and the wave distribution. This process in turn increases the density perturbations in a positive feedback loop via

33
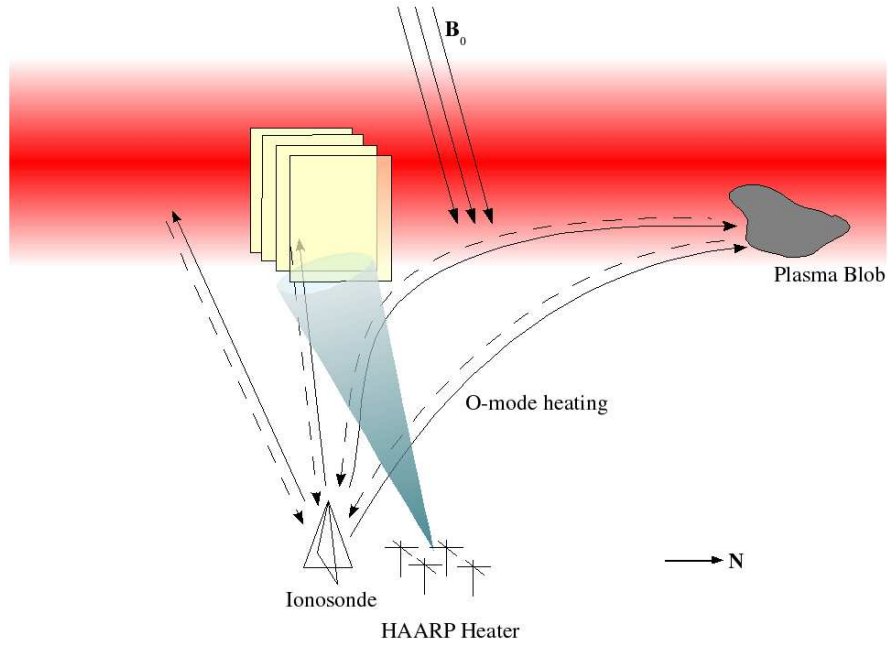
Figure 4-3: Configuration of O-mode induced plasma sheets parallel to meridional plane [Cohen et al., 2008].
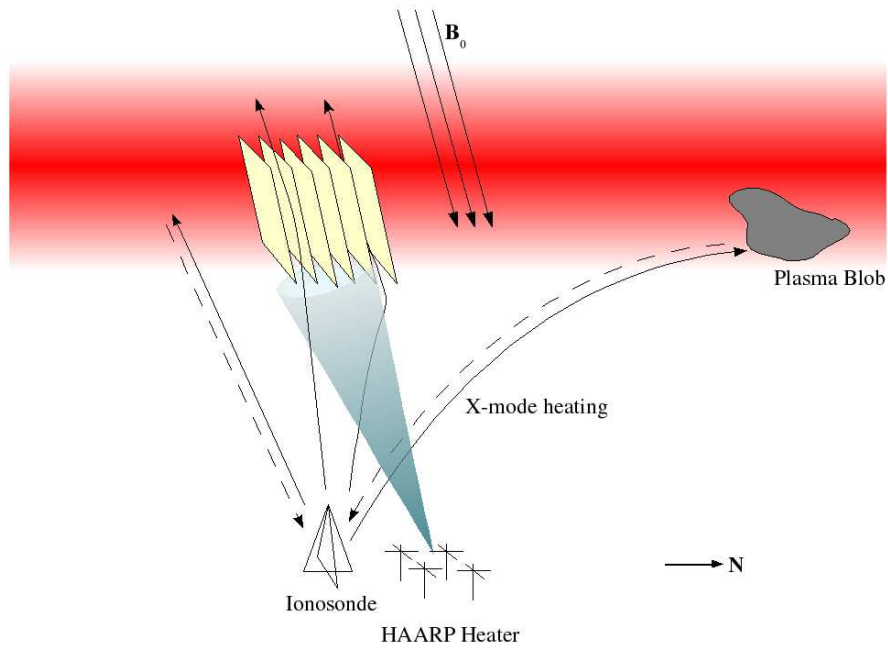
Figure 4-4: Configuration of X-mode induced plasma sheets orthogonal to meridional plane [Cohen et al., 2008].

thermal filamentation instabilities. The meaning of thermal refers to the differential joule heating as the dominant nonlinearity in exciting the instabilities [J.A. Cohen, B.S. Thesis, MIT, 2007]. The frequency of the perturbations is determined by the modulation frequency ω_m on the HF heater in the ON-OFF operations.

4.3 Mechanism Producing Plasma Density Fluctuations (δn) and Geomagnetic Field Fluctuations ($\delta \mathbf{B}$)

The differential joule heating, resulting from the interactions of HF heater waves and excited high frequency sidebands, yields a thermal pressure force ($\mathbf{f_T}$) on electrons, but not on ions, because only electrons can respond to high-frequency waves. The thermal pressure force ($\mathbf{f_T}$) is across the geomagnetic field, pointing to the direction along the y-axis ("Geomagnetic East" as specified in Figure 4-5 for O-mode heating) or along the x-axis ("Geomagnetic North and upward" as specified in Figure 4-6 for O-mode heating). Note that the thermal pressure force ($\mathbf{f_T}$) gives rise to electron density fluctuations (δn) with wave vectors ($\mathbf{k}$) pointing along the same direction [the y-axis (i.e. "Geomagnetic East") for O-mode heating or the x-axis (i.e. "Geomagnetic North and upward") for X-mode heating]. Hence, spatial variations of plasma density fluctuations (δn) are generated across (or within) the meridional plane for O-mode (or X-mode) heating. In other words, large plasma sheets are produced within (or orthogonal to) the meridional plane for the O-mode (X-mode) heating cases, as vividly illustrated in Figures 4-5 and 4-6, respectively.

The thermal pressure force ($\mathbf{f_T}$) leads to a $\mathbf{f_T} \times \mathbf{B_0}$ drift motion of electrons and, consequently, induces a net electron drift current along the x-axis (as shown in Figure 4-5 for O-mode heating) or along the y-axis (as shown in Figure 4-6 for X-mode heating). The direction of the current is perpendicular to both the background magnetic field $\mathbf{B_0}$ and the wave vector $\mathbf{k}$ of the excited plasma density irregularities. Therefore, magnetic field fluctuations ($\delta \mathbf{B}$) are excited along the background magnetic

δJ

δA

δE

f_T

Meridional
Plane

x'

x

y,y'

z'

z

$\delta\mathbf{B} \propto \nabla \times \delta\mathbf{A}$

$\mathbf{B_0}$

Geomagnetic
North

$\delta\mathbf{B_D}$

$\delta\mathbf{B_Z}$

Geomagnetic
East

HAARP
Magnetometer
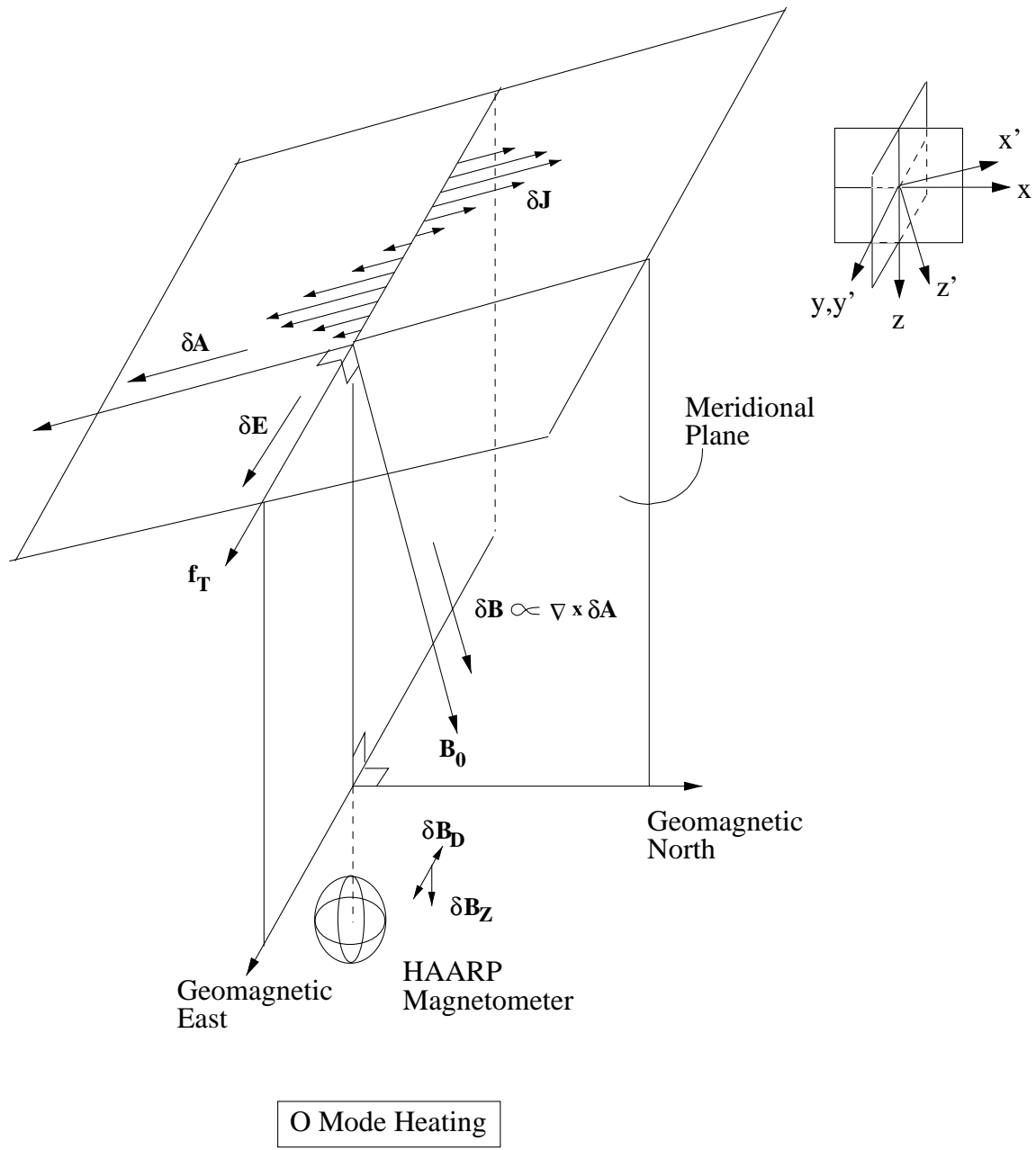
O Mode Heating

Figure 4-5: Simultaneous excitation of ionospheric plasma density fluctuations (δn) and geomagnetic field fluctuations ($\delta\mathbf{B}$) in O-mode heating experiments.

