Detecting Discontinuities from in-Situ Space Measurements: Method and FPGA Implementation

Costel Munteanu[1], Dan Cristian Turicu[2], Octavian Creț[3], and Marius Echim[4]

[1]Institute of Space Science
[2]Technical University of Cluj-Napoca
[3]Computer Science Department, Technical University
[4]Belgian Institute for Space Aeronomy

November 24, 2022

Abstract

The analysis in real time of space data variability is essential for scientists and space mission controllers. Automated tools designed to extract key descriptors of variability are needed and solutions to adapt such algorithms for on-board computers are rare. This paper describes the design of an automated system for detecting directional discontinuities of a physical quantity and its implementation in Field-Programmable Gate Array (FPGA). The system is currently adapted for solar wind or terrestrial magnetosheath magnetic field directional discontinuities, i.e. sharp changes of the magnetic field directionality. Our detection algorithm uses analysis windows of adjustable width and averaging procedures in order to reduce the effects of random fluctuations. A sliding-window approach is designed for continuous monitoring and detection of magnetic directional discontinuities. A software implementation of the algorithm was tested using in-situ magnetic field measurements, and emphasised improvements of performance when using analysis windows of adjustable width. The FPGA implementation of the detection algorithm is built on DILIGENT Nexys 4 DDR featuring a comercial Xilinx Artix-7 device and is designed to be ported to space qualified infrastructure. The FPGA system was tested with synthetic and laboratory signals, and provides results in very good agreement with the software implementation. The FPGA system provides an efficient real-time monitoring solution using minimal computational and energy resources, and reducing the main on-board computer utilization.

Detecting Discontinuities from in-Situ Space Measurements: Method and FPGA Implementation

C. Munteanu[1], D. C. Turicu[2], O. Creţ[2], and M. Echim[1, 3]

[1]Institute for Space Science, Măgurele, 077125, Romania.

[2]Technical University of Cluj-Napoca, Cluj-Napoca, 400114, Romania.

[3]Royal Belgian Institute for Space Aeronomy, Brussels, Belgium.

Corresponding author: D. C. Turicu (dan.turicu@cs.utcluj.ro)

Key Points:

- We designed an algorithm to detect directional discontinuities from in-situ space measurements, suitable for automated application
- A Field-Programmable Gate Array (FPGA) implementation of the algorithm, designed to be adapted for on-board operations, is provided
- Tests with laboratory and space measurements of the FPGA and software implementations of the detection algorithm, give excellent results

Abstract

The analysis in real time of space data variability is essential for scientists and space mission controllers. Automated tools designed to extract key descriptors of variability are needed and solutions to adapt such algorithms for on-board computers are rare. This paper describes the design of an automated system for detecting directional discontinuities of a physical quantity and its implementation in Field-Programmable Gate Array (FPGA). The system is currently adapted for solar wind or terrestrial magnetosheath magnetic field directional discontinuities, i.e. sharp changes of the magnetic field directionality. Our detection algorithm uses analysis windows of adjustable width and averaging procedures in order to reduce the effects of random fluctuations. A sliding-window approach is designed for continuous monitoring and detection of magnetic directional discontinuities. A software implementation of the algorithm was tested using in-situ magnetic field measurements, and emphasised improvements of performance when using analysis windows of adjustable width. The FPGA implementation of the detection algorithm is built on DILIGENT Nexys 4 DDR featuring a comercial Xilinx Artix-7 device and is designed to be ported to space qualified infrastructure. The FPGA system was tested with synthetic and laboratory signals, and provides results in very good agreement with the software implementation. The FPGA system provides an efficient real-time monitoring solution using minimal computational and energy resources, and reducing the main on-board computer utilization.

1 Introduction

A clear understanding of data variability recorded in space is vital for scientists and space mission controllers. Consequently, automatic tools designed to extract relevant key-descriptors of variability are extremely useful. Nevertheless, solutions to adapt such algorithms for on-board computers are still rare. In this paper we describe an algorithm that detects directional discontinuities of in-situ measured variables, and its implementation on Field-Programmable Gate Array (FPGA) devices with an application on directional discontinuities of the interplanetary magnetic field (IMF).

The abrupt changes in the orientation of the IMF, referred to as directional discontinuities (DDs), are known to trigger geomagnetic storms and substorms, with significant impact on ground-based and spaceborne technologies. DDs are important when estimating the solar wind propagation time from an upstream solar wind monitor to a downstream target (e.g., Mailyan et al., 2008; Haaland et al., 2010; Munteanu et al., 2013). IMF discontinuities play a key role in understanding the micro-scale structure of the solar wind. With an average occurrence rate of one or two per hour, IMF discontinuities are abundant structures in the solar wind (e.g., Newman et al., 2020) and represent an omnipresent source of variability for the terrestrial plasma environment.

Two general classes of idealized Magnetohydrodynamic (MHD) discontinuities can be distinguished: stationary structures, i.e. discontinuities that do not propagate with respect to the ambient plasma (tangential discontinuities (TDs) and contact discontinuities), and propagating discontinuities (rotational discontinuities (RDs) and shocks). The most frequent small-scale discontinuities in the interplanetary space are the abrupt changes in the direction of the magnetic field, predominantly expected for TDs and RDs (e.g., Paschmann et al., 2013).

Two main classes of algorithms to detect solar wind discontinuities are proposed in the literature. The first class includes algorithms searching for changes in the magnetic field direction (Burlaga, 1969; Lepping and Behannon, 1986; Li, 2008; Borovsky, 2008, 2010; Miao et al., 2011; Perri et al., 2012; Zhdankin et al., 2012 a, b); the second class includes algorithms searching for changes in the amplitudes of the magnetic field components and/or magnitude (Tsurutani and Smith, 1979; Vasquez et al., 2007; Greco et al., 2008; Greco and Perri, 2014; Greco et al., 2016, 2018).

The term "directional discontinuity" was originally introduced by Burlaga (1969) to denote a variation of solar wind magnetic field direction larger than 30 degrees in less than 30 seconds. Since then, this definition was used in many variant algorithms. Li (2008) developed an algorithm to identifying discontinuities based on this definition. Borovsky (2010) studied the spectral effects of solar wind DDs detected using the definition above. Chian and Munoz (2011) used the Li (2008) detection method to study the relation between discontinuities, turbulence, and magnetic reconnections at the leading edge of an interplanetary coronal mass ejection. Miao et al. (2011) further developed the algorithm by Li (2008) and introduced a way of determining the discontinuity thickness.

Vasquez et al. (2007) developed an original detection algorithm which is independent of the directional changes, but relies on changes in the amplitude of the magnetic field components. They found that the daily occurrence rate of strong solar wind discontinuities obtained with their original detection algorithm is comparable with the daily occurrence rate obtained using an algorithm based on directional changes. Tsurutani and Smith (1979) also showed that the method used by Burlaga (1969) provides similar results to their method based on changes of the amplitude of field components. Burkholder and Otto (2019) describe yet another original detection algorithm based on changes of the amplitude of field components.

Due to various computational difficulties encountered when implementing automated detection methods, even very recent studies still use visual inspection to identify discontinuities (Mailyan et al., 2008; Munteanu et al., 2013; Artemyev et al., 2018, 2019 a, b). Note that the automated detection algorithm of Burkholder and Otto (2019) still uses visual inspection to eliminate events that are not isolated from other structures in the time series. For datasets of only a few hundred events/discontinuities, the detection by visual inspection can be an acceptable option, but, for large-scale statistical studies, visual inspection-based methods are certainly not suitable.

In this study we designed and implemented an original algorithm based on the principles described by Li (2008) (see also Borovsky, 2008). We adopt a discontinuity detection algorithm based on directional changes because: (i) traditionally, angular changes were the preferred detection method (Burlaga, 1969), which renders our approach compatible with previous ones, (ii) many authors have recently started to use angular changes in order to improve results from algorithms based on amplitude changes of the field components (e.g., Greco et al., 2018), (iii) it is an efficient and less complex approach, thus computationally less intensive and hence less power consuming compared to most algorithms based on amplitude changes of the field components. Since we aim to provide a hardware implementation of our algorithm, the directional changes approach leads to a more reliable and robust FPGA implementation. (iv) In many cases, the two approaches are equivalent, especially for discontinuities with rotation angles larger than 30° (e.g., Greco et al., 2018).

State of the art space plasma instruments on-board recent terrestrial or interplanetary missions, e.g., MMS (Burch et al., 2016) and Parker Solar Probe (Fox et al., 2016), provide high time-resolution in-situ measurements, while being constrained by limited telemetry possibilities. Thus, the on-board implementation of computations is critical for taking advantage of the full set of collected data. The on-board discontinuity detector would allow for the computation of magnetic field rotation angles from high-resolution measurements without downloading the entire dataset. The on-board computed angular changes can also be used to identify interesting events and activate triggers for temporary on-board storage and subsequent download of high-resolution data (selective downlink).

Most of the algorithms discussed above are capable of automatically detecting IMF directional discontinuities, but they were designed only for on-ground data analysis. To our knowledge, no other algorithm was designed to be implemented in an FPGA device. This study is part of a broader effort devoted to building a complex semi-autonomous digital signal processing library, able to apply on-board various digital signal processing techniques. The current version of the library already includes modules devoted for the spectral and statistical analysis of fluctuations (Deak et al., 2018; Opincariu et al., 2019; Deak et al., 2021; Turicu et al., 2022). Here we discuss a new feature, allowing to detect directional discontinuities.

The paper is structured as follows. Section 2 provides a theoretical background and a conceptual description of the discontinuity detection algorithm. It also discusses reconfigurable FPGA devices in the context of on-board data processing. Section 3 gives an overview of the system and describes the FPGA implementation of the discontinuity detection algorithm. Section

4 presents the main tests and validation procedures of both the algorithm and its FPGA implementation; the section shows results obtained using synthetic datasets, laboratory magnetic field measurements, and also in-situ interplanetary magnetic field measurements. Section 5 provides a summary and perspective.

