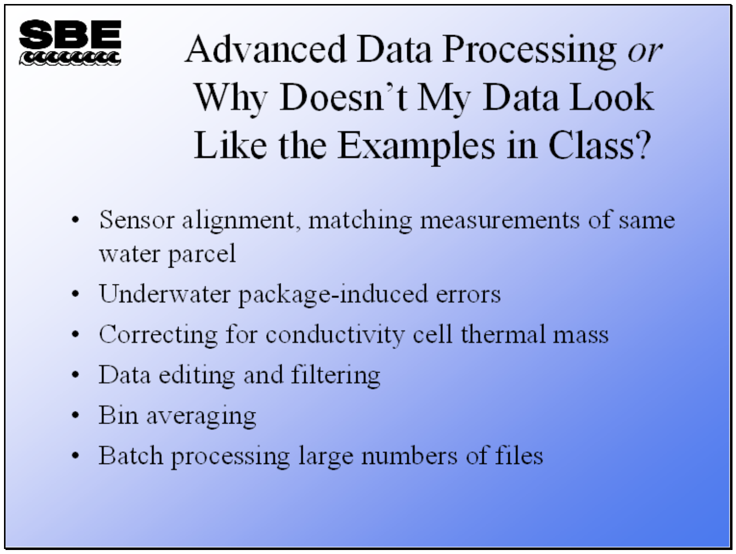
Module 9

Advanced Data Processing

Overview



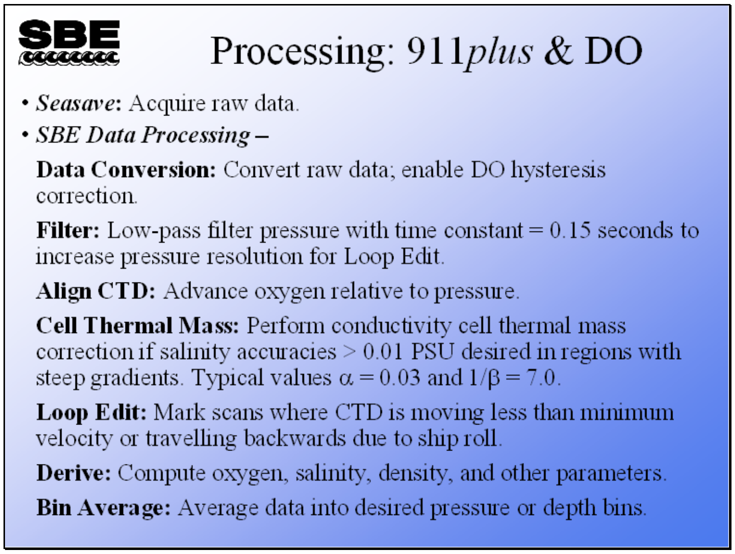
This section of the course is the final topic in profiling. Some of it is fine tuning of your data to remove small artifacts of frequency counting, plumbing, and sensor physics. We will also discuss the removal of the fairly gross effects of ship heave. Understanding these topics will help explain most of the peculiar things that you might observe in your data if you look closely.

Finally, we will talk about bin averaging your final data, and batch processing of large numbers of files.

When we finish this module you should be able to:

- Align your conductivity and temperature data relative to pressure.
- Filter your conductivity data so it matches the time response of your temperature data in an SBE 19*plus*.
- Align your dissolved oxygen data relative to pressure.
- Remove the effects of conductivity cell thermal mass from your data.
- Remove data artifacts caused by ship heave.
- Bin Average to reduce your data set, producing data at even pressure or depth values.
- Batch process large number of files.

Data Processing Steps: SBE 9*plus* / 11*plus*

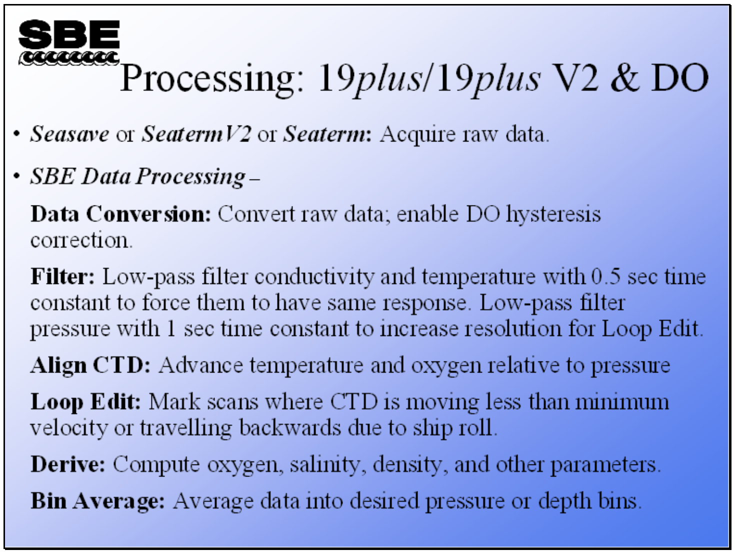


This is an ordered list of the steps in acquisition and processing of CTD data gathered with the 911*plus*. This list has quite a few more steps than our earlier discussion of the basics. We will work our way through the list, first discussing the cause of the artifact that we are interested in applying some computational energy to, and then discussing the tool to apply it.

Notes:

1. Enable the oxygen deep-water hysteresis correction on the Miscellaneous tab in Data Conversion.
2. Data is bin averaged after the major derived quantities are computed. Salinity, DO density, etc, are functions of T, C, and P; these are calculated on the final values of T, C, and P rather than the intermediate values. We'll talk a bit more about this at the end of the module.

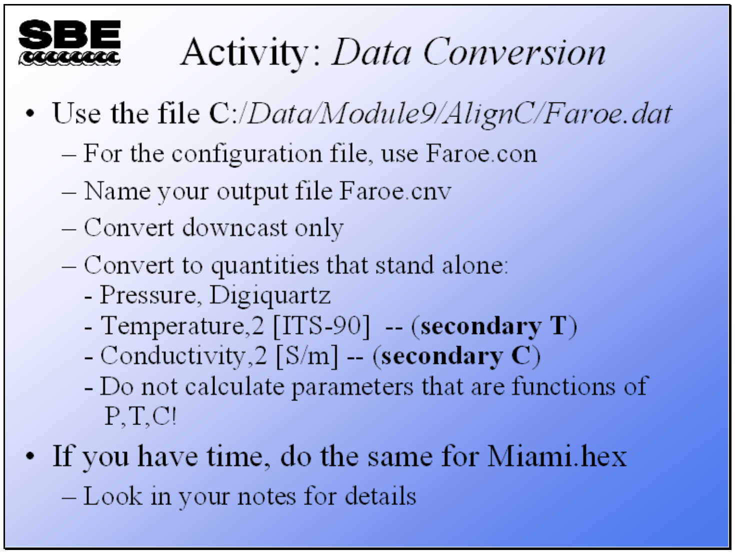
Data Processing Steps: SBE 19*plus* or 19*plus* V2



The processing list for the 19*plus* or 19*plus* V2 is shorter because of the lower expectations of precision and the different acquisition electronics.

- Use Seasave if acquiring real-time data.
- Use one of the terminal programs if uploading data from the instrument memory: SeatermV2 for the 19*plus* V2, Seaterm for the 19*plus*.

Activity



We are preparing to operate on the data with an application that moves the T and C data streams relative to the pressure data stream. Calculation of parameters that are functions of T, C, and P is not useful at this stage. Further, it will complicate and confuse things to have them in the data set before we are ready for them.

If you have time, process the 19*plus* data in C:\Data\Module9\AlignC\SBE19plus\Miami.hex, using Miami.con. The 19*plus* has no secondary sensors. Your *Data Conversion* setup is:

- Downcast only
- Convert… Pressure, Strain Gauge [db], Temperature [ITS-90], Conductivity [S/m]
- Name your output file Miami.cnv

Conductivity Time Constant

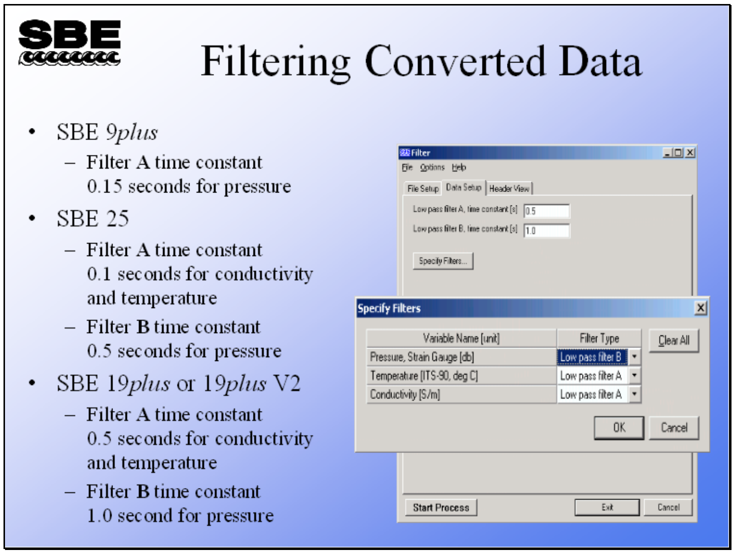
Conductivity Time Constant (Tau)

- Conductivity cell has a time constant (Tau) that depends on pumping rate
- SBE 9*plus* pump and TC duct constrain Tau of conductivity to match Tau of temperature
- SBE 25 pump is slower (2000 vs 3000 rpm) than pump used with 9*plus*; filtering of T and C is recommended
- Taus for SBE 19*plus* and 19*plus* V2 are not as well matched, and require filtering of T and C

It is very desirable to match the time constants of the temperature and conductivity sensors. This improves salinity data in conditions of sharp gradients. The conductivity sensor has a time constant that depends on pumping rate; it can range from 10 milliseconds at a fast pumping rate to very large if no water is moving through the cell.

- For the 9*plus* with a TC duct and standard plumbing, a pumping rate of 25 ml/s brings the conductivity sensor's time constant in line with the temperature sensor's time constant of ~70 ms.
- The SBE 25 uses the same sensors as the 9*plus*, but the SBE 25 system usually includes a slower pump (2000 rpm vs 3000 rpm). A small amount of filtering of temperature and conductivity is usually recommended.
- The 19*plus* and 19*plus* V2 temperature sensor has a much slower time constant than its conductivity sensor, as we saw in the first part of the course. Because of the way the filtering algorithm operates, the best match of temperature and conductivity is obtained by filtering both channels with a filter having the same time constant as the temperature channel.