δJ

f_T

δE

Meridional
Plane

δA

$\delta\mathbf{B} \propto \nabla \times \delta\mathbf{A}$

$\mathbf{B_0}$

Geomagnetic
North

$\delta\mathbf{B_H}$

$\delta\mathbf{B_Z}$

HAARP
Magnetometer

Geomagnetic
East

x'

x

y,y'

z'

z
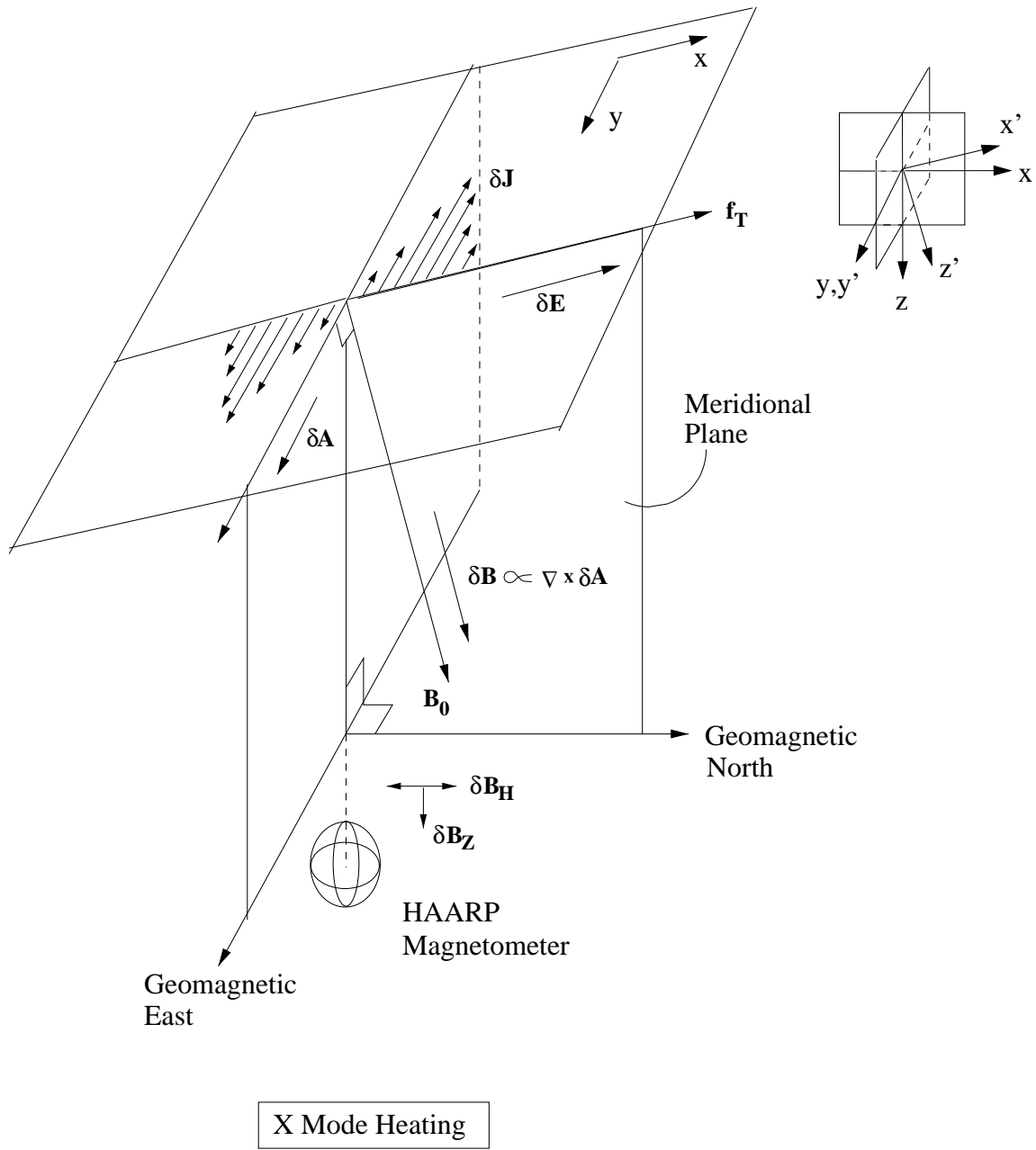
X Mode Heating

Figure 4-6: Simultaneous excitation of ionospheric plasma density fluctuations (δn) and geomagnetic field fluctuations ($\delta\mathbf{B}$) in X-mode heating experiments.

field ($\mathbf{B_0}$ designated as the z-axis) simultaneously with the density irregularities in the filamentation instability in both O- and X-mode heating processes. Note that the background geomagnetic field ($\mathbf{B_0}$) has a dip angle of 75.8°. Thus, it has three components [designated as $\mathbf{B}_{0D}$, $\mathbf{B}_{0H}$, and $\mathbf{B}_{0Z}$] along East-West, North-South, and downward directions, respectively. Therefore, the excited magnetic field fluctuations ($\delta \mathbf{B}$) also have the corresponding three components [designated as $\delta\mathbf{B}_D$, $\delta\mathbf{B}_H$, and $\delta\mathbf{B}_Z$].

The plasma density fluctuations (δn) and geomagnetic fluctuations ($\delta \mathbf{B}$) are related by

$$\frac{\delta n}{n_0} = \left[1 + \frac{(1 + \nu_e/\gamma)(k^2 c^2 + \gamma^2)}{\omega_{pe}^2} \right] \left(\frac{\delta \mathbf{B}}{\mathbf{B}_0} \right)$$

where $\gamma, \nu_e, \delta\mathbf{B}, \mathbf{B}_0, n_0, \omega_{pe},$ and ω_m are the instability growth rate, electron collision frequency, "in-situ" magnetic field fluctuations, electron plasma frequency, and the heater modulation frequency. In the derivation of the above ($\delta n/n_0$) versus ($\delta \mathbf{B}/\mathbf{B}_0$) relation, $\omega_m^2 << \gamma^2$ is assumed and justified. After I plug in the following relevant ionospheric parameters appropriate for the space plasma conditions at Gakona, I find that $(\delta n/n_0) \approx 2(\delta \mathbf{B}/\mathbf{B}_0)$.

$\gamma \approx 0.3\text{sec}^{-1}$ (T $\approx$ 3.5 sec) and $k \equiv 2\pi/\lambda$. We assume $k \approx 10^{-3}\text{m}^{-1}$, so $k^2 c^2 = 9 \times 10^{10}$. Also, $\nu_e \approx 500\text{sec}^{-1} >> \gamma$. Thus, $k^2 c^2/\omega_{pe}^2 \approx (9 \times 10^{10}/10^{14}) \approx 10^{-3}$ and $\nu_e/\gamma \cdot 10^{-3} \approx 500/0.3 \cdot 10^{-3} = 1.6$.

Note that the magnetometer data is recorded on the ground. Thus, using Ampere's law we can infer the "in-situ" magnetic field fluctuations in the source region where the HF heater excited both δn and $\delta \mathbf{B}$ simultaneously. This result carries an important application to ground-based diagnoses of ionospheric plasma turbulence [viz., $(\delta n/n_0)$], which can be verified by satellite scintillation measurements. Using a near-field $1/r^3$ magnetic field dependence for these long-wavelength magnetostatic modes, we begin with a measured $\delta\mathbf{B}/\mathbf{B}_0 \approx 2 \times 10^{-4}$ on the ground to infer a value in the ionosphere at 200 km of $\delta\mathbf{B}/\mathbf{B}_0 \approx 0.10$ or 10%. This implies $\delta n/n_0$ on the order of 10%, which is a very reasonable predicion and can be verified in future scintillation measurements.

4.4 Experimental Verifications

Based on my theoretical analyses and illustrations of the simultaneous excitations of δn and $\delta\mathbf{B}$ in Figures 4-5 and 4-6, aided by the delineation of sheet-like configurations in Figures 4-3 and 4-4, we can expect that $\delta\mathbf{B}_D$ and $\delta\mathbf{B}_Z$ [or $\delta\mathbf{B}_H$ and $\delta\mathbf{B}_Z$] will be highly correlated in O-mode [or X-mode] heating experiments. Our data analyses are presented in the following.

In analyzing magnetometer data, our approach has been to look for correlations between two of three orthogonal components of the earth's magnetic field guided by our theory. The earth's magnetic field is decomposed into a vertical component (denoted $\mathbf{B}_Z$), an east-west component (denoted $\mathbf{B}_D$), and a north-south component (denoted $\mathbf{B}_H$). The time series of these components is fit to a low-order polynomial (usually 5th order) and this fit is subtracted to remove long time scale drifts or natural phenomena. The resulting fluctuations $\delta\mathbf{B}_Z$, $\delta\mathbf{B}_D$, and $\delta\mathbf{B}_H$ are analyzed to yield three correlations: C_{DZ} (between the D and Z components), C_{HZ} (between the H and Z components), and C_{HD} (between the H and D components). Letting $i, j,$ and k denote any cyclic permutation of the magnetic field components, the correlations are computed using the formula:

$$C_{ij} = \frac{\delta\mathbf{B}_i \cdot \delta\mathbf{B}_j}{\|\delta\mathbf{B}_i\|\|\delta\mathbf{B}_j\|} \tag{4.1}$$

Finally, the background is subtracted from each expected signal to provide two net correlations, $\widetilde{C}_{DZ}$ and $\widetilde{C}_{HZ}$, according to the formula $\widetilde{C}_{ij} = C_{ij} - C_{ik} \cdot C_{jk}$.

According to our theory, O-mode heating should produce perturbations predominantly in the $\mathbf{B}_D$ and $\mathbf{B}_Z$ components, while X-mode heating should produce perturbations predominantly in the $\mathbf{B}_H$ and $\mathbf{B}_Z$ components. Note that due to the mechanism producing these magnetostatic fluctuations, they are not very sensitive to the magnetic dip angle. As a first pass in analyzing our heating experiments, we assume an ideal geometry where each heater wave produces precisely the expected predictions, and calculate a net correlation by subtracting a background correlation. For example, in O-mode heating we examine C_{DZ} correlation while considering both

C_{HZ} and C_{HD} correlations to be background (perhaps related to natural phenomena). This yields two net correlations for our O-mode and X-mode heating periods, $\widetilde{C}_{DZ}$ and $\widetilde{C}_{HZ}$, respectively.

In the following analysis, we note that a positive experimental outcome ($\widetilde{C}_{DZ} > \widetilde{C}_{HZ}$ for O-mode heating and $\widetilde{C}_{HZ} > \widetilde{C}_{DZ}$ for X-mode heating) provides very strong support for our theory, while an ambiguous or negative outcome does not indicate clearly the relationship between heating and magnetic field fluctuations.

