HemoLab Manual

Harald M. Stauss, MD, PhD

August 15, 2012

Contents

List of Figures

Chapter 1

Installation

1.1 Download HemoLab Software

The HemoLab software is freely available at the HemoLab homepage. The URL for the HemoLab homepage is:

http://www.haraldstauss.com/HemoLab/HemoLab.html

On this URL, you will find a download link for the setup file of the HemoLab software. The download link appears similar to:

HemoLab Ver. 7.5 (July 21, 2009)

Right click on the download link and select "Save Target As...". A new window will pop up that allows you to select a directory to save the HemoLab setup file. As of this writing, the filename of the setup file is:

HemoLab_7.5.zip

1.2 Unzip the Setup File

The HemoLab setup file comes in a compressed zip archive. Therefore, the zip file needs to be unzipped. Simply right click on the zip archive file (e.g., HemoLab_7.5.zip) and select "Extract All...". This will create a new directory (e.g., HemoLab_7.5). In this new directory, you will find the executable setup file for the HemoLab software (e.g., setup.exe).

1.3 Run the Setup File

It is important that you have system administrator rights when starting the setup program. As administrator, double click the setup.exe file and follow the instructions. After installation, you will find the HemoLab software in the

start menu (All Programs) of MS-Windows. All three components of the HemoLab software (Analyzer, Batch Processor, WinAD, WinStat) can be accessed from the start menu. In addition, the software can be uninstalled by selecting "Uninstall".

1.4 Update to a Newer Version

It is a good idea to frequently check the HemoLab homepage for updates of the HemoLab software. Update to a new version consists of two steps:

1. Uninstall previous version
2. Install new version

It is really important to uninstall the previous version before installing a new version. This ensures that older files are deleted and that the registry entries are updated correctly. To uninstall the previous version login as system administrator. You cannot update the software from a limited user account. As system administrator simply select "Uninstall" from the HemoLab Program Group in the "All Programs" menu from the MS-Windows "start menu". Then, simply install the new version of the HemoLab software as outlined above.

Chapter 2

Supported File Formats

The HemoLab software supports 6 different file formats, including a text or ASCII format. In the remainder of the manual, these file formats are designated as:

- **MAD-format** (Motif Analogue to Digital format). Motif is the name of a window manager for the UNIX X-window system. Originally, the HemoLab software was written for UNIX and was using the Motif window manager. Recently the software was converted to Windows, but the name of the file format remained the same for consistency. The MAD-format is a binary format used by the WinAD data acquisition software. MAD files can be read but not written by the Analyzer software.
- **TSA-format** (Time Series Analysis format). This is another binary data format that can be used to save data files from within the Analyzer software. TSA files can be read and written by the Analyzer software.
- **ASC-format** This is a regular text or ASCII format. The individual channels are aligned in columns, separated by tabulator (TAB) or space (blank) characters. ASCII files can be read and written by the Analyzer software. The ASC files used by the HemoLab software do not use any headers.
- **DSI-format** Data Sciences International generously made the data format of their Dataquest A.R.T. files available for the HemoLab project. So far the Analyzer software can only read DSI waveform (and not parameter) files.
- **LabChart** The LabChart file format (*.adicht) is the proprietary format used by ADInstruments. ADInstruments currently develops a programming SDK. The HemoLab import filter for LabChart files was programmed using a beta version of the SDK. ADInstruments does not want me to distribute the required DLL of the beta version. Once the final version of the SDK is released by ADInstruments, I may be able to release the LabChart

import filter for HemoLab. Please contact your ADInstruments Representative if you need the import filter for LabChart files.

- **WinDAQ, WDQ-format** The WinDAQ format is used by DATAQ Instruments to save files recorded by its proprietary software. DATAQ Instruments (http:\\dataq.com) sells a very nice line of low-cost AD-converters (DI-145, DI-149) that are supported by their free WinDaq/Lite recording software. However, these low-cost AD-converters also work with the WinAD data acquisition software included with HemoLab. WinDaq/Lite only supports a maximum sampling rate of 240 Hz (divided by the number of active channels). By using the WinAD software, you can use the DI-149 AD-converter at a sampling rate of 1,000 Hz for each of the 4 channels. Thus, for a low-cost data acquisition system, the DATAQ DI-149AD-converter together with the WinAD data acquisition software is a very nice combination.

- **EndoPAT-format** The EndoPAT is a commercial device that claims that it can determine endothelial function based on finger plethysmograms. The device generates a file with the extension ***.S32**. These files contain 7 channels: a time channel and 3 channels for each hand. The 3 channels look like they are the raw finger plethysmogram and a derived channel that may be the estimated aortic pressure waveform. The 3rd channel looks like a high-pass filtered aortic pressure channel. I am not sure if this is the correct interpretation of the 3 channels per hand. However, the quality of the signals is good enough to derive heart rate and to perform heart rate variability analysis.

Chapter 3

WinAD

3.1 What is WinAD

WinAD is a data acquisition software for MS-Windows. The main window of the WinAD software is shown in Fig. 3.1. Although, it was designed for hemodynamic data acquisition in our physiology lab, it may also proof useful for other data acquisition purposes.

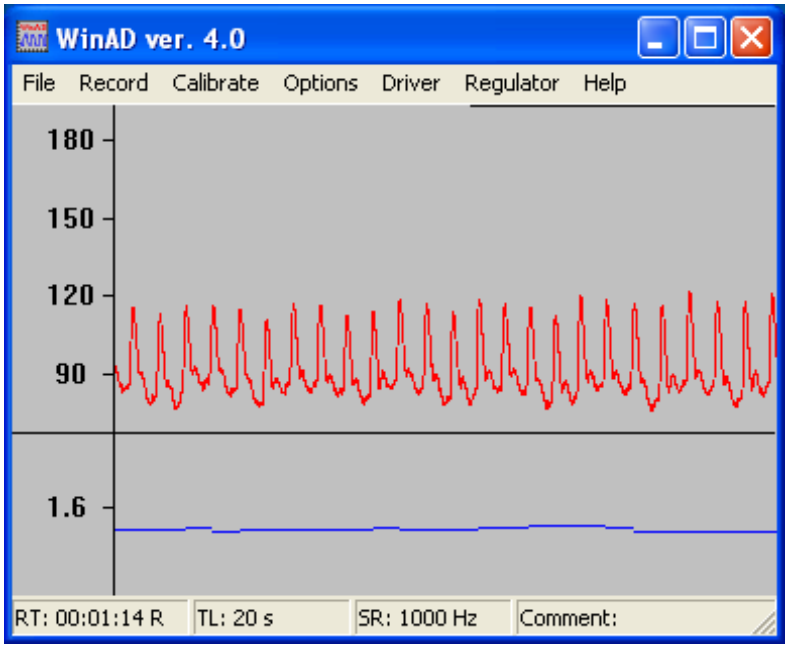

Figure 3.1: WinAD, main window

Currently, the following drivers are included:

- A dummy driver to test WinAD
- MAX186 AD-converter with DirectIO (best solution)
- MAX186 AD-converter using Windows API (slower than DirectIO)
- Pic12F683 driver (AD-converter that connects to the COM port)
- Pic18F2553 driver (AD-converter that connects to the USB port)
- Telemetry driver (an experimental telemetry system driver)
- DLP-IO8 driver (AD-converter connects to the USB port)
- Gameport driver for joysticks or a gameport adapter
- Soundcard driver
- DI-145 (DATAQ) driver
- DI-149 (DATAQ) driver
- DI-155 (DATAQ) driver

3.2 WinAD Drivers - AD converters

3.2.1 Dummy Driver

The dummy driver is included to test the software. It provides up to 8 channels.

3.2.2 MAX186 Driver

This is the most important driver for HemoLab. This driver supports an A/D-converter board that is connected to the serial port of the computer. The A/D-converter board is based on the Maxim MAX186 A/D-converter chip. The specifications for this AD-board are:

- 8 channels
- unipolar or bipolar mode
- input voltage: unipolar: 0.000V ... +4.096V, bipolar: -2.048V ... +2.048V
- sampling rate ca. 3000 Hz (sum of all channels) with DirectIO
- sampling rate ca. 500 Hz (sum of all channels) with Windows API

A schematic for the electrical circuit for the MAX186 AD-board is shown in Fig. 3.2 and is also included as PDF file ("Max186Circuit.pdf") in the HemoLab install directory (typically c:\Program Files\HemoLab). If you are serious about using WinAD, I highly recommend building this A/D adapters.

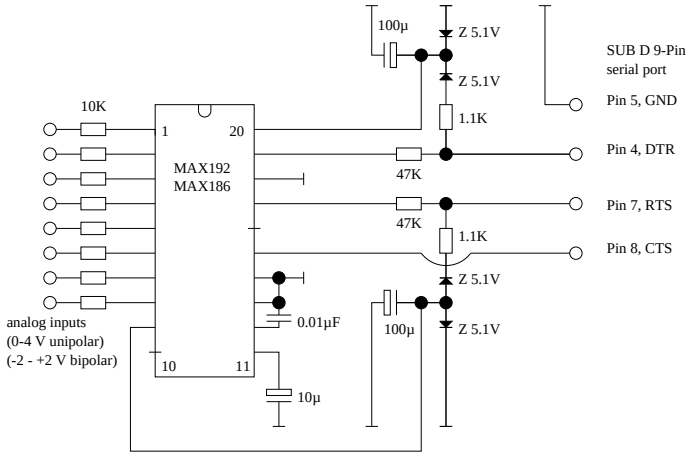

Figure 3.2: Electrical circuit for the MAX186 A/D-adapter
The adapter is based on the Maxim MAX186 A/D-converter chip.

3.2.3 DirectIO

DirectIO is a driver that allows direct access to the serial port of the computer under the MS-Windows XP operating system. DirectIO need to be installed separately from HemoLab. The setup file is included in the HemoLab install directory (typically c:\Program Files\HemoLab). The filename of the DirectIO setup file is "directio.exe". Please note that DirectIO is **not** free software. You need to register the software at the DirectIO homepage (http://www.direct-io.com/). This software is absolutely worth its money. It allows for relatively high sampling rates with the MAX186 A/D-converter under HemoLab.

