DFO Team Whale

Beaked Whale Detection Procedure

Wilfried Beslin and Joy Stanistreet, April 2023

Table of Contents

[1. INTRODUCTION 2](#_Toc133233250)

[2. WALKTHROUGH 2](#_Toc133233251)

[2.1 File Setup 2](#_Toc133233252)

[2.2. General Click Detection using Triton 2](#_Toc133233253)

[2.2.1 STS -- > Hi Res Proc Stream 2](#_Toc133233254)

[2.2.2 Clean up metadata (optional) 3](#_Toc133233255)

[2.3. Automatic Event Detection 3](#_Toc133233256)

[2.4. Event Validation and Species Identification 4](#_Toc133233257)

[2.5. Post-Processing 6](#_Toc133233258)

# 1. INTRODUCTION

The purpose of this document is to provide a quick, step-by-step walkthrough of our process for detecting beaked whale events from acoustic recordings, using Triton and our in-house beaked whale detection code (BWD). For a comprehensive description of each feature in BWD, refer to the Reference Manual.

# 2. WALKTHROUGH

## 2.1 File Setup

The first steps in preparing to analyze a new dataset should be to do the following:

* Create a folder to store all WAV files to be analyzed. Move all WAV files within the period of interest to that folder.
* Create an output folder for the dataset to be analyzed. This can have any name you want, but best practice is to name it based on the deployment name (e.g. *MGL\_2015\_05*).
* Within this output folder, create another folder called *metadata*. This will store the output produced by Triton (.c and .cTg files).

## 2.2. General Click Detection using Triton

Initial click detection with Triton is done in two parts: “Batch Short-Time Spectrum”, which will find general locations of clicks, and “Batch High-Res Clicks”, which will find specific start and end times of clicks based on Teager-Kaiser energy estimation. Using Triton version 1.0 2021 09 21 (available at <https://github.com/MarineBioAcousticsRC/Triton>), these two processes can be set up to run sequentially as a single “process stream.”

To run Triton, open MATLAB and type “triton” in the command window

* Note: if this does not work, make sure the Triton code folder is on the MATLAB path

Required step for first time use: add the click detector to Triton. Triton comes with a number of add-on tools, called “Remoras”, including the click detector which is used in Section 2.2.1 below. To add the Detector Remora to Triton:

* In the Triton control window, select *Remoras 🡪 Add Remora*, then navigate to the “Detector” folder inside the Triton/Remoras folder, and click *Select Folder.*
* Triton will prompt you to restart in order to load the Remora. The *Remoras* menu should now have a *Detector* submenu.

### 2.2.1 STS -- > Hi Res Proc Stream

* Select *Remoras 🡪 Detector 🡪 STS -- > Hi Res Proc Stream*
* In *Short Time Spectrum Detection Parameters*, select *load*, then enter the path to the *Triton\_detParams\_8dB.spec.prm* file (located in the BWD folder)
* In *Metadata*, enter the path to the *metadata* folder you created in the output directory
* In *Files*, click *Specific files*, then set *Base Folder* to the path containing your WAV files
* A complete list of WAV files should appear – this may take several seconds to show up, depending on the number of files
* Click large *Okay* button on bottom left
* Click large *Okay* button on bottom left of next window (*Associate labels with files*)
* In the next window, click *Specific files*, then set *Base Folder* to the path containing your WAV files
* A complete list of WAV files should appear – this may take a few seconds to show up
* Click large *Okay* button on bottom left
* In the next window, set *Output Folder* to your *metadata* path, then click *Okay*

Triton will now run the batch short time detection process to detect echolocation clicks and generate .c files within the *metadata* folder, containing rough time stamps for each detected click. When that step is complete, Triton will run the batch high res click detection process on the rough time stamps from the initial detections, using a Teager-Kaiser energy detector to produce .ctg files with precise start/end times of clicks.

### 2.2.2 Clean up metadata (optional)

At this point it is useful to run the *cleanMetadata.m* script located in the *Supplementary\_Scripts* folder to remove extra files created by Triton that will not be used in the click analysis (only .c and .cTg files are used). To do this, open the script and edit the path to the *metadata* folder containing the Triton output, then run the script.