Conductivity Time Constant (*continued*)



Filtering is done for two reasons:

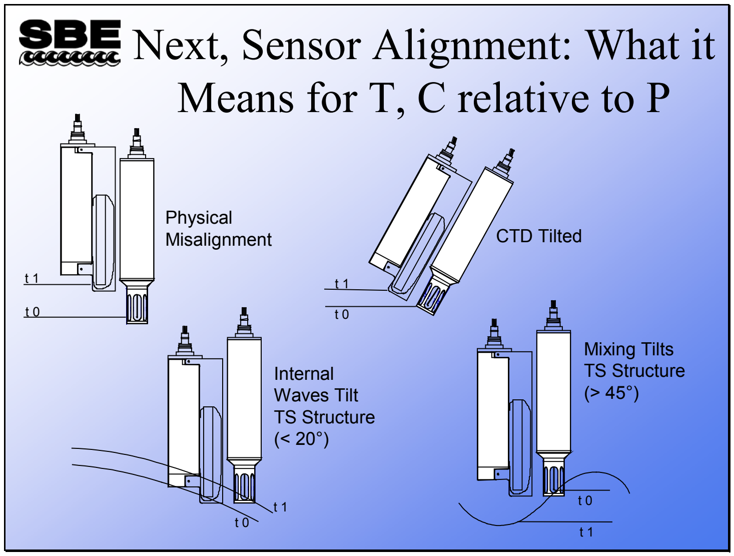
- To match the time constants of the temperature and conductivity sensors.
- To smooth the pressure signal to minimize digitization noise in preparation for removing *loops* in the data with *Loop Edit*. When smoothing pressure, use a filter that is approximately four times the sample rate:

  - SBE 9*plus* – samples at 24 Hz (0.04167 sec), 4 x 0.04167 ≈ 0.15 sec

  - SBE 25 – samples at 8 Hz (0.125 sec), 4 x 0.125 = 0.5 sec

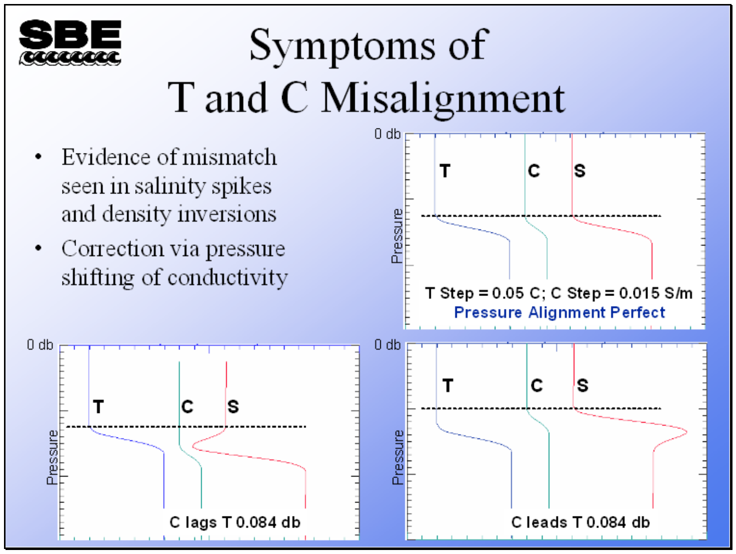
  - SBE 19*plus* or 19*plus* V2 – samples at 4 Hz (0.25 sec), 4 x 0.25 = 1 sec

Illustrating Sensor Misalignment



These illustrations show alignment problems for an instrument **not** using a TC duct (such as an SBE 19, which was typically sold without a TC duct). Here the temperature and conductivity sensors can *see* very different water.

Demonstration of Misalignment Effects



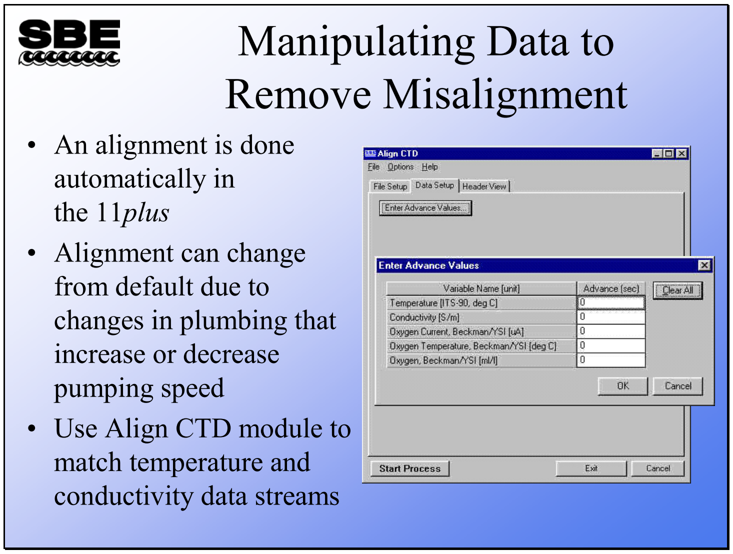
Here is an artificial data set with a step change in temperature and conductivity. Temperature is the blue trace, conductivity is the green, and salinity is the red.

- In the top plot, T and C are perfectly matched, yielding a plot of salinity that is as expected.
- In the bottom left plot, C lags by 0.084 decibars (this is 2 scans at the 9*plus* data rate, at a typical 1 m/s lowering rate). You can see that a negative spike shows up in the salinity data.
- In the bottom right plot, C leads T by 0.084 decibars, yielding a positive spike in salinity.

This behavior can be present in any CTD system built by any manufacturer. It is caused by a mismatch of T and C measurements in relation to pressure. This is not a sensor artifact; any T and C pair using any technology will produce an error in salinity if a scan contains measurements from different water parcels.

A note about the direction of the salinity spike: The direction of the salinity spike is also dependent on whether T and C are increasing or decreasing with increasing pressure. For the examples above, if T and C were decreasing rather increasing, the salinity spike would be in the opposite direction.

Removing Misalignment

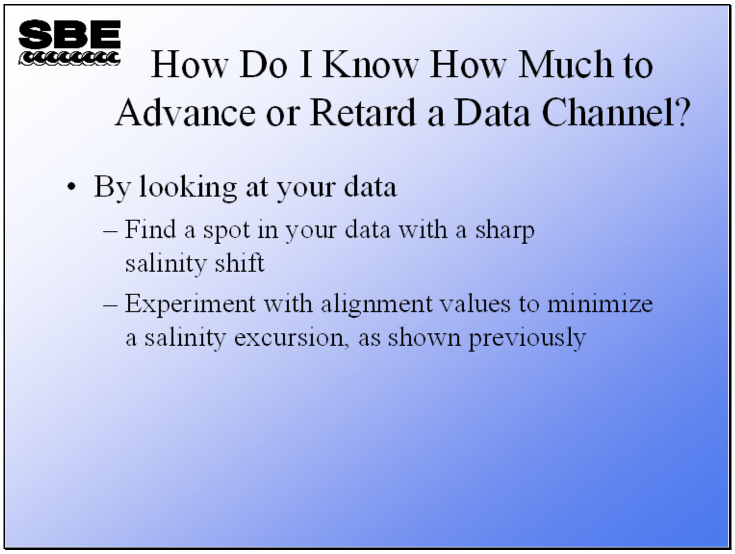


Because the 911*plus* system is well characterized, an alignment of the data stream is done automatically in the 11*plus* before data is transmitted to your computer. With the TC duct in place, an alignment of 1.75 data scans (or 0.073 seconds) is done on incoming data. A linear interpolation between scans is done to implement the alignment of a non-integer number of scans.

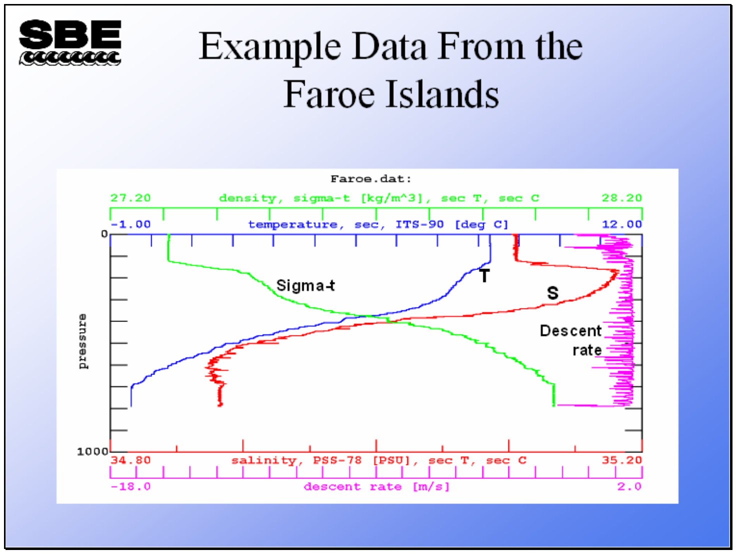
Misalignment that differs from the nominal values can arise from plumbing changes, which can influence pumping speed. A slower pumping speed increases a water parcel's residence time in the TC plumbing, and will require a larger shift in data scans. A faster pumping speed will decrease the residence time in the TC plumbing and require a smaller alignment value.

Note that the advance values are given in seconds and are relative to the pressure channel.

Removing Misalignment (*continued*)



Removing TC Misalignment: Example

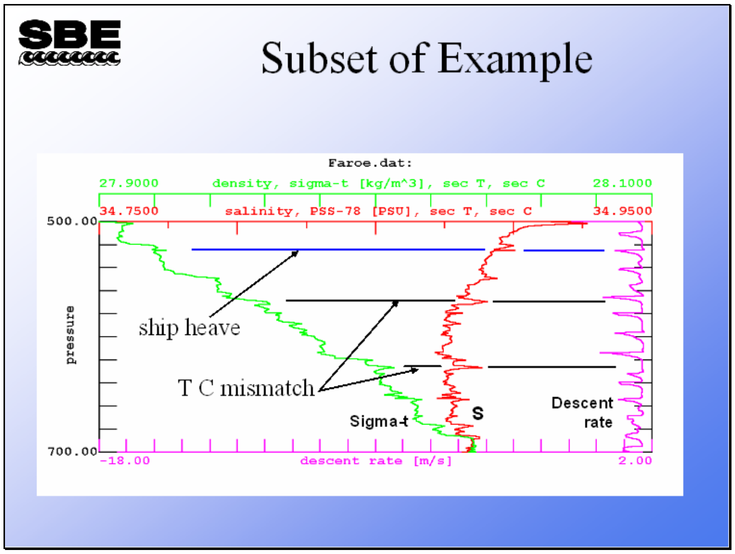


This data set was collected with the secondary T and C sensors of a 911*plus* off the Faroe Islands in 1995. Note the spiky salinity data and the density inversions. These arise from two phenomena: a mismatch between temperature and conductivity samples, and ship heave. The ship heave causes water to move from around the instrument package down to the sensors during deceleration.

We will enlarge part of the plot for a closer look, and do some experimenting with sensor alignment.

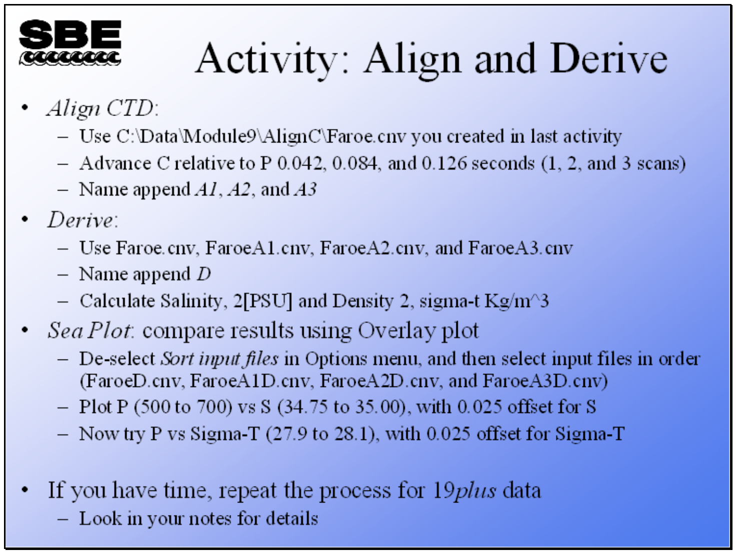
A note about sensor alignment in the 911*plus*: We mentioned earlier that the 11*plus* Deck Unit performs an automatic alignment of T and C. However, in old versions of the 11*plus*, the automatic alignment was done only on the output from the primary T and C sensors; for secondary sensors we must do the alignment in post-processing.

Removing TC Misalignment: Example (*continued*)



It is important to plot descent rate as well as density and salinity, because ship heave can cause errors in your data set that are completely different from alignment errors. To align your data, plot a small subset of the data that has sharp changes in temperature and/or conductivity. Look for spikes in the salinity and density that do not correspond to rapid descent rate decreases, which are indicative of ship heave.