3.2.4 Pic12F683 Driver

This driver is for an AD converter with 4 analog channels and a maximum sampling rate for 1000 Hz per channel (4000 Hz total sampling rate). The AD converter connects to the serial (COM) port of the computer or - via a serial to USB converter - to the USB port. The AD converter can be purchased through the HemoLab webpage.

3.2.5 Pic18F2553 Driver

This driver is for an AD converter with 8 analog channels and a maximum sampling rate of 1000 Hz per channel (8000 Hz total sampling rate). The AD converter connects to the computer via the USB port. This AD converter is currently only available via special request.

3.2.6 Telemetry Driver

This driver is for an experimental telemetry system that is currently under development.

3.2.7 DLP-IO8 Driver

This AD-converter has 8 analog channels with a sampling rate of ∼250 Hz total (for all channels). It connects to the USB port of the computer. The AD-converter can be obtained from Mouser Electronics (http://www.mouser.com).

3.2.8 Gameport Driver

This driver supports 4 analog and 4 digital (on/off) channels with a maximum sampling rate of 500 Hz for each channel. Basically, the resistances of the potentiometers and the states of the "fire-buttons" of the joysticks are recorded. To try this driver, you can simply connect one or two joysticks to the gameport of your computer. For data acquisition, when accuracy is not a major concern, a voltage to current converter can be used. I designed a gameport adapter that is basically a 4-channel voltage to current converter (emulating 2 joysticks with x and y axes each). This adapter works reasonably well with the HemoLab software. A schematic for the gameport adapter is shown in Fig. 3.3 and is also included as PDF file ("GamePortCircuit.pdf") with the HemoLab software. The file can be found in the HemoLab install directory (typically c:\Program Files\HemoLab). Please note that there are issues with linearity when using this simple adapter and, therefore, accuracy is not great.

3.2.9 Soundcard Driver

The soundcard driver allows recording from the soundcard. Two analog channels with sampling rates in the range from 8000 Hz to 44,100 Hz are available. Of course, only AC signals can be recorded with this driver.

3.2.10 DI-145 (DATAQ) driver

DATAQ Instruments (http:\\www.dataq.com) sells a series of very nice low-cost AD-converters that they market as "Starter Kits". The DI-145 has 4 differential input channels, 10 bit resolution, and an input voltage range of -10V to +10V. The drawback of this AD-converter is that it only allows for a total sampling rate of 240 Hz (all channels together). Thus, if all 4 channels are recorded, the sampling rate per channel is only 60 Hz. However, this AD-converter only costs $29.00 (as of this writing). Thus, for applications where the sampling rate is not critical, this low-cost AD-converter may be an excellent choice.

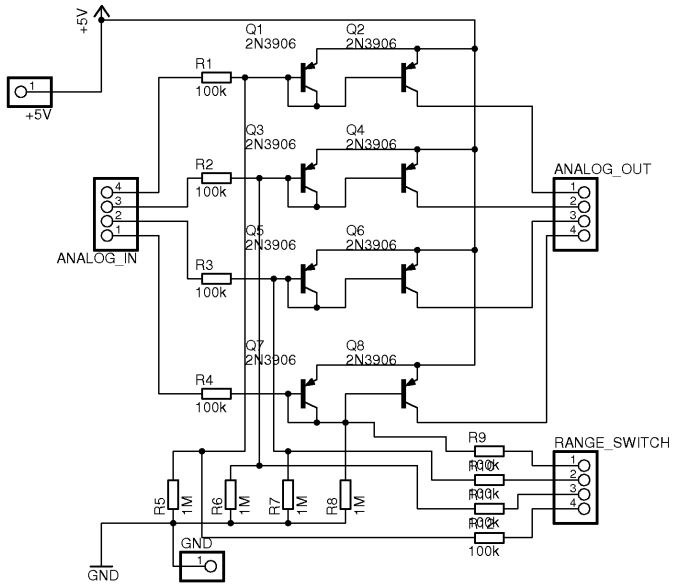

Figure 3.3: Electrical circuit for the gameport A/D-adapter
The adapter is based on a 4-channel voltage to current converter. The pins from the range switch connect to on/off switches. The other pins from the on/off switches connect to ground. In addition, the fire buttons of the gameport adapter can be used as digital inputs by switching them from +5V to ground.

3.2.11 DI-149 (DATAQ) driver

DATAQ Instruments (http:\\www.dataq.com) sells a series of very nice low-cost AD-converters that they market as "Starter Kits". The DI-149 has 8 differential input channels, 10 bit resolution, and an input voltage range of -10V to +10V. The maximal sampling rate of this AD-converter is 10,000 Hz (for all channels together). WinAD sets the internal sampling rate for this AD-converter to 1,000 Hz for each individual channel. If different sampling rates are selected in WinAD, the originally sampled 1,000 Hz data are interpolated to the requested sampling rate. This is a very nice AD-converter and - as of this writig - costs only $59.00. Thus, for applications where a higher sampling rate (compared to the DI-145) is needed, this low-cost AD-converter may be an excellent choice.

3.2.12 DI-155 (DATAQ) driver

DATAQ Instruments (http:\\www.dataq.com) sells a series of very nice low-cost AD-converters that they market as "Starter Kits". The DI-155 has 4 differential input channels, 13 bit resolution, and selectable input voltage ranges of ±2.5V, ±3.125V, ±5V, ±6.25V, ±10V, ±12.5V, ±25V, ±50V. Each of the 4 channels can have a different input voltage range. The maximal sampling rate of this AD-converter is 10,000 Hz (for all channels together). WinAD sets the internal sampling rate for this AD-converter to 1,250 Hz for each individual channel. If different sampling rates are selected in WinAD, the originally sampled 1,250 Hz data are interpolated to the requested sampling rate. This is a very nice AD-converter and - as of this writig - costs only $149.00. Thus, for applications where a higher sampling rate (compared to the DI-145) and selectable input voltage ranges are needed, this low-cost AD-converter may be an excellent choice.

3.3 Data Acquisition with WinAD

1. Connect AD-converter to computer. At this time you may have to install Windows driver for the AD-converter in use. See manual for AD-converter for information on driver installation.

2. Connect recording equipment (e.g., amplifiers) to AD-converter.

3. Start WinAD

4. Select the appropriate Driver in WinAD (e.g., *Driver - DLP-IO8*).

5. **Determination of a useful sampling rate:** The sampling rate should be at least twice (better several times) the highest frequency of contained in the recorded signal. For example, the heart rate may be considered the highest frequency in a blood pressure signal. The heart rate in humans can be as high as 200 bpm (=3.3 Hz). However, since the blood pressure waveform is not sinusoidal, one has to consider the harmonics included

in the waveform that can have frequencies up to ∼50 Hz. The noise coming from AC power lines that are sometimes picked up from recording equipment has a frequency of 60 Hz in the US. If that frequency should be resolved, the sampling rate should be at least 120 Hz. Due to these considerations, a reasonable sampling rate for the blood pressure signal in humans is 250 Hz or higher.

6. The sampling rate in WinAD is always determined by the settings in the *Options - Sampling Rate* window. This sampling rate is shown on the status line at the bottom of the WinAD window. If an AD-converter is used that does not support as high a sampling rate, WinAD physically samples at the highest sampling rate supported by the AD-converter and then interpolates the values to achieve a final sampling rate identical to the settings in the *Options - Sampling Rate* window. The data files are saved at that (interpolated) sampling rate.

7. Select sampling rate: *Options - Sampling Rate* (e.g., 250 Hz).

8. Select the trace length: *Options - Trace Length.* The trace length is the time period of the recording that is shown in the WinAD window during recording. For example, a value of 60 s means that the last 60 s of the recording are always visible in the WinAD window.

9. Setup primary (direct) channels: *Options - Channel Setup.* Activate the appropriate channel (depending on which channel is connected to your recording devices) by activating the appropriate check boxes. Select a color (e.g., red) and a relative width (e.g., 2) for the signal. For some AD converters bipolar vs. unipolar recording modes can be selected.

10. Setup derived channels: *Options - Channel Setup.* For each primary (or direct) channel, up to 7 derived channels can be recorded. Derived channels are derived online during the recording from the primary channels based on the periodic features of the signal. For example, from a blood pressure signal the following derived channels can be obtained: Heart rate, systolic, mean, and diastolic blood pressure, pulse pressure amplitude, the maximum and minimum of the first derivative of the pressure signal.
In the "Derived Channels Setup" window the waveform of the recorded signal is visible. Often noise prevents a clear waveform to be seen. This can be solved by using the "Moving Average" slider. If the signal is too low, the "Amplification" slider can be used to amplify the signal. If the periodic features are too fast to see a clear periodic waveform, the "Trace length" slider can be adjusted. Finally, the vertical slider on the left side must be used to select the threshold for detection of the periodic features, such that the horizontal line goes through the periodic features of the recorded signal. Again, the color and width of the channel for the derived signal can be selected.

11. Calibration: *Calibrate - Channel x.* Set your recording equipment (amplifier etc.) to a low value and enter this value under "Real value:". Then start sampling by clicking "Sample On". Once the A/D values are stabilized, turn sampling off by clicking on "Sample Off". Repeat this process for a higher value (after activating the "High" check box. Click "OK" and the recording channel is calibrated.

12. Saving the setup. After calibration it is a good idea to save the setup (including the calibration data). Simply use *File - Save Setup.* **The Setup file is very important for data recovery in the rare case that WinAD crashes during the recording**.

13. Start Recording: *Record - Start Recording.*

14. Enter Comments during recording: With the WinAD window active, just type text using the keyboard. The text will appear in the line at the bottom of the WinAD window. Hitting the "Enter" key on the keyboard puts the Comment in the file.

15. Monitor Mode: *Options - Monitor Mode.* In monitor mode, WinAD does not save the data. It just shows the data on the screen without saving! In the status line at the bottom of the WinAD window, a letter "M" (for monitor) appears after the recording time. To return to recording mode just use *Options - Monitor Mode* again. The letter "M" after the recording time will switch back to "R" (for recording).