2 Theoretical background

2.1 Directional discontinuity detection algorithm for automatic use

Let us consider a set of in-situ measurements of a plasma or field variable. Take, for instance, the triaxial measurements of the interplanetary magnetic field, B_x, B_y and B_z, in an arbitrary reference system. Magnetic directional discontinuities observed in interplanetary space and in the terrestrial magnetosheath are characterized by sharp changes in the direction of the magnetic field vector $\boldsymbol{B} = [B_x\ B_y\ B_z]$. Magnetic directional changes are computed as:

$$\varphi(t_k) = \left(\frac{180}{\pi}\right) \cos^{-1}\left(\frac{\boldsymbol{B_1}\cdot\boldsymbol{B_2}}{|\boldsymbol{B_1}|\cdot|\boldsymbol{B_2}|}\right) \quad (1)$$

where φ (in degrees) is computed at time t_k, and $\boldsymbol{B_1}$ and $\boldsymbol{B_2}$ are defined as:

$$\begin{aligned} \boldsymbol{B_1} &= \langle \boldsymbol{B} \rangle_{\tau_1} = \langle [B_x\ B_y\ B_z] \rangle_{\tau_1} \\ \boldsymbol{B_2} &= \langle \boldsymbol{B} \rangle_{\tau_2} = \langle [B_x\ B_y\ B_z] \rangle_{\tau_2} \end{aligned} \quad (2)$$

with the symbol $\langle\cdot\rangle_\tau$ denoting time averaging. A window W centered at time t_k is defined: $W = [t_{k-l/2}, t_{k+l/2}]$, with l denoting the length of this window. Within this window, the two averaging intervals in (2) are defined as: $\tau_1 = [t_{k-l/2},\ t_k]$ and $\tau_2 = [t_k,\ t_{k+l/2}]$. Clearly, τ_1 and τ_2 contain the same number of data samples ($\tau_1 = \tau_2 = l/2$), thus, for brevity, we will refer to either one of these two intervals as TAU. According to (1), φ takes values between 0° (parallel orientation) and 180° (antiparallel orientation).

Other authors used similar definitions for $\boldsymbol{B_1}$ and $\boldsymbol{B_2}$ (e.g., Borovsky 2008, 2010). Li (2008) and Miao et al. (2011) used only the instantaneous vector measurements at the two edges of the window W. Mailyan et al. (2008) (see also Munteanu et al., 2013) used two time intervals: one for computing the averages, and an additional time interval for separating $\boldsymbol{B_1}$ from $\boldsymbol{B_2}$. For the purpose of this study, we will use $\boldsymbol{B_1}$ and $\boldsymbol{B_2}$ as defined by (2).

Figure 1a depicts a synthetic dataset with three components of 1000 data samples each. The *x* and *y* components include sharp discontinuities added onto a constant signal with superposed white Gaussian noise; the *z*-component is white noise. Figure 1b shows the result obtained with (1) for a window W of 128 data samples centered on the discontinuity. Figure 1b also shows an illustration of the time intervals τ_1 and τ_2 from (2). The two sharp changes in the *x* and *y* components correspond, as expected from geometrical considerations, to an angular change of $\varphi \cong 90°$.

For the analysis of a real signal, continuously collected in-situ, we developed a sliding-window algorithm which computes the angular changes for windows W centered at each time instance t_k. An illustrative example is shown in Fig. 1c. The algorithm starts with the analysis of the first 128 samples of the signal and computes the value $\varphi(t_{64})$ corresponding to the center of this first instance of the analysis window. In the next step the window is moved by one sample, and the procedure is repeated to compute the value $\varphi(t_{65})$. The algorithm continues until the last value $\varphi(t_{936})$ is computed. Figure 1c shows that φ takes values close to 0° up to sample number 436, then it starts to increase as the window position approaches the discontinuity, reaching a maximum value of $\varphi \cong 90°$ at the discontinuity center (at sample 500); then it starts to decrease back towards $\varphi \cong 0°$ (at sample 564). This increasing (decreasing) trend of angular changes is due to the relative position of the sliding window with respect to the actual position of the discontinuity, resulting in amplitude changes as the window moves closer to (away from) the discontinuity. Figure 1 was generated using the software analysis tool Integrated Nonlinear Analysis (INA) (Munteanu, 2017; see also http://www.storm-fp7.eu/index.php/data-analysis-tools).

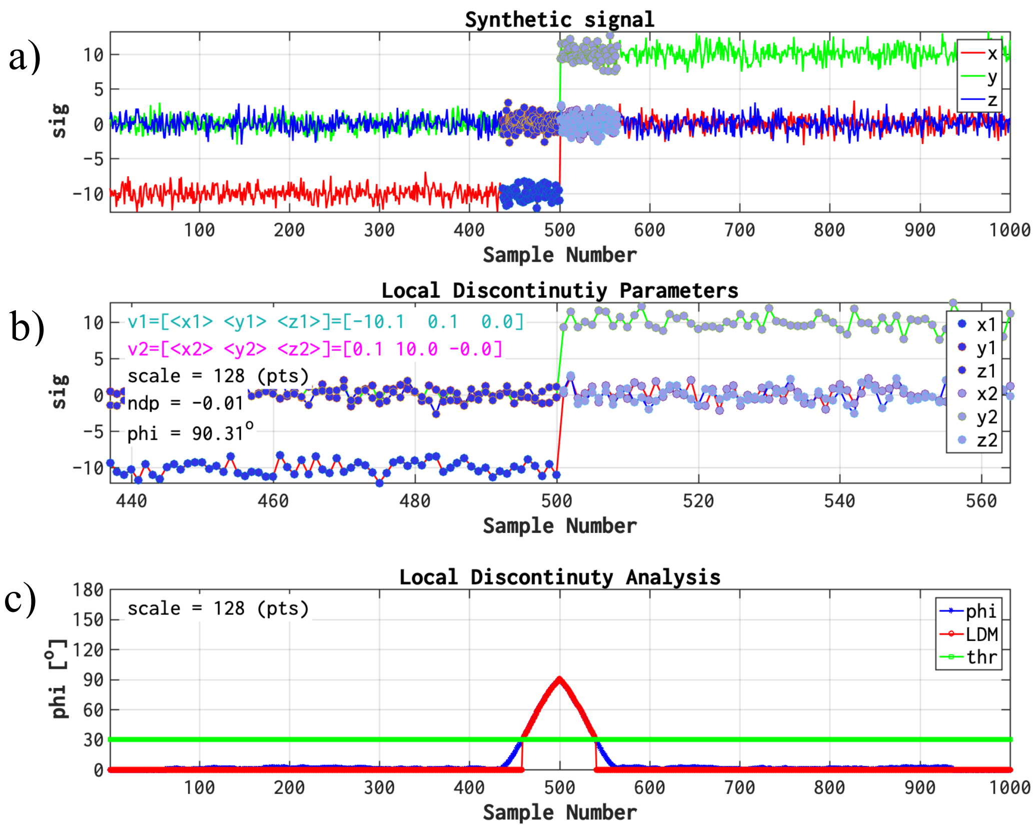

Figure 1. a) Synthetic dataset with three components of 1000 data samples each. b) Zoom-in on the 128-sample interval highlighted in the top panel; τ_1 and τ_2 are marked in dark-blue and light-blue, respectively. c) Local discontinuity analysis of the synthetic dataset depicted in panel (a), using a sliding-window of 128 data samples: $\varphi(t_k)$ (from (1); blue line labeled "phi"), local discontinuity measure LDM$^{(\mathrm{deg})}$ (from (3); red line) and $\varphi_c = 30°$ (green line labeled "thr"). This figure, and also Figs. 3, 4 and 5, were generated using the Integrated Nonlinear Analysis (INA) library (Munteanu, 2017; see also http://www.storm-fp7.eu/index.php/data-analysis-tools).

The discontinuity detection algorithm we propose here is based on a critical value of the angular change. This value, denoted in the following as φ_c, is set to 30°. We define a local discontinuity measure (LDM) which is equal to the value of the rotation angle φ, if this is larger than φ_c, and zero otherwise:

$$LDM^{(deg)}(t_k) = \begin{cases} \varphi(t_k), & if\ \varphi(t_k) \geq \varphi_c \\ 0, & otherwise \end{cases} \tag{3}$$

$LDM^{(deg)}$ is used as a quantitative measure for the presence of directional discontinuities. Figure 1c shows the $LDM^{(deg)}$ values computed using (3) for the synthetic signal. As expected, $LDM^{(deg)}$ is different from zero (and exactly equal to φ) only when $\varphi \geq 30°$.

For an on-board FPGA implementation of the sliding-window algorithm described above, there is no need to apply the $\cos^{-1}$ function and multiply by $(180/\pi)$, as in (1). If needed, these operations can be later performed on ground, using calibrated results. Thus, a simplified version of the method can be implemented in the FPGA device, based solely on the normalized dot product (ndp):

$$ndp(t_k) = \frac{\boldsymbol{B_1} \cdot \boldsymbol{B_2}}{|\boldsymbol{B_1}| \cdot |\boldsymbol{B_2}|} \tag{4}$$

with $\boldsymbol{B_1}$ and $\boldsymbol{B_2}$ defined by (2). From (1) and (4), it follows that angular changes of $180°$ and $0°$, correspond to normalized dot product values of -1 and $+1$, respectively. Similarly, the value of $ndp = 0$ will correspond to an angular change of $\varphi = 90°$. For future reference, the threshold value $\varphi_c = 30°$, corresponds to $ndp_c \cong 0.87$.