Activity



For this activity, we start with the .cnv file you created in *Data Conversion* that contains temperature and conductivity, and then do some advancing on the file with *Align CTD*, trying a few advances. We then use *Derive* to calculate salinity and density from the original .cnv file and from each aligned file. We should end up with the following files, which we'll plot in Sea Plot to see which advance gives the best results:

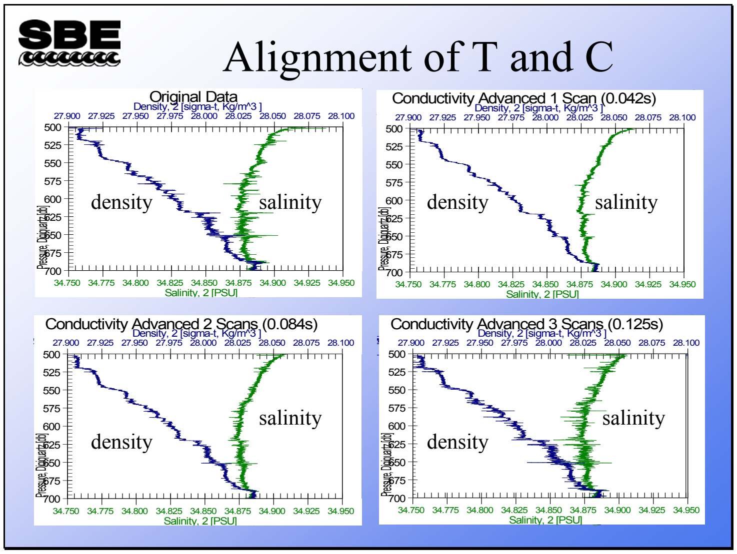
- FaroeD.cnv -- Original data with salinity and density derived
- FaroeA1D.cnv -- C advanced 1 scan (0.042 sec), with salinity and density derived
- FaroeA2D.cnv -- C advanced 2 scans (0.084 sec), with salinity and density derived
- FaroeA3D.cnv -- C advanced 3 scans (0.126 sec), with salinity and density derived

---

If you have time, align the data in C:\Data\Module9\AlignC\SBE19plus\.

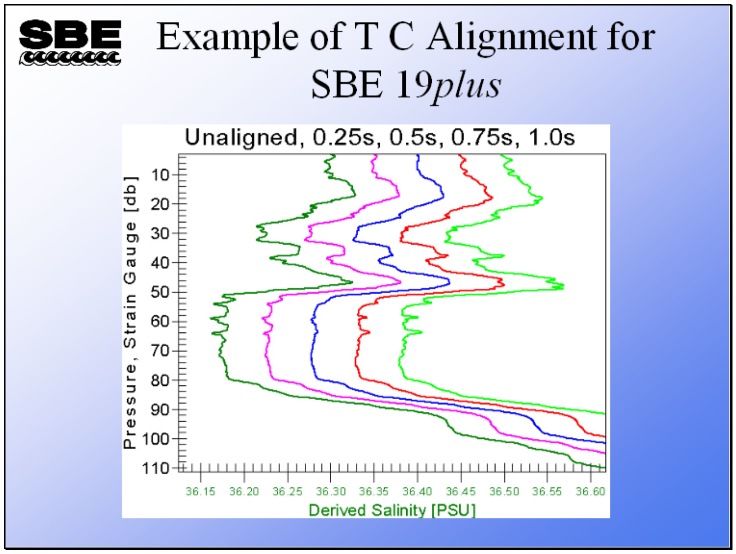
- Filter the temperature and conductivity channels to match the sensor response, using the Filter module. On the *Data Setup* tab, set Low pass filter A to a time constant of 0.5 seconds. Click *Specify Filters…* and select *none* for pressure and *Low pass filter A* for temperature and conductivity.
- Advance temperature against pressure; try whole scan *values* of 0.25, 0.5, 0.75, and 1.0 seconds.
- Use Derive to calculate salinity and density.
- Check your results with *Sea Plot* (try P from 0 to 110, S from 36.15 to 36.6, and offset 0.05).

Removing TC Misalignment: Example (*continued*)



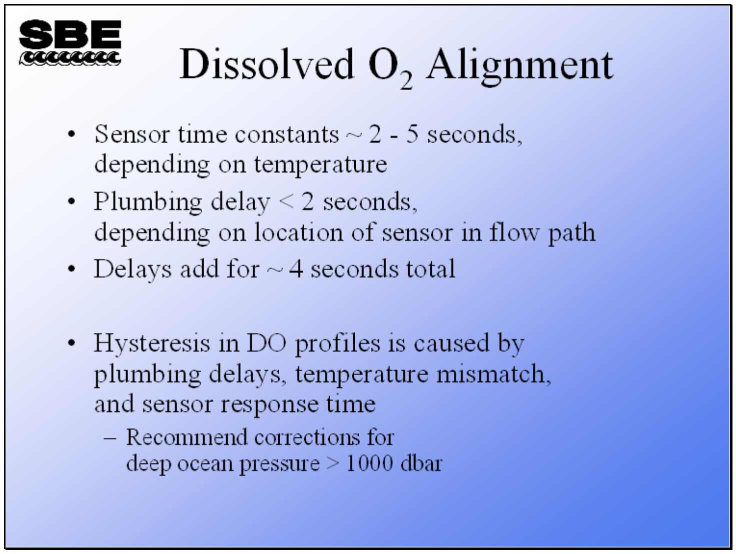
This is from the Faroe data set. The data on the upper left is unaligned, raw data. The upper right has the conductivity channel advanced relative to pressure 0.42 seconds (1 scan); spiking shows considerable improvement. The plot on the lower left has conductivity advanced 0.084 seconds (2 scans) and shows some of the spikes going the other direction. The plot on the lower right has conductivity advanced 0.125 seconds (3 scans), and the spikes have reversed direction and are beginning to get longer. Note that you can align by a non-integer scan interval. In fact, the SBE 11*plus* performs an alignment for you of 1.75 scans (0.073 seconds).

Removing TC Misalignment: Example (*continued*)



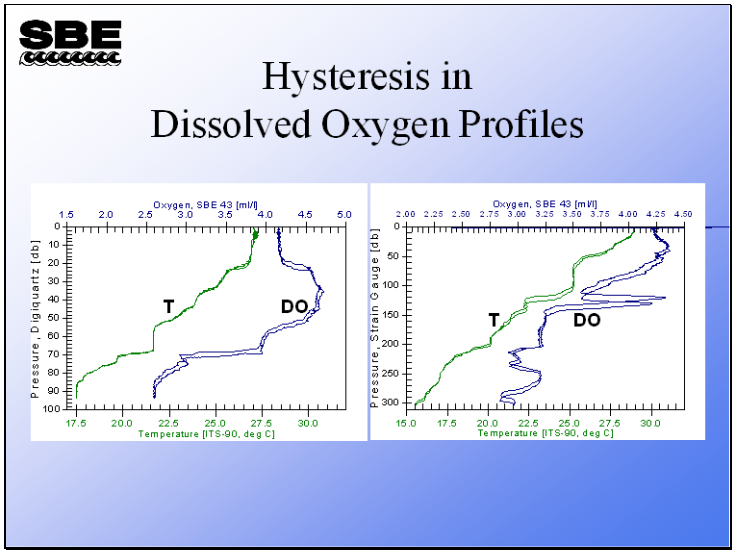
This is from the extra credit data set. The left most trace is the original unaligned data and the others, from left to right, are aligned from 1 to 4 scans. The optimal alignment value may lie between 0.25 and 0.5 seconds; it is left for the reader to experiment and find it.

Removing Misalignment in Dissolved Oxygen



Aligning oxygen current and temperature in relation to pressure can improve hysteresis (mismatch) in dissolved oxygen profiles. The SBE 43 has a faster time constant and shows improvement in hysteresis over the Beckman- or YSI-type of sensor. Deep-ocean hysteresis corrections are advised at depths greater than 1000 dbar.

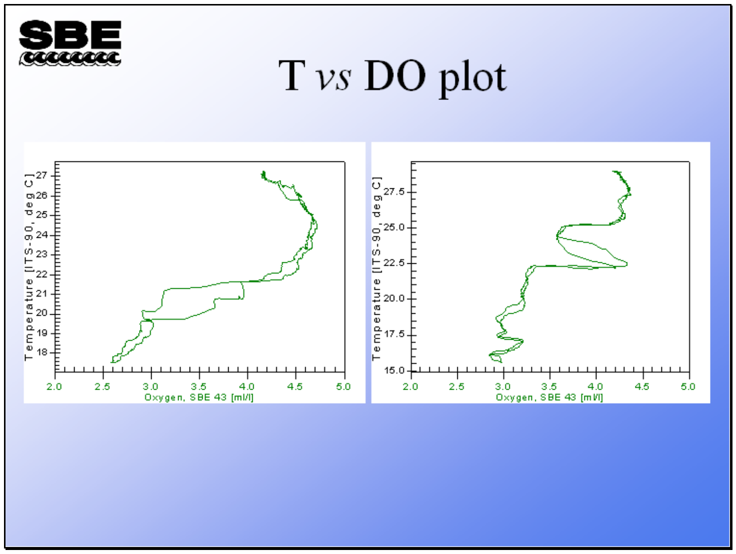
Removing Misalignment in Dissolved Oxygen (*continued*)



While hysteresis is easily observed in pressure *vs*. oxygen profiles, it is also easy to confuse hydrographic phenomenon with hysteresis. The plot on the right is a shallow water plot < 1000 dbar, so is not experiencing deep-ocean hysteresis here. However, it does show hysteresis in both temperature and oxygen; the peak in oxygen has a sharp gradient in temperature associated with it. The cast was taken near the Gulf Stream, and it is likely that the ship drifted during the cast. The CTD downcast probably moved through the hydrographic feature at a different depth than the upcast. A hydrographic phenomenon should not be removed with data manipulation.

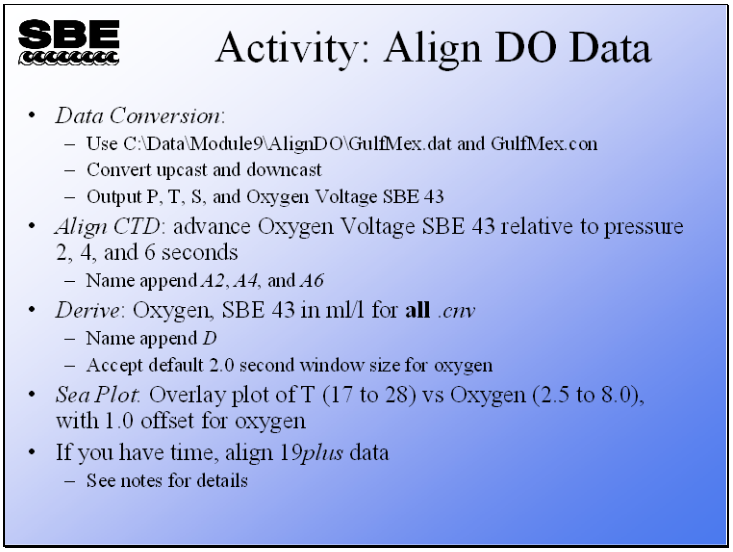
Pressure-induced hysteresis can explain differences between downcast and upcast oxygen data at depths greater than 1000 meters. For these exercises, we will not worry about deep-ocean hysteresis effects, as this was covered in Modules 7 and 8.

Removing Misalignment in Dissolved Oxygen (*continued*)



Viewing hysteresis in this manner is an effective way of eliminating the confusion of hysteresis and hydrographic phenomenon.