There are a few subtleties in examining our heating experiments. One is that to further avoid background signals, it is preferable to conduct experiments when the magnetic field is relatively quiescent to small perturbations (on the order of 1 to 10 nT), or during periods where any perturbations have long time scales compared to the heating cycle being used. Another is that we are correlating using each On/Off cycle of the heater as a window (yielding one data point), so shorter heating cycles will tend to decrease the effect of natural phenomena on our data; however, this needs to be balanced with heating for long enough that the thermal filamentation instability can develop fully and reach saturation. This is not a problem at HAARP, where the estimated heater electric field strength at F-region altitudes far exceeds the thermal filamentation threshold. Our analysis also does not take into account any effects of ionospheric preconditioning, which could be complicated to model. Finally, we do not here account for the temporal growth and decay phases of thermal filamentation-induced magnetic field fluctuations, rather counting each data point concurrently with the heating mode being used at the time of acquisition. This is clearly increasingly valid with longer heating periods, which are not desirable as just discussed.

Taking all these considerations into account, it is preferable to heat with relatively short ON/OFF periods for a sustained period of time, and look at the statistical correlation of observed magnetic field fluctuations. We should note that the preponderance of data we have using this scheme utilizes effective X-mode heating, which can explain why the expected correlation for X-mode heating is seen more often in our datasets than that for O-mode heating. Presented here in relative order from most significant to less significant are four heating experiments we conducted where our magnetometer

Figure 4-7: October 23, 2008. The dots represent OFF periods and the crosses indicate periods of X mode heating. Black shaded areas show the duration of X mode heating.

analysis shows good agreement with our theory.

Figure 4-7 presents data from a heating experiment using only an X-mode ON/OFF cycle. The repeating heating scheme used was 2 min. ON, 1 min. OFF. The dots represent OFF periods and the crosses indicate periods of X mode heating. Black shaded areas show the duration of X mode heating. 30 correlation points are derived from the 45 minutes of magnetometer data. The results show that 73% of the time $\widetilde{C}_{HZ} > \widetilde{C}_{DZ}$, where we have looked for such correlations in OFF periods as well as ON (X-mode) periods. Looking only at X-mode heating periods, 87% show the expected correlation with only 13% yielding a null result ($\widetilde{C}_{DZ} > \widetilde{C}_{HZ}$). The OFF periods themselves show a closer though still favorable distribution, with 60% giving positive correlation and 40% giving a null result.

Figure 4-8 presents data from a heating experiment using only an X-mode ON/OFF cycle. The repeating heating scheme used was 2 min. ON, 1 min. OFF, 3 min. ON, 2 min. OFF, 1 min. ON, 1 min. OFF. Again, the dots represent OFF periods and the crosses indicate periods of X mode heating, while black shaded areas show the duration of X mode heating. 20 correlation points are derived from the 34 minutes of magnetometer data. The results show that 80% of the time $\widetilde{C}_{HZ} > \widetilde{C}_{DZ}$, where we

Figure 4-8: October 28, 2008. The dots represent OFF periods and the crosses indicate periods of X mode heating. Black shaded areas show the duration of X mode heating.

have looked for such correlations in OFF periods as well as ON (X-mode) periods. Looking only at X-mode heating periods, 90% show the expected correlation with only 10% yielding a null result ($\widetilde{C}_{DZ} > \widetilde{C}_{HZ}$). The OFF periods themselves show a closer though still very favorable distribution, with 70% giving positive correlation and 30% giving a null result.

Figure 4-9 presents data from a heating experiment using a mixed O-mode and X-mode heating cycle. The repeating heating scheme used was 2 min. O-mode, 2 min. X-mode, 2 min. O-mode, 2 min. OFF. The dots represent OFF periods and the crosses indicate periods of X mode heating. Circles represent periods of O mode heating. Black shaded areas show the duration of X mode heating, while blue shaded areas show the duration of O mode heating. 8 correlation points are derived from the 15 minutes of magnetometer data. The results show that 62% of the time $\widetilde{C}_{HZ} > \widetilde{C}_{DZ}$, where we have looked for such correlations in OFF periods as well as heating periods. Looking only at heating periods, 67% show the expected correlation with 33% yielding a null result ($\widetilde{C}_{DZ} > \widetilde{C}_{HZ}$). The off periods themselves show an equal distribution, with 50% giving positive correlation and 50% giving a null result.

42
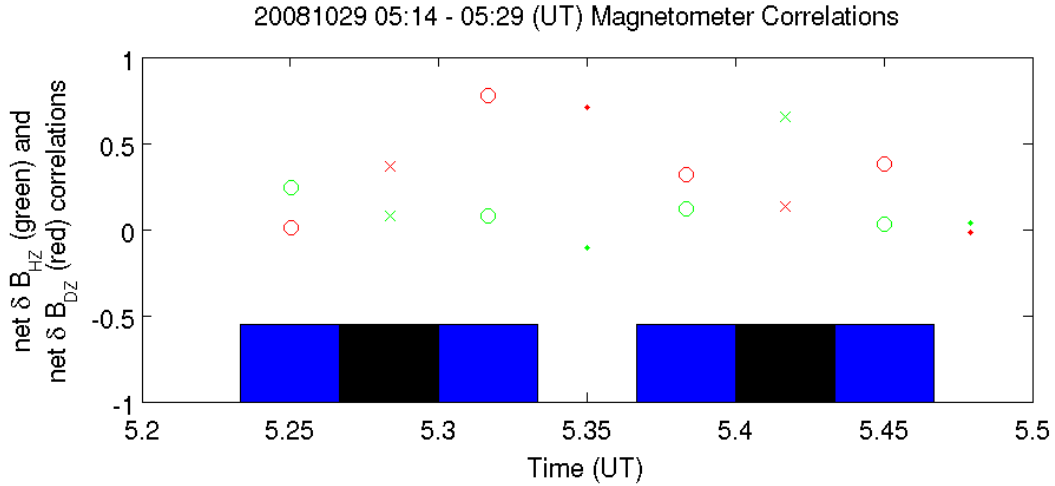
Figure 4-9: October 29, 2008. The dots represent OFF periods and the crosses indicate periods of X mode heating. Circles represent periods of O mode heating. Black shaded areas show the duration of X mode heating, while blue shaded areas show the duration of O mode heating.

Figure 4-10 presents data from a heating experiment using only an X-mode ON/OFF cycle. The repeating heating scheme used was 4 min. ON, 1 min. OFF. The dots represent OFF periods and the crosses indicate periods of X mode heating. Black shaded areas show the duration of X mode heating. 24 correlation points are derived from the 60 minutes of magnetometer data. The results show that 58% of the time $\widetilde{C}_{HZ} > \widetilde{C}_{DZ}$, where we have looked for such correlations in OFF periods as well as ON (X-mode) periods. Looking only at X-mode heating periods is slightly better; 67% show the expected correlation and 33% yield a null result $(\widetilde{C}_{DZ} > \widetilde{C}_{HZ})$. The OFF periods themselves show an equal distribution, with 50% giving positive correlation and 50% giving a null result.
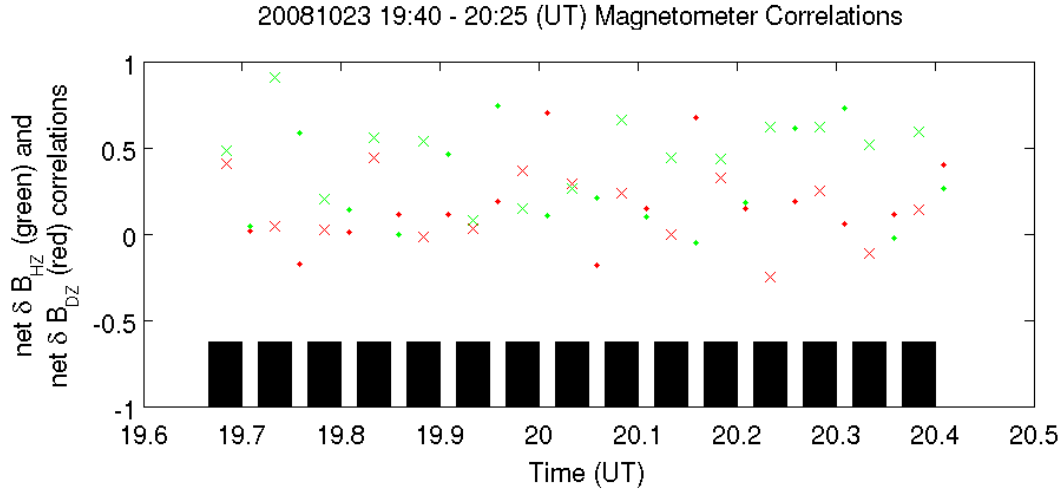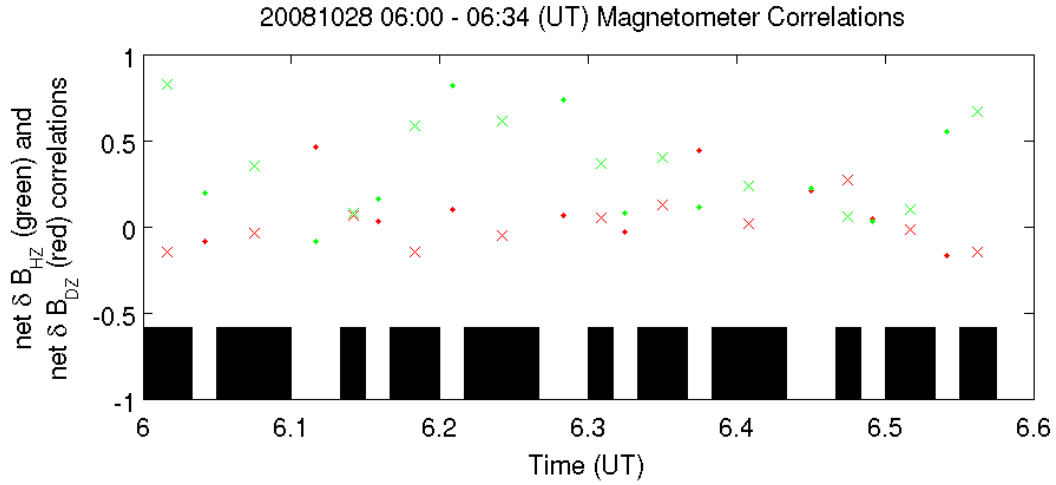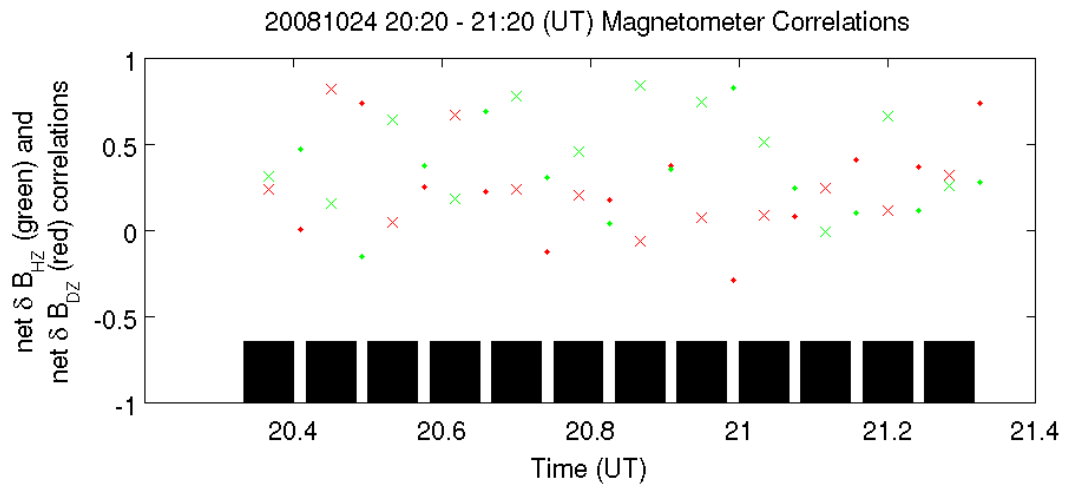
Figure 4-10: October 24, 2008. The dots represent OFF periods and the crosses indicate periods of X mode heating. Black shaded areas show the duration of X mode heating.