Based on the ndp parameter given in (4), a new localized discontinuity measure can be defined:

$$LDM^{(ndp)}(t_k) = \begin{cases} ndp(t_k), & if\ ndp(t_k) \leq ndp_c \\ 0, & otherwise \end{cases} \tag{5}$$

$LDM^{(ndp)}$ will be further used in the FPGA implementation of the detection algorithm.

2.2 Reconfigurable FPGA devices for on-board processing

Modern architectures designed for on-board analysis of data make use of reconfigurable FPGA technology (Pingree, 2010; Kuwahara, 2009; French et al., 2018; Huber et al., 2007; Hanafi et al., 2017). In space, the electronic systems are prone to failures caused by radiations generated by high-energy particles. The FPGA devices are even more vulnerable since radiations can affect their configuration logic and applications data. To mitigate the effect of radiations, specific software and hardware techniques are used for space-qualified FPGA designs. The radiation tolerant FPGA devices from Xilinx (Virtex, 2014; Virtex, 2018; Kintex, 2020) or Microsemi (Microsemi, 2015) mitigate the effect of space radiations and eliminate the requirement of using dedicated mitigation techniques, like TMR (triple modular redundancy). Our current design/prototype relies on commercial FPGAs and can be ported to radiation hardened architectures.

The development flow for FPGA devices is based on a Register Transfer Level (RTL) description of the circuits using a hardware description language. The High-Level Synthesis (HLS) technology simplifies the digital signal processing algorithms implementation in FPGA devices allowing the description of the algorithm’s functionality using a classical programming language and the generation of the RTL description.

We used HLS to generate an RTL description for the discontinuity detector and an RTL description for the rest of the system components.

3 System overview and implementation

The spacecraft's main on-board computers execute multiple critical tasks. Our prototype demonstrates that some tasks can be retargeted to be executed on FPGA devices even at instrument level, thus reducing the main on-board computers' utilization and saving energy resources.

The functionality of the system was validated with synthetic signals, and also with real measurements received from a magnetic sensor, as discussed below. The laboratory tests were performed by continuously monitoring the measurements received from the magnetic sensor and calculating the local discontinuity measures. The output of the system can be used to notify the on-board computer to perform additional analyses or to confirm a *status quo* in the evolution of the field parameters.

3.1 System implementation

The directional discontinuity detector based on a local discontinuity measure defined by (4) is implemented on a Digilent Nexys 4 DDR development board featuring a commercial Xilinx Artix-7 FPGA device. The architecture of the system, shown in Fig. 2a, includes a Digilent CMPS2 module attached to the development board to measure the magnetic field. The laboratory prototype includes its own data provider which is a Memsic MMC34160PJ magnetic sensor (www.memsic.com/magnetometer-4) able to perform measurements of the magnetic field on 3 axes within the full-scale range of ±16 Gauss. The sensor can perform measurements every 10 ms when the resolution of the analog-to-digital convertor is set to 16 bits. The magnetic sensor provides an *I2C* communication channel for configuration, control and to read the measurements.

The discontinuity detector that implements our proposed detection algorithm is implemented in the Xilinx Artix-7 FPGA device. When directional discontinuities are detected, the *Discontinuity Detected* output notifies the main on-board computer or other instruments; this notification can be used to perform, for instance, additional analyses. The *Serial Connection* with the host computer is used only for the validation of the results produced by the system. In a production setup for the system to be deployed on the satellite's on-board FPGA device, the *Serial Connection* will be removed or replaced with other communication protocols to receive the measurements from the on-board magnetic instrument and the computed discontinuity measures.

3.2 FPGA system design

The design implemented in the FPGA device provides the following functionalities: an *I2C* communication unit to interface with the attached data provider (magnetic sensor), the *Directional Discontinuity Detector* component that analyses the measurements received from the sensor and calculates the discontinuity measures, an *Output* component that notifies the main on-board computer when discontinuities are detected, a *Data Buffer* that aggregates the magnetic sensor measurements and the computed discontinuity measures, and a *Serial Unit* to interface with the host computer for the validation of the system. The architecture of the FPGA design is shown in Fig. 2b.

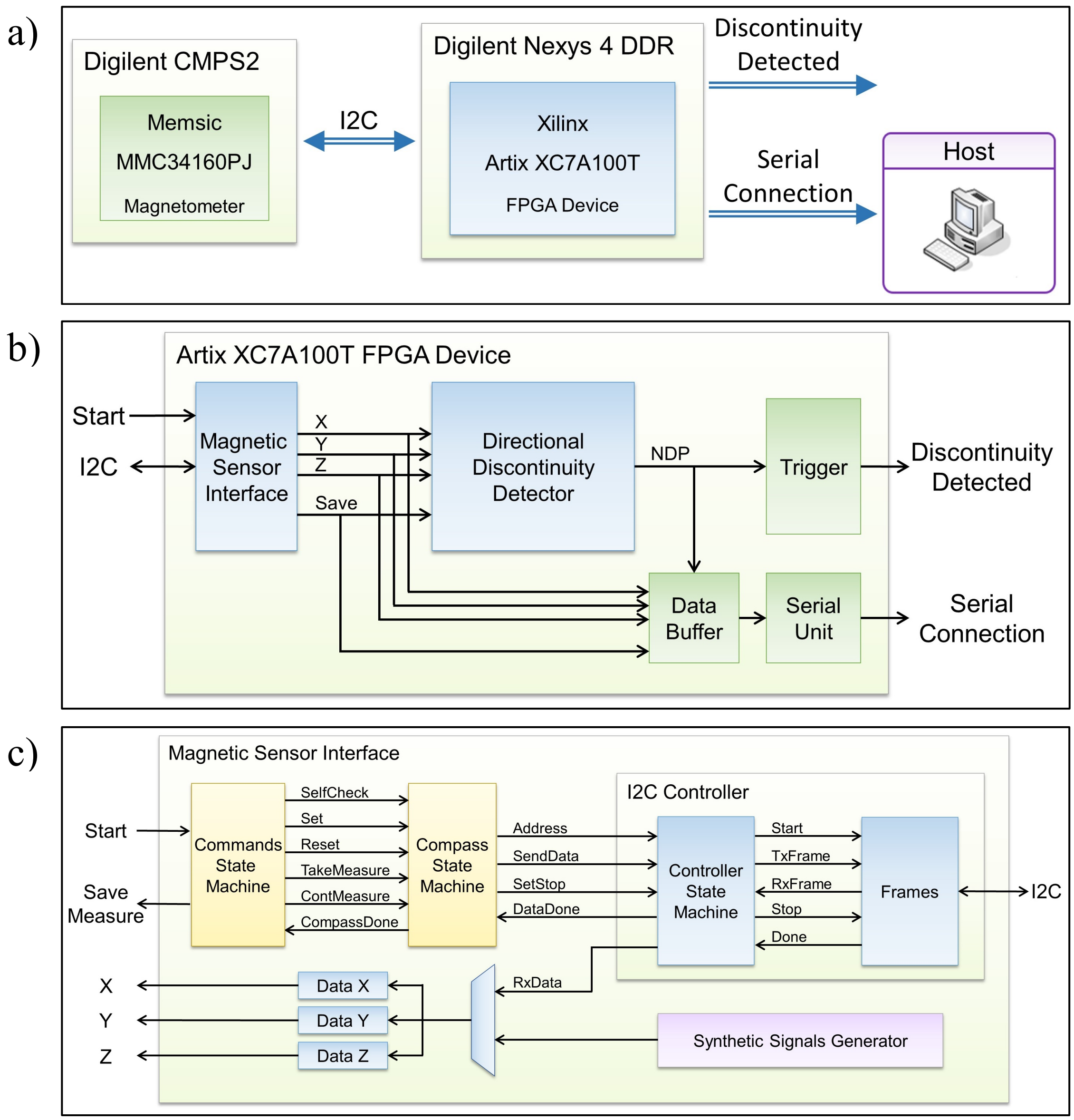

Figure 2. a) System Block Diagram. b) FPGA System Architecture. c) Magnetic Sensor Interface Architecture.

3.2.1 Magnetic sensor interface

The *Magnetic Sensor Interface* implements the communication protocol with the Memsic magnetic sensor. The implementation is based on an *I2C* controller which provides a generic implementation of the protocol, and on additional custom circuits to configure and control the magnetic sensor.

Figure 2c shows the architecture of the *Magnetic Sensor Interface*. The *Commands State Machine* and *Compass State Machine* components implement the operations specific to the Memsic magnetic sensor according to the sensor datasheet (www.memsic.com/magnetometer-4), *e.g.* take one measurement, activate continuous measurement mode, and calibration specific commands. The interface also includes the *Synthetic Signals Generator* that generates synthetic signals for the validation of the discontinuity detector.

The communication with the magnetic sensor is initiated by the assertion of the *Start* input signal. A sequence of commands is sent to configure and control the sensor: 1) execute the self-check test, 2) measure and save the calibration offsets, and 3) activate continuous measure mode. After the initial self-check test, the sensor performs two measurements to determine the calibration offsets. Once the calibration phase is finished, the continuous measurements mode is activated and the sensor performs measurements of the three axes every 12 ms. The measurements for each axis are returned on the X, Y, and Z outputs.

3.2.2 Directional discontinuity detector

The *Directional Discontinuity Detector* was implemented in C++ and its corresponding RTL description was generated using the high-level synthesis tool from Xilinx. Since the current design assumes a sampling rate of the magnetic sensor of 100 Hz, which is reasonable for on-board triaxial fluxgate magnetometers and is also very low compared to the 100 MHz clock signal driving the FPGA device, the goal set for the high-level synthesis process is to reduce the amount of programmable logic resources required for the design. Note however that the sampling rate is an adjustable parameter of the system.