Activity



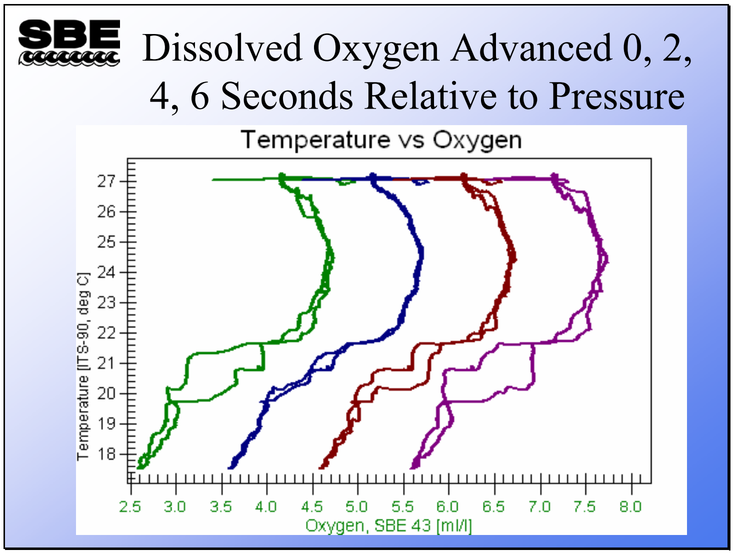
You should end up with the following files to plot in Sea Plot to see which advance gives the best results:

- GulfMexD.cnv original data, not advanced
- GulfMexA2D.cnv dissolved oxygen advanced 2 seconds
- GulfMexA4D.cnv dissolved oxygen advanced 4 seconds
- GulfMexA6D.cnv dissolved oxygen advanced 6 seconds

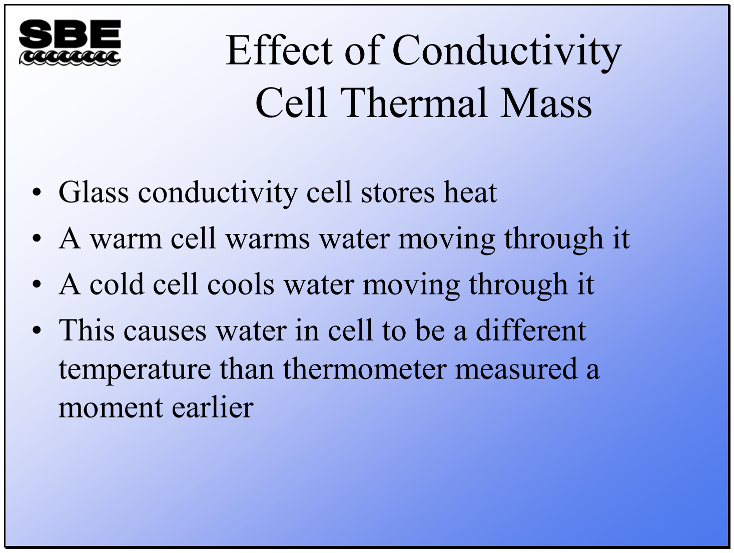
If you have time, align the data in C:\Data\Module9\AlignDO\SBE19plus\ and derive oxygen, using the same procedure.

- Check your results with Sea Plot (try T from 15 to 30, Oxygen from 2.2 to 5.5, and offset 0.3).

Removing Misalignment in Dissolved Oxygen (*continued*)

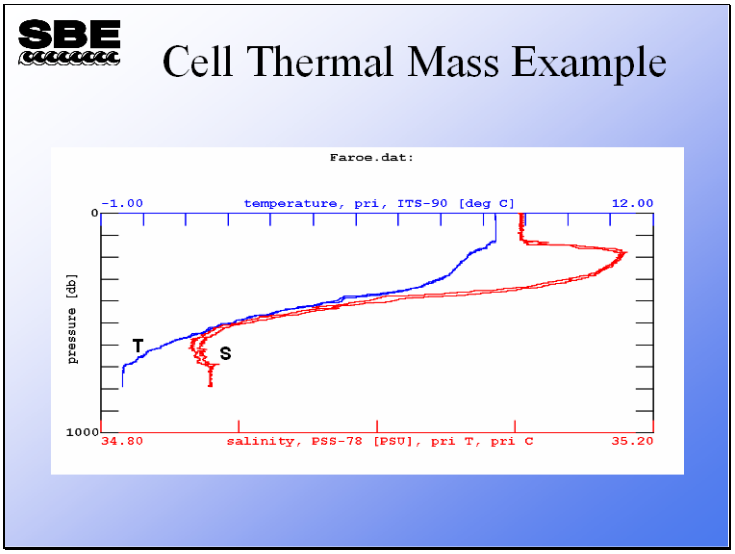


Conductivity Cell Thermal Mass



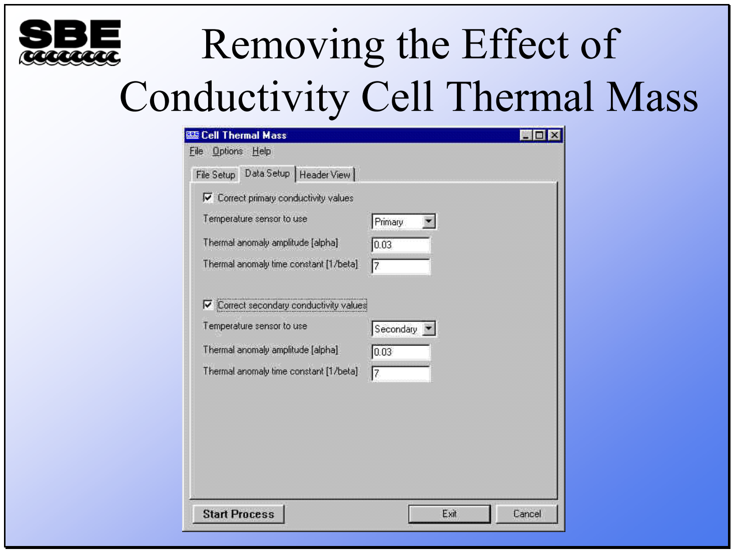
The conductivity measurement has temperature dependence. The conductivity cell itself is constructed of glass and plastic, and as such has a thermal mass. When the cell goes from warm water into cold, the water that passes through the cell is slightly warmed as it transits the cell, resulting in a conductivity measurement that is high of correct. Conversely, when the cell comes up from cold water into warmer water, the water that passes through the cell is cooled slightly, resulting in a conductivity measurement that is low of correct. This heat transfer can be modeled and corrected.

Conductivity Cell Thermal Mass (*continued*)



The data shown in this plot has already been aligned (using Align CTD), reducing salinity spikes associated with misalignment. We see what appears to be salinity hysteresis, but there is no temperature hysteresis. The feature that looks like salinity hysteresis is actually caused by the effect of cell thermal mass. Another way to check for cell thermal mass errors is to create a T vs S plot; the feature to look for in that plot is an open curve.

Compensating for Conductivity Cell Thermal Mass



SBE Data Processing has a *Cell Thermal Mass* module. The thermal mass correction is made with the equation shown below, which is a function of amplitude (alpha) and time constant (1 / beta). Like many of the sensor-related phenomena we have considered, the heat transfer within the cell has a time constant.

Thermal mass correction:

$$Corrected\ Conductivity = C + ctm$$

Where:

C = uncorrected conductivity

$$ctm = -1.0 \times b \times previous\ ctm + a \times \left(\frac{dc}{dT}\right) \times dT$$

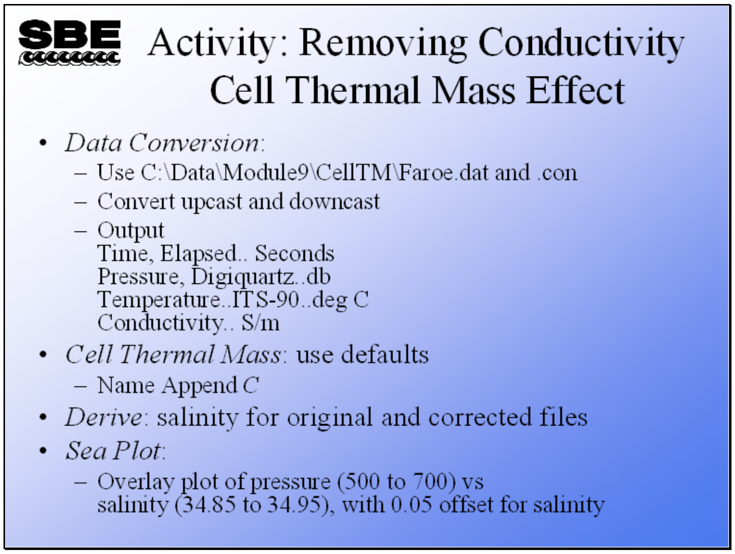
dT = temperature - previous temperature

$$a = 2 \times \frac{alpha}{sample\ interval \times beta + 2}$$

$$b = 1 - \left(2 \times \frac{a}{alpha}\right)$$

$$\frac{dc}{dT} = 0.1 \times (1 + 0.006 \times [temperature - 20])$$

Activity

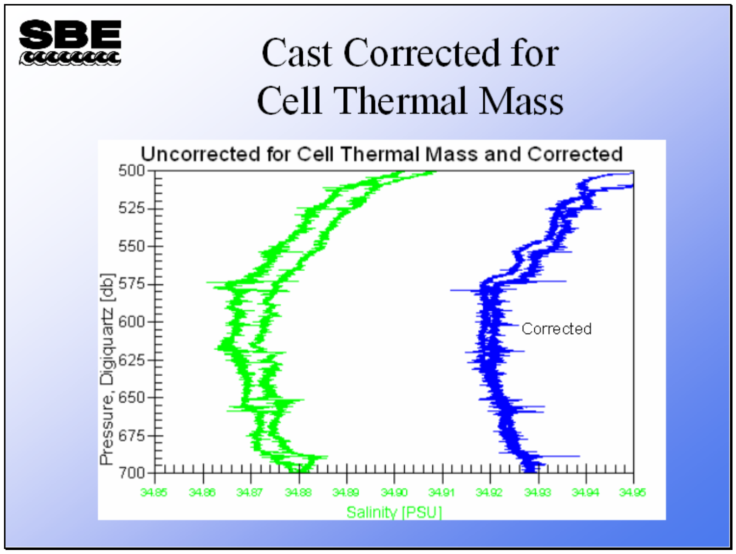


You should end up with the following files, which you'll plot in Sea Plot to see whether the cell thermal mass correction improves the data:

- Faroe.cnv for original data
- FaroeC.cnv for corrected data

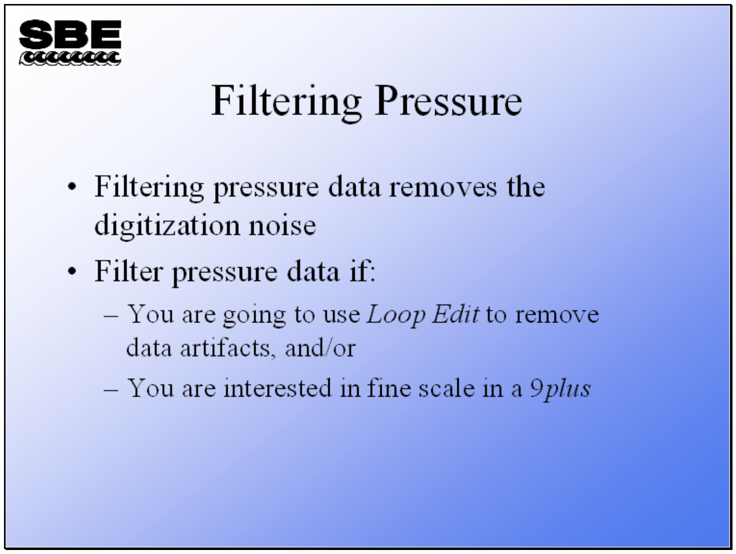
Note: To save time, we aren't running Align CTD as part of this activity, so the resulting plot of the data corrected with Cell Thermal Mass will still show salinity spiking. Typically, you would run Align CTD to eliminate salinity spikes before running Cell Thermal Mass.

Activity (*continued*)



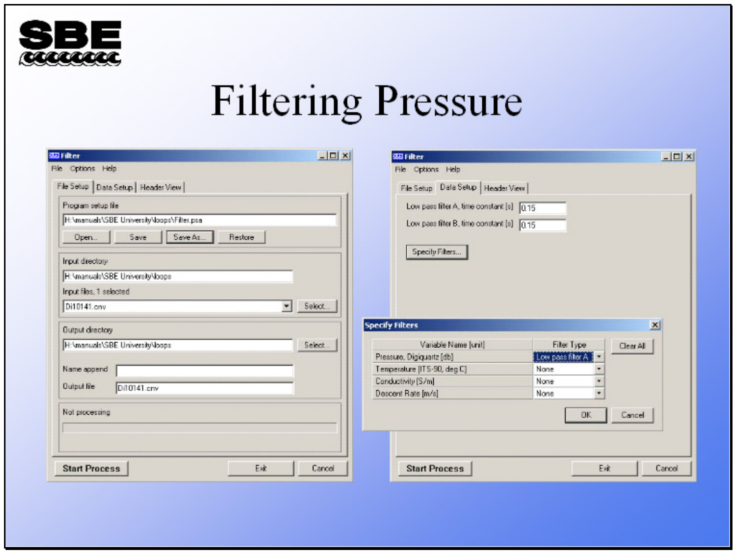
Note that the *corrected* curve still shows what looks like salinity hysteresis, so the correction could perhaps use some refinement. You could repeat the process for other values of alpha and beta, to find the best result.