Chapter 5

Summary and Conclusions

Two research subjects have been investigated for my M.S. Thesis: (1) excitation of "forced ion acoustic waves," and (2) simultaneous excitation of plasma density fluctuations and geomagnetic field fluctuations. The experiments were conducted at Gakona, Alaska from summer 2007 to winter 2008, using DoD/NSF and our own optical and radio facilities aided by GPS satellites and AMISR radar at Poker Flat, Alaska. "Forced ion acoustic waves" detected by MUIR radar on Oct. 29 during 6:20–6:30 UT are suggested to arise from keV electron precipitation associated with the occurrence of green aurora. This work shows, for the first time, that MUIR radar is suitable for probing naturally occurring space plasma processes and is not limited to HF heater-induced effects. This would extend the usage of MUIR for the investigation of space weather together with AMISR radar at Poker Flat, to advance our knowledge in space plasma turbulence.

The research on "simultaneous excitation of plasma density fluctuations and geomagnetic field fluctuations" is an extension of my B.S. thesis research on thermal filamentation instability [J.A. Cohen, B.S. Thesis, MIT, 2007], which started in our summer Gakona experiments in 2005. The most significant progress was made in our 2007–2008 summer and winter experiments. Large plasma sheets (or sheet-like filaments) can be excited by HF O-mode and X-mode heater waves via thermal filamentation instability. The dominant nonlinearity is provided by the differential joule heating acting on electrons [J.A. Cohen, B.S. Thesis, MIT, 2007], which subsequently

gives rise to a cross-field thermal pressure force, to generate spatially varying plasma density fluctuations and electric current. This induces electric current is the source to create geomagnetic field fluctuations. This physical process provides a mechanism to induce both spatially varying plasma density fluctuations and geomagnetic field fluctuations. It is interesting to find, as discussed in detail in Section 4.3, that the fractional density fluctuations $(\delta n/n_0)$ are approximately equal to the fractional magnetic field fluctuations $(\delta \mathbf{B}/\mathbf{B}_0)$. This gives us the theoretical basis to use ground-based magnetometer measurements to infer the density fluctuations in space plasma turbulence. Such a remote sensing technique for probing the space plasma is much more effective and economic than using a beacon satellite.

Bibliography

[1] J.A. Cohen. Investigation of thermal filamentation instability over gakona, alaska. B.s. thesis, M.I.T., Department of Physics, June 2007. Co-supervisors: M.C. Lee and R.J. Temkin.

[2] J.A. Cohen, L.M. Burton, Rezy P., A. Labno, M.C. Lee, and S.P. Kuo. Generation of large plasma irregularity sheets by haarp heater via thermal filamentation instability. In *URSI-Ottawa*, URSI-Ottawa, URSI-Ottawa, July 22-26 2007a.

[3] J.A. Cohen, L.M. Burton, R. Pradipta, A. Labno, M.C. Lee, S.P. Kuo, B.J. Watkins, and S. Oyama. Ionospheric ducts and plasma waves induced by hf heater over gakona. In *Proceedings of URSI General Assembly*, URSI General Assembly, Chicago, U.S.A., 2008.

[4] J.A. Cohen, L.M. Burton, Rezy Pradipta, A. Labno, M.C. Lee, and S.P. Kuo. Diagnoses of large plasma sheets excited by heater via thermal filamentation instability at gakona, alaska. In *Proceedings of "Turbulent Mixing and Beyond" Workshop*, "Turbulent Mixing and Beyond" Workshop, Trieste, Italy, August 18-26 2007b.

[5] J.A. Cohen, R. Pradipta, D. Mabius, and M.C. Lee. Polar aeronomy radio science (pars) summer experiments for the investigation of pdi-excited langmuir wave turbulence, July 28–August 3 2007c.

[6] J.M. Kindel and C.F. Kennel. Topside current instabilities. *Journal of Geophysical Research*, 76:3055, 1971.

[7] S.P. Kuo and M.C. Lee. Cascade spectrum of hf enhanced plasma lines generated in hf heating experiments. *Journal of Geophysical Research*, 110:A01309, 2005.

[8] A. Labno, R. Pradipta, M. C. Lee, M. P. Sulzer, L. M. Burton, J. A. Cohen, S. P. Kuo, and D. L. Rokusek. Cascade spectrum of hf enhanced plasma lines generated in hf heating experiments. *Journal of Geophysical Research*, 112:A03306, doi:10.1029/2006JA0112089, 2007.

[9] M.C. Lee and S.P. Kuo. Simultaneous excitation of large-scale geomagnetic field fluctuations and plasma density irregularities by powerful radio waves. *Radio Science*, 20:539, 1985.

[10] M.C. Lee, D.T. Moriarty, R. Riddolls, S.M. Murphy, C. Yoo, K.D. Vilece, R.F. Duraski, D. Beals, and R.R. Parker. Plasma chamber and active space experiments to investigate ionospheric plasma turbulence. In *Proceedings of the International Beacon Satellite Symposium*, URSI General Assembly, Aberystwyth, Wales, July 11–15 1994.

[11] M.C. Lee, R.J. Riddolls, W.J. Burke, M.P. Sulzer, S.P. Kuo, and E.M.C. Klien. Generation of large sheet-like ionospheric plasma irregularities at arecibo. *Geophysical Research Letters*, 25:3067, 1998.

[12] R. Pradipta, A. Labno, M. C. Lee, W. J. Burke, M. P. Sulzer, J. A. Cohen, L. M. Burton, S. P. Kuo, and D. L. Rokusek. Electron precipitation from the inner radiation belt above arecibo. *Geophys. Res. Lett.*, 34:L08101, doi:10.1029/2007GL029807, 2007.

[13] M.J. Starks. *Measurement of the conjugate propagation of VLF waves by matched filter and application to ionospheric diagnosis.* Ph.D. dissertation, M.I.T., (EECS), 1999. Supervisor: M.C. Lee.

[14] M.J. Starks and M.C. Lee. Matched filtering for the measurements of conjugately ducted vlf transmissions. *Radio Science*, 35:351, 2000.

[15] M.J. Starks, M.C. Lee, and P. Jastrzebski. Interhemispheric propagation of vlf transmissions in the presence of ionospheric hf heating. *Journal of Geophysical Research*, 106:5579, 2001.

Appendix A

Overview of Magnetometer Analysis Environment

I have developed a suite of MATLAB functions and scripts that provide an environment to facilitate the analysis of magnetometer data, in particular from the Geophysical Institute Magnetometer Array (GIMA). In view is the collection of data during HF heating experiments which can be used for statistical studies. The data is provided in netCDF format (network Common Data Format), a well-known data file format for distribution of scientific data. Executable binaries exist for extracting the data into various environments. Here all the MATLAB scripts use a GIMA function *rdnetcdf.m* (which depends on the binary file *netcdf* from the netCDF toolbox) to import the data initially [see http://magnet.gi.alaska.edu/scripts/rdnetcdf.m and http://mexcdf.sourceforge.net, respectively]. Once the data has been imported, it provides a number of parameters describing the instrument location and operation, as well as arrays containing the time series data of Earth magnetic field measurements along three orthogonal directions, as explained in Section 4.4.

There are two key principles behind the Magnetometer Analysis Environment. The first is modularity of data import, analysis, and display functions. This allows functions to be grouped according to their respective purposes and facilitates clean development of data processing algorithms and routines that can create production-ready graphs and plots. The goal was to have a MATLAB command-line accessible

environment with a limited set of human-readable commands that could extract, process, and display any required time interval of magnetometer data. The second principle is the use of Experimental Definition Files (EDFs) to collect and store relevant information for each experiment that is carried out during a campaign. Each EDF file contains all the necessary information to identify an experiment's essential parameters, locate and load desired data for the appropriate time period, and return a MATLAB object containing these parameters and data. Subsequent calls to Magnetometer Analysis Environment functions then operate on and update this object (for reference hereafter called an Experiment Object). This is extremely handy for organizing and working on a potentially large set of experiment blocks (and, e.g., if the HF heater scheme is changed several times for different objectives during a several hour experiment, each scheme can be defined in a separate EDF file). Appendix A describes the organization and contents of Magnetometer Analysis Environment functions and EDF files, including an example session, while a listing of relevant source code is provided in Appendix B.