The raw measurements received from the magnetic sensor are first converted to Gauss and then passed to the Directional Discontinuity Detector. The pseudocode for the C++ implementation of the detector is depicted in Algorithm 1. The WX, WY and WZ parameters store the last measurements received from the magnetic sensor. These parameters are implemented as buffers in which each new measurement is saved and the oldest measurement is discarded. The first loop, lines 2-5, computes the sum of the TAU measurements defined by (2) up to the point for which the discontinuity measure (1) is computed. Lines 6-8 calculate the components of the B1 vector as the mean of all these measurements. The second loop, lines 10-13, computes the sum of the TAU measurements received after the point for which the discontinuity measure is computed, and similarly lines 14-16 calculate the components of the B2 vector as the mean of these measurements (see definitions (2)). Line 17 computes the scalar product between the vectors B1 and B2, while lines 18 and 19 compute the norm for each vector. Finally, line 20 computes the discontinuity measure *ndp* defined by (4).

3.2.3 Trigger component and serial unit

The *Trigger* component of the FPGA system generates the output of the discontinuity dection system for the main on-board computer or other instruments. The discontinuity measures *ndp* computed by the discontinuity detector are compared against a predefined threshold value. If the value exceeds the threshold defined by (5), the output is asserted to notify the main on-board computer or other devices. Based on the requirements of the on-board computer, the *Trigger* component can be further customized. While this component generates the main output of the system, the *Serial Unit* sends the measurements received from the magnetic sensor and the computed discontinuity measures to a host computer for the validation of the system and visualization of the results.

Algorithm 1. Pseudocode for the C++ implementation of the discontinuity detector.

```
function Discontinuity_Detector(WX, WY, WZ, TAU):
sum_X ← 0, sum_Y ← 0, sum_Z ← 0
for i = 1 to TAU do
   sum_X ← sum_X + WX[i]
   sum_Y ← sum_Y + WY[i]
   sum_Z ← sum_Z + WZ[i]
B1_X ← sum_X / TAU
B1_Y ← sum_Y / TAU
B1_Z ← sum_Z / TAU
sum_X ← 0, sum_Y ← 0, sum_Z ← 0
for i = TAU to 2*TAU do
   sum_X ← sum_X + WX[i]
   sum_Y ← sum_Y + WY[i]
   sum_Z ← sum_Z + WZ[i]
B2_X ← sum_X / TAU
B2_Y ← sum_Y / TAU
B2_Z ← sum_Z / TAU
product ← B1_X * B2_X + B1_Y * B2_Y + B1_Z * B2_Z
norm_B1 ← SQRT(B1_X*B1_X + B1_Y*B1_Y + B1_Z*B1_Z)
norm_B2 ← SQRT(B2_X*B2_X + B2_Y*B2_Y + B2_Z*B2_Z)
NDP ← product / (norm_B1 * norm_B2)
return NDP
```

4 Experimental results

4.1 Computer validation of the discontinuity detection algorithm

4.1.1 Testing the effects of discontinuity strength, signal-to-noise ratio, discontinuity thickness and analysis window width

We define the "strength" of a directional discontinuity as the amplitude of the angular change across the discontinuity (see (1)). Figure 3a shows the results of the LDM algorithm applied on a synthetic three-component vector dataset comprised of 3600 data samples per component. Abrupt amplitude changes in the *x* and *y* components are included such that they correspond to angular changes of $\varphi = 180°$ (at sample number 900), $\varphi = 90°$ (at sample 1800) and $\varphi = 45°$ (at sample 2700). The LDM algorithm uses a window W of width $l = 128$ data samples, in this case.

Figure 3a demonstrates that the three predefined angular changes are accurately detected by our proposed discontinuity detection algorithm. Fig. 3a shows how the temporal profile of angular changes depends on discontinuity strength. For an angular change of $\varphi = 45°$, one can observe a rather slow increase (decrease) of $\varphi(t_k)$ as we approach (move away from) the center of the discontinuity. For $\varphi = 90°$, the increasing and decreasing trends are faster compared to the $\varphi = 45°$ case. At $\varphi = 180°$ these increasing and decreasing trends almost break down, and we see rather abrupt jumps from $\varphi \cong 0°$ to $\varphi \cong 180°$. These results were obtained for a signal with a small-amplitude white noise whose standard deviation is equal to 0.1.

Figure 3b shows the results of our LDM algorithm when the noise level is increased by adding a white Gaussian noise with a standard deviation equal to 20. Note that the signal amplitude jumps are of 20 units (for $\varphi = 180°$) and 10 units (for $\varphi = 90°$ and $\varphi = 45°$). Fig. 3b demonstrates that a high noise level, with standard deviation of the order of the signal amplitude jumps, can easily generate spurious discontinuities above the threshold level of $\varphi_c = 30°$.

Figure 3c illustrates the performance of the LDM algorithm applied on a signal including discontinuities with larger width. This figure demonstrates that when a small analysis window is applied on signals containing thick discontinuities, the algorithm fails to accurately detect discontinuities.

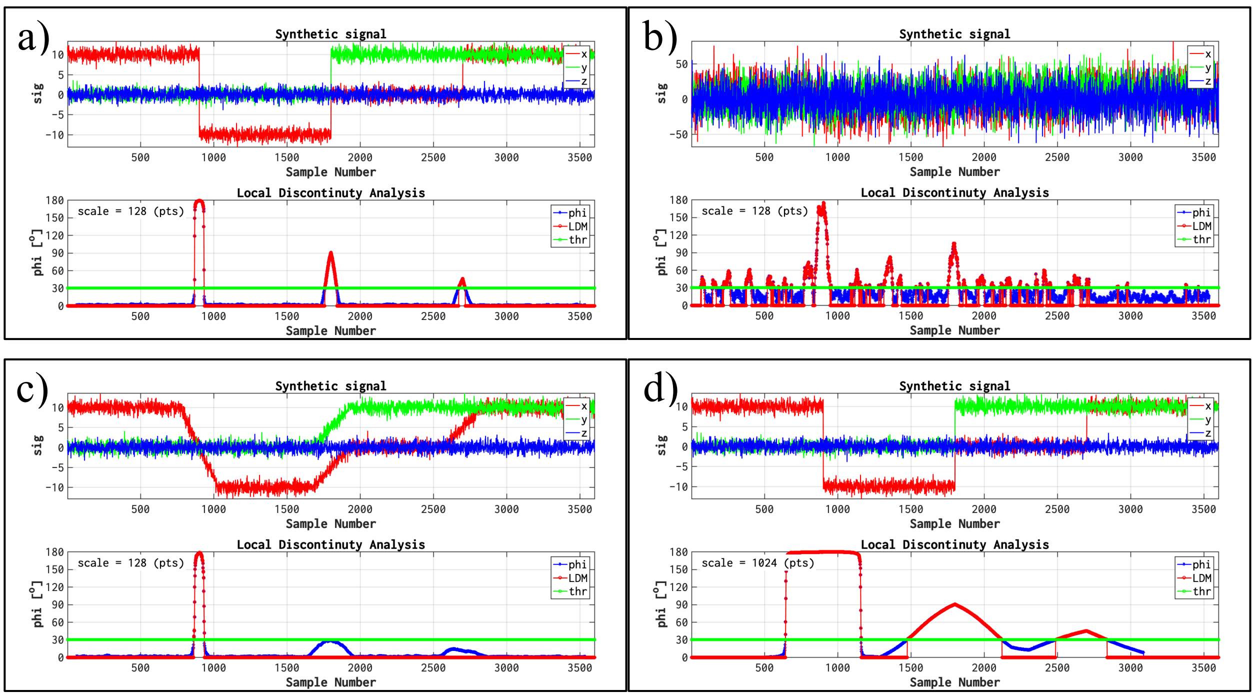

Figure 3. a) Effect of discontinuity strength. b) Effect of poor signal-to-noise ratio. c) Effect of discontinuity width. d) Effect of window length. In each panel, top plots depict synthetic datasets with three components of 3600 samples each, and bottom plots depict the local discontinuity analysis using the setup in Fig. 1c. Amplitude changes are introduced in all synthetic datasets so as to correspond to angular changes of 180° (at sample number 900), 90° (at 1800) and 45° (at 2700). Panel a) depicts the results for a low noise-level and abrupt (1 sample) amplitude changes, in panel b) the noise level is increased, and in panel c) the amplitude changes are widened to span 256 samples each. Local discontinuity analysis is performed using windows of 128 samples (panels a, b and c) and 1024 samples (panel d).

Figure 3d depicts the effect of enlarging the length of the analysis window. In this case the resulting temporal profiles of $\varphi(t_k)$ are much wider around each discontinuity, compared to those in Fig. 3a. When the length of the analysis window is comparable to the distance between two adjacent discontinuities, as is the case here, the detection profiles corresponding to each individual discontinuity start to merge and become indistinguishable from each other. By enlarging the analysis window even more, the algorithm will eventually detect only one broad discontinuity. Figure 3d shows results for an analysis window spanning 1024 samples, which is slightly larger than the separation distance of 900 samples between adjacent discontinuities. One can observe that the discontinuities characterized by angular changes of 90° and 45° are almost fully merged, and undistinguishable by the LDM algorithm.

4.1.2 Improving detection accuracy of the LDM algorithm

Figures 3b, 3c and 3d identified cases when the discontinuity detection algorithm failed to work properly for one of the following reasons: (i) due to poor signal-to-noise ratio (Fig. 3b), (ii) due to discontinuities being thicker than the analysis window (Fig. 3c), and (iii) due to the analysis window being larger that the time interval between adjacent discontinuities (Fig. 3d). In this section we discuss how to address these issues and improve the detection accuracy of the algorithm.