Filtering Pressure to Remove Digitization Effects



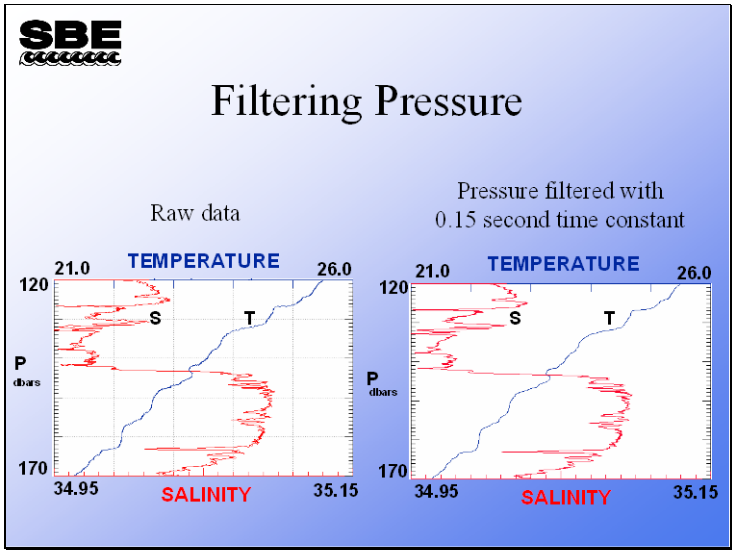
In measuring the pressure sensor signal, there is a digitization error that may be removed by filtering the signal with a low-pass filter. This has the effect of improving the resolution of the pressure signal by smoothing the digitization jitter.

Filtering Pressure to Remove Digitization Effects (*continued*)



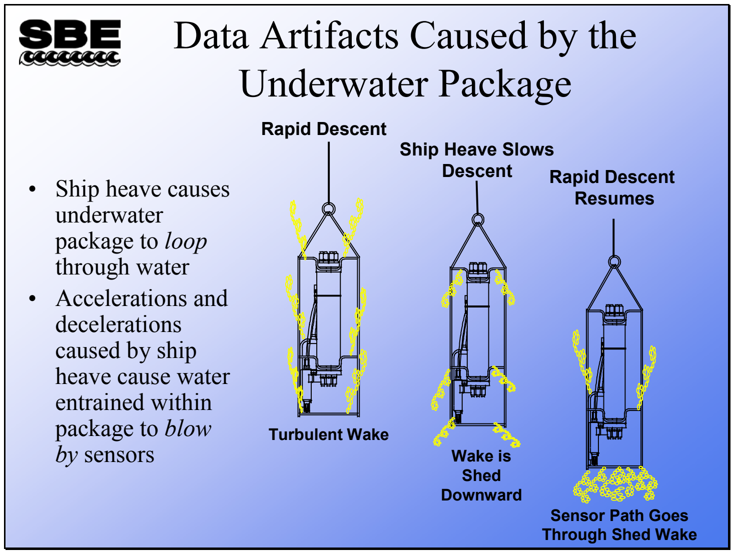
Typically, we filter pressure with a time constant that is 4 times the CTD sample rate.

Filtering Pressure to Remove Digitization Effects (*continued*)



The left plot shows some *dithering* in the data that is eliminated by filtering the pressure.

Data Artifacts Induced by Ship Heave

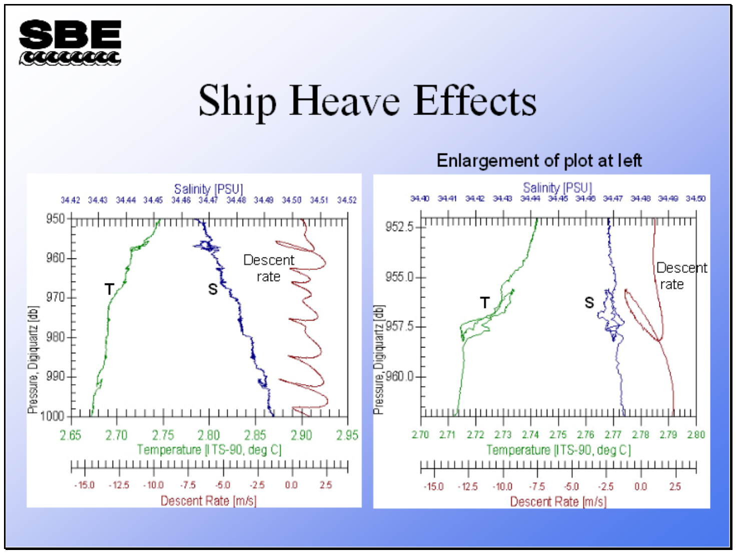


Ship heave is the rocking motion of the ship. Most CTD deployments are made with a small boom or an A-frame that leans out from the ship, giving some distance between the sea cable and the side of the ship. Ship rocking has the effect of pulling up on the sea cable when the ship rocks in one direction and slackening the sea cable when it rocks in the other. This heaving action causes the underwater package to decelerate when the sea cable is pulled up and accelerate when it goes slack. Most instrument packages have sufficient cross section that the deceleration effect is more pronounced than the acceleration.

As the instrument decelerates, water that is entrained within the package can continue downward past the sensors. This water is of different temperature and conductivity than the water at the bottom of the package, and it causes a sampling error.

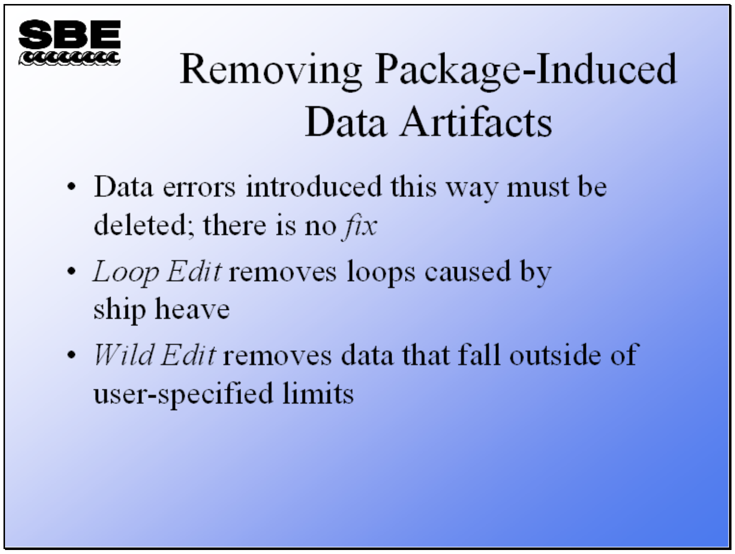
Further, in cases of radical ship heave, the instrument package can have a trajectory through the water column that describes loops. It goes without saying that this sort of behavior causes sampling errors.

Data Artifacts Induced by Ship Heave (*continued*)



These two plots show the effect of ship heave. Both plots show descent rate in brown. The plot on the left shows that each time the descent rate drops, the temperature and salinity traces are disturbed. The plot on the right is an enlargement of a portion of the left plot, showing the loop trajectory that was mentioned previously.

Removing Data Artifacts Induced by Ship Heave



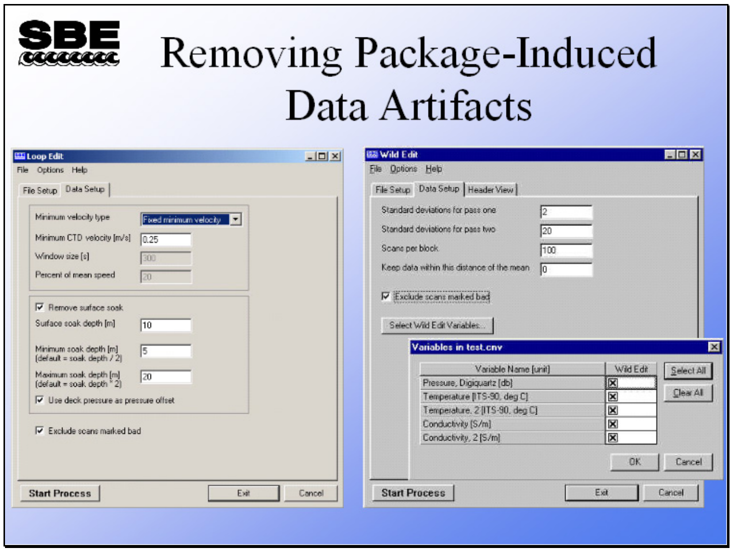
The error caused by ship heave comes from the instrument package disturbing the water that it is trying to sample. Because of this, there is no numerical solution for the problem. SBE Data Processing has two editing modules that remove the offending data. As winch technology improves, we can expect to see vessels equipped with motion compensation capability, which will greatly reduce this problem.

Loop Edit marks data collected when the CTD loops through the water or decelerates sharply.

Wild Edit marks data that falls outside of user-specified limits, given as standard deviations of a window of data; this *bad* data may be caused by a telemetry problem or perhaps a critter or piece of debris going through the conductivity cell.

Data that is marked by these modules can be omitted in subsequent processing steps.

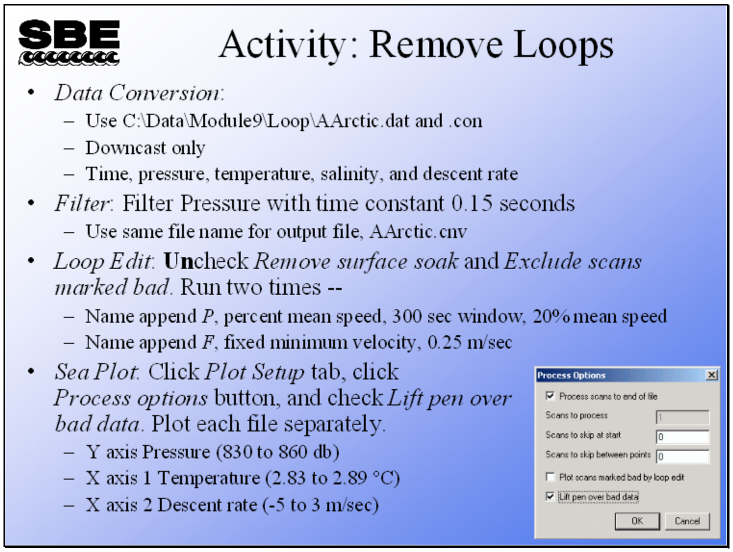
Removing Data Artifacts Induced by Ship Heave (*continued*)



For Loop Edit:

- The default minimum velocity is 0.25 m/sec, which is 25% of the typical nominal descent rate of 1 m/sec. Typically, you should use the *Percent of mean speed* algorithm if the cast had a variable descent rate; otherwise, use *Fixed minimum velocity*.
- If *Remove surface soak* is selected, the scans related to the surface soak are also marked. See the SBE Data Processing manual or Help files for details on setting up the soak depth parameters. The *Use deck pressure as pressure offset* selection relates only to the marking of surface soak data, and has no effect on the pressure data in the file.

Activity



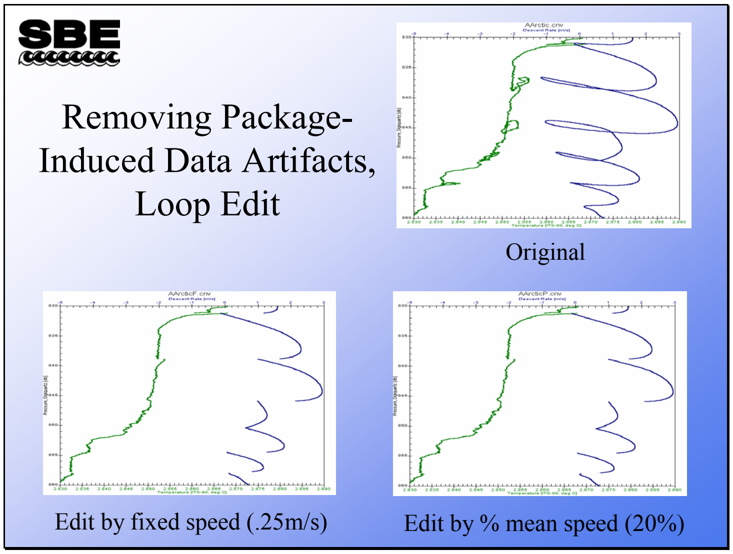
After processing AArctic.cnv in Loop Edit, you should have the following files to look at in Sea Plot:

- AArctic.cnv original file, not corrected for loops
- AArcticP.cnv corrected using percent of mean speed
- AArcticF.cnv corrected using fixed velocity

Create a separate plot for each file.