16. End Recording: *Record - End Recording.*

17. Saving the recording: *File - Save Data.* It is important to use *File - Save Data* and not *File - Save Setup.* There are 4 File formats in which the recorded data can be saved: ASCII format

3.4 WinAD Modules

Currently WinAD includes the following modules:

1. Stimulator Module
2. Trigger Module
3. PID Regulator Module
4. Telemetry Module (experimental)

3.4.1 Stimulator Module

Manual not written yet.

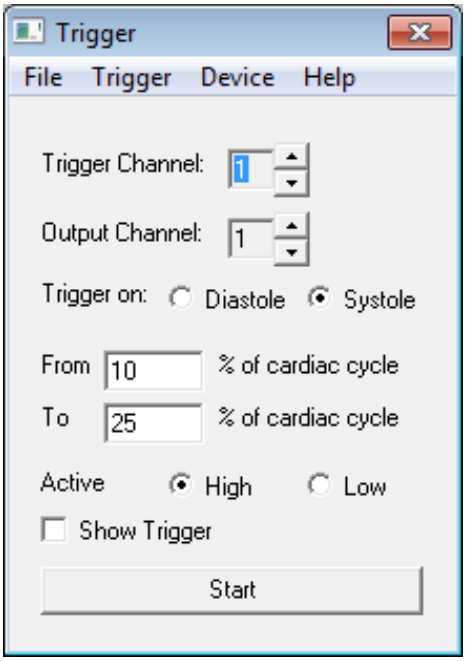

Figure 3.4: Trigger Dialog Box

3.4.2 Trigger Module

The Trigger Module allows to output a trigger signal depending on the phase of a periodic input signal. For example, if you record arterial blood pressure waveforms, you can setup the trigger function to generate a trigger signal during systole or diastole. The trigger output can be a pin on the parallel port or an output channel of the DLP-IO8 (USB-based AD-converter board). The trigger output can then used to operate other devices, such as an electrical stimulator. This would allow to perform experiments, such as nerve stimulation during specific phases of the cardiac cycle. Fig. 3.4 shows the *Trigger Dialog Box* used to setup the trigger.

In the following I assume the Trigger Module is used to generate a trigger signal in a predetermined phase of the cardiac cycle based on blood pressure waveform recordings. The Trigger Channel is the recorded channel on which the trigger is based. It is important to setup the trigger parameters in the *Options - Channel Setup - derived* dialog box (see section 3.3). The Output Channel is the channel of the output device (parallel port or DLP-IO8) that outputs the trigger signal. It is possible to trigger based on diastole or systole. This means a cardiac cycle is defined to start at the diastolic or systolic blood pressure value. The From and To values define the time point in the cardiac cycle during which the trigger signal is on. These values are given as percent of the duration of

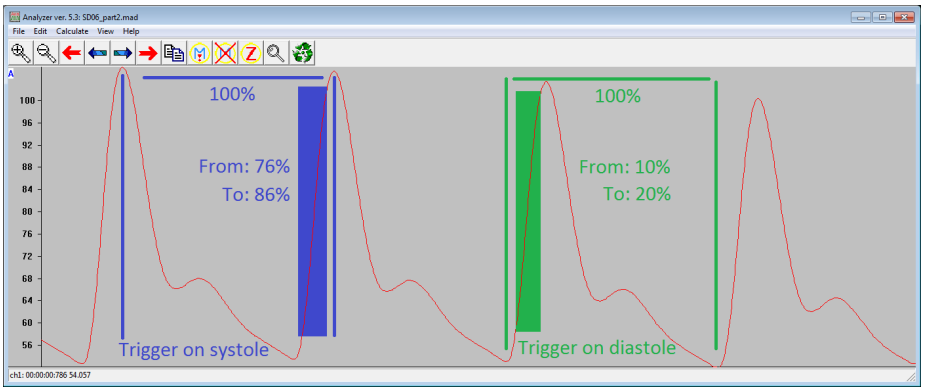

Figure 3.5: Timing of the Trigger Module

the cardiac cycle. From or To values of 0% or 100% should be avoided. For example, to trigger during systole the From value could be 10% and the To value could be 20% (because the duration of the systole is $\sim$ 33% of the cardiac cycle) if the trigger is based on diastole (cardiac cycle is defined to start at the diastolic blood pressure value). If the trigger is based on systole, a From value of 76% and a To value of 86% would also trigger during systole. Active *High* or *Low* means that the output of the output device is either high (e.g., 5V) or low (e.g., ground) during the time when the trigger is on. If the *Show Trigger* checkbox is checked, red vertical lines will be drawn in the recording when the trigger is on. Using the *File Menu* it is possible to save and load the parameters for the trigger for later reuse. Fig. 3.5 illustrates the timing of the Trigger Module. Fig. 3.6 shows screen shots of the Trigger Module in action. In the top screen shot, the trigger is active during systole and in the bottom screen shot the trigger is active during diastole. The respective settings in the Trigger Module dialog box are also shown.

3.4.3 PID Regulator Module

Manual not written yet.

3.4.4 Telemetry Module

Manual not written yet.

3.5 Data Recovery if WinAD crashes

No larger software packages are without errors that can cause the software to crash. With WinAD crashes can occur but in general, crashes are rare. If the

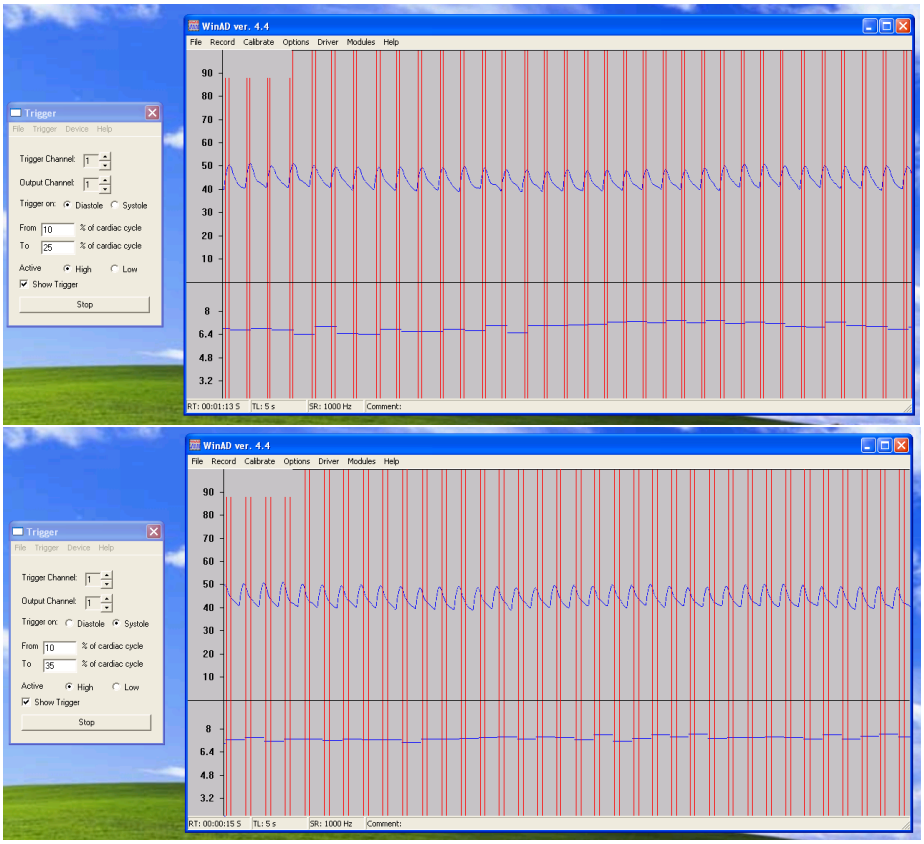

Figure 3.6: Screen shots of the Trigger Module in action

software crashes in the middle of an recording it is **only** possible to recover the data if WinAD is not started again after the crash.

1. DO NOT START RECORDING WITH WINAD AFTER A CRASH IF YOU WANT TO RECOVER THE DATA FROM THE RECORDING PRIOR TO THE CRASH!

2. Each MS-Windows user has a folder (or directory) on the hard disk for application data. Typically, this folder is hidden from the user. To show hidden folders use "Folder Options" in the "Control Panel" of MS-Windows. Under the "View" tab activate "Show hidden files and folders".

3. Find the HemoLab folder. On Windows XP this folder is at: `C:\Documents and Settings\User\Application Data\HemoLab`. Make a new folder within this HemoLab folder and rename it to "temp". Then copy all files from the HemoLab folder into the new "temp" folder.

4. Only after you have copied all files from the "HemoLab" folder into the "temp" folder, start WinAD and load the setup file from the previous recording (the one that crashed). If you did not save the setup you have to enter all options and calibration data manually.

5. Then start recording: *Record - Start Recording* and after about 5 s of recording stop the recording: *Record - End Recording*.

6. Now copy all files from the new "temp" folder back into the HemoLab application folder overwriting all files in the HemoLab folder with the files from the "temp" folder.

7. Now you can save your data from the previous recording (the one that crashed) by: *File - Save Data*.

Chapter 4

Analyzer

4.1 What is Analyzer

The main window of the Analyzer software is shown in Fig. 4.1.

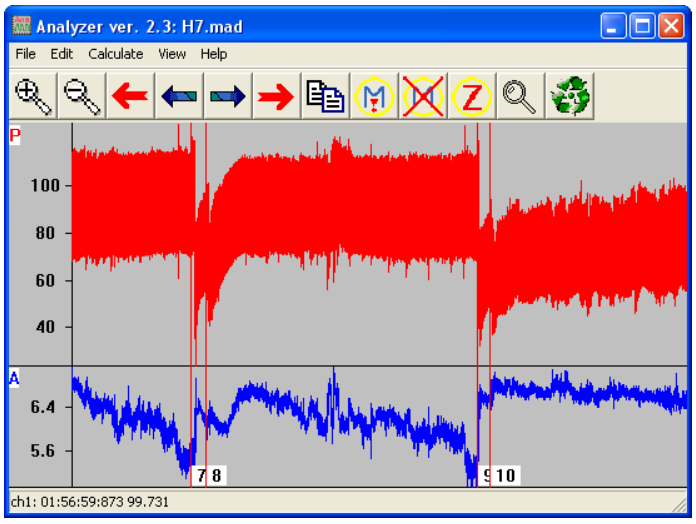

Figure 4.1: Analyzer, main window

Analyzer is a data analysis software specifically designed for hemodynamic data. This software was originally written for UNIX/Linux using the X-window Motif window manager. The first version of the software was released almost a decade ago. Since then, additional features were included constantly and the software was converted to the Windows operating system. As a result, the current version of Analyzer is very mature and stable, and offers a wide variety of data analysis features as listed below.