A.1   Magnetometer Data Analysis Example

Here is a brief example of a session where one experiment is loaded and the magnetometer time series are processed to find correlations between magnetic field fluctuations and HF heater operation. A description of the EDF files themselves and an overview of the MATLAB file organization follow. The MATLAB output has been condensed for brevity (indicated by ellipses), and the example demonstrates: listing available experiments, loading an experiment from March 2006, performing polynomial fitting and subtraction, and finally calculating correlations. Comments are added and denoted by a % character. These commands will produce a fully annotated plot of magnetometer correlations with HF heating.

```
>> cd research/campaigns
>> edf     % list all available files
1. EDF_Gak_Oct2008_15_1.m
```

2. EDF_Gak_Oct2008_15_2.m

3. EDF_Gak_Oct2008_15_3.m

4. EDF_Gak_Oct2008_15_4.m

5. EDF_Gak_Oct2008_15_5.m

6. EDF_Gak_Oct2008_4_16_1.m

7. EDF_Gak_Oct2008_4_16_2p1.m

8. EDF_Gak_Oct2008_4_16_2p2.m

9. EDF_Gak_Oct2008_4_16_2p3.m

10. EDF_Gak_Oct2008_4_16_3p2.m

11. EDF_Gak_Oct2008_4_16_4p2.m

12. EDF_Gak_Oct2008_4_16_5p1.m

13. EDF_Gak_Oct2008_4_16_5p3.m

14. EDF_Gak_Oct2008_4_16_6.m

15. EDF_march2006_TF_0326.m

16. EDF_pars2005_TF_0821.m

17. EDF_pars2006_TF_0809.m

18. EDF_pars2008_TF_0728_1.m

19. EDF_pars2008_TF_0729.m

20. EDF_pars2008_TF_0731.m

```
>> exp = edf(15);     % load EDF file number 15
>> exp = exp.getData(exp, exp.instruments(1));     % load data
You requested data from magnetometer
starttime = 2006     3     26     6     54     0
endtime = 2006     3     26     8     18     26
...
filename = 2006Gakona087-06.nc
reading /mit/jacohen/research/2006Gakona087-06.nc NetCDF file
filename = 2006Gakona087-07.nc
reading /mit/jacohen/research/2006Gakona087-07.nc NetCDF file
filename = 2006Gakona087-08.nc
```

```
reading /mit/jacohen/research/2006Gakona087-08.nc NetCDF file
3242 to 8307 of 10800      % Data from hours 6--8 UT have been imported.
                           % Here the appropriate interval is extracted.
>> exp = EDFanalysis_magnetometer_polyfit(exp);  % polynomial
                                                 % fitting and subtraction
>> EDFanalysis_magnetometer_corr(exp)            % find correlations
Array time() runs from 6.9003 to 8.3072          % experiment duration
% what follows is a summary of the times() array, where each line
% shows the start and end times of a Heating or Off period.
...partitions->1 of 3, times([i,i+1]): 6.9 to 7.1583
...partitions->2 of 3, times([i,i+1]): 7.1583 to 8.0667
...partitions->3 of 3, times([i,i+1]): 8.0667 to 8.3072
>>
```

A.2   EDF File Contents

All EDF files are .m files and have filenames that begin with the string "EDF_". The last part of the string can be any valid string, and should be chosen to be descriptive of the particular experiment or campaign. An example (template) EDF file is given in Appendix B.2. The parameters of an experiment are stored in EDF files and are described below. Here parameters are grouped in headings which correspond to components of an Experiment Object.

- Experiment Object (EO)

    - *location* (typically a string identifying the experiment facility)

    - *dateStart* (a MATLAB date vector with UT year, hour, day, hour, minute, and sec)

    - *dateEnd* (also MATLAB date vector)

    - *day1UT* (UT day from 1 to 366 of the day given in dateStart)

52

- *daysUT* (a vector of UT days which the experiment period spans)

- *times* (a vector specifying the start and end times of each Heating or Off period. EO.times and EO.modes are typically filled by a subroutine heatingscheme() in the current EDF file.)

- *modes* (a cell array of strings specifying the heater operation during each period defined by two adjacent entries in EO.times)

- *exclude* (contains three vectors with lists of points to remove from $\mathbf{B}_H$, $\mathbf{B}_D$, and $\mathbf{B}_Z$ time series in case of data contamination)

    * *h* (a vector of decimal start and stop times bracketing each interval to exclude)

    * *d* (for example, EO.exclude.d = [[7.5215,7.5225];[7.9146,7.9158]])

    * *z* (to include all data, set h, d, and z to [[];[]])

- *getData* (a pointer to the function getData defined in the current EDF file, which is called to load the actual data into the EO object)

- *data* (contains all sets of data, both imported raw data and intermediate or final processed data. Functions will read data from here and, depending on their design, add new data arrays with processed data.)

- *instruments* (an array of instrument objects, each with the following components:)

    - *name* (instrument name, e.g. 'magnetometer')

    - *class* (for cases where multiple instruments of the same type are used concurrently, perhaps in different locations. Examples are ISR, magnetometer, digisonde.)

    - *subclass* (for example, "Narod fluxgate" type of magnetometer)

    - *location* (where the instrument is located and operated)

- *dataDir* (the filesystem directory where netCDF (.nc) files for the current experiment are located)

- *plotDir* (the default filesystem directory where output such as plots and graphs can automatically be saved)

A.3 MATLAB Organization

The MATLAB functions which load, operate on, and display data are stored in a folder named EDF, which is accessible on the MATLAB path. Functions are named with a leading "EDF" followed by a string denoting their function ("data" for working with raw data, "analysis" for data processing, or "plot" for display functions). Then comes the name of the intended instrument whose data each function is designed to work with, offset by underscore characters. Finally, a string identifying the specific task of each function is appended. (For functions intended only by other functions in the Magnetometer Analysis Environment and not by users, a final underscore is added, such as (2) below.) The current set of such functions available includes:

1. EDFanalysis_magnetometer_corr

2. EDFanalysis_magnetometer_exclude_

3. EDFanalysis_magnetometer_FTI

4. EDFanalysis_magnetometer_polyfit

5. EDFdata_magnetometer

6. EDFplot_magnetometer_psd

7. EDFplot_magnetometer_heatingscheme

The single data function (5) extracts time series data given an experiment described by an EDF file. A common first task is to fit the series to a polynomial and subtract to remove offsets and long-time variations (when looking for fluctuations due

to short-time HF heating periods). This is accomplished with function (4). Other analysis functions include (3), constructing an FTI (Frequency-Time-Intensity) plot for each magnetic field component to examine the time evolution of the data in Fourier space. The correlation function (1) was used predominantly for this thesis. The two output functions (6) and (7, respectively, plot the signal power as a function of time and print a formatted output of the HF heating time periods and heating modes for a particular experiment.

The EDF files are all stored in a folder named CAMPAIGNS, though in principle they could be saved anywhere. After, changing the current working directory to this folder, all available EDF files can be listed and enumerated by running the function *edf.m*. This function also takes an optional numerical argument specifying one particular EDF file; if provided, that file is read and an Experiment Object (EO) is returned. For easy repetition of a task requiring a common set of commands in the Magnetometer Analysis Environment, these commands can be collected in a script, for example *edfTemp.m*. Such a script is demonstrated in Appendix B.3.1. In this particular case the file number is provided and the function:

1. loads the Experiment Object into a global variable called "exp"

2. runs a polynomial fitting routine on the three Earth magnetic field components and then calls the a function to compute correlations of magnetic field fluctuations with HF heater operation

3. constructs a string (global variable savePlots) of system commands which can be executed with the native MATLAB *system* function. This will automatically save the correlation plots just produced in (2) in .pdf, .png, and .epsc (color eps) formats.

4. clears the MATLAB command window and displays formatted output describing the experiment just loaded.

Batch operations to process a large number of EDF files can also be implemented, for example in Appendix B.3.2. It should be noted that the current organization of

the Magnetometer Analysis Environment is function-based. This provides a versatile and extendable environment, but it would also be possible to move the framework from being function-based to being class-based. In implementing classes for the Magnetometer Analysis Environment, it would be prudent to use MATLAB handle classes instead of value classes for Experiment Objects (EOs), so that data is not copied each time the object is operated on or sent to function.

Appendix B

Magnetometer Analysis Environment: Source Code Listing

B.1 Main EDF program loop

```
function exp = edf(varargin)
%% Experiment Definition File
% This file defines an experiment and the data
% that is available for it.

files = dir('EDF_*.m');

if (nargin==1)
    if (isnumeric(varargin{1}))
        eval(['exp = ' files(varargin{1}).name(1:end-2) '();']);    10
    end
else
    for i=1:length(files)
        disp([num2str(i) '. ' files(i).name])
    end
end
```

B.2 Example: EDF template

```
function experiment = EDF_EXPERIMENT-TAG() %% name the experiment here
%% Experiment Definition File
```

```matlab
% This file defines an experiment and the data
% that is available for it.

%% Setup
% Here basic information about the experiment is
% defined.   Note that each of these files is well
% suited for one block of a larger experimental
% campaign.

experiment.location  = 'Gakona';
experiment.dateStart = datenum([2006 03 26 06 54 00]);
experiment.dateEnd   = datenum([2006 03 26 08 18 26]);
experiment.day1UT    = 87;
experiment.daysUT    = daysUT(experiment);

% here the function heatingScheme will need to be updated case by case
[experiment.times, experiment.modes] = heatingScheme();

% optionally, vectors of time domains to exclude can remove spikes, etc.
experiment.exclude.h = [[7.5215,7.5225];[7.9146,7.9158]];
experiment.exclude.d = [[7.5215,7.5225];[7.9146,7.9158]];
experiment.exclude.z = [[7.5215,7.5225];[7.9146,7.9158]];

experiment.getData = @getData;

%% Instrument 1
instrument.name = 'magnetometer';
instrument.class = 'magnetometer';
instrument.subclass = 'Narod fluxgate';
instrument.location = 'HAARP';
instrument.dataDir = '/mit/jacohen/research';
instrument.plotDir = '/mit/jacohen/research';
experiment.instruments(1) = instrument;

experiment.data = {};

%
% FUNCTION daysUT
%

function days = daysUT(experiment)
%% return vector of UT days for experiment
span = experiment.dateEnd−experiment.dateStart;
days = experiment.day1UT + [0 1:floor(span) ];

%
% FUNCTION heatingScheme
%

function [times, modes] = heatingScheme()
```