## 2.3. Automatic Event Detection

* If Triton is open, close it by selecting *File* *🡪 Exit* in the *Control* window
* Open the script *BWD\_master.m* in the MATLAB editor
  + you can do this by clicking on the script file in MATLAB’s *Current Folder* window, or from the command window by typing *edit BWD\_master*; if the command window option does not work (i.e. it creates a blank page), then the BWD directory is not on the MATLAB path
  + If the correct version of BWD/TWD is not on the MATLAB path, it can be added by running the script *TWDPathTool* located in the TWD folder
* Edit the lines where prompted to set input parameters, according to the following table (Based on Reference Manual **Table 4.6**):

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| depName | Name of the deployment/dataset |
| dirPath\_analysis | Full path to your output directory |
| dirPath\_audio | Full path to your WAV file directory |
| dirName\_detResults | Name of the results folder |
| detProtocol | Name of the detection protocol you want to use (one of the folders within *DetectionCriteria*)  ***Suggested default*:** GeneralWithDiscriminators for multi-species beaked whale analysis in a typical dataset. |
| segDur | This parameter controls segmentation of audio files for continuous data. Must be specified as a MATLAB duration object, which uses the syntax *duration(h,m,s)*.   For example, to divide a recording into 1-minute segments, type *duration(0,1,0)*.  To use recording durations as segments, type *duration(Inf,Inf,Inf)*.  ***Suggested default*:** duration(0,1,0) for any dataset where .wav files are longer than 1 minute; duration(Inf,Inf,Inf) for datasets where .wav files are approximately 1 minute or shorter in length. |
| nfft | Number of points to use for FFT  ***Suggested default*:** 512 for data collected at sampling rates between 192 and 256 kHz. |
| nMATClicksMax | Maximum number of clicks that can be stored in an output MAT file. If a recording has more clicks than this, it will be divided into multiple MAT files. |

* Run the script

This will create a results directory in your analysis folder, which will contain one or more Excel spreadsheets labelled “RawEvents”. The names and number of sheets depends on the detection protocol you selected.

## 2.4. Event Validation and Species Identification

* Open the script *BWDValidate\_master.m* in the MATLAB editor (again, you should be able to do this by typing *edit BWDValidate\_master* in the command window)
* Edit the lines where prompted to set input parameters, according to the following table (Based on Reference Manual **Table 4.7**):

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| dirPath\_analysis | Full path to your output directory.  Make sure this is consistent with what you entered in *BWD\_master*. |
| dirName\_detResults | Name of the results folder |
| targetName | Name of target to analyze (e.g., ‘beaked’ or ‘Ha’). There must be a *RawEvents* and/or *Validated* spreadsheet containing this target name. |
| eventMergeOpt | A string that controls which definition to use for the start and end time of a detection event. Options are as follows:  ‘none’ – use raw segment durations as the basis for event durations.  ‘timegap’ – segments with positive detections that occur within a certain time period of one another will be merged into single large events  ‘timebin’ – use specified time bin duration as the basis for event durations. A given time bin will be considered a detection event if it contains at least one segment with beaked whale detections.  ‘calendar’ – use fixed calendar durations as the basis for event durations. A given calendar period will be considered a detection event if it contains at least one segment with beaked whale detections. |
| eventMergVal | Complementary parameter to *eventMergeOpt*. It represents different things depending on what you entered for *eventMergeOpt*:  If *eventMergeOpt* = ‘none’, then this parameter is not used.  If *eventMergeOpt* = ‘timegap’, then this parameter is a duration object (syntax is (*h,m,s*) ) representing the minimum separation that two segments with positive detections may have to be considered separate events.  If *eventMergeOpt* = ‘timebin’, then this parameter duration object (syntax is (*h,m,s*) ) representing the desired time bin. For example, to analyze click events on the basis of 10-minute time bins, use duration(0,10,0).  If *eventMergeOpt* = ‘calendar’, then this parameter is a string specifying the calendar duration. Valid options are the same as for the *unit* argument of MATLAB’s *dateshift* function; this includes ‘hour’, ‘day’, ‘week’, ’month’, and ‘year’. |
| nClicksMax | Maximum number of clicks to include for event validation. Set to *Inf* to disable. |
| fRange | Range of frequencies (in Hertz) to display on click spectrogram plots, set as a two-element numeric vector (e.g., [0, 100]).  ***Suggested default*:** [0, 100] |

* Run the script

This will start the interactive event validation/species identification application. On first run, this will produce a “Validated” Excel spreadsheet from the “RawEvents” sheet. The application will update this sheet directly as you work through it.

Among other options, the interactive application will let you to enter a code for each event specifying which species you identify as being present. For guidance on how to properly enter codes and how to use the application in general, consult the Reference Manual (Section 3. Manual Event Validation and Species Identification).

## 2.5. Post-Processing

After the “Validated” spreadsheet has been filled out, there are a few more operations you can do:

* Use the script *subsetValidated.m* in the *Supplementary\_Scripts* folder to isolate events of interest and create duplicates of the “Validated” sheet containing those events only. This can be useful for going back to unknowns.
* Use the script *createPresenceTable.m* in the *Supplementary\_Scripts* folder to convert numeric species ID codes into presence/absence scores

These are discussed further in the Reference Manual (section 4.4 Additional Scripts).