- Mean, Minimum, and Maximum values
- Area under the curve (AUC)
- Derive new parameter from direct pulsatile blood pressure or left ventricular pressure signals, such as systolic, mean, diastolic blood pressure, heart rate, dP/dtmax, dP/dtmin, and blood pressure amplitude.
- Filtering of data (Butterworth filter, moving average)
- Spline interpolation
- Special functions for sympathetic nerve activity analysis
- Arithmetic calculations based on the data of two channels (e.g., calculate vascular conductance or resistance from blood pressure and blood flow data).
- Spectral analysis using Fast Fourier Transform (FFT) or autoregressive modeling.
- Determination of baroreceptor reflex function by the *sequence technique*
- Detrended fluctuation analysis
- Automatic and manual artifact removal

4.2 File Formats

Currently Analyzer can open the following file formats:

- ASCII or Text files
- TSA files (a binary format used in Analyzer and Batch processor)
- MAD files (the format of the WinAD data acquisition software)
- WAV files (yes, that's an audio format)
- DSI files (Data Sciences International - telemetry files)
- WDQ files (WinDAQ, DATAQ Instruments)
- S32 files (EndoPAT file format)

4.2.1 ASCII or Text files

I have received reports from some users who had problems opening text files. In most cases the problem was related to the file name ending in ".txt". For example, a file with the file name "Exp01.txt" will not be automatically detected as an ASCII (text) file. Analyzer uses the file extension ".asc" to identify ASCII (text) files. The easiest way to deal with this is to rename the files from e.g., "Exp01.txt" to "Exp01.asc". ASCII (text) files may contain comments.

4.3 A word about average HR calculated from beat-by-beat data files

Because of the longer pulse intervals at lower heart rates, the time duration of a given number of heart beats is longer at a low HR than at a high HR. Thus, in calculating average HR, the beat-by-beat HR values need to be weighted based on the pulse interval duration. The arithmetic average of the beat-by-beat sampled HR values (without weighing for pulse interval duration) overestimates the true average HR, because high HR values are (incorrectly) given the same weight as low HR values.

The Analyzer software deals with this problem by calculating average heart rate from beat-by-beat data files by the ratio of number of heart beats and recording duration. However, to identify beat-by-beat sampled heart rate channels, the channel must be marked as:

Name: frequency

Unit: bpm or Hz

Spacing: beat-by-beat

These parameters can be checked by clicking with the right mouse button on a channel and selecting "Channel Details". Analyzer automatically sets these parameters correctly, when deriving HR using "Calculate - Derived Channels by Threshold" or when importing tsa files. However, when importing text or ASCII files, these parameters may not be set correctly and need to be changed manually in "Channel Details".

4.4 Derived Channels by Threshold

Analyzer can derive new hemodynamic parameters from data files that contain periodic signals, such as heart rate from EKG or systolic, mean, and diastolic blood pressure from a pulsatile blood pressure waveform signal. It can also determine contractility index dP/dt_{max} and dP/dt_{min} from left ventricular pressure waveform signals.

1. Load the periodic signal (e.g., blood pressure waveform or EKG) into Analyzer (*File - Load Data*). Typically, these are equidistant sampled time series (e.g., sampling rate = 1000 Hz).

2. Activate the periodic signal (middle mouse button or right mouse button and context menu, red "P" for passive turns into a blue "A" for active).

3. Select *Calculate - Derived Channels by Threshold*. In the "Derived Channels by Threshold" (see Fig. 4.2) select what type of parameters are to be derived from the input signal.

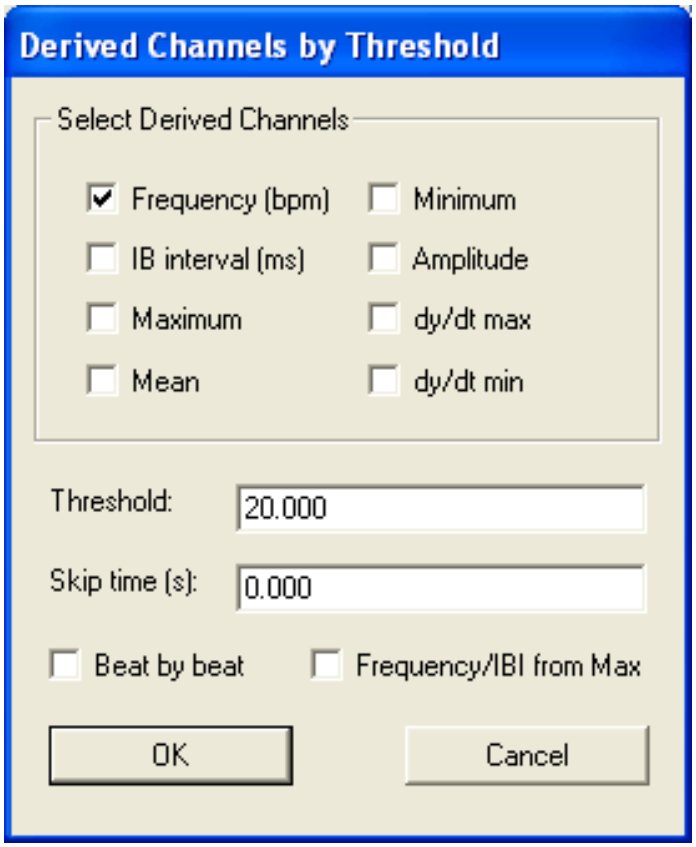

Figure 4.2: Analyzer, Dialog for derived channels by threshold

4. The threshold value is the amount by which the signal must continuously increase from a local minimum (e.g., diastolic blood pressure value) in order to detect a heart beat. Typically, half of the pulse pressure amplitude (for blood pressure signals) or half of the height of the R wave (for EKG signals) works well. The software suggests a "Threshold" value based on the local minima and maxima in the time series. If this suggested threshold does not work well (too many artifacts), you can zoom out on a section of the time series, so that just a few individual heart beats are visible. Zoom on a section that appears to have the smallest amplitude (e.g., lowest R-wave). You can then determine a threshold value based on the pulse pressure or hight of the R-wave. One may have to play a little around with the threshold on a file by file basis to obtain the best results (least number of artifacts).

5. The software-suggested "Threshold" value is determined from the section

of the input signal that is selected by a "red rectangle". Thus, before you open the *Calculate - Derived Channels by Threshold* dialog box, you may want to select a red rectangle (drag the mouse with the left mouse button pressed) around a portion of the input signal that is artifact free and that appears to be the best section to determine the threshold value.

6. Enter a "Skip time" in seconds. From the time point at which a periodic event has been triggered (e.g., beginning of systole in BP signal or R-wave in EKG) the next periodic event (e.g., next systole or next R-wave) is only searched starting the "Skip time" after the beginning of the previous periodic event. This feature can be used if there are marked dicrotic waves (in BP signals) or marked T-waves (in EKGs) that may be mistaken as heart beats. By introducing a "Skip time" the dicrotic wave or the T-wave can be skipped before the next heart beat is identified.

7. Select "Beat by beat" if the derived channels should be saved on a beat-by-beat basis, i.e., one value per heart beat. If this checkbox is activated, the sampling rate of the derived channels is not equidistant. If this check box is not activated, the derived channels are sampled at the same equidistant sampling rate as the periodic input signal. There is a function in Batch processor to convert equidistant time series into beat-by-beat time series.

8. If the "Frequency/IBI from Max" checkbox is activated the "Frequency (Hz)" (e.g., heart rate) and "IB interval (s)" (interbeat interval, RR interval etc.) are calculated based on the time difference between two consecutive local maxima (e.g., peaks of R-waves) rather than from two consecutive local minima (e.g., diastolic blood pressure values). Local maxima typically work better for EKG signals, local minima typically work better for blood pressure signals.

9. Finally hit "OK" and the derived channels will show up as new channels in the main window of Analyzer.

10. If the signal appears "noisy" you may want to apply a low-pass Butterworth filter with a corner frequency of ∼20-40 Hz before calculating derived channels. Activate the channel to be used to derive the HR time series by clicking in the channel with the middle mouse button (some mice use the left and right button simultaneously to simulate the middle mouse button, on some mice the middle mouse button is integrated in the wheel). The red "P" (for passive) should change to a blue "A" (for active). Then select *Calculate - Butterworth Filter*. Select "Low Pass Filter" and enter a "Corner Frequency" of 20 Hz. Leave the "Filter order" at 4. Then click "OK". The low pass filter will be applied and the noise should be mostly gone.

4.5 Derived Channels by Trigger

This function was specifically implemented in Analyzer to study baroreceptor-sympathetic nerve activity reflex function. It is also useful if studying baroreceptor-heart rate reflex function if an ECG and BP recording is available. Assuming you recorded the arterial blood pressure waveform together with renal sympathetic nerve activity and want to assess how much the nerve activity changes for a given change in blood pressure. You could extract the systolic blood pressure on a beat-by-beat basis (using *Calculate - Derived Channels by Threshold* and then use *Derived Channels by Trigger* to calculate the nerve activity during each heart beat (average of the nerve signal during the time interval of each heart beat) to obtain a "beat-by-beat nerve activity time series". You could then apply the sequence technique (*Calculate - Baroreflex*) to study the baroreceptor-sympathetic nerve activity reflex function.