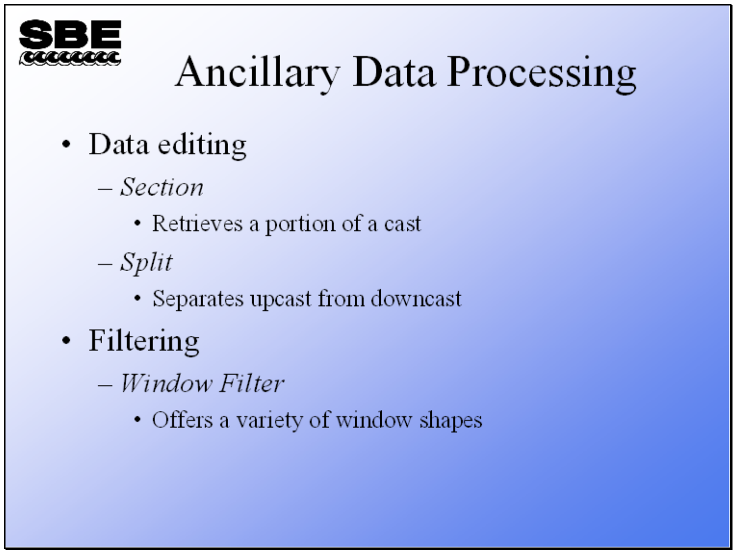
An alternate way to do this exercise is to use an overlay plot to plot all three files on one plot, but you may decide that the plot is too *busy* to be useful.

Removing Data Artifacts Induced by Ship Heave (*continued*)



Here is the example of loopy data that we showed earlier. The bottom two plots have been edited by the two means available. Both plots show very similar results. The bottom left plot is made by editing out data that drop below a fixed speed, in this case 0.25m/s. The bottom right plot is made by editing data that drops below 20% of the mean speed calculated over a 5-minute (300-second) window; this method gives you a bit more flexibility.

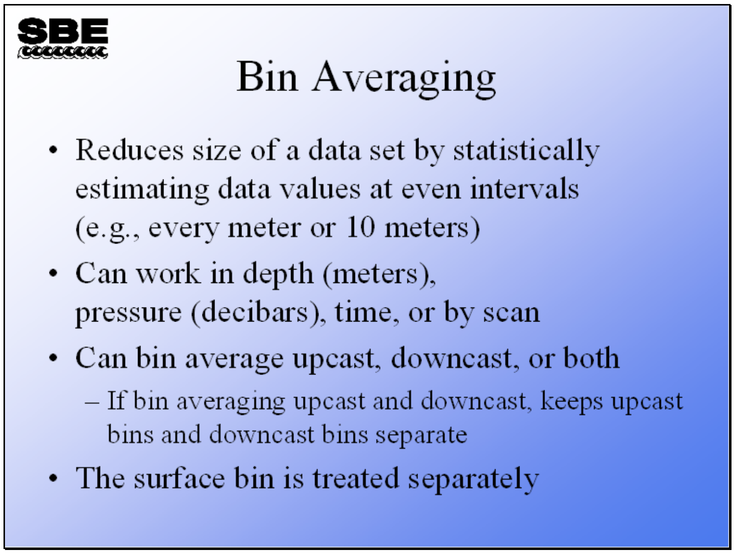
Ancillary Data Processing



In addition to the data processing modules and procedures we have talked about, there are other modules available.

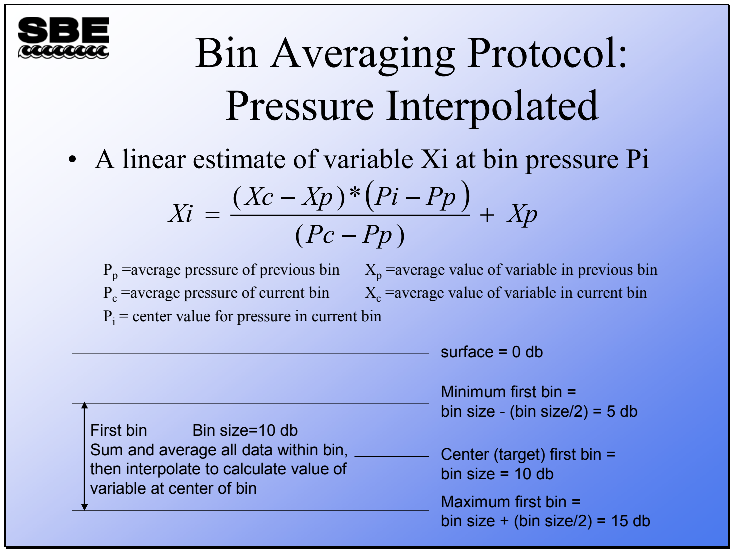
- Clip out part of your data with *Section*.
- Separate your data into upcasts and downcasts with *Split*.
- Use various shaped windows (boxcar, cosine, Gaussian, median, or triangle) to filter your data with *Window Filter*. The median filter is particularly good at preserving sharp steps in the data while rejecting noise.

Bin Averaging



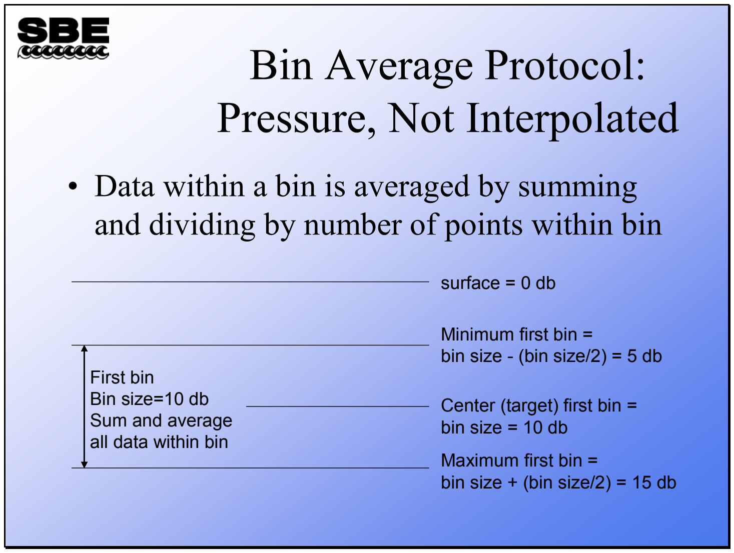
Bin averaging is a means of reducing your data set to a more tractable, and perhaps a more meaningful, size. The Bin Average module makes a statistical estimate of data values at a user-prescribed interval based on the surrounding data. You can bin data on the even meter or 10 meters. You can bin data with a bin size that represents the resolution of your instrument. For time series measurements, you can bin on time interval.

Bin Averaging: Processing Protocol



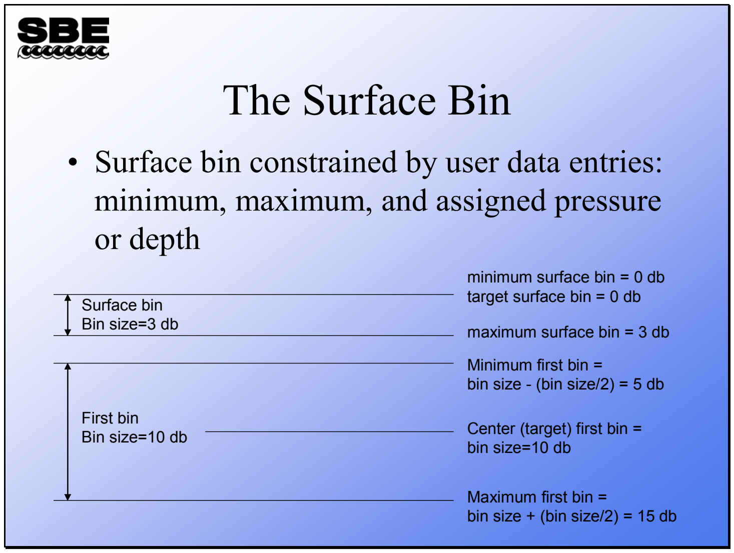
An estimate of each variable is made using the average value of that variable and pressure in the previous bin, and the average values of the variable and pressure in the current bin. Bin averaging with interpolation provides output data at regular intervals (for example, 10 meters, 20 meters, etc.).

Bin Averaging*:* Processing Protocol



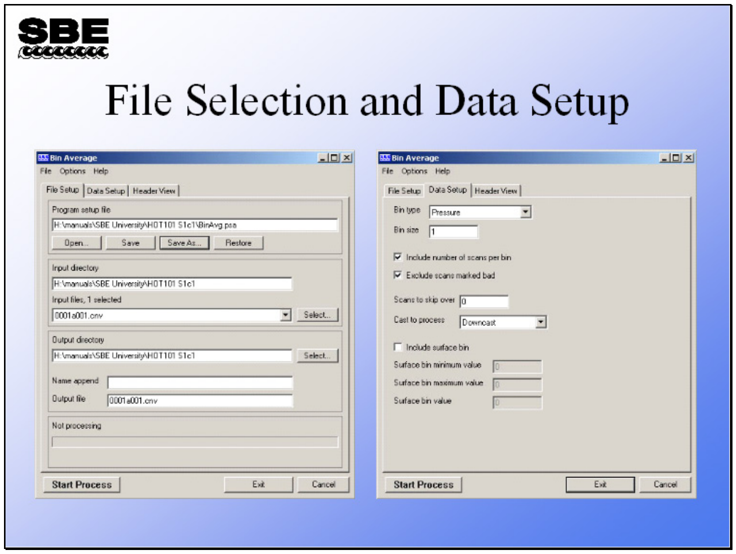
This protocol averages all the data within the bin, producing uneven bin pressures or depths. For example, if you are binning on 10-meter intervals, the first bin start is 5 meters and the end is 15 meters. All data within this window is averaged, producing a bin depth of approximately 10 meters (e.g., 10.123 meters).

Bin Averaging*:* Surface Bin



The surface bin is handled differently because the previous bin would be up in the air. The surface bin is assigned a beginning pressure or depth, an ending pressure or depth, and a target pressure or depth.

Bin Averaging: File Selection and Data Setup



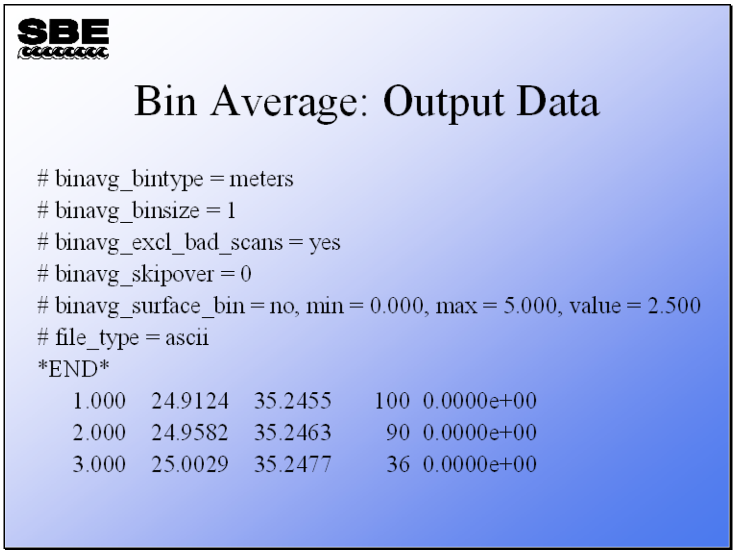
The Data Setup tab allows your choice of pressure, depth, time, or scan bins. You can include the number of scans per bin in the output file, which is useful for evaluating data from instruments with a low sample rate. Bins with 1 sample in them are not very accurate statistically.