58

```
% examples of heating scheme definitions ... note that
% length(modes) = length(times) - 1 always.
%                                                                              60
% —— 1. explicitly enter a heater sequence ——
% times=[6+54/60,7+9/60+30/3600,8+4/60,8+18/60+26/3600];
% modes={'O','Off','X'};
%
% —— 2. use a loop to construct the array times() ——
% in this case, the heater operates on/off CW O mode in periods
% of 5 minutes, beginning from 04:00 UT and ending at 04:20 UT
% for i=0:4
%    times(i+1)=4+i/12;
% end                                                                          70
% modes = {'O','Off','O','Off'};
%
% —— 3. another loop to construct times() ——
% for i=0:6
%    times(i+1)=2+i/6;
% end
% modes = {'O','Off','X','Off','O','Off'};
%
% —— 4. explicitly enter a heater sequence ——
% here a manual sequence is entered since the heating periods are              80
% irregular.  For clarity, on/off cycles are specified one at a time
% times=[10,10+45/60+30/3600,11+5/60+30/3600];
% modes={'O','Off'};
% times(end+1:end+2)=[11+8/60+5/3600,11+9/60];
% modes(end+1:end+2)={'X','Off'};
% times(end+1:end+2)=[11+35/60,11+35/60+30/3600];
% modes(end+1:end+2)={'X','Off'};
% times(end+1:end+2)=[11+50/60,11+50/60+30/3600];
% modes(end+1:end+2)={'X','Off'};
% times(end+1:end+2)=[12,12+30/3600];                                          90
% modes(end+1:end+2)={'O','Off'};
% times(end+1:end+2)=[12+10/60];
% modes(end+1:end+2)={'O'};
%
% —— 5. a complex on/off sequence that repeats ——
% durations() gives the lengths of each period (either on or off). Here
% the entire sequence repeats twice, beginning at 04:20 UT. The modes()
% array is then given explicitly.
% durations=[3,4,3,10,3,5,5,7];
% durations=[durations, durations];                                            100
% times = 4+20/60;
% for i=1:length(durations)
%      times(end+1)=times(end)+durations(i)/60;
% end
% modes={'O','Off','O','Off','O','Off','O','Off','O','Off','O','Off','O','Off','O','Off'};
%
% —— 6. a given sequence repeated until end of experiment ——
% the arrays steps and stepmodes are iterated through and the arrays
% times and modes are correspondingly filled until endtime is reached
% 5 min. TF sequence                                                          110
```

```matlab
% steps = [5/60,5/60,5/60,5/60];
% stepmodes = {'O','Off','X','Off'};
% times = 4+15/60;
% endtime = 4+34/60+30/3600; %final endtime is 5+30/60, but this is first period
% modes = {};
% i=1;
% modN=length(steps);
% counter=1+mod(i-1,modN);
% while( times(end) + steps(counter) <= endtime )
%     times(end+1) = times(end) + steps(counter);
%     modes(end+1) = stepmodes(counter);
%     i=i+1;
%     counter=1+mod(i-1,modN);
% end
% Note: this is needed to close times in the case of an early HF turnoff
% if( times(end) + steps(counter) > endtime)
%     times(end+1) = endtime;
%     modes(end+1) = stepmodes(counter);
% end


%
% ——— 7. Two repeated sequences automatically generated ———
% here an extra 'Off' is inserted between sequences to be complete
% 5 min. TF sequence
% steps = [5/60,5/60,5/60,5/60];
% stepmodes = {'O','Off','X','Off'};
% times = 4+15/60;
% endtime = 4+34/60+30/3600; %final endtime is 5+30/60, but this is first period
% modes = {};
% i=1;
% modN=length(steps);
% counter=1+mod(i-1,modN);
% while( times(end) + steps(counter) <= endtime )
%     times(end+1) = times(end) + steps(counter);
%     modes(end+1) = stepmodes(counter);
%     i=i+1;
%     counter=1+mod(i-1,modN);
% end
% now we have to check if a leftover time period needs to be labelled
% if( (times(end) + steps(counter) > endtime) )
%     if(~strcmp('Off',stepmodes(counter)))
%         times(end+1) = endtime;
%         modes(end+1) = stepmodes(counter);
%         capEnd = 1;
%     else
%         capEnd = 2;
%     end
% else
%     capEnd = 0;
% end
%
% 10 min. TF sequence
% steps = [10/60,10/60,10/60,10/60];
% stepmodes = {'O','X','O','Off'};
```

```matlab
% times2 = 4+35/60;
% endtime = 5+30/60;
% if( capEnd>0 )
%     modes(end+1)='Off';
% else
%     times = times(1:end-1);
% end
% i=1;
% modN=length(steps);
% counter=1+mod(i-1,modN);
% while( times2(end) + steps(counter) <= endtime )
%     times2(end+1) = times2(end) + steps(counter);
%     modes(end+1) = stepmodes(counter);
%     i=i+1;
%     counter=1+mod(i-1,modN);
% end
% if( times2(end) + steps(counter) > endtime)
%     times2(end+1) = endtime;
%     modes(end+1) = stepmodes(counter);
% end
%
% times = [times,times2];


%
% FUNCTION getData
%


function experimentWData = getData(exp,instrument)
%% Retrieve data for a particular instrument
experimentWData = exp;
disp(['You requested data from ' instrument.name])
switch instrument.name
    case 'magnetometer'
        data = EDFdata_magnetometer(exp.dateStart,exp.dateEnd, ...
            exp.daysUT,instrument.dataDir);
        data.instrument = instrument.name;
    try data = EDFanalysis_magnetometer_exclude_(data,exp.exclude);
    catch disp('there was an error excluding points...');
    end
end
experimentWData.data{end+1} = data;
experimentWData.dataRef.(instrument.name) = data;
```

B.3   Scripts and Batch Processing

B.3.1   Command Script

```matlab
function edfTemp(n)

global exp
global plotDir
global tempDir
global savePlots

tempDir='/mit/jacohen/matlab/research/EDF/Temp';

%% This code gets magnetometer data and the relevant plotDir

exp = edf(n);
exp = exp.getData(exp,exp.instruments(1));
exp = EDFanalysis_magnetometer_polyfit(exp);
EDFanalysis_magnetometer_corr(exp); % this is now the second corr() routine, no 'out' argument

plotDir = exp.instruments(1).plotDir;

dateStart = datevec2strcell( datevec( exp.dateStart ) );
dateEnd = datevec2strcell( datevec( exp.dateEnd ) );
filename = ['stem_' dateStart{1} dateStart{2} dateStart{3} '_HDZ_' dateStart{4} ...
    dateStart{5} '_' dateEnd{4} dateEnd{5}];
% saveas(gcf,[plotDir '/' filename '.png']) % good for sharing and basic reports
% saveas(gcf,[plotDir '/' filename '.pdf']) % nice full-page centered svg output
% saveas(gcf,[plotDir '/' filename '.epsc']) % color eps figure tightly bounded
pngStr = ['saveas(gcf,''' plotDir '/' filename '.png'');'];
pdfStr = ['saveas(gcf,''' plotDir '/' filename '.pdf'');'];
epscStr = ['saveas(gcf,''' plotDir '/' filename '.epsc'');'];
savePlots = ['set(gcf,''PaperPosition'',[1,1,6.5,9]);' pngStr pdfStr epscStr];

%% Display the results of loading EDF experiment 'n'

clc
disp('exp:')
disp(exp)
disp(['tempDir: ' tempDir])
disp(['plotDir: ' plotDir])
disp(['filename base: ' filename])
disp(['savePlots: ' savePlots])

function cell = datevec2strcell(mat)
cell = {};
for i=1:length(mat)
    if(i==1 || mat(i)>9)
        cell{i} = num2str(mat(i));
    else
        cell{i} = ['0' num2str(mat(i))];
    end
end
end
```

B.3.2 Example: Batch Processing

```
function out2 = edfTempBatch()

global savePlots

nums = [1:17,19,20];

%out2 = []

for i=1:length(nums)
    edfTemp(nums(i)); % for second corr() routine, no 'out' argument          10
    % out2(1+(i-1)*2:i*2,1:4) = out;
    eval(savePlots)
end
```

B.4   EDF Functions

B.4.1   Data Retrieval

```
function data = EDFdata_magnetometer(dateStart,dateEnd,daysUT,dataDir)
%% Returns the compiled .nc files for a given experiment block
% This is meant to be called from a .edf file-generated object

evalin('base','warning off MATLAB:sprintf:InputForPercentSIsNotOfClassChar');

starttime = datevec(dateStart)
endtime   = datevec(dateEnd)
starttime = starttime(4:6)
endtime   = endtime(4:6)                                                        10
daysUT

time=[];
hcomp=[];
dcomp=[];
zcomp=[];
header={};

for day = 1:length(daysUT)
%% This block gets necessary .nc files for each UT day                          20
    year = datestr(dateStart,'yyyy');
    filenamebase = fileNameBase([year 'Gakona'],daysUT(day));

    if day==1; innerstarttime = starttime(1);
    else innerstarttime = 0;
    end
    if day==length(daysUT); innerendtime = endtime;
    else innerendtime = [23 59];
```

```matlab
    end

    for i=innerstarttime(1):innerendtime(1)
        %% Read in .nc files and populate data arrays
        if(i<10) hour=['0' num2str(i)];
        else hour=num2str(i);
        end

        filename=[filenamebase hour '.nc']
        [innertime, innerhcomp, innerdcomp, innerzcomp, innerheader] = ...
            rdnetcdf([dataDir '/' filename],'False',1);
        innertime=(innertime-daysUT(1))*24;

        time  = [time;innertime];
        hcomp = [hcomp;innerhcomp];
        dcomp = [dcomp;innerdcomp];
        zcomp = [zcomp;innerzcomp];
        header= [header,innerheader];
    end

end

index1 = find(time>=starttime(1)+starttime(2)/60+starttime(3)/3600);
index1 = index1(1);
%index1 = ceil(1+starttime(2)*60+starttime(3));
index2 = find(time<=endtime(1)+endtime(2)/60+endtime(3)/3600);
index2 = index2(end);
%index2 = floor((59 - endtime(2))*60 + endtime(3));
%if(index2 > length(time)-1 )
%      index2 = length(time)-1;
%end
disp([num2str(index1) ' to ' num2str(index2) ' of ' num2str(length(time))])
time=time(index1:index2); % changed from end-index2
hcomp=hcomp(index1:index2);
dcomp=dcomp(index1:index2);
zcomp=zcomp(index1:index2);

data.time = time;
data.hcomp = hcomp;
data.dcomp = dcomp;
data.zcomp = zcomp;

evalin('base','warning on MATLAB:sprintf:InputForPercentSIsNotOfClassChar');

function filenamebase = fileNameBase(base,day)
%% construct a filename base string

if(day<10) filenamebase=[base '00' num2str(day) '-' ];
elseif(day<100) filenamebase=[base '0' num2str(day) '-' ];
else filenamebase=[base num2str(day) '-' ];
end
```