When the signal-to-noise ratio is decreased, the accuracy of the LDM algorithm can be maintained by computing the angular changes (1) using a larger number of samples. This can be explained using the standard error of the mean (SEM), a commonly used statistical measure for the differences between the mean value of a sample dataset and the mean value of the population from which the sample was drawn. By definition SEM is directly proportional to the standard deviation of the sample dataset and inversely proportional to the square root of the sample length. Thus, using a smaller dataset length increases SEM, and this increases the statistical uncertainty of the average values used in computing the angular changes. This is even more important in case of poor signal-to-noise ratio, where the larger standard deviation of the dataset increases the statistical uncertainty (decreases the statistical significance) of the averages even more. Increasing the width of the analysis window adds more data samples to the analysis and, assuming that no other strong discontinuities are added within this enlarged analysis window, leads to an increased accuracy.

When the width of the analysis window is smaller than the discontinuity thickness, our algorithm computes angular changes using samples only from inside the discontinuity. Thus, assuming that the discontinuity has a simple ramp-like structure (as in Fig. 3c), the angular changes computed using small windows will always be smaller than the angular changes computed using windows covering the full width of the discontinuity. As in the previous case, increasing the width of the analysis window will increase the accuracy, assuming again that no other discontinuities are included in the enlarged analysis window.

When more than one discontinuity is present inside the analysis window, we are in a case similar to that depicted in Fig. 3d, where the discontinuity detection algorithm fails because the analysis window is wider that the time interval between adjacent discontinuities. In this case, accuracy can be increased by decreasing the width of the analysis window.

Figure 4 illustrates an example on how to improve the algorithm's accuracy by using a larger analysis window. The signal includes three wide discontinuities identical to those in Fig. 3c, with the addition of a white Gaussian noise with standard deviation equal to 20, as in Fig. 3b. It is shown that increasing the width of the window allows the detection of discontinuities that are missed by shorter windows.

To conclude, a compromise has to be made between enlarging the analysis window in order to increase detection accuracy for wider discontinuities and for those buried in noise, and keeping the analysis window small enough so as to be capable of distinguishing between two adjacent discontinuities separated by a small time-interval.

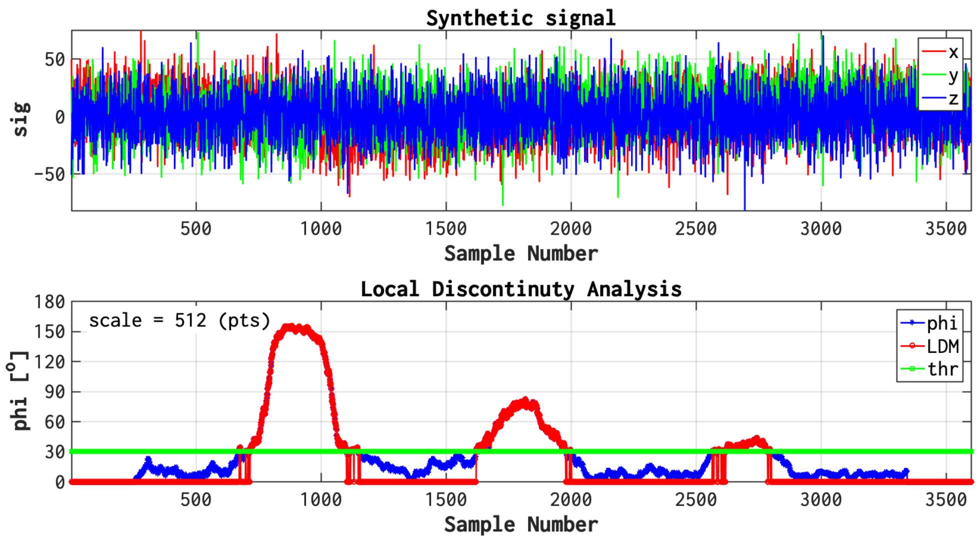

Figure 4. Example on how to mitigate the effects of increased noise-level and discontinuity thickness, by using larger windows. Top plot: synthetic data containing three 256-samples-wide discontinuities superposed onto white Gaussian noise with standard deviation equal to 20 units. Bottom plot: local discontinuity analysis with windows of 512 samples, depicted using the setup in Fig. 1c.

4.1.3 Computer validation of the LDM algorithm using interplanetary magnetic field measurements

In the study by Munteanu et al. (2013) we compiled a database consisting of 365 solar wind discontinuities, identified as clear magnetic field rotations by visually examining interplanetary magnetic field measurements. We select a sample from that database, consisting of magnetic field measurements from the Advanced Composition Explorer (ACE) spacecraft in January 06, 2003. Figure 5 illustrates the results of the discontinuity detection algorithm applied on these real-life, in-situ measurements.

Several specific features are observed in Fig. 5. Let us examine the discontinuity around sample 3250. As we increase the window width, the LDM profile corresponding to this discontinuity becomes wider, as expected. In this case, the peak value of $\varphi(t_k)$ computed by our algorithm remains almost unchanged as we increase the width of the analysis window: the discontinuity amplitude is slightly above 100° for the first two windows, and slightly below 100° for the last two windows. The situation is very different for the discontinuity centered around sample 4000. In this latter case, $\varphi(t_k)$ decreases systematically, from a value around 120° for the window whose length is set to 32 samples, to a value of 60° for the window whose length is equal to 256 samples.

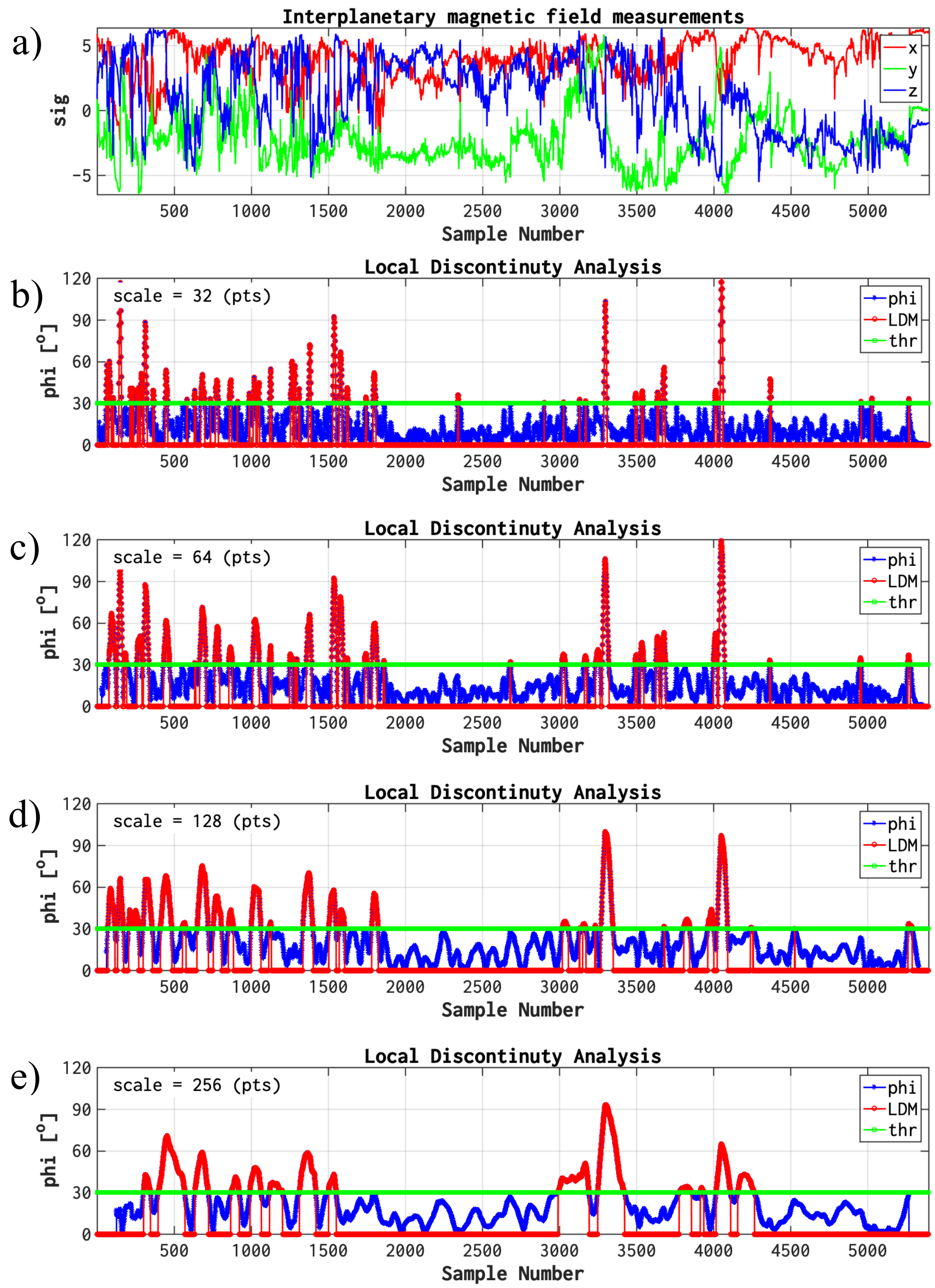

Figure 5. Computer validation of the directional discontinuity detector using interplanetary magnetic field (IMF) measurements from the Advanced Composition Explorer (ACE) spacecraft in January 06, 2003. a) IMF components as function of sample number. b-e) Local discontinuity analysis: $\varphi(t_k)$ (from (1); blue line labeled "phi"), $\mathrm{LDM}^{(\mathrm{deg})}$ (from (3); red line) and $\varphi_c = 30°$ (green line labeled "thr"). The panels depict results using windows of 32 samples (panel b), 64 (panel c), 128 (panel d) and 256 samples (panel e), respectively.