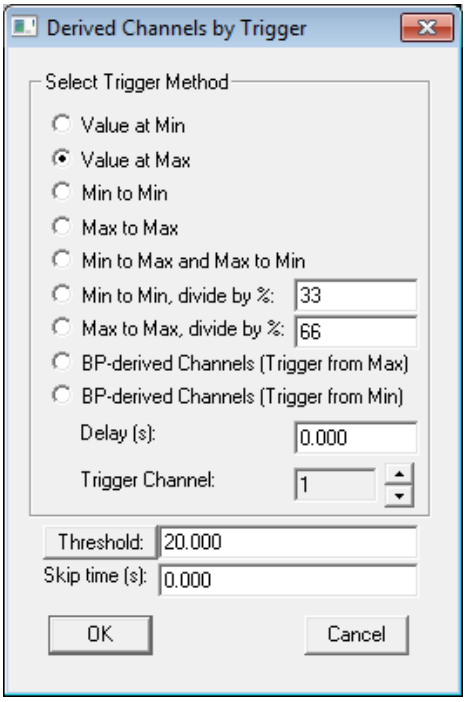

Figure 4.3: Derived Channels by Trigger Dialog

In the example above, you would generate "beat-by-beat nerve activity time series" by first activating the nerve activity channel (you can also activate multiple channels to extract beat-by-beat data on multiple channels simultaneously). You would then select *Calculate - Derived Channels by Trigger* from the menu to bring up the Derived Channels by Trigger dialog box (Fig. 4.3). There are 7

different methods to extract the beat-by-beat data. Five of them are illustrated in Fig. 4.4.

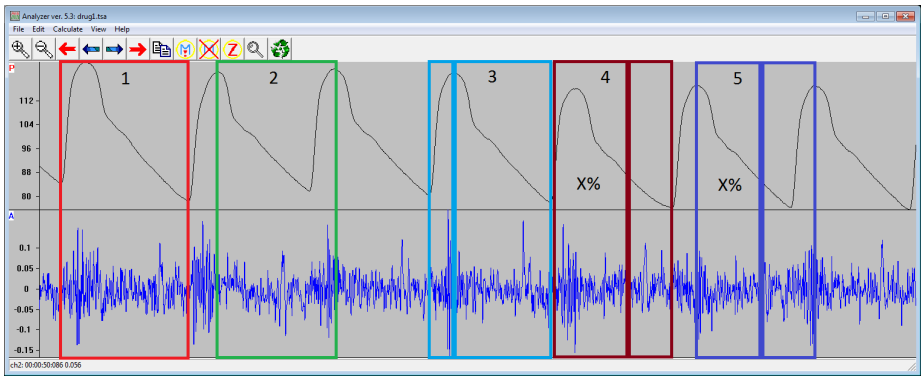

Figure 4.4: Different Methods to Derive Data by Trigger

Method 1 (Value at Min) takes the values of the active channels at the time (± delay) of the minimum of the trigger event (continuous increasing section of the trigger channel that increases at least by the threshold). Method 2 (Value at Max) is similar but takes the values of the active channels at the time (± delay) of the maximum of the trigger event. With method 3 (Min to Min), the signals of the active channels are averaged between two local minima (e.g., from diastolic to diastolic blood presssure). Method 4 (Max to Max) is similar but the data are averaged between two local maxima (e.g., from systolic to systolic blood pressure). Methods 5 (Min to Max and Max to Min), 6 (Min to Min, divide by %), and 7 (Max to Max, divide by %) generate two derived channels for each active channel. These methods will average the signals from a local minimum to the next local maximum and then from the local maximum to the next local minimum (method 5). In methods 6 and 7 the active channels are averaged over time periods of a given number of percent of the distance between a local minimum to the next local minimum (method 6) or between a local maximum and the next local maximum (method 7). For example with method 6, you can derive the average of the active channels for the first 75% and the second 25% of the interbeat interval starting at the diastolic blood pressure.

Methods 9 and 10 (BP-derived Channels (Trigger from Max/Min)) are specifically designed for use by the sequence technique to study baroreceptor-heart rate reflex function. Typically the trigger channel would be an ECG and the active channel would be a blood pressure (BP) waveform. These methods would then derive the RR-interval from the ECG and also derive heart rate, interbeat-interval, systolic, mean, and diastolic blood pressure for each heart beat following a R-wave in the ECG. This technique avoids inconsistent beat-by-beat time series if beat detection for the ECG and beat detection for the BP waveform result in unequal number of heart beats. This can happen if the detection algo-

rithm does not detect a heart beat properly due to artifacts in the ECG or BP waveform recordings.

You can also enter a time delay by which the active channels are precede (negative time delay) or lag (positive time delay) the trigger channel. The spin button next to the text field for the Trigger Channel allow you to select a trigger channel (e.g., the blood pressure waveform or an EKG recording). Finally, you need to enter a threshold and skip time that have the same purpose as in the *Derived Channels by Threshold* function (see above). Note that you can click on the "Threshold" button to obtain a suggested threshold value specific for the selected trigger channel.

4.5.1 Derive LV-EDP from LV pressure waveform data

Another useful application of the "Derived Channels by Trigger" function is to calculate left ventricular end-diastolic pressure from left ventricular pressure waveform data. Fig. 4.5 illustrates how to do this:

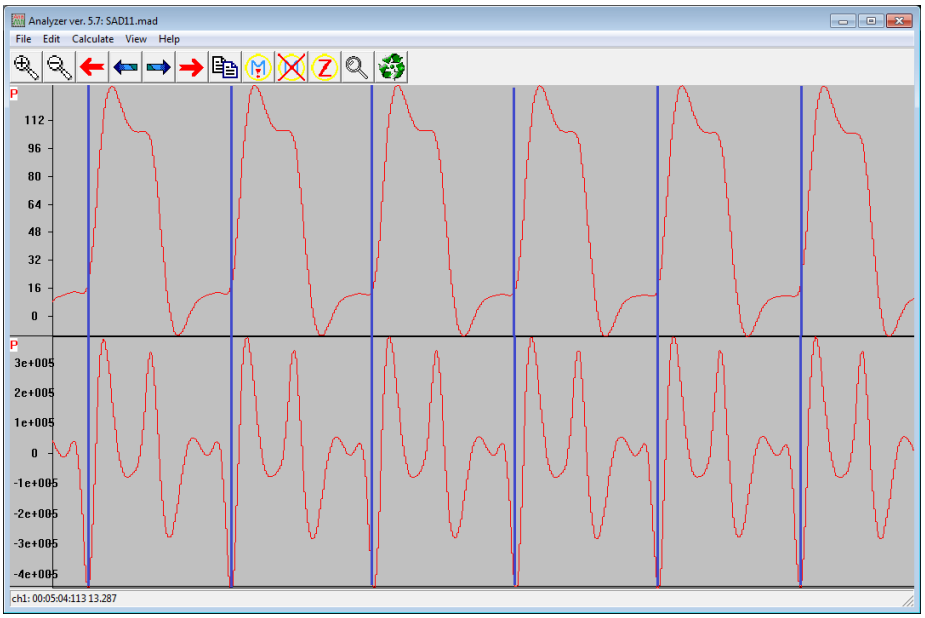

Figure 4.5: Derive LV-EDP from LV pressure waveform data

1. Activate the LV pressure waveform channel and use *Edit - Copy Channel*.
2. Only activate the newly copied channel.
3. Use *Calculate - Differentiate* to calculate the first derivative of the LV pressure waveform.

4. The 1^{st} derivative sometimes is a little noisy. Therefore, apply a low-pass Butterworth filter with a corner frequency of 40 Hz. (*Calculate - Butterworth Filter*, Filter Order 4).

5. Use *Calculate - Differentiate* once more to calculate the second derivative of the LV pressure waveform.

6. Since the "Derived Channels by Trigger" function uses a continuously increasing section of the time series for triggering, we take the inverse of the 2^{nd} derivative. Use *Calculate - Invert*.

7. Now the end-diastolic left ventricular pressure is marked by a local minimum in the inverted 2^{nd} derivative of the left ventricular pressure waveform file (see Fig. 4.5).

8. Finally, we can use "Derived Channels by Trigger" to derive the left ventricular end-diastolic pressure. First, we activate the original left ventricular pressure waveform channel. Then, we use *Calculate - Derived Channels by Trigger* and select "Value at Min" (because the local minimum in the inverted 2^{nd} derivative marks the end-diastolic pressure). For "Trigger Channel" we select the channel number that corresponds to the inverted 2^{nd} derivative channel and we select the default "Threshold" by clicking on the "Threshold" button. A "Skip time" is useful to avoid triggering on one of the other peaks in the inverted 2^{nd} derivative file.

9. Clicking on "OK" will result in a new beat-by-beat channel of the left ventricular end-diastolic pressure.

4.6 Detrended Fluctuation Analysis

Detrended fluctuation analysis (DFA) is a method for determining the statistical self-similarity of a signal that is frequently used in chaos theory [9]. As a non-linear technique, DFA can be applied to non-stationary time series, which are time series that change over time. The result of DFA is the fluctuation exponent α, which provides information on the correlation of the time series with itself. Th Analyzer software calculates the first order DFA only.

4.6.1 Calculation of the fluctuation exponent α

The time series x_i is integrated according to:

$$X_t = \sum_{i=1}^{t} (x_i - \bar{x})$$

Next, X_t is divided into segments of length L samples, a linear regression line is fitted to the data pairs $\{t, X_t\}$; $t\epsilon\{1 \dots L\}$, and the fluctuation (F) is

calculated from the linear regression lines of all segments of length L according to:

$$F(L) = \sqrt{\frac{1}{L}\sum_{i=1}^{L}(Xi - ai - b)^2}$$

The F(L) values from all segments of length L are averaged into one single F(L) value for this specific value of L. These steps are then repeated for all values of $L\epsilon\{L_1 \dots L_2\}$, resulting in $(L_2 - L_1 + 1)$ F(L) values.

Finally, a log-log graph of L against F(L) is constructed. A straight line on this log-log graph indicates statistical self-similarity. The scaling exponent α is calculated as the slope of a straight line fit to the log-log graph of L against F(L). In fact, the Analyzer software calculates two α exponents as the slopes of the linear regression lines between points $\{L_{\alpha_1 1}, F(L_{\alpha_1 1})\}$ and $\{L_{\alpha_1 2}, F(L_{\alpha_1 2})\}$ for α_1 and $\{L_{\alpha_2 1}, F(L_{\alpha_2 1})\}$ and $\{L_{\alpha_2 2}, F(L_{\alpha_2 2})\}$ for α_2.

