

# BRAIN – Getting Started

*Paul Wilcox, University of Bristol, May 2020*

## Introduction and installation

In lieu of a formal manual, this is a short guide to getting started with **BRAIN**. Matlab commands look like this; file names, **folders and items in BRAIN look like this**.

The most recent release of BRAIN is downloadable as a zip file from <https://github.com/ndtatbristol/brain1/releases>.

Download the zip file and unzip it. There will be a folder called something like **brain1-1.90** (exact name depends on the version) and depending on how you unzipped it, there may be an identically folder below that. Eventually you should get to a folder containing about 9 subfolders (**Analysis, array processing**, etc.) and lots of Matlab files and functions, one of which is called **brain.m**, which is the entry point for **BRAIN**. To start **BRAIN**, you need to run **brain.m** from within the Matlab environment. To do this, the folder containing **brain.m** and all its subfolders need to be on the Matlab path so before you run brain for the first time type:

```
addpath(genpath(BRAIN_FOLDER_NAME))
```

at the Matlab command prompt, where `BRAIN_FOLDER_NAME` is the full path to the folder containing **brain.m** enclosed in single quotes, e.g. `'C:\Users\someone\Desktop\brain1-1.90\brain1-1.90'`. This step only needs to be repeated if you reset the Matlab path – in general the path is persistent between sessions unless you have a Matlab start-up script that resets it.

Subsequently you just need to type `brain` at the Matlab command prompt to start **BRAIN**. Note that the version of **BRAIN** generates a new persistent folder in the current Matlab working directory (i.e. where you were when you typed `brain`) called **Brain (local data)**. The subfolders with contain user-defined arrays, materials etc. These will be accessible if **BRAIN** is run again from the same working directory, but if **BRAIN** is run from a different working directory a new **Brain (local data)** folder is created and only the contents of this folder will be accessible in **BRAIN**.

## Acquisition

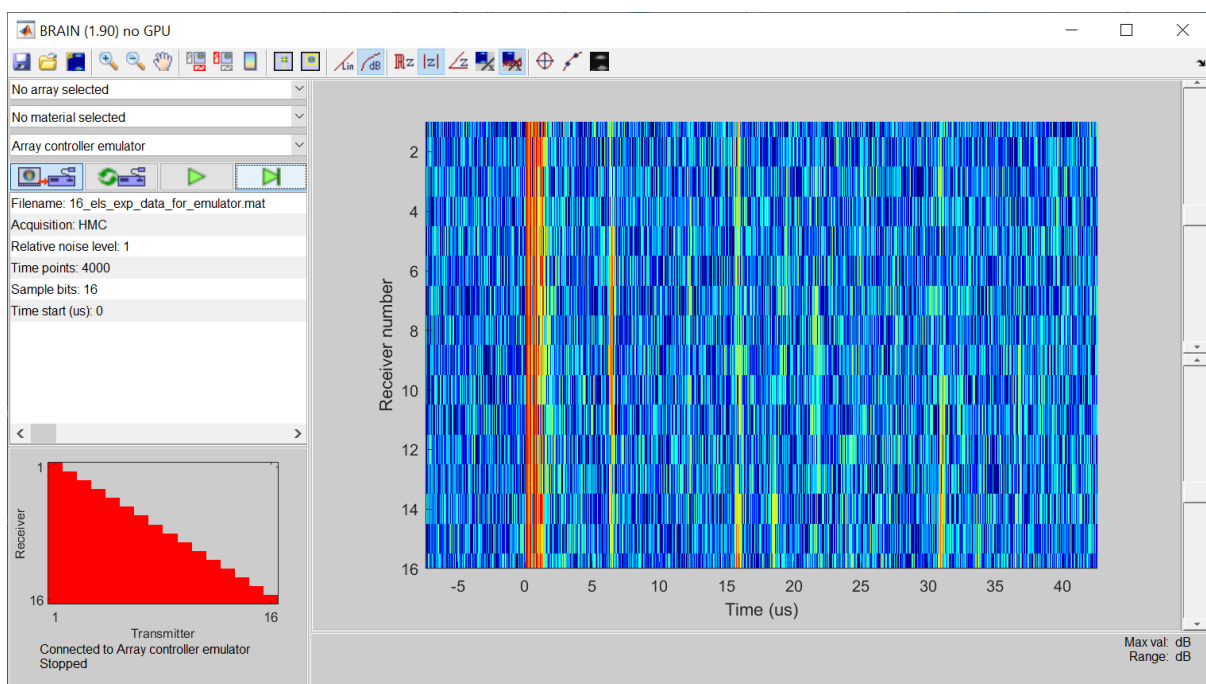
The first thing that should appear after launching **BRAIN** is the **Acquisition window** shown below. On the left of the screen is a group of four large buttons:



To get started, click the leftmost button (**Connect**). This will 'connect' **BRAIN** to an array controller emulator, at which point the other three buttons in the set will be enabled. The meaning of the four buttons is described in the table below:

Button	Description
	Connect/disconnect to selected array controller
	Reset array controller
	Start/stop continuous acquisition of frames of raw data
	Acquire single frame of raw data

To 'acquire' raw data from the emulator, click the third button (**Start continuous**) and the **Acquisition window** should now look like this with the main image updating continuously:



The main image in the **Acquisition window** shows a subset of the current raw data (in general the pulse-echo A-scans from each element in the array, which is equivalent to the A-scans in the leading diagonal of the full matrix of array data). This image updates each time a new frame of data is acquired. The red triangle at the bottom left of the **Acquisition window** provides a graphical snapshot of the data raw data acquired, in terms of which A-scans in the full matrix are present. Here, half matrix capture has been used so only half the matrix is populated. The colour indicates the peak amplitude of the data in each A-scan (if the cross-hair cursor is on it is the peak amplitude of data later in time than the position of the cross-hair cursor – see below). The two sliders on the right hand side of the **Acquisition window** control the amplitude range shown in the main image: the lower slider is the dynamic range in dB (i.e. the difference between the smallest and largest value shown)

and the upper slider sets the upper limit of the amplitude range in dB (relative to whatever the image is currently normalised to).

The main menu bar in the **Acquisition window** looks like this:



The meaning of the buttons is described in the table below (many of these are also used on **Imaging windows** showing processed images):

Button	Description
	Saves current frame of raw data
	Opens a previously saved file of raw data
	Launches a new imaging process
	Zoom in (mouse acts as a cursor to draw a zoom box)
	Zoom out (to extent of available data)
	Pan (mouse changes to a hand tool for dragging and panning image)
	Toggle horizontal cross section of image at cross-hair on/off
	Toggle vertical cross section of image at cross-hair on/off
	Toggle image colour bar on/off
	Toggle cross-hair (point) selection on/off
	Toggle box (rectangular region) selection on/off
	Linear image scale
	Logarithmic image scale (dB)
	Show image of real part of data (also switches to linear image scale)
	Show image of modulus (amplitude) of data
	Show image of argument (phase angle) of data
	Normalise image according to values at current cross-hair selection or in box selection
	Toggle automatic normalisation of image to peak value on/off
	Toggle unity aspect ratio on/off
	Toggle Fourier interpolation on/off
	Toggle greyscale image on/off

You can investigate the operation of these buttons with the example data currently on the screen.

Below the four larger buttons is a table with acquisition parameters related to the current array controller. The contents of this table depend on the array controller that is in use, but typically allow control of sampling frequency, gain, A-scan points etc. For the emulator, only a few parameters can be changed.

Filename: 16\_els\_exp\_data\_for\_emulator.mat

Acquisition: HMC

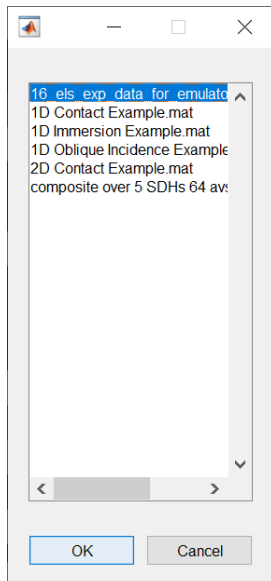
Relative noise level: 1

Time points: 4000

Sample bits: 16

Time start (us): 0

To change an item in such a table (which are used throughout the **BRAIN** user interface), just click on it. This will bring up a dialogue box where a new value can be entered or selected, depending on the sort of parameter. For example, in the case of the emulator, if you click on the **Filename** line, you will be able to select a different data file for the emulator's raw data:

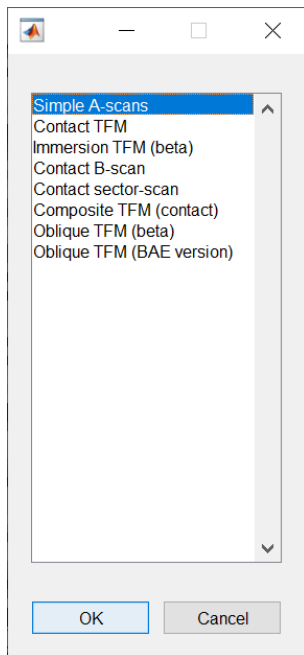


*Note: The emulator files are located in **BRAIN\_FOLDER\_NAME\Instruments\Emulator data**. Under normal circumstances, existing data files are access through the file open button in the main menu. However, FMC or HMC files can be added to the emulator data folder if desired and they will then also be accessible from the emulator menu. Raw data files accessed through the emulator can be manipulated before processing through the emulator menu (for example, by having random noise added, the bit-depth altered etc.) where as raw data files access through the open button cannot be altered.*