Other features can also be observed in Fig. 5. The first part of the series, up to sample number 2000, shows very different scaling properties in terms of the number of discontinuities, compared to the second part of the series. The smallest window detects the largest number of discontinuities, and, as we increase the width of the analysis window, more and more discontinuities that are close to each other start to merge. See for example the part of the signal centered around sample 1500. Three distinct discontinuities are detected by the 32-samples analysis window. Using a 64-samples window, the weakest discontinuity starts to merge with the middle amplitude one. Using 128-samples, only two discontinuities are detected, and both are much weaker compared to the strongest one detected using the 32-samples window. The 256-samples window detects only one weak discontinuity around sample number 1500.

The results described above, are similar to the results discussed by Greco et al. (2016). They also observed complex “break ups” and “ramifications” going from singular large-scale discontinuities to multiple small-scale ones, giving rise to a tangled network of primary and secondary structures. We retrieve same types of structures in our LDM results obtained for the group of discontinuities observed around sample 1500 of the signal depicted in Fig. 5.

4.2 Implementing the discontinuity detector in FPGA

The initial prototype of the directional discontinuity detection algorithm was implemented in INA (Munteanu, 2017), a MATLAB-based software analysis tool. The algorithm was tested with series of signals that highlight different kinds of discontinuities, as described in Section 4.1.

For the FPGA implementation of the detection system, we used high-level synthesis to generate the RTL implementation of the discontinuity detection algorithm as described by (2)-(5). Since the Xilinx Vivado HLS 2019.1 (Vivado, 2019 a) does not support MATLAB as programming language for the description of the algorithms, the discontinuity detection algorithm was first rewritten in C++.

To validate the C++ implementation of the algorithm, we designed and implemented a testbench in Vivado HLS which generates the series of synthetic signals and the series of measurements captured by the magnetic sensor as inputs for the discontinuity detector, and then compares the results with the ones generated by the MATLAB prototype for the same inputs.

4.2.1 Data types analysis

The Xilinx Vivado HLS provides support for all the standard C++ datatypes and also supports additional data types which can generate more efficient hardware implementations for the algorithms. Several data types were analyzed for the implementation of the discontinuity detector. The algorithm uses advanced mathematical functions which are supported by the Vivado HLS math library.

Table 1 shows the analyzed data types and the computed errors between the results generated by the MATLAB implementation and the results of the C++ implementation for the series of synthetic signals described in Fig. 6a, and the measurements captured by the magnetic sensor described in Fig. 6b. The *Max Error* column describes the maximum absolute difference between the discontinuity measures computed by the MATLAB and the C++ implementations for the same series of inputs. The *Errors Count* column reports the number of absolute differences which are higher than 0.001, an arbitrarily selected threshold. Based on these results, we selected the float data type (single-precision) for the implementation of the discontinuity detector as it provides a good trade-off between precision and efficiency of the hardware implementation.

Table 1. Analyzed data types and resulting errors.

Test Signal	Data Type	Max Error	Errors Count
Synthetic	Float	0.000000	0
Capture	Float	0.000000	0
Synthetic	Half	0.756262	90
Capture	Half	0.302604	3976
Synthetic	FXP<32,16>	1.984288	7402
Capture	FXP<32,16>	1.818750	4259
Synthetic	FXP<32,14>	1.999315	593
Capture	FXP<32,14>	1.967990	4235
Synthetic	FXP<32,20>	0.017650	4
Capture	FXP<32,20>	0.998943	5588
Synthetic	FXP<32,22>	0.170812	10
Capture	FXP<32,22>	1.435630	5910

4.2.2 Discontinuity detector analyses

The Xilinx Vivado HLS allows a direct inference of the RTL component from the C++ source code, thus ensuring a high-quality implementation in the FPGA device. In order to improve the quality of the generated RTL implementation, several code optimizations were analyzed and evaluated after circuit synthesis, based on the timing and resource utilization estimates provided by the Xilinx Vivado HLS tool.

These optimizations were not automatically generated but had to be developed by the research team. In the final version of the discontinuity detection algorithm we replaced the regular C++ arrays which store the last received measurements with shift registers. The sums of the measurements in buffers are maintained in variables which are adjusted for every new measurement received by adding the value of the new measurement and subtracting the value of the measurement leaving the buffer. This way, we eliminate the algorithm loops (see Algorithm 1) which compute the sums of the measurements in buffers.

Table 2 describes the timing estimates for the implementation of the discontinuity detector with different values for the TAU parameter. The implementation targets a clock period of 10 ns, due to the 100 MHz oscillator present on the development board which clocks the FPGA device. This timing constraint is met for all tested values of the TAU parameter.

The *Initiation Interval* defines the minimum and maximum number of clock cycles before new measurements can be accepted, while the *Latency* specifies the minimum and maximum number of clock cycles required to compute the discontinuity measure *ndp* (see (5)). Considering that we configured the magnetic sensor to take a new measure every 12 ms, these requirements are also satisfied. Also, the results show that with the shift registers optimization the timing parameters are not influenced by the size of the TAU parameter.

Table 2 also describes the estimated programmable logic resources required for the implementation of the discontinuity detector in the FPGA device with different values for the TAU parameter. The results show that the TAU parameter influences the required number of Block-RAM resources of the FPGA device, while the amount of the other programmable logic resources remains constant.

The discontinuity detectors for two arbitrarily selected values of the TAU parameter, 50 and 100, were generated and exported from Vivado HLS to be integrated with the rest of the system components (see Section 3.2) for validation.

Table 2. Vivado HLS timing and utilization estimates for the discontinuity detector.

TAU *samples*	Timing Estimates						Utilization Estimates			
	Clock		**Initiation Interval**		**Latency**		**BRAM**	**DSP**	**FF**	**LUT**
	Target	**Estimated**	*clock cycles*		*clock cycles*					
	ns	*ns*	*min*	*max*	*min*	*max*				
50	10	9.384	74	88	74	88	0	24	7562	7949
100	10	9.384	74	88	74	88	0	24	7562	7949
500	10	9.384	74	88	74	88	6	24	7400	7577
1000	10	9.384	74	88	74	88	12	24	7403	7577
5000	10	9.384	74	88	74	88	96	24	7412	7580

4.3 Validation of the discontinuity detector in FPGA

The RTL description of the discontinuity detector generated by the Xilinx Vivado HLS was integrated with rest of the system as described in Section 3. The system implemented in the Artix-7 FPGA device using the Xilinx Vivado 2019.1 (Vivado, 2019 b) suite was tested with synthetic signals and real measurements of the magnetic field received from the magnetic sensor.

The measurements and the computed discontinuity measures *ndp* are sent to the host machine for visualization and the results were compared with the ones obtained with the MATLAB implementation (adapted according to (4)). The maximum absolute error between the two implementations is 0.

Figure 6a shows the local discontinuity measures computed using FPGA for three synthetic signals with 10,000 samples each. Note that the threshold value between weak and strong discontinuities used in the MATLAB-based results from Section 4.1, i.e. $\varphi_c = 30°$, corresponds to a normalized dot product value of $ndp_c \cong 0.87$ (see Section 2.1). Also note the thresholding of φ stated that all angular changes larger $30°$ corresponded to strong discontinuities. In the case of the ndp parameter (defined in (4)) the situation is somewhat different: all normalized dot product values in the interval $[0.87, 1]$ are "below" the threshold value, and are considered weak discontinuities, while ndp values in the interval $[-1, 0.87]$, are "above" the threshold, and correspond to strong discontinuities.

Figure 6a demonstrates that the *ndp* measure is capable of accurately detecting all directional discontinuities included in the synthetic dataset. In particular, three strong discontinuities dominate the results: the first one around sample 3500, with $ndp = -1$, i.e. $\varphi = 180°$ (see (1) and (4)); the second one around sample 5000, with $ndp = 0$, i.e. $\varphi = 90°$; and the third one around sample 6000, with $ndp \cong 0.71$, i.e. $\varphi = 45°$. These strong discontinuities are all accurately detected "above" the threshold value (see the previous paragraph).

The synthetic dataset of Fig. 6a also includes a few additional weak discontinuities: a set of two around sample 2500 and another set around sample 7500. One can observe that the *ndp* measure is able to detect even these weak discontinuities, but, since they are "below" the threshold value, they will not activate the *Trigger Component* of the FPGA system (see Section 3.2.3).

Another noteworthy result in Fig. 6a is the fact that all directional discontinuities detected by the FPGA device, even the weakest ones, show a persistency of the value of the *ndp* measure as we increase the analysis window from TAU = 50 to TAU = 100 samples. This is consistent with the results illustrated in Figs. 3a and 3d (see also Fig. 5), and is due to the fact that all discontinuities in Fig. 6a are isolated structures, i.e., the time intervals between adjacent discontinuities are larger than the width of the analyzing window. This is not the case for the signal in Fig. 6b, as we describe next.