4.6.2 Interpretation of the fluctuation exponent α

$\alpha < 0.5$ the time series is anti-correlated with itself within the time window corresponding to L1 and L2.

$\alpha \simeq 0.5$ the time series is uncorrelated with itself within the time window corresponding to L1 and L2 (similar to white noise).

$\alpha > 0.5$ the time series is correlated with itself within the time window corresponding to L1 and L2.

$\alpha \simeq 1$ the time series is characterized by pink noise (1/f noise) within the time window corresponding to L1 and L2.

$\alpha > 1$ the time series is unbounded or non-stationary (random-walk like) within the time window corresponding to L1 and L2.

$\alpha \simeq 1.5$ the time series is characterized by Brownian noise within the time window corresponding to L1 and L2.

4.6.3 Step-by-step instructions for DFA

1. Start Analyzer and open a data file using *File - Load Data*.
2. Activate the channel for DFA (red "P" should change to a blue "A") using the middle mouse button (or clicking with the wheel, or on two button mice clicking both mouse buttons at the same time) or use the context menu (right mouse button).
3. Determine the number of data points in the active channel by clicking with the right mouse button on the active channel and selecting *Channel Details* from the context menu. The number of data points is provided as *Number of Values*. Remember this value.

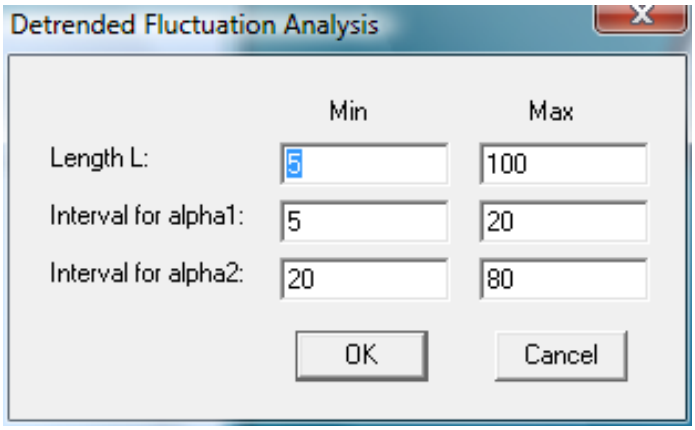

Figure 4.6: Detrended Fluctuation Analysis Dialog

4. Select the time period for which DFA shall be carried out by drawing a red rectangle around the section of the data to be analyzed. Drag the mouse with the left button pressed, a red rectangle should appear. The rectangle only marks the time. Thus, the height of the rectangle doesn't matter. If the complete BP signal should be used, make sure you are on total view (*View - Total View*) and then select the complete recording.

5. Use *Calculate - Detrended Fluctuation Analysis*. The dialog box shown in Fig. 4.6 appears.

6. Enter the boundaries for L, α_1, and α_2 (minimum must be ≥ 3 and maximum cannot exceed the nunber of data points in the active channel as determined above). Then click on OK.

7. A new window with the results of the DSA will show up as shown in Fig. 4.7. The new window shows the log-log plot of L against F(L) and provides the boundaries for L, α_1, and α_2 in number of data points as well as in time units (seconds). The values for α_1 and α_2 are provided as the slopes. The correlation coefficient R and the significance level for the linear correlation of the data points are also provided.

8. In the output window (Fig. 4.7) the boundaries for α_1 and α_2 can be changed and the analysis repeated by using *Edit - Refresh*. The results can be copied and pasted to other software (e.g., MS-Excel) using *Edit - Copy Header to Clipboard* and *Edit - Copy Data to Clipboard*.

9. Finally, the graph of the log-log plot and the data points for the log-log plot can be saved by using *File - Save Image (GIF)*. This function saves two files: (1) the image of the log-log plot in the GIF format (*.GIF) and the data points for the log-log plot as an ASCII file (*.asc).

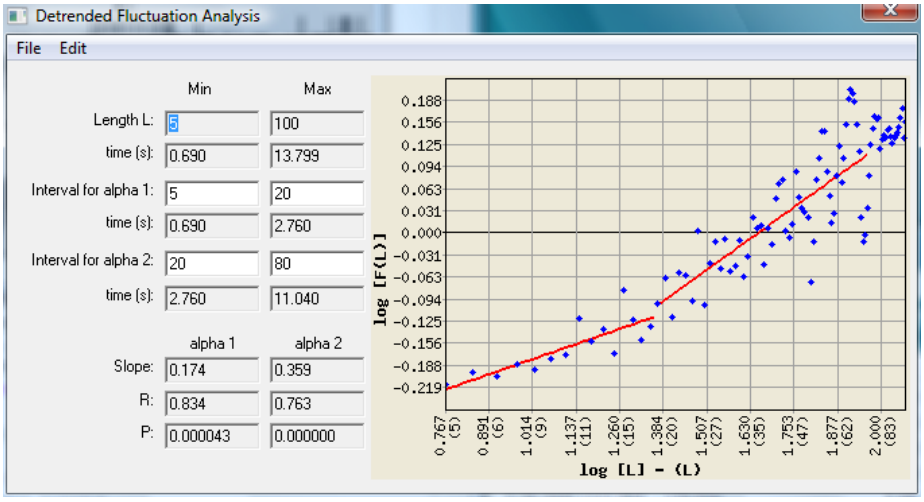

Figure 4.7: Detrended Fluctuation Analysis Results

4.6.4 Pitfalls in the interpretation of DFA

It is always possible to calculate values for α_1 and α_2. However, this does not automatically imply that the time series is self-similar. Self-similarity requires that the points on the log-log plot are sufficiently collinear between the boundaries for α_1 and α_2. Thus, the Analyzer software provides the R and P values for the linear regression lines used to calculate the slopes (α-values) of the log-log plot between the boundaries for α_1 and α_2. Only use the values for α_1 and α_2 if the R values are reasonably high (e.g., R>0.8) and the P values are significant (e.g., P<0.05)! Using these criteria, α_2 may not be used in the example shown in Fig. 4.7 because R<0.8.

4.7 Joint Symbolic Dynamics

Symbolic dynamics allows a simplified description of the dynamics of a system with a limited amount of symbols [3]. Joint symbolic dynamics tries to interpret symbolic dynamics of two (or more) simultaneously recorded time series (e.g., RR-intervals and systolic blood pressure values) to study the interaction between these time series. Currently, only symbolic dynamics (for one time series) with a word length of 3 symbols is implemented in Analyzer. However, it is possible to export the time series of the words for further analysis, such as joint symbolic dynamics analysis.

4.7.1 Computation

A time series X is transformed in a symbol time series S by the following transformation:

$$X = \{x_1, x_2, ..., x_n\}\, x_i \in R$$

$$S = \left\{ \begin{array}{lcl} 0 : (x_n - x_{n-1} & \leq & 0 \\ 1 : (x_n - x_{n-1} & > & 0 \end{array} \right\}$$

Thus, the length of the symbol time series S is n-1. Next, the symbol time series S is subdivided into words with a length of 3 symbols. Thus, 8 (2^3) different words are possible (000, 001, 010, 011, 100, 101, 110, 111). Each single word is obtained by a shift of one within the symbol string S. Thus, the symbol time series S (n-1 elements) is converted into a word time series W with n-3 elements. Finally, the absolute and relative (in percent) occurrence of each of the 8 different words is counted and presented in the output window (Fig. 4.8).

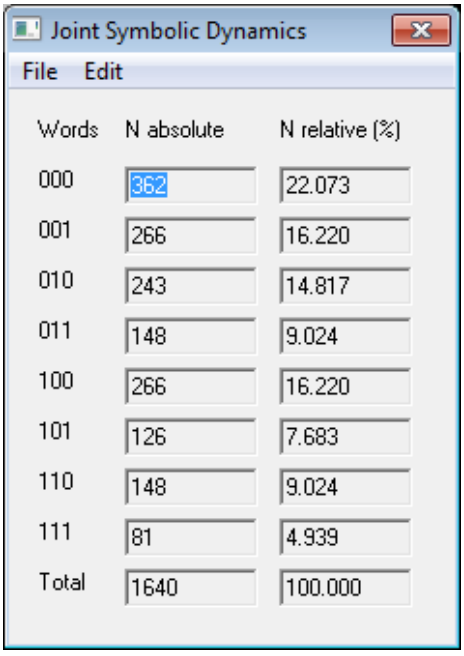

Figure 4.8: Joint Symbolic Dynamics Analysis Results

The menu of the output window allows to save the time series of the words as text (ASCII) file (*File - Save Words Time Series*). It also allows to copy the results in the clipboard (*Edit - Copy Header* and *Edit - Copy Results*) in order to paste them into other software such as Microsoft Excel.

4.7.2 Application

Joint Symbolic Dynamics analysis should be performed from beat-by-beat blood pressure (e.g., systolic blood pressure), heart rate, or RR-interval time series. Some investigators apply a moving average filter to the beat-by-beat time series before calculation of the Joint Symbolic Dynamics analysis to filter out respiration-related fluctuations. However, because this practice effectively removes respiratory sinus arrhythmia from heart rate or RR interval time series, it will no longer be possible to study the effect of parasympathetic modulation of heart rate because parasympathetic modulation of heart rate affects mostly the respiratory sinus arrhythmia. Therefore, I do not recommend filtering of heart rate of RR interval time series before calculation of Joint Symbolic Dynamics analysis.

4.8 Baroreflex Analysis - Sequence Technique

4.8.1 General Remarks

The sequence method, as first described by DiRienzo et al. [5] and Bertinieri et al. [4], identifies sequences of four or more heart beats, where blood pressure (BP) and pulse interval change in the same direction.