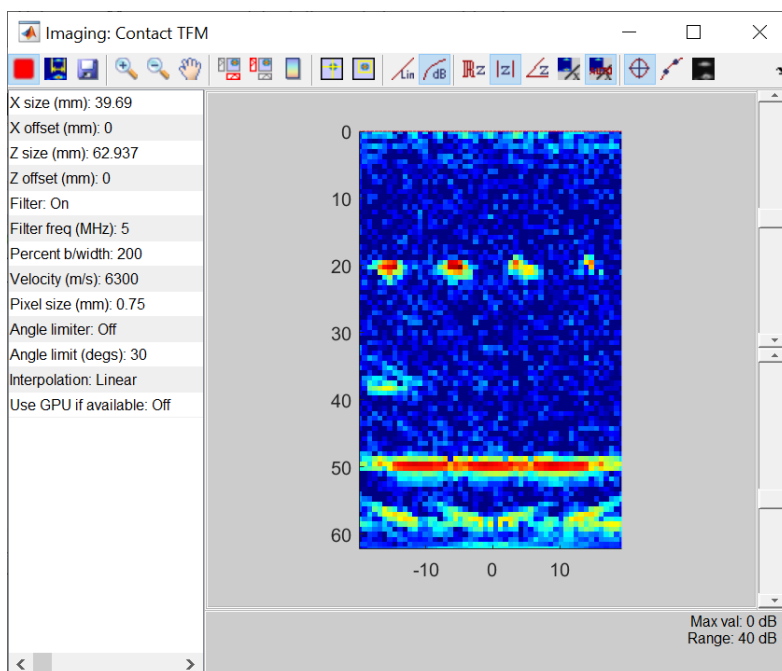
Try selecting **1D Contact Example.mat** for the emulator data file. You will see that the raw data no longer fills the image region. **BRAIN** (intentionally) does not alter the extents of the image displayed unless told explicitly to do so. Click the Zoom out button so the full extent of the data fills the image.

## Imaging



For real array controllers, it is necessary to select the array and sample material prior to imaging, but for the emulator this information is already in the data file. Click the **Launch imaging window** button in the main menu in the **Acquisition window** and a list of available imaging algorithms will appear in a dialogue box.




Here, select **Contact TFM** from the list and wait for a new **Imaging window** to appear. The frames/second rate shown at the bottom left of the **Acquisition window** will drop significantly due to the extra processing required to form an image (less significantly if a compatible GPU has been detected).



The buttons in the menu bar in the **Imaging window** are similar to the **Acquisition window**, with some additions and changes in functionality:

Button	Description
	Saves either the entire window as a Matlab figure file, or the image data itself as a Matlab data file.
	Pauses the refreshing of the image window when new raw data is acquired.

	Launches a new analysis process related to the imaging window (e.g. defect sizing)
---	--

The table in the **Imaging window** shows the current parameters for the imaging algorithm. Again, these can be edited by clicking on the relevant line in the table. Some of the initial values for the imaging parameters are estimated from the raw data, such as **X size** and **Z size** of the image (relates to temporal duration of A-scans and number of elements in the array) and the **Filter freq.** (set to match the nominal centre frequency of the array).

Note that the image in an **Imaging window** is refreshed whenever an imaging parameter is changed. On slow computers or for complex imaging algorithms this can be quite tortuous. As the computational burden of many imaging algorithms scales with the number of pixels, it makes sense to adjust other parameters before increasing resolution (by reducing **Pixel size**).








Multiple **Imaging windows** using either the same or different imaging algorithms can be launched in parallel. Each imaging window will refresh each time a new frame of data is acquired (unless the window is paused) and hence the more imaging windows open, the slower the overall frame rate. The values for any imaging parameters common to the most recently-launched imaging algorithm will be used as the initial values in a new **Imaging window**.

## Analysis

From an imaging window, click the **Launch analysis** button. This will bring up a dialogue box showing the available analysis functions. Note that some require regions in images to be selected first in an **Imaging window**. Also, note that the data used by an **Analysis window** is static; unlike the data in an **Imaging window** it does not refresh when new raw data is acquired.

## Connecting to real array controllers

Launch **BRAIN** and select the desired array controller from the third dropdown menu on the left of the **Acquisition window**. The table below will update to show parameters that depend on the array controller, for example:

No array selected	▼
No material selected	▼
Peak NDT Micropulse TCPIP	▼
    	
IP address: 10.1.1.2	
Port number: 1067	
Acquisition: HMC	
Sample frequency (MHz): 25	
Pulse voltage (V): 100	
Pulse width (ns): 80	
Time points: 1000	
Sample bits: 8	
Gain (dB): 40	
Filter number: 4	
Maximum PRF (kHz): 2	
Averages: 1	
Time start (us): 0	
Instrument delay (ns): 0	
 	

Some of these (in this case **IP address** and **Port number**) relate to the connection to the array controller. Ensure these are set correctly and then press the **Connect** button. If all goes well, a note will appear at the bottom of the **Acquisition window** that says that **BRAIN** is connected to the array controller.