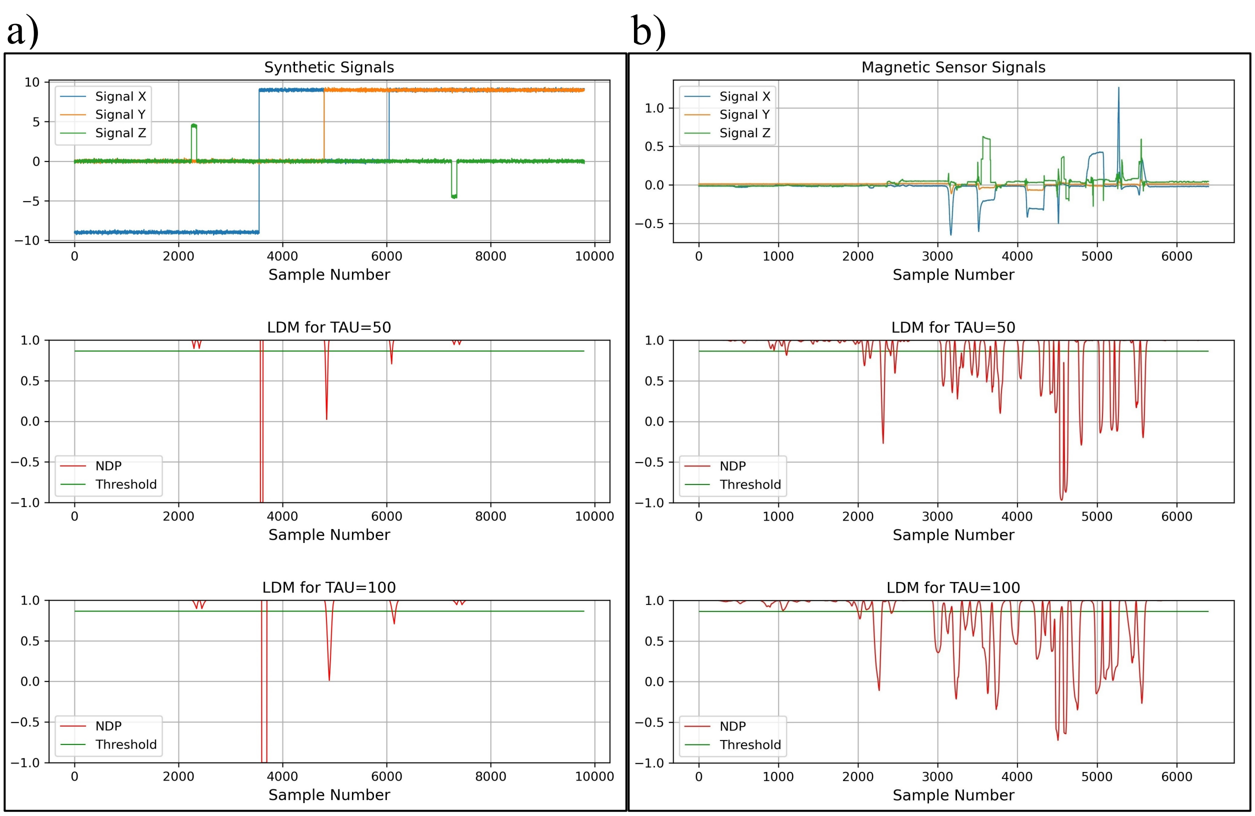

Figure 6. Validation of the discontinuity detector in FPGA using: a) synthetic signals, and b) measurements received from the magnetic sensor. In both panels, top plot shows the analysed dataset; middle plot depicts *ndp* (from (4); red line) for TAU=50 samples; the *ndp* threshold (green line) is at $ndp_c \cong 0.87$; bottom plot, same as middle plot, but for TAU=100 samples.

Figure 6b shows the results obtained for measurements received from the magnetic sensor. To artificially modify the magnetic field measured by the sensor, a magnet was used to produce the sharp variations observed in the measured magnetic field. Figure 6b demonstrates that the *ndp* measure computed by the FPGA device is capable of accurately detecting directional discontinuities even for these highly fluctuating measurements received from the magnetic sensor. One can observe that multiple weak discontinuities are detected using the small analysis window (TAU = 50 samples), but some of them are not detected as distinct events when the larger analysis window (TAU = 100 samples) is used. This is because the weaker discontinuities have merged into a single profile (similar to the case around sample 1500 in Fig. 5). A clear example can be seen around sample 2500: for TAU = 50 samples, two relatively weak discontinuities with *ndp* values slightly above the threshold limit are observed; for TAU = 100, only one of these two discontinuities is detected, with a corresponding value for the *ndp* measure equal to the threshold limit.

Other interesting features evidenced by Fig. 6b are the large changes of the peak *ndp* values as we change the width of the analysis window. This is due to the fact that adjacent discontinuities are separated mostly by time intervals which are smaller than the width of the analysis window. See, e.g., the two strongest discontinuities around sample 4500: for TAU = 50 samples, they have *ndp* values slightly below -1, i.e. $\varphi \cong 180°$; while for TAU = 100, their *ndp* values are only slightly “above” -0.5, i.e. $\varphi \cong 120°$.

4.4 Energy efficiency and device utilization

Table 3 describes the amount of programmable logic resources used by the implementation of the whole system in an Artix XC7A100T-CSG324 FPGA device for the selected values of the *TAU* parameter, 50 and 100 samples. These results, generated after implementation, show that the system uses just a small part of the FPGA device resources, allowing for other functionalities to be implemented in the same device.

We also computed power estimates for the whole system implemented in the FPGA device. Our device power analysis showed that, from a total On-Chip Power of 0.261 W (100%), the Device Static Power was 0.098 W (37%) and the Dynamic Power was 0.163 W (63%). The power analysis for the design was performed using the default environmental settings in Xilinx Vivado 2019.1 environment.

Table 3. FPGA device resource utilization.

Resource	Utilization		Available	Utilization %	
	TAU 50	*TAU* 100		*TAU* 50	*TAU* 100
LUT	7110	7491	63400	11.21	11.82
LUTRAM	522	809	19000	2.75	4.26
FF	6165	6165	126800	4.86	4.86
BRAM	3	3	135	2.22	2.22
DSP	24	24	240	10.00	10.00

5 Summary and conclusions

We designed an algorithm to continuously compute changes in the direction of a three-component vector quantity. The algorithm was first implemented in the MATLAB-based software tool Integrated Nonlinear Analysis - INA (Munteanu, 2017; see also http://www.storm-fp7.eu/index.php/data-analysis-tools). This software implementation was used for the validation and testing of the algorithm for various synthetic datasets and in-situ magnetic field measurements. The algorithm's functionality is demonstrated for magnetic discontinuities in space, but, due to its adjustable parameters, the detection method can be used to analyze any vector quantity for which rapid direction changes are important.

In space physics, the term "directional discontinuity" was originally used to denote a change in interplanetary magnetic field (IMF) direction of more than 30 degrees in less than 30 seconds. The exact threshold between weak and strong discontinuities and also their duration can vary significantly depending on the specific datasets being used and/or specific science questions being addressed. Random fluctuations in the direction of the IMF are ubiquitous, due to either natural variability and/or instrumental noise, making it almost impossible to use such fixed threshold values (e.g. 30 sec. for duration, or 30° for strength). Our algorithm uses adjustable thresholds, adjustable widths for the analysis windows and also averaging procedures in order to reduce the adverse effects of the background variability on the detection accuracy. A sliding-window algorithm is also designed in order to perform real-time monitoring and continuous discontinuity detection.

The directional discontinuity detection algorithm is adapted for and implemented on a Field-Programmable Gate Array (FPGA) device. In order to optimize the FPGA implementation, the complex calculations of angular changes used by the software implementation, which required the use of trigonometric functions, were replaced by much simpler calculations based solely on dot products, which require only basic arithmetic operations.

The FPGA design was extensively validated and tested using multiple synthetic datasets and real-life laboratory magnetometer measurements. Detailed experimental results regarding energy efficiency, power consumption, and resource utilization demonstrate the usefulness of the design and the feasibility of an FPGA-based discontinuity detector for a real-time monitoring of observables on-board spacecraft.

We also report the optimizations applied on the FPGA design in order to minimize the computational resources and ensure an efficient utilization of the limited computational and energy resources available on-board. The simplicity of the design flow, the flexibility and the short time to market of the FPGA devices, compared to ASIC, can constitute a pathway for future data analysis strategies.

The discontinuity detection algorithm described in this paper is part of a broader effort meant to build an integrated library for autonomous on-board digital signal processing. This library already includes modules designed for spectral and statistical analysis of on-board data and is able to produce a set of key data descriptors when the entire data stream cannot be sent to the ground for further analysis.

Acknowledgments

This work was supported by the Romanian Ministry of Research and Innovation via a PCCDI Grant 18PCCDI/2018 and PROGRAM NUCLEU LAPLAS. The work of M.E. was supported by ESA PRODEX CLUSTER and Belgian Solar Terrestrial Center of Excellence (STCE). The work of C.M. was supported by ESA PRODEX MISION.

Data Availability Statement

In Section 4.1.3 we tested the directional discontinuity detector using magnetic field data from the MAG instrument (Smith et al., 1998) on-board the Advanced Composition Explorer (ACE) spacecraft. We used a 24-hour time interval, at 16 sec time resolution, covering the whole day of January 06, 2003. The data are publicly available from the Coordinated Data Analysis Web (CDAWeb; https://cdaweb.gsfc.nasa.gov) maintained by the the National Aeronautics and Space Administration (NASA).