B.4.2 Analysis

Correlation

```
function EDFplot_magnetometer_corr(experiment)


time = experiment.dataRef.magnetometer_psd.time;
hcomp2 = experiment.dataRef.magnetometer_psd.hcomp;
dcomp2 = experiment.dataRef.magnetometer_psd.dcomp;
zcomp2 = experiment.dataRef.magnetometer_psd.zcomp;

times = experiment.times;
modes = experiment.modes;                                               10

dateString=[ datestr(experiment.dateStart,'yyyymmddHHMM') ...
             datestr(experiment.dateEnd,'HHMM') ];

partitions = [];
numIntervals = length(times)−1;

disp(['Array time() runs from ' num2str(time(1)) ' to ' num2str(time(end))]);
for i = 1:numIntervals
    disp([ 'EDFplot_magnetometer_corr()→partitions→' num2str(i) ' of ' ...     20
           num2str(numIntervals) ', times([i,i+1]): ' num2str(times(i)) ' to ' ...
           num2str(times(i+1)) ]); % For debugging purposes
    indices = find( time >= times(i) );
    partitions(i) = indices(1);
end

indices = find( time <= times(end) );

if length(indices) > 1
    partitions(end+1) = indices(end) + 1;                               30
else
    partitions(end+1) = length(time) + 1;
end

corrTime = [];
corrHD = [];
corrHZ = [];
corrDZ = [];

for i=1:numIntervals                                                    40

    subTime = time( partitions(i):partitions(i+1)−1 );
    corrTime(i) = mean(subTime);

    subHcomp = hcomp2( partitions(i):partitions(i+1)−1 );
    subDcomp = dcomp2( partitions(i):partitions(i+1)−1 );
    subZcomp = zcomp2( partitions(i):partitions(i+1)−1 );
```

```matlab
    hNorm = sqrt(sum(subHcomp.^2));
    dNorm = sqrt(sum(subDcomp.^2));                                          50
    zNorm = sqrt(sum(subZcomp.^2));

    corrHD(i) = sum(subHcomp.*subDcomp) / (hNorm * dNorm);
    corrHZ(i) = sum(subHcomp.*subZcomp) / (hNorm * zNorm);
    corrDZ(i) = sum(subDcomp.*subZcomp) / (dNorm * zNorm);

end


% Here we look at the heating modes to adjust the plot marker              60

indicesX = strmatch('X',modes,'exact');
indicesO = strmatch('O',modes,'exact');
indicesOff = strmatch('Off',modes,'exact');

adjCorrO = corrDZ − corrHZ.*corrHD; % correlation assuming ideal O-mode heating
adjCorrX = corrHZ − corrDZ.*corrHD; % correlation assuming ideal X-mode heating

figure(1)
                                                                          70
clf

subplot(3,1,1)

s1X=stem(corrTime(indicesX),adjCorrO(indicesX),'rx');
hold on
s1O=stem(corrTime(indicesO),adjCorrO(indicesO),'ro');
s1Off=stem(corrTime(indicesOff),adjCorrO(indicesOff),'r.');
ylabel('net \delta B_{DZ} correlation')
                                                                          80
title([ dateString(1:8) ' ' dateString(9:10) ':' dateString(11:12) ' - ' ...
      dateString(13:14) ':' dateString(15:16) ' (UT) Magnetometer Correlations' ])

subplot(3,1,2)

s2X=stem(corrTime(indicesX),adjCorrX(indicesX),'gx');
hold on
s2O=stem(corrTime(indicesO),adjCorrX(indicesO),'go');
s2Off=stem(corrTime(indicesOff),adjCorrX(indicesOff),'g.');
ylabel('net \delta B_{HZ} correlation')                                   90


subplot(3,1,3)

plot(corrTime(indicesX),adjCorrO(indicesX),'rx');
hold on
plot(corrTime(indicesO),adjCorrO(indicesO),'ro');
plot(corrTime(indicesOff),adjCorrO(indicesOff),'r.');

plot(corrTime(indicesX),adjCorrX(indicesX),'gx');                         100
plot(corrTime(indicesO),adjCorrX(indicesO),'go');
plot(corrTime(indicesOff),adjCorrX(indicesOff),'g.');
```

66

```
ylabel({'net \delta B_{HZ} (green) and','net \delta B_{DZ} (red) correlations'})

xlabel('Time (UT)')

% draw in heating modes

minDataY = min([adjCorrX, adjCorrO]);
minAxesY = min(get(gca,'YLim'));                                              110
rectH = minDataY − minAxesY;
for i=1:length(indicesX)
    rectXY = times([ indicesX(i), indicesX(i)+1 ]);
    rectW = rectXY(2) − rectXY(1);
    rectangle('Position',[rectXY(1) minAxesY rectW rectH/2],'FaceColor','black');
end
for i=1:length(indicesO)
    rectXY = times([ indicesO(i), indicesO(i)+1 ]);
    rectangle('Position',[rectXY(1) minAxesY rectW rectH/2],'FaceColor','blue');
end                                                                           120
```

Exclude

```
function data = EDFanalysis_magnetometer_exclude_(dataOld,exclude)

    data = dataOld;
    for i = 1:size(exclude.h,1)
        interval=exclude.h(i,:);
        index1 = find(data.time>interval(1));
        index2 = find(data.time<interval(2));
        p = polyfit( data.time([index1(1) index2(end)]), ...
                data.hcomp([index1(1) index2(end)]) , 1 );
        data.hcomp(index1(1):index2(end)) = ...                               10
            polyval( p , data.time(index1(1):index2(end)) );
    end
    for i = 1:size(exclude.d,1)
        interval=exclude.d(i,:);
        index1 = find(data.time>interval(1));
        index2 = find(data.time<interval(2));
        p = polyfit( data.time([index1(1) index2(end)]), ...
                data.dcomp([index1(1) index2(end)]) , 1 );
        data.dcomp(index1(1):index2(end)) = ...
            polyval( p , data.time(index1(1):index2(end)) );                  20
    end
    for i = 1:size(exclude.z,1)
        interval=exclude.z(i,:);
        index1 = find(data.time>interval(1));
        index2 = find(data.time<interval(2));
        p = polyfit( data.time([index1(1) index2(end)]), ...
                data.zcomp([index1(1) index2(end)]) , 1 );
        data.zcomp(index1(1):index2(end)) = ...
            polyval( p , data.time(index1(1):index2(end)) );
```

```
                 end                                                          30
```

FTI

```
function experimentNew =MagPolyFFT2008(experiment)

%
% Get the relevant magnetometer data
%

close all

experiment = EDFanalysis_magnetometer_polyfit(experiment);
                                                                              10
time = experiment.dataRef.('magnetometer_polyfit').time;
hcomp = experiment.dataRef.('magnetometer_polyfit').hcomp;
dcomp = experiment.dataRef.('magnetometer_polyfit').dcomp;
zcomp = experiment.dataRef.('magnetometer_polyfit').zcomp;

nowDir=pwd;
%cd(experiment.instruments(1).plotDir);

%
% prepare the data for FFT, populate the frequency array              20
% note: length is scaled to the next power of 2
%

Fs = 1;                    % Sampling frequency
L = length(time);                 % Length of signal
NFFT = 2^nextpow2(L); % Next power of 2 from length of y
f = Fs/2*linspace(0,1,NFFT/2);
Hcomp = fft(hcomp,NFFT)/L;
Dcomp = fft(dcomp,NFFT)/L;
Zcomp = fft(zcomp,NFFT)/L;                                           30

%
% plot H component single-sided FFT spectrum...take this out?
%

figure

subplot(3,1,1)
plot(f,2*abs(Hcomp(1:NFFT/2)))
title('Single-Sided Amplitude Spectrum of h(t)')               40
xlabel('Frequency (Hz)')
ylabel('|H(f)|')

subplot(3,1,2)
plot(f,2*abs(Dcomp(1:NFFT/2)))
title('Single-Sided Amplitude Spectrum of d(t)')
xlabel('Frequency (Hz)')
ylabel('|D(f)|')
```

```matlab
subplot(3,1,3)
plot(f,2*abs(Zcomp(1:NFFT/2)))
title('Single-Sided Amplitude Spectrum of z(t)')
xlabel('Frequency (Hz)')
ylabel('|Z(f)|')

%
% prepare dateString and pad arrays if ending second is 60
%

dateString=[datestr(experiment.dateStart,'yyyymmddHHMM') datestr(experiment.dateEnd,'HHMM')];

%
% FIGURE 1 - H, D, and Z components
%

figure(1)
cd(experiment.instruments(1).plotDir);
filename=[dateString(1:8) '_Mag_HDZ_' dateString(9:12) '_' dateString(13:16)];
%print('-dpng',[filename '.png']);
% print('-depsc',[filename '.eps']);
%saveas(gcf,[filename '.eps'])
%saveas(gcf,[filename '.png'])
%saveas(gcf,[filename '.pdf'])
cd(nowDir);

%
% perform FFT on each component - parameters are
% time, comp, intPeriod, Fs, L, dateString, compStr, dcCutoff, doPrint
%

integrateFFT(time,hcomp,60,Fs,L,dateString,'B_H',1,'NoPrint');
integrateFFT(time,dcomp,60,Fs,L,dateString,'B_D',1,'NoPrint');
integrateFFT(time,zcomp,60,Fs,L,dateString,'B_Z',1,'NoPrint');

%
% FIGURE 7 - compile all plots
%

cd(experiment.instruments(1).plotDir);
filename=[dateString(1:8) '_Mag_All_' dateString(9:12) '_' dateString(13:16)];

figure(1)
a1=gca;
figure(3)
a2=gca;
figure(4)
a3=gca;
figure(5)
a4=gca;

a2clim=get(a2,'clim');
a3clim=get(a3,'clim');
```

50

60

70

80

90

100

69

```matlab
a4clim=get(a4,'clim');
newClim=[min([a2clim(1),a3clim(1),a4clim(1)]) max([a2clim(2),a3clim(2),a4clim(2)])];
%set(a2,'clim',newClim);
%set(a3,'clim',newClim);
%set(a4,'clim',newClim);

figure(7)
clf                                                                          110

set(gcf,'units','inches')
set(gcf,'Position',[1,1,10,7.5])

global axes4

h=copyobj(a1,gcf);
set(h,'OuterPosition',[0,0.5,0.5,0.5])
legend({'H','D','Z'})
colorbar('peer',h)                                                           120
axes4(1)=h;

h=copyobj(a2,gcf);
set(h,'OuterPosition',[0.5,0.5,0.5,0.5])
colorbar('peer',h)
axes4(2)=h;

h=copyobj(a3,gcf);
set(h,'OuterPosition',[0,0,0.5,0.5])
colorbar('peer',h)                                                           130
axes4(3)=h;

h=copyobj(a4,gcf);
set(h,'OuterPosition',[0.5,0,0.5,0.5])
colorbar('peer',h)
axes4(4)=h;

orient landscape
set(gcf,'inverthardcopy','off')
%saveas(gcf,[filename '.eps'])                                               140
%saveas(gcf,[filename '.png'])
%saveas(gcf,[filename '.pdf'])
%print('-dpng',[filename '.eps'],'-r300');
%print('-depsc',[filename '.eps'],'-r300');
%[status,result]=system(['ps2pdf ' filename '.eps']);
cd(nowDir);

experimentNew = experiment;


%———————————————————————%                                                    150
%    FUNCTION integrateFFT - FFT of one magnetometer component
%———————————————————————%



function integrateFFT(time,comp,intPeriod,Fs,L,dateString,compStr,dcCutoff,doPrint)
```