You can skip data that you acquired while checking out your instrument before the cast started. Similar to previous processing, you can process the upcast, downcast, or both.

Earlier in this module, we discussed techniques for removing suspect data. These data are marked in the data set as *bad scans*. When setting up *Bin Averaging*, you may exclude scans marked bad by previous processing steps.

As mentioned earlier, the surface bin is handled separately. Note that in our example the surface bin is not included, because we are binning on a 1-meter interval. If you bin on a small interval, it is very difficult to calculate a surface bin. For example, with 1-meter bins, a surface bin would run from 0 to 0.5 meters with value 0.25 meters; depending on the profiling and sampling speeds, there would be few samples within that depth. The surface bin is useful for a coarser bin size. For example, with 10-meter bins, the first bin starts at 5 meters and runs to 15 meters. You can succeed in calculating a surface bin that runs from 0 to 5 meters with value 2.5 meters.

Bin Averaging: Output Data

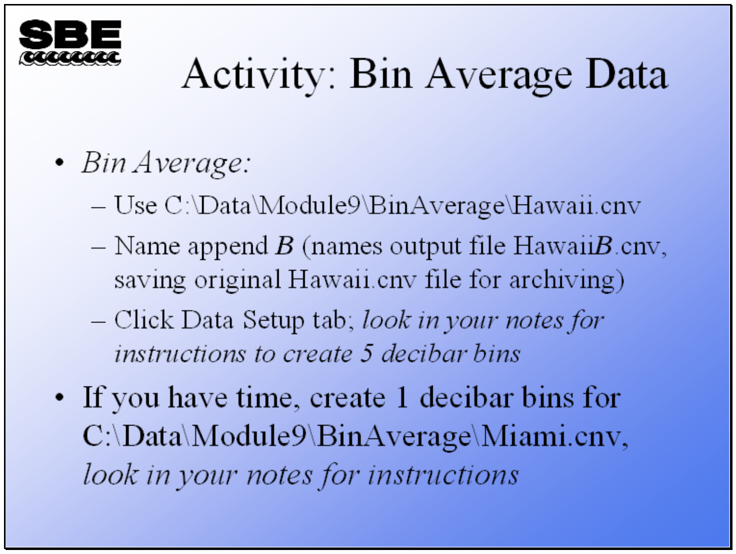


Bin Average processes all variables in the input .cnv file, and inserts a column before the error flag column if you selected *include number of scans per bin*.

The output columns for the example bin averaged data above are:

- Bin depth
- Temperature (°C)
- Salinity (PSU)
- Number of scans per bin (only if *Include number of scans per bin* was selected)
- Error flag

Activity



After you have made the selections on the File Setup tab, click the Data Setup tab:

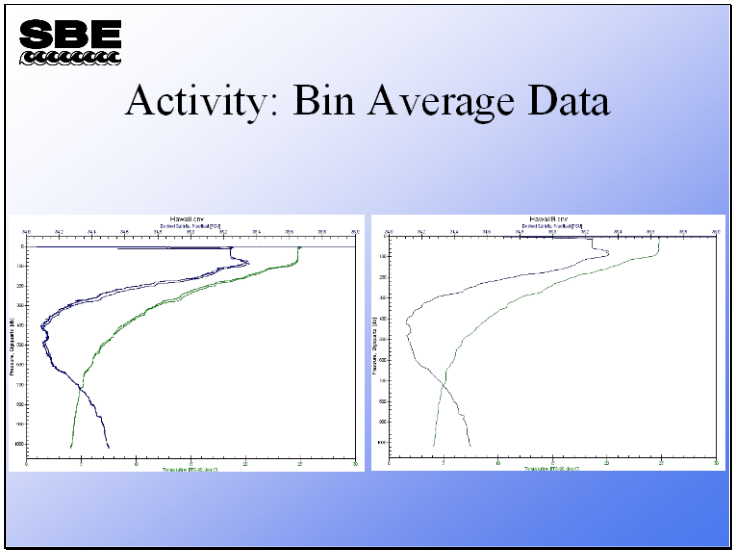
Choose Pressure for *Bin Type*
Enter *Bin size* of 5
Check *Include number of scans per bin*
Check *Exclude scans marked bad*
Skip over 0 scans
Process the downcast
Include the Surface bin
Surface bin minimum value 0
Surface bin maximum value 5
Surface bin value 2.5

Click Start Process.

Open HawaiiB.cnv in Notepad or Wordpad and take a look at the header and data. If you have time, open 2 copies of SBE Data Processing; plot Hawaii.cnv in one, and HawaiiB.cnv in the other.

If you have time, repeat this for C:\Data\Module9\BinAverage\Miami.cnv, using 1 decibar bins and including the surface bin (surface bin minimum value 0, surface maximum value 1, and surface bin value 0.5).

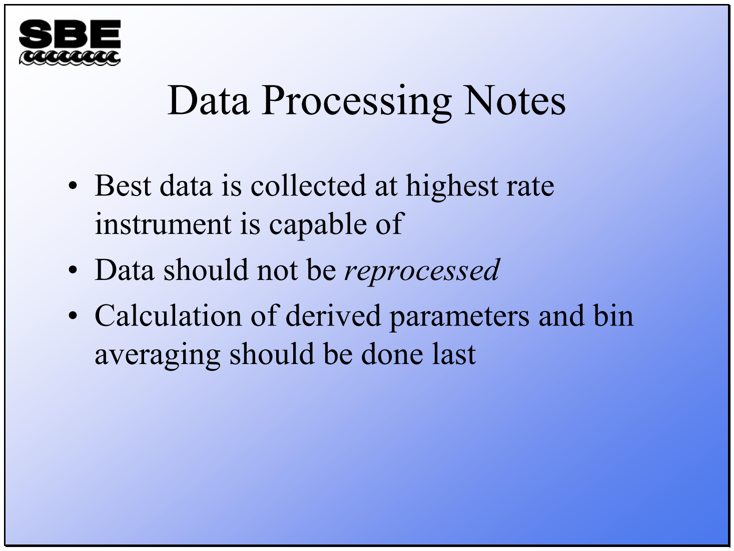
Activity *(continued)*



The plot on the left is one we looked at in Module 3, for data that had been converted using Data Conversion, but not yet *processed* using the advanced processing techniques we reviewed today. That plot shows both the upcast and downcast.

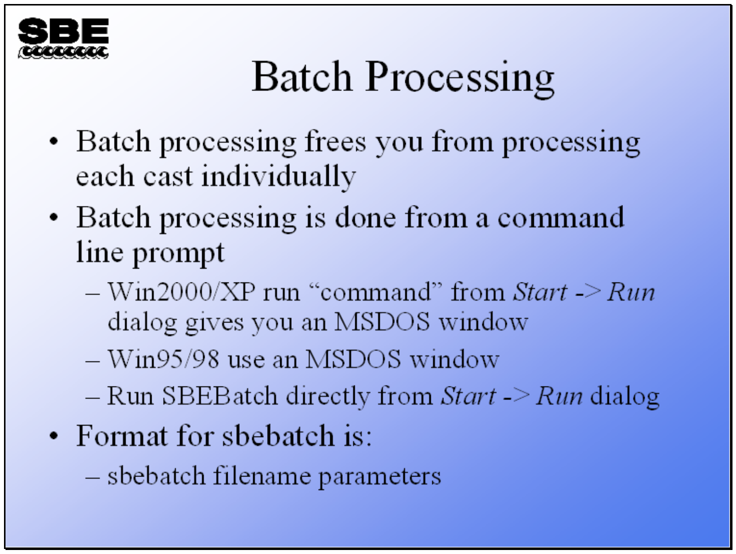
The plot on the right is for the same data, but bin averaged. Notice that many of the features are smoothed out when looking at Bin Averaged data. If you will be using any of the advanced processing techniques we discussed today, **do not Bin Average data before you do the advanced processing.**

Data Processing Tips



A final note: Collect your data at the highest speed you can. Do not reprocess data; if you advance data channels and bin average them or derive other parameters from them, do not advance them again. Derivation of salinity, density, etc. and bin averaging should be the last step after you process and edit your data. The decision to *Derive* and then *Bin Average* or to *Bin Average* and then *Derive* is yours. If you *Bin Average* first you will be *Deriving* from statistical estimates made from your data. If you *Derive* and then *Bin Average*, you will be creating statistical estimates of your derived quantities.

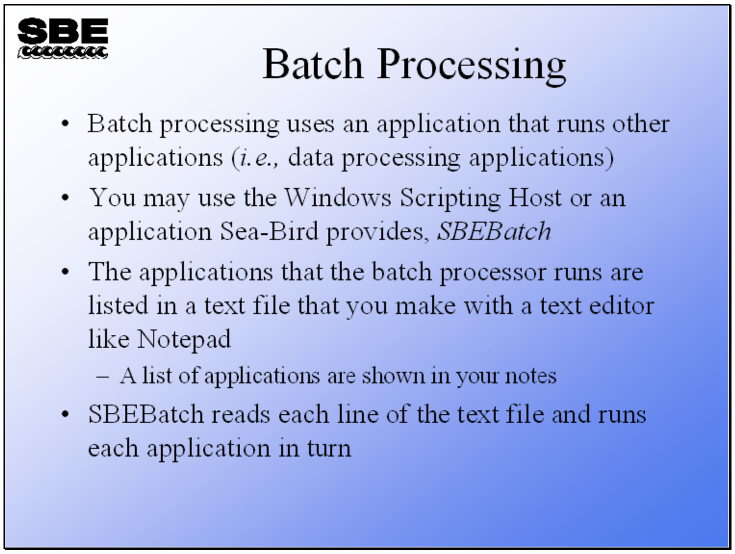
Data Processing of Large Numbers of Files



For processing large sets of cast data, batch mode processing automates the job. You can use the windows scripting host or a program provided with SBE Data Processing, SBEBatch.exe. Your batch file can take advantage of command line parameters and wild card characters.

You can run SBEBatch from a DOS window or from the Windows Run dialog (Start -> Run). In all the examples we'll use today (and the examples in the SBE Data Processing manual), we're assuming you are running from the Windows Run dialog box.