For all individual sequences of BP in mmHg (x-axis) and pulse interval in ms (y-axis) values linear regression lines are calculated. The average of the slopes of all individual regression lines is then used as an index of baroreceptor-heart rate reflex sensitivity. Accordingly, the unit of this index is ms/mmHg.

The Analyzer software can use a direct pulsatile BP signal (e.g., from an arterial catheter, a telemetric BP recording system, or a Finapres device) or it can use beat-by-beat sampled BP and pulse interval (or RR interval) time series. In the latter case, the beat-by-beat values for BP and pulse interval need to be matched up. Please refer to the section on "Derived Channels by Trigger" (4.5) for information on how to properly extract beat-by-beat time series from the ECG and blood pressure waveforms.

4.8.2 Baroreflex from direct pulsatile BP signals

1. Start Analyzer and open the direct pulsatile BP signal using *File - Load Data*.
2. If the BP signal is "noisy" you may have to apply a low pass Butterworth filter *Calculate - Butterworth Filter* with a corner frequency of ∼20 Hz to smooth the signal. Detection of heart beats does not work properly if the BP signal is "noisy".
3. Activate the BP channel (red "P" should change to a blue "A") using the middle mouse button (or clicking with the wheel, or on two button mice clicking both mouse buttons at the same time) or use the context menu that comes up with the right mouse button.

4. Select the time period from which the baroreflex should be analyzed by drawing a red rectangle around the section of the BP recording. Drag the mouse with the left button pressed, a red rectangle should appear. The rectangle only marks the time. Thus, the height of the rectangle doesn't matter. If the complete BP signal should be used, make sure you are on total view (*View - Total View*) and then select the complete recording.

5. Use *Calculate - Baroreflex* to bring up the "Baroreflex according to Bertinieri" Window.

6. Leave "Channels for y-axis:" on "0". A zero for this parameter tells the software to derive the pulse intervals from the pulsatile BP channel (active channel).

7. Enter a "Threshold:" for detection of heart beats. The threshold value is the amount by which the signal must increase from a local minimum (i.e., diastolic BP value) in order to detect a heart beat. Typically, half of the pulse pressure amplitude works well.

8. Enter a "R for inclusion:" value. For each sequence of BP - pulse interval values, a linear regression line is calculated. Only those sequences are used for the final estimation of baroreflex sensitivity that have an R-value that is greater than the value for this parameter. Typically, 0.8 is a good choice.

9. Select a "Delay in beats". The baroreflex affects heart rate only after a certain time delay that is related to the neuronal transmission of the signal from the baroreceptors to the brain and back to the sinus node of the heart. Parasympathetic modulation of sinus node function is very fast (within one heart beat in humans). However, sympathetic transmission is slower and can take several heart beats. Thus, a delay between the BP values and the pulse interval values can be used when detecting the sequences. In mice a delay of 0 or 3 beats has been suggested [6]. In rats a delay of 0 beats often works well. In humans a delay in the range of 0 to 5 beats may be appropriate. The software can also identify the delay that results in the largest number of sequences (BRR max. N sequences) or the delay that results in the largest gain of the reflex (BRR max. gain).

10. Select if you want to use the Systolic, Mean, or Diastolic BP for calculation of baroreflex sensitivity. Most investigators use the systolic BP.

11. Check "IBI from Max" if you want to determine the pulse intervals from two consecutive systolic BP values rather than from two consecutive diastolic BP values.

12. Finally, click "OK".

4.8.3 Baroreflex from beat-by-beat BP and pulse interval signals

1. Load the beat-by-beat BP and pulse interval time series into Analyzer using *File - Load Data*. You can also derive these time series in Analyzer from a direct pulsatile BP waveform signal.

2. Use *View - Show all channels* to show all channels. Then count the channels from top to bottom and determine the channel number of the pulse interval channel. For example, if you only have the BP channel on top and the pulse interval channel on bottom, the channel number for the pulse interval channel is 2.

3. Make sure the pulse interval channel is in milli seconds (ms) and not in seconds. If it is in seconds, you can convert it to ms using *Calculate - Linear Transformation* with the "Parameter: m" on 1000 and the "Parameter b:" on 0.

4. Activate only the BP channel by clicking on it with the middle mouse button (or the wheel on the mouse, or both button on 2-button mice) or by using the context menu that shows up using the right mouse button. **Do not activate the pulse interval channel.**

5. Select the time period from which the baroreflex should be analyzed by drawing a red rectangle around the section of the recording. Drag the mouse with the left button pressed, a red rectangle should appear. The rectangle only marks the time. Thus, the height of the rectangle doesn't matter. If the complete recording should be used, make sure you are on total view (*View - Total View* or the "recycle icon") and then select the complete recording.

6. Select *Calculate - Baroreflex* from the Analyzer menu.

7. Enter the channel number of the pulse interval channel for "Channel for y-axis" using the up- and down-arrows.

8. The "Threshold:" parameter is not used because the pulse intervals are not derived from a direct pulsatile BP signal.

9. Enter a "R for inclusion:" value. For each sequence of BP - pulse interval values, a linear regression line is calculated. Only those sequences are used for the final estimation of baroreflex sensitivity that have an R-value that is greater than the value for this parameter. Typically, 0.8 is a good choice.

10. Select a "Delay in beats". The baroreflex affects heart rate only after a certain time delay that is related to the neuronal transmission of the signal from the baroreceptors to the brain and back to the sinus node of the heart. Parasympathetic modulation of sinus node function is very fast

(within one heart beat in humans). However, sympathetic transmission is slower and can take several heart beats. Thus, a delay between the BP values and the pulse interval values can be used when detecting the sequences. In mice a delay of 0 or 3 beats has been suggested [6]. In rats a delay of 0 beats often works well. In humans a delay in the range of 0 to 5 beats may be appropriate. The software can also identify the delay that results in the largest number of sequences (BRR max. N sequences) or the delay that results in the largest gain of the reflex (BRR max. gain).

11. Select if you want to use the Systolic, Mean, or Diastolic BP for calculation of baroreflex sensitivity. Most investigators use the systolic BP.

12. The "IBI from Max" check box is not used because the pulse intervals are not derived from a direct pulsatile BP signal.

13. Finally, click "OK".

4.8.4 The Baroreflex Analysis Output Window

1. The Baroreflex Analysis window is organized in 2 major columns. One is for the baroreflex sequences and one is for so-called "Non-BRR Sequences" or "Feed-forward Sequences". The Non-BRR Sequences are sequences where the BP and pulse interval go in opposite directions. For example, synchronous increases or decreases in BP and heart rate would result into Non-BRR Sequences. It is believed that these sequences are caused by activation or deactivation of the sympathetic nervous system acting on the blood vessels and the sinus node of the heart.

2. The graphics on top of the window shows all sequences that were detected. One can scroll through the sequences (the green sequence is the current sequence) using the `<<<` and `>>>` buttons. Sequences that do not "look right" (e.g., due to artifacts) can be excluded from the analysis by using the `+/-` button. Excluded sequences appear in yellow. The x-axis in the diagrams is the BP in mmHg, the y-axis is the pulse interval in ms (not heart rate!). Thus, the units of the sensitivity (or gain) of the reflex is ms/mmHg. In the table below the graphs the results of the analysis is listed for all sequences pooled and for the up and down sequences analyzed separately.

3. The results of the data analysis can be copied into other software, such as MS-Excel using the *Edit - Copy Header Row* and *Edit - Copy Data* menus. In the other software, simply use "Paste" to copy the results from the clipboard.

4. A very detailed report of the baroreflex analysis can be obtained by *File - Save*.

4.9 Pulse Wave Analysis

4.9.1 General Remarks

Recently, there has been an increased interest in pulse wave analysis, because it has been suggested that parameters derived from the waveform of the arterial pulse are powerful independent predictors for cardiovascular events [2, 7]. Two of these derived parameters are (1) the central (aortic) pulse wave velocity (PWV) and (2) the central augmentation index (AI). Both of these parameters are derived from the waveform of the pressure pulse in the ascending aorta. High central PWV (in humans above 9-10 m/s) and large central AI (in humans above 20%) indicate greater stiffness of the aorta (e.g., due to atherosclerosis) and greater cardiovascular risk. It is also important to note that the central PWV appears to have better predictive power to estimate cardiovascular risk than the peripheral PWV (i.e., PWV derived from pulse waveforms obtained in the brachial or radial artery). Ideally, one would record the pulse (pressure) waveforms in the ascending aorta (or in the carotid artery) and at the distal end of the aorta (or in the femoral artery). Central PWV can then be calculated as the ratio of the length of the aorta and the time delay between the two pulse waveforms.

4.9.2 Reflected Waves

The pressure wave, generated during systole by the ejection of the stroke volume into the aorta travels along the aorta until it reaches arterial branching points, or smaller resistance arteries. At these distal points, the forward pressure wave is reflected and a reflected pressure wave travels back towards the heart. Once the reflected pressure wave arrives at the ascending aorta, the forward and backward (reflected) waves add together to form the total pressure waveform that can be recorded with a tip catheter placed in the ascending aorta. This phenomenon is illustrated in Fig. 4.9.

4.9.3 Non-invasive assessment of central PWV and AI

Ideally, the pressure waveform in the ascending aorta should be recorded directly in order to determine central PWV and AI. However, since this requires invasive procedures, indirect techniques have been developed that allow reconstruction of the ascending aortic (central) pressure waveform from peripherally measured waveforms, such as the finger blood pressure recorded from a Finapres device or even from pulse oxymeter recordings that are proportional to finger blood flow.

