

Bathymetry.jl : Bathymetry Processing, Visualization, and Export

This script demonstrates a full workflow using the EcoSons package for sonar bathymetry processing. It loads raw sonar data, detects the seafloor (bottom), computes bathymetry, applies optional corrections, plots results, and exports processed data. All parameters are driven by a JSON config file.

1. Compute Bathymetry (Line 20)

```
baths = compute_bathymetry(data)
```

- Calculates bathymetric depths for all pings based on bottom picks.

2. Prepare for Optional Tide Correction and Transect Selection (Lines 22–30)

```
dir_tide = joinpath(@__DIR__, "..", "data", "tide.dat")
transect = config["transect"]
bath = baths[transect]
plot_bathymetry_line(1:length(bath.depth), -bath.depth)

# tideCorrection!(baths, dir_tide) # currently commented out
```

```
bath = baths[transect]
plot_bathymetry_line(1:length(bath.depth), -bath.depth)
```

- Builds path to tidal correction data (unused but ready).
- Selects a specific transect from the bathymetry results.
- Plots the bathymetry profile (depth values negated to reflect downward depth).
- (Optional) tide correction is available but disabled here.

3. Save Bathymetry Data (Lines 32–36)

```
JLD2_path = joinpath(@__DIR__, config["JLD2_dir"]["bath"])
saveJLD2(JLD2_path, baths)
```

- Constructs output path from config.
- Saves the bathymetry data in JLD2 format for efficient future loading.

4. Prepare Data for Export and Export (Lines 38–44)

```
ntr, utmCoords, xCoord, yCoord, znCoord, depth, time =  
preproc_transects(  
    baths, data; sel = config["transects"]["sel"], use_utm =  
    config["transects"]["use_utm"])
```

```
bathymetry_path = joinpath(@__DIR__,  
    config["bathymetry"]["output_file"])  
export_bathymetry(ntr, utmCoords, xCoord, yCoord, znCoord, depth,  
    time, bathymetry_path)
```

- Processes selected transects into coordinate arrays, optionally using UTM projections.
- Defines export path and writes the bathymetry and metadata in text or structured format for downstream use.

Summary

- Config-driven loading of raw sonar files and channel selection.
- Bottom detection applied with user-configurable parameters.
- Bathymetry computed per transect and plotted for inspection.
- Optional tide correction supported (commented out).
- Saves processed data to .jld2 binary and exports human-readable bathymetry data.
- Modular and scalable design suitable for batch sonar survey processing.