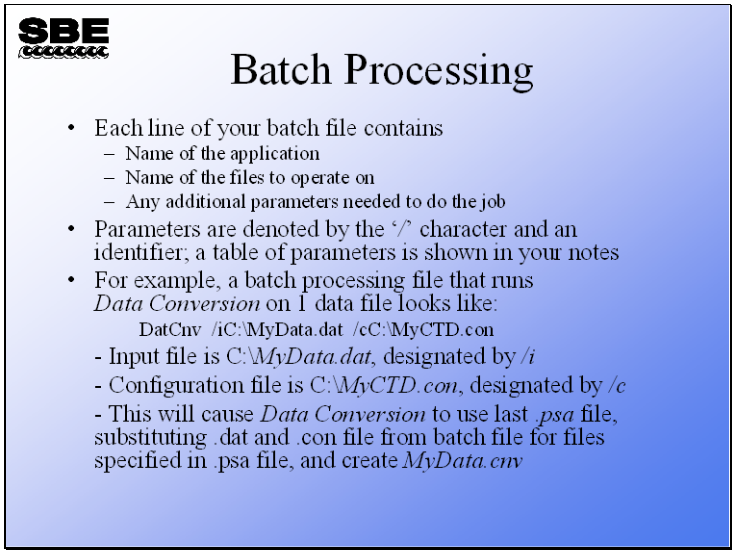
Data Processing of Large Numbers of Files (*continued*)



Applications:

Module	**Process Name**
Align CTD	Alignctd
ASCII In	Asciin
ASCII Out	Asciiout
Bin Average	Binavg
Bottle Summary	Bottlesum
Buoyancy	Buoyancy
Cell Thermal Mass	Celltm
Data Conversion	Datcnv
Derive	Derive
Filter	Filter
Loop Edit	Loopedit
Mark Scan	Markscan
Sea Plot	Seaplot
Section	Section
Split	Split
Strip	Strip
Translate	Trans
Wild Edit	Wildedit
Window Filter	Wfilter

Data Processing of Large Numbers of Files (*continued*)

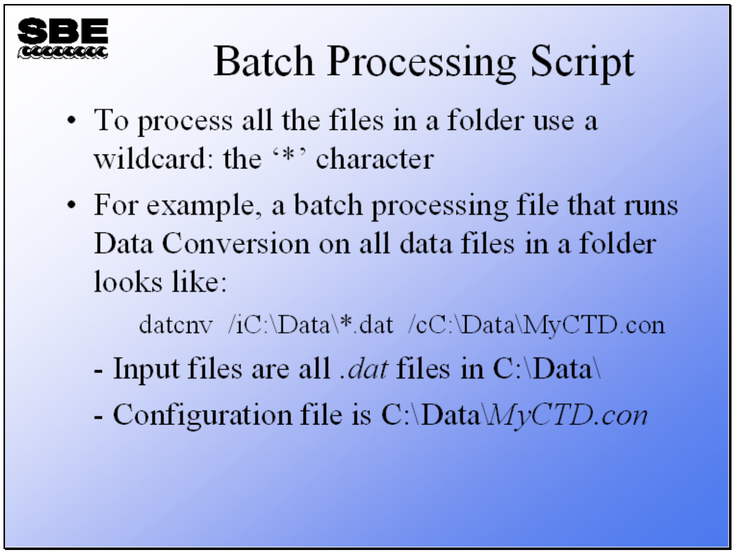


A list of the most commonly used parameters follows; see the SBE Data Processing manual for a complete list:

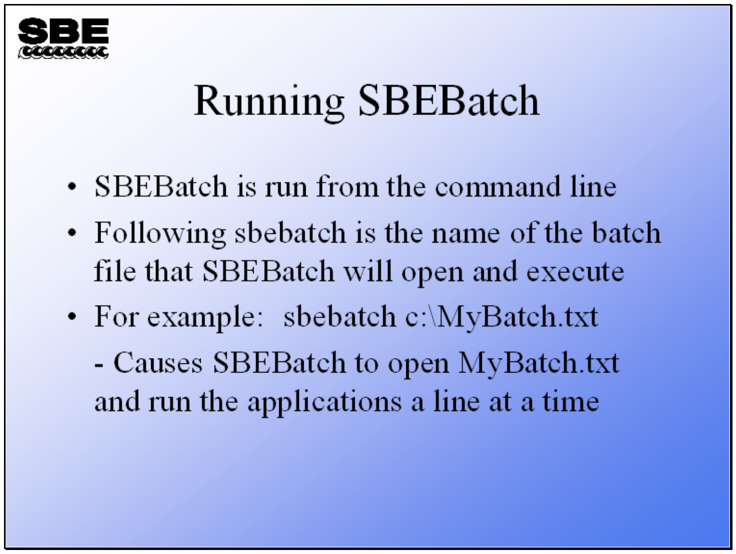
Parameter	Description
/cString	Use *String* as instrument **configuration (.con or .xmlcon) file**. *String* must include full path and file name. Note: If using /cString, must also specify input file name (using /iString).
/iString	Use *String* as **input file** name. *String* must include full path and file name. This parameter supports standard wildcard expansion: • ? matches any single character in specified position within file name or extension • * matches any set of characters starting at specified position within file name or extension and continuing until end of file name or extension or another specified character
/oString	Use *String* as **output directory** (not including file name).
/fString	Use *String* as **output file name** (not including directory).
/aString	**Append *String* to output file name** (before extension).
/pString	Use *String* as **Program Setup (.psa)** file. String must include full path and file name.
/xModule: String	Use *String* to define an additional parameter to pass to Module. Not all modules have x parameters; see module descriptions for details. If specifying multiple x parameters, enclose in double quotes and separate with a space. *Example*: Run Data Conversion, telling it to skip first 1000 scans: /xdatcnv:skip1000
#m	Minimize SBE Data Processing window while processing data, allowing you to do other work on computer.

If specifying multiple parameters, insert a space between each parameter in the list.

Data Processing of Large Numbers of Files (*continued*)



Data Processing of Large Numbers of Files (*continued*)



Here's an example of how to use batch processing to run Data Conversion and Derive to process all the files in C:\Data:

1. Run Data Conversion, entering the desired choices in the File Setup and Data Setup dialog boxes. Upon completing setup, press Save or Save As on the File Setup tab. The configuration is stored in the .psa file.
   Repeat for Derive.

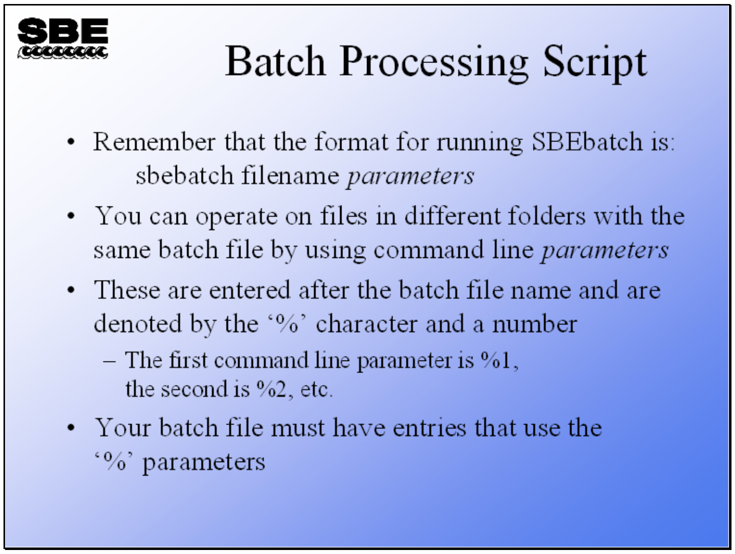
2. Create a batch file named batch.txt in C:\Data, which contains the following lines:
   @Lines starting with @ are comment lines, and have no effect on the result
   @Use these to document what you are doing in the batch file
   @Processing data from February 2006 Cruise
   datcnv /iC:\Data*.dat /cC:\Data\MyCTD.con
   derive /iC:\Data*.cnv /cC:\Data\MyCTD.con

3. Select Run in the Windows Start menu. The Run Dialog box appears.
   Type in the batch processing program name and the .txt file name:
   sbebatch c:\Data\batch.txt

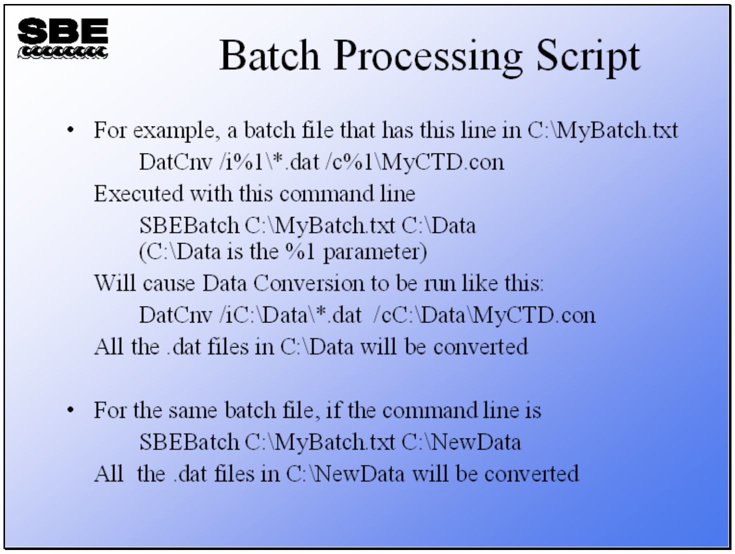
The results:

1. Data Conversion uses its last .psa file, substituting the .con file from the batch file for the .con file specified in the .psa file, and processes **all** .dat files in C:\Data, creating a .cnv file from each .dat file.
2. Derive uses its last .psa file, substituting the .con file from the batch file for the .con file specified in the .psa file, and processes **all** .cnv files in C:\Data (which were just created by Data Conversion), creating a .cnv file from each .cnv file.

Data Processing of Large Numbers of Files (*continued*)



Data Processing of Large Numbers of Files (*continued*)



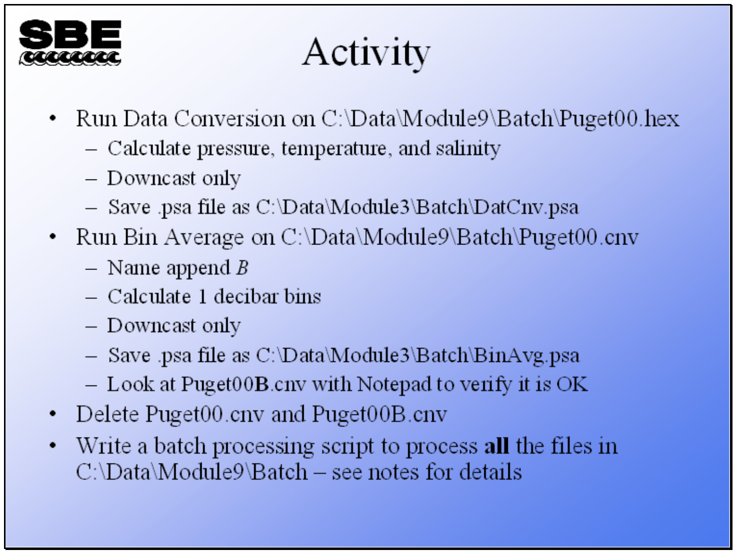
Now let's add a bit more flexibility to the process. Here's an example of how to use batch processing to run Data Conversion and Derive to process all the data files in C:\Data\Leg1, C:\Data\Leg2, and C:\Data\Leg3:

1. Run Data Conversion, entering the desired choices in the File Setup and Data Setup dialog boxes. Select *Match instrument configuration to input file* on the File Setup tab. Upon completing setup, press Save or Save As on the File Setup tab. The configuration is stored in the .psa file.
   Repeat for Derive.
2. Create a batch file named batch.txt in C:\Data, which contains the following lines:
   ```
   @Processing data from 3 legs of February 2006 Cruise
   datcnv /i%1\*.dat
   derive /i%1\*.cnv
   ```
3. Select Run in the Windows Start menu. The Run Dialog box appears.
   Type in the batch processing program name, the .txt file name, and the %1 parameter:
   ```
   sbebatch C:\Data\batch.txt C:\Data\Leg1
   ```
   Repeat for the files in Leg2 and Leg3:
   ```
   sbebatch C:\Data\batch.txt C:\Data\Leg2
   sbebatch C:\Data\batch.txt C:\Data\Leg3
   ```

The results:

1. Data Conversion uses its last .psa file, substituting the matching .con file for the .con file specified in the .psa file, and processes **all** .dat files in C:\Data\Leg1, creating a .cnv file from each .dat file.
2. Derive uses its last .psa file, substituting the matching .con file for the .con file specified in the .psa file, and processes **all** .cnv files in C:\Data\Leg1 (which were just created by Data Conversion), creating a .cnv file from each .cnv file.
3. Steps 1 and 2 are repeated for the files in C:\Data\Leg2 and C:\Data\Leg3.

Activity



1. The steps detailed on the slide are used to prepare Data Conversion and Bin Average to operate with your batch file, by setting up the desired output parameters, bin size, etc. The saved .psa files contain all the information on the setup.
   - We delete the output files – Puget00.cnv and Puget00B.cnv – because they will be recreated by the batch processing script.

2. Create a batch processing script to process **all** files in C:\Data\Module9\Batch (Puget00.hex, Puget01.hex, etc.):
   - Use Notepad to write your batch file, using a %1 parameter for the file locations.
   - Save your batch file as C:\Data\Module9\Batch\MyBatch.txt

3. Launch a command line session: Click Start > Run.

4. Run your batch file with the command line:
   Sbebatch C:\Data\Module9\Batch\MyBatch.txt C:\Data\Module9\Batch

5. Check to see that you were successful with Notepad