70

```matlab
NFFT2 = 2^nextpow2(intPeriod);
f2 = Fs/2*linspace(0,1,NFFT2/2);
Comp=[];
timeseries=[];                                                              160
for i=1:intPeriod:L-mod(L,intPeriod)-1
    time2=time(i:i+intPeriod-1);
    timeseries(end+1)=time2(1);
    comp2=comp(i:i+intPeriod-1);
    Comp(:,end+1)=fft(comp2,NFFT2)/intPeriod;
end
figure

% New - can try to make dB plots
                                                                            170
surf(timeseries,f2(dcCutoff:end),2*(abs(Comp(dcCutoff:NFFT2/2,:))) )
view([0,90])
colorbar
xlabel('Time (UT)','fontsize',12)
ylabel('Frequency (Hz)','fontsize',12)
title([dateString(1:8) ' UT - Fluctuations in ' compStr],'fontsize',14,'fontweight','bold')
colorAxesBlack
if(strcmp(doPrint,'Print'))
    %cd(experiment.instruments(1).plotDir);
    filename=[dateString(1:8) '_' compStr '_FFT_' dateString(9:12) '_' dateString(13:16)];   180
    %print('-dpng',[filename '.png'],'-r300');
    % pause
    %print('-depsc',[filename '.eps']);
    %[status,result]=system(['ps2pdf ' filename '.eps']);
    %saveas(gcf,[filename '.eps'])
    %saveas(gcf,[filename '.png'])
    %saveas(gcf,[filename '.pdf'])
    %cd(currDir);
end
                                                                            190


%
% FUNCTION colorAxesBlack - set the colors of FFT plots for printing
%


function colorAxesBlack
                                                                            200
whitebg(gcf,'black')
set(gcf,'color','white')
shading flat
set(get(gca,'Title'),'color','black')
set(get(gca,'xlabel'),'color','black')
set(get(gca,'ylabel'),'color','black')
set(gca,'xcolor','black')
set(gca,'ycolor','black')
set(gca,'zcolor','black')
set(gcf,'inverthardcopy','off')                                             210
```

71

polyfit

```
function experimentNew = EDFanalysis_magnetometer_polyfit(experiment)

%
% Get the relevant magnetometer data
%

time = experiment.dataRef.('magnetometer').time;
hcomp = experiment.dataRef.('magnetometer').hcomp;
dcomp = experiment.dataRef.('magnetometer').dcomp;
zcomp = experiment.dataRef.('magnetometer').zcomp;                          10

nowdir=pwd;
cd(experiment.instruments(1).plotDir);

indexoffset=1;
time=time(indexoffset:end);
hcomp=hcomp(indexoffset:end);
dcomp=dcomp(indexoffset:end);
zcomp=zcomp(indexoffset:end);
means = '';                                                                 20

hstyle='r-';
dstyle='g-';
zstyle='b-';

means = {[ 'Mean Values: H: ' num2str(round(mean(hcomp))) ...
          ' D: ' num2str(round(mean(dcomp))) ..
          ' Z:' num2str(round(mean(zcomp))) ' (nT)' ]};

hcomp=hcomp−mean(hcomp);                                                    30
dcomp=dcomp−mean(dcomp);
zcomp=zcomp−mean(zcomp);

hcompOffset=max(dcomp)−min(hcomp);
zcompOffset=min(dcomp)−max(zcomp);

interval=length(time);
order=5;

close all                                                                   40
figure

pval_h=fitPolynomial(time,hcomp,interval,order);
pval_d=fitPolynomial(time,dcomp,interval,order);
pval_z=fitPolynomial(time,zcomp,interval,order);


%pval_h=fitPolynomialCustom(time,hcomp,[3400,8000],[5,5,5]);
```

72

%pval_d=fitPolynomialCustom(time,dcomp,[2800,10000],[3,3,5]);
%pval_z=fitPolynomialCustom(time,zcomp,[2800],[order]);

%time=1:length(time);

%
% plot the components (separated) and the polynomial fits
%

plot(time,hcomp+hcompOffset,hstyle,time,pval_h+hcompOffset,'k-.')
hold on
plot(time,dcomp,dstyle,time,pval_d,'k-.')
plot(time,zcomp+zcompOffset,zstyle,time,pval_z+zcompOffset,'k-.')

%pause

%
% plot the residuals from the polynomial fits (separated)
%

hfitOffset=max(dcomp−pval_d)−min(hcomp−pval_h);
zfitOffset=min(dcomp−pval_d)−max(zcomp−pval_z);
clf
plot(time,hcomp−pval_h+hfitOffset,hstyle)
hold on
plot(time,dcomp−pval_d,dstyle)
plot(time,zcomp−pval_z+zfitOffset,zstyle)

%pause

%
% filter and plot the filtered data (separated)
%

clf
width=0;
filtered_h=medianFilter(hcomp−pval_h,width);
filtered_d=medianFilter(dcomp−pval_d,width);
filtered_z=medianFilter(zcomp−pval_z,width);
hfilteredOffset=max(filtered_d)−min(filtered_h);
zfilteredOffset=min(filtered_d)−max(filtered_z);
plot(time,filtered_h+hfilteredOffset,hstyle)
hold on
plot(time,filtered_d,dstyle)
plot(time,filtered_z+zfilteredOffset,zstyle)

%
% prepare dateString and, optionally, print the final labelled figure
%

%dateString=DateString(day,starttime(1),starttime(2));
%if(endtime(2)==60)
%   sysendtime=59;
%else

```
%        sysendtime=endtime(2);
%end
%result=DateString(day,endtime(1),sysendtime);
%dateString=[dateString(1:end-1) result(9:12)]
%[status, titleString]=system([ 'date -d jan1+' num2str(day-1) 'days' ...
%    num2str(starttime(1)) ':' num2str(starttime(2)) ' +%h.\ %d,\ %Y' ]);
titleString = datestr(experiment.dateStart,'mmm. dd, yyyy');
```

```
legend({'H','D','Z'})
title({[titleString ' Gakona Magnetometer'],means{1:end}},'fontsize',16,'fontweight','bold')
xlabel('Time (UT)','fontsize',14)
ylabel('Relative Magnetic Field Fluctuation (nT)','fontsize',14)


% print -dpng 'MagStudies2008.png'
% print -depsc 'MagStudies2008.eps'


cd(nowdir);
```

```
experimentNew = experiment;
experimentNew.dataRef.('magnetometer_polyfit').time = time;
experimentNew.dataRef.('magnetometer_polyfit').hcomp = filtered_h;
experimentNew.dataRef.('magnetometer_polyfit').dcomp = filtered_d;
experimentNew.dataRef.('magnetometer_polyfit').zcomp = filtered_z;

experimentNew.dataRef.('magnetometer_psd').time = time;
experimentNew.dataRef.('magnetometer_psd').hcomp = abs(filtered_h).^2;
experimentNew.dataRef.('magnetometer_psd').dcomp = abs(filtered_d).^2;
experimentNew.dataRef.('magnetometer_psd').zcomp = abs(filtered_z).^2;
```

```
%
% FUNCTION fitPolynomial
%

function [pval]=fitPolynomial(time,comp,interval,order)

numPeriods=floor(length(time)/interval)
pval=[];
for i=1:numPeriods
```

```
    indices=[1+(i-1)*interval:i*interval];
    pfit=polyfit(time(indices),comp(indices),order);
    pval=[pval;polyval(pfit,time(indices))];
end
if (mod(length(time),interval))
    indices=[1+interval*numPeriods:length(time)];
    pfit=polyfit(time(indices),comp(indices),order);
    pval=[pval;polyval(pfit,time(indices))];
end
```

```
%
% FUNCTION fitPolynomialCustom
%

function [pval]=fitPolynomialCustom(time,comp,intervals,orders)
```

74

```
intervals=[1,intervals,length(time)];
if (length(orders)==1)
    orders=ones(1,length(intervals)−1)*orders;
end                                                                          160

numPeriods=length(intervals)−1;
pval=[comp(1)];
for i=1:numPeriods
    indices=[intervals(i):intervals(i+1)];
    pfit=polyfit(time(indices),comp(indices),orders(i));
    pval=[pval;polyval(pfit,time(indices(1:end−1)))];
end

%                                                                            170
% FUNCTION medianFilter
%

function [filteredData]=medianFilter(data,width)

dataMedian=median(data);
    %width=dataMedian*50;
disp(dataMedian);
dataOut=abs(data−dataMedian)<width*dataMedian;
filteredData=data;                                                           180
filteredData(find(dataOut))=0;
```

B.4.3   Plotting

Export Heating Schemes

```
function EDFplot_magnetometer_heatingscheme(experiment)

times = experiment.times;
modes = experiment.modes;

for i=1:length(times)−1
    str = [time2str(times(i)) ' to ' time2str(times(i+1)) ', ' modes{i}];
    disp(str)
end
                                                                             10




function displayStr = time2str(time)
    hours = floor(time);
    mins = (time − hours)*60;
    if (mins>=60)
        hours = hours + 1;
        mins = mins − 60;                                                    20
    end
```

```
secs = round((mins − floor(mins))∗60);
mins = floor(mins);
if (secs>=60)
    mins = mins + 1;
    secs = secs − 60;
end

if (hours<10)
    if (hours<1)                                                    30
        hh = '00';
    else
        hh = ['0' num2str(hours)];
    end
else
    hh = num2str(hours);
end

if (mins<10)
    if (mins<1)                                                     40
        mm = '00';
    else
        mm = ['0' num2str(mins)];
    end
else
    mm = num2str(mins);
end

if (secs<10)
    if (secs<1)                                                     50
        ss = '00';
    else
        ss = ['0' num2str(secs)];
    end
else
    ss = num2str(secs);
end

displayStr = [hh ':' mm ':' ss];
```

Power Spectral Density

```
function handles = EDFplot_magnetometer_psd(experiment)

time = experiment.dataRef.magnetometer_psd.time;
hcomp2 = experiment.dataRef.magnetometer_psd.hcomp;
dcomp2 = experiment.dataRef.magnetometer_psd.dcomp;
zcomp2 = experiment.dataRef.magnetometer_psd.zcomp;

dateString=[datestr(experiment.dateStart,'yyyymmddHHMM') datestr(experiment.dateEnd,'HHMM')];

clf                                                                  10
```

```matlab
color = 'k';

subplot(3,1,1)
plot(time,hcomp2)
hold on
a1=gca;
xlabel('Time (UT)')
ylabel('h_2^2(t)')
title([dateString(1:8) ' UT - H,D,Z components'],'fontsize',10,'fontweight','bold')      20


subplot(3,1,2)
plot(time,dcomp2)
hold on
a2=gca;
xlabel('Time (UT)')
ylabel('d_2^2(t)')


                                                                                          30


subplot(3,1,3)
plot(time,zcomp2)
hold on
a3=gca;
xlabel('Time (UT)')
ylabel('z_2^2(t)')

h1=markTime(a1,experiment.times,color,1);
markHeaterMode(a1,experiment.times,experiment.modes,0.95,10,color);                       40
h2=markTime(a2,experiment.times,color,1);
markHeaterMode(a2,experiment.times,experiment.modes,0.95,10,color);
h3=markTime(a3,experiment.times,color,1);
markHeaterMode(a3,experiment.times,experiment.modes,0.95,10,color);
handles={h1,h2,h3};
```