References

Artemyev, A. V., Angelopoulos, V., J. S. Halekas, Alexander A. Vinogradov, I. Y. Vasko, and L. M. Zelenyi (2018) Dynamics of Intense Currents in the Solar Wind, ApJ 859 95, https://doi.org/10.3847/1538-4357/aabe89

Artemyev, A. V., Angelopoulos, V., Vasko, I. Y., Runov, A., Avanov, L. A., Giles, B. L., et al. (2019) (a). On the kinetic nature of solar wind discontinuities. Geophysical Research Letters, 46, 1185-1194. https://doi.org/10.1029/2018GL079906

Artemyev, A. V., Angelopoulos, V., & Vasko, I. Y. (2019) (b). Kinetic properties of solar wind discontinuities at 1 AU observed by ARTEMIS. Journal of Geophysical Research: Space Physics, 124, 3858-3870. https://doi.org/10.1029/2019JA026597

Burlaga, L.F. (1969) Directional discontinuities in the interplanetary magnetic field. Sol Phys 7, 54–71. https://doi.org/10.1007/BF00148406

Borovsky, J. E. (2008), Flux tube texture of the solar wind: Strands of the magnetic carpet at 1 AU? J. Geophys. Res., 113, A08110, doi:10.1029/2007JA012684

Borovsky, J. E. (2010), Contribution of Strong Discontinuities to the Power Spectrum of the Solar Wind, Phys. Rev. Lett., 105 (11), https://link.aps.org/doi/10.1103/PhysRevLett.105.111102

Burch, J.L., Moore, T.E., Torbert, R.B. et al. (2016) Magnetospheric Multiscale Overview and Science Objectives. Space Sci Rev 199, 5–21, https://doi.org/10.1007/s11214-015-0164-9

Burkholder, B. L., and Otto, A. (2019). Magnetic reconnection of solar flux tubes and coronal reconnection signatures in the solar wind at 1 AU. Journal of Geophysical Research: Space Physics, 124, 8227-8254. https://doi.org/10.1029/2019JA027114

Chian, A. C.-L. and Muñoz, P. R. (2011) Detection of Current Sheets and Magnetic Reconnections at The Turbulent Leading Edge of An Interplanetary Coronal Mass Ejection, ApJL 733 L34, https://doi.org/10.1088/2041-8205/733/2/L34

Deak, N.; Creţ, O.; Echim, M.; Teodorescu, E.; Negrea, C.; Văcariu, L.; Munteanu, C.; Hângan, A. (2018), Edge computing for space applications: Field programmable gate array-based implementation of multiscale probability distribution functions, Review of Scientific Instruments, vol. 89, no. 12, p. 125005, https://doi.org/10.1063/1.5044425

Deak, N., Creț, O., Munteanu, C., Teodorescu, E., & Echim, M. M. (2021) FPGA design for on-board measurement of intermittency from in-situ satellite data, Earth and Space Science, 8, https://doi.org/10.1029/2021EA001678

Fox, N.J., Velli, M.C., Bale, S.D. et al. (2016) The Solar Probe Plus Mission: Humanity's First Visit to Our Star. Space Sci Rev 204, 7–48, https://doi.org/10.1007/s11214-015-0211-6

French, M. et al. (2018) SpaceCubeX2 - Heterogeneous On-board Processing for Distributed Measurement and Multi-Satellite Missions, NASA Earth Science Technology Forum (ESTF).

Greco, A., Chuychai, P., Matthaeus, W. H., Servidio, S., and Dmitruk, P. (2008), Intermittent MHD structures and classical discontinuities, Geophys. Res. Lett., 35, L19111, doi:10.1029/2008GL035454

Greco, A. and Perry, S. (2014) Identification of high shears and compressive discontinuities in the inner heliosphere, ApJ 784 163, https://doi.org/10.1088/0004-637X/784/2/163

Greco, A., S. Perri, S. Servidio, E. Yordanova, and P. Veltri (2016) The complex structure of magnetic field discontinuities in the turbulent solar wind, ApJL 823 L39, https://doi.org/10.3847/2041-8205/823/2/L39

Greco, A., Matthaeus, W.H., Perri, S. et al. (2018) Partial Variance of Increments Method in Solar Wind Observations and Plasma Simulations. Space Sci Rev 214, 1, https://doi.org/10.1007/s11214-017-0435-8

Haaland, S., Munteanu, C., and Mailyan, B. (2010), Solar wind propagation delay: Comment on "Minimum variance analysis-based propagation of the solar wind observations: Application to real-time global magnetohydrodynamic simulations" by A. Pulkkinen and L. Raststätter, Space Weather, 8, S06005, doi:10.1029/2009SW000542

Hanafi, A., M. Karim, I. Latachi, T. Rachidi, S. Dahbi and S. Zouggar (2017) FPGA-based Secondary On-Board Computer System for Low-Earth-Orbit Nano-satellite, Conference on Advanced Technologies for Signal and Image Processing (ATSIP), doi: 10.1109/ATSIP.2017.8075514.

Huber, F., P. Behr, H. P. Röser and S. Pletner (2007) FPGA Based On-Board Computer System for the "Flying Laptop" Micro-Satellite, Proceedings of DASIA 2007, European Space Agency.

Kintex (2020) RT Kintex UltraScale FPGAs for Ultra High Throughput and High Bandwidth Applications WP523 (v1.0)

Kuwahara, T. (2009) FPGA-based Reconfigurable On-board Computing Systems for Space Applications, Ph.D. Thesis, Institute of Space Systems University at Stuttgart.

Lepping, R. P., and Behannon, K. W. (1986), Magnetic field directional discontinuities: Characteristics between 0.46 and 1.0 AU, J. Geophys. Res., 91 (A8), 8725-8741, doi:10.1029/JA091iA08p08725

Li, G. (2008). Identifying Current-sheet-like Structures in the Solar Wind. ApJ 672, L65–L68, https://doi.org/10.1086/525847

Mailyan, B., Munteanu, C., and Haaland, S. (2008) What is the best method to calculate the solar wind propagation delay? Ann. Geophys., 26, 2383–2394, https://doi.org/10.5194/angeo-26-2383-2008

Miao, B., Peng, B., and Li, G. (2011) Current sheets from Ulysses observation, Ann. Geophys., 29, 237–249, https://doi.org/10.5194/angeo-29-237-2011

Microsemi (2015) - Mitigation of Radiation Effects in RTG4 Radiation-Tolerant Flash FPGAs (2015) WP0191 White Paper.

Munteanu, C., Haaland, S., Mailyan, B., Echim, M., and Mursula, K. (2013) Propagation delay of solar wind discontinuities: Comparing different methods and evaluating the effect of wavelet denoising, J. Geophys. Res. Space Physics, 118, 3985– 3994, doi:10.1002/jgra.50429

Munteanu, C. (2017) Turbulent fluctuations and discontinuities in the solar wind: statistical properties and possible effects on the terrestrial plasma environment, Ph.D. Thesis, Faculty of Physics, Doctoral School of Physics, University of Bucharest, Romania.

Newman, R., Vainchtein, D. and Artemyev, A. (2020) Solar wind transient currents: statistical properties and impact on Earth's magnetosphere. Sol Phys 295, 129, https://doi.org/10.1007/s11207-020-01695-z

Opincariu, L., Deak N., Creţ O., Echim M., Munteanu, C. and L. Văcariu (2019), Edge computing in space: Field programmable gate array-based solutions for spectral and probabilistic analysis of time series, Review of Scientific Instruments, 90, 114501, https://doi.org/10.1063/1.5119231

Paschmann, G., Haaland, S., Sonnerup, B., and Knetter, T. (2013) Discontinuities and Alfvénic fluctuations in the solar wind, Ann. Geophys., 31, 871–887, https://doi.org/10.5194/angeo-31-871-2013

Perri, S., Goldstein, M. L., Dorelli, J. C. and Sahraoui, F. (2012) Detection of Small-Scale Structures in the Dissipation Regime of Solar-Wind Turbulence, Phys. Rev. Lett., 109, 19, https://link.aps.org/doi/10.1103/PhysRevLett.109.191101

Pingree, P.J. (2010) Advancing NASA's on-board processing capabilities with reconfigurable FPGA technologies: Opportunities and implications, IEEE International Symposium on Parallel Distributed Processing, Workshops, and PhD Forum (IPDPSW), Atlanta, GA.

Smith, C. W., J. L'Heureux, N. F. Ness, M. H. Acuña, L. F. Burlaga, and J. Scheifele (1998) The ACE magnetic fields experiment, Space Sci. Rev., 86, 613–632, doi:10.1023/A:1005092216668.

Turicu, D. C., O. Creţ, M. Echim and C. Munteanu (2022) An FPGA-Based Solution for Computing a Local Stationarity Measure from Satellite Data, IEEE Access, 10, 9668-9676, doi: 10.1109/ACCESS.2022.3143239.

Tsurutani, B. T., and Smith, E. J. (1979), Interplanetary discontinuities: Temporal variations and the radial gradient from 1 to 8.5 AU, J. Geophys. Res., 84(A6), 2773-2787, doi:10.1029/JA084iA06p02773

Vasquez, B. J., Abramenko, V. I., Haggerty, D. K., and Smith, C. W. (2007), Numerous small magnetic field discontinuities of Bartels rotation 2286 and the potential role of Alfvénic turbulence, J. Geophys. Res., 112, A11102, doi:10.1029/2007JA012504.

Virtex (2014) Space-Grade Virtex-4QV Family Overview DS653 (v2.1).

Virtex (2018) Radiation-Hardened, Space-Grade Virtex-5QV Family Data Sheet: Overview DS192 (v1.6).

Vivado (2019) (a) - Vivado High-Level Synthesis User Guide 2019.1 (2019), https://www.xilinx.com/support/documentation/sw_manuals/xilinx2019_1/ug902-vivado-high-level-synthesis.pdf

Vivado (2019) (b) - Vivado Design Suite User Guide 2019.1. (2019), https://www.xilinx.com/support/documentation/sw_manuals/xilinx2019_2/ug901-vivado-synthesis.pdf

Zhdankin, V., Boldyrev, S., Mason, J. and Perez, J. C. (2012) (a), Magnetic Discontinuities in Magnetohydrodynamic Turbulence and in the Solar Wind, Phys. Rev. Lett., 108, 17, https://link.aps.org/doi/10.1103/PhysRevLett.108.175004

Zhdankin, V., Boldyrev, S. and Mason, J. (2012) (b), Distribution of magnetic discontinuities in the solar wind and in magnetohydrodynamic turbulence, ApJL, 760, 2, https://doi.org/10.1088/2041-8205/760/2/L22