Reconstruction of the central pressure waveform from peripheral wave forms requires application of so-called transfer functions that describe the transformation from peripheral to central pulse waveforms. Such a transfer function for the reconstruction of central (aortic) pulse waveforms from tonometrically recorded radial artery pressure waveforms has been described by Sunagawa's research group from Osaka, Japan [13]. This transfer functionis implemented

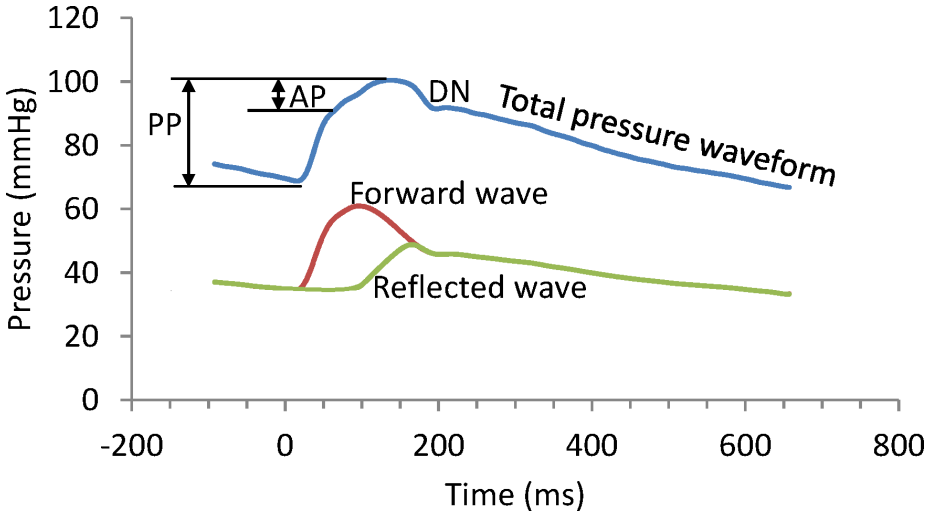

Figure 4.9: Reflected waves. AP-Augmentation Pressure, PP-Pulse Pressure, DN-Dicrotic Notch (closure of aortic valve)

in the *Analyzer* software (*Pulse Wave Analysis - Transfer function peripheral pulse − > central pulse (Humans)*).

Sometimes, it is more convinient to record peripheral flow than peripheral pressure, because flow can be easily obtained by an ultrasound Doppler device or even a cheap pulse oximeter. However, since total flow is the difference between forward and backward flow, while pressure waves are additive, the flow signal must be converted into a pressure signal before applying any pulse waveform analysis algorithms. The *Analyzer* software includes a function to convert peripheral flows in peripheral pressures (*Pulse Wave Analysis - Convert peripheral flow − > peripheral pressure (Humans)*).

Once a pressure waveform from the ascending aorta has been obtained through direct invasive recording or from conversion of a peripheral pressure waveform, central PWV and AI can be extracted from the aortic pressure waveform. The algorithm for this extraction has been published by Qasem and Avolio [10] and is implemented in the *Analyzer* software (*Pulse Wave Analysis - PWV and AI: from one pressure waveform*).

Of course, PWV can also be determined from two simultaneously recorded pressure or flow waveforms. This is done by dividing the distance between the two measuring points by the time delay between the two signals. The *Analyzer* software provides a function that does exactly that (*Pulse Wave Analysis - PWV: two pressure/flow waveforms*). Sometimes cardiac left ventricular pressure is measured together with a peripheral pressure (e.g., with a femoral artery catheter). In this case the central PWV can also be derived from these two signals (*Pulse Wave Analysis - PWV: LVP and one arterial pressure waveform*).

4.9.4 Reconstruction of the central pressure waveform from peripheral waveforms

The transfer function that converts peripheral pressure waveforms to ascending aortic (central) pressure waveforms is based on a three parameter Windkessel model. The three parameters for the model have been determined by the group of Sunagawa [13] for the conversion of radial artery pressure waveforms to ascending aortic pressure waveforms. The parameters are entered in the dialog box shown in Fig. 4.10. Two of the parameters (CR and Zc/R) are very robust and changing these parameters has little effect of the result of the transfer function. However, the parameter Td corresponds to the transmission delay from the ascending aorta to the measuring point at the radial artery. This delay should be adjusted individually for each subject. This delay can also be adjusted for measuring points other than the radial artery (e.g., brachial artery). The Td parameter is entered in units of seconds.

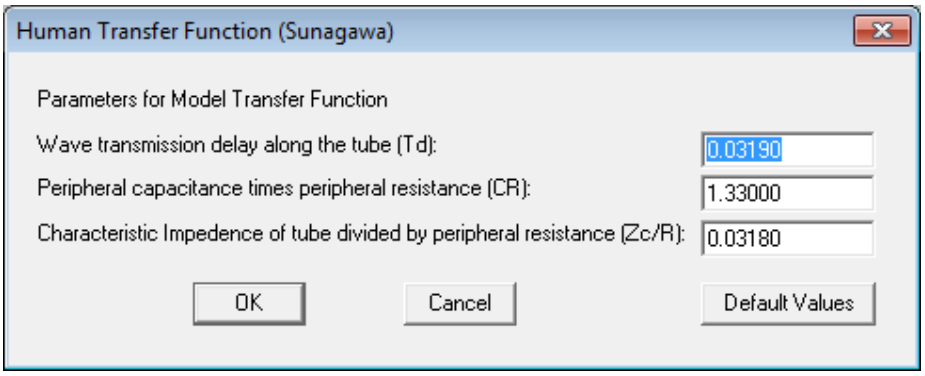

Figure 4.10: Human Transfer Function Dialog

4.9.5 Convert peripheral flow to peripheral pressure

As outlined above, if PWV and AI are to be computed from peripheral flow recordings, the flow must first be converted to pressure before this peripheral pressure waveform can then be converted into a central (ascending aortic) pressure waveform. For determination of PWV and AI the pressure waveform does not need to be calibrated. Thus, relative flow signals are sufficient for this type of analysis.

An example is shown in Fig. 4.11. The flow signal shown in the upper channel was obtained from the radial artery using a Doppler device. Please note that the flow signal contains positive and negative flow values. The negative (backward) flow is caused by the reflected wave traveling from the periphery (Hand) back to the measuring site at the radial artery. Since forward and backward flows a subtracted, a negative flow results. This radial artery blood flow

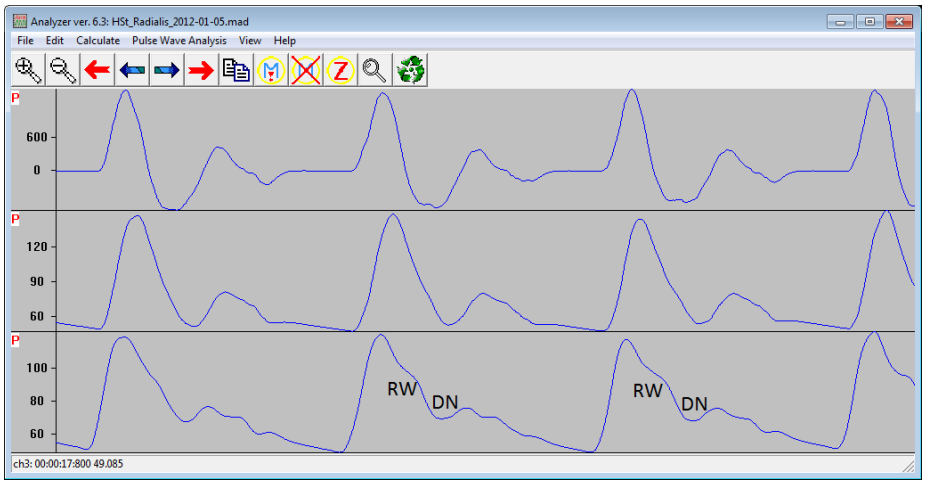

Figure 4.11: Conversion of peripheral flow (top trace) to pressure (middle trace) and conversion of peripheral pressure (middle trace) to central (ascending aortic) pressure (bottom trace). The reflected waves (RW) and dicrotic notches (DN) are marked.

signal (upper trace) was converted to a peripheral pressure signal using *Pulse Wave Analysis - Convert peripheral flow − > peripheral pressure (Humans)*. Note that the pressure signal (middle channel) contains only positive pressures. Finally, the peripheral (radial artery) pressure signal was converted to a central (ascending aortic) pressure waveform using *Pulse Wave Analysis - Transfer function peripheral pulse − > central pulse (Humans)*. The central pressure waveform is shown in the bottom trace. The reflected wave (RW) and the dicrotic notch (DN), marking the closing of the aortic valve, are clearly visible in the reconstructed central pressure waveform.

4.9.6 PWV and AI from central pressure waveform

To calculate central PWV and central augmentation index (AI) from an ascending aortic pressure waveform, you can use the *Pulse Wave Analysis - PWV and AI: from one pressure waveform* menu item. The dialog box for this function is shown in Fig. 4.12. The distance parameter refers to the effective reflecting distance (EfRD, roughly the length of the aorta), which must be determined individually for each subject. The threshold and skip time parameters have the same meaning as described for *Derived Channels by Threshold* and are used to identify individual heart beats (see section 4.4 on page 27. You can select whether you want to see markers for the start of the heart beat, the peak of the forward wave, and the dicrotic notch, as well as the forward and backward pressure waves. You can also choose to calculate the time delay of the forward

and backward wave (Delay in ms), the PWV (PWV in m/s), and the central augmentation index (Central AI in %).

The effective reflecting distance, EfRD, can be calculted based on the age, height, and weight of the subjects.

$$EfRD = 0.17 * age + 0.6649 * BMI + 34.55\,cm \tag{4.1}$$

BMI refers to the body mass index (weight [kg]/height [m]^2). This equation is based on a multiple regression analysis using true PWV values obtained by Sphygmocor. To calculate EfRD, click on the *Distance (cm):* button (see Fig. 4.12).

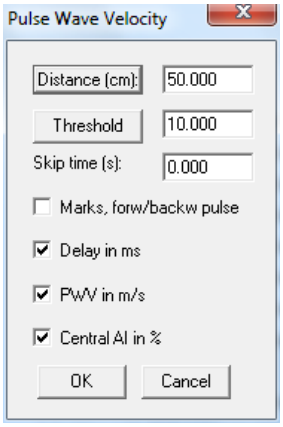

Figure 4.12: PWV from central pressure waveform dialog box

4.9.7 PWV from two pressure or flow waveforms

Calculation of PWV from two pressure waveforms is relatively easy. You activate the two channels that contain the waveforms and and open the dialog box shown in Fig. 4.13 by using the *Pulse Wave Analysis - PWV: two pressure/flow waveforms* menu item. The distance parameter is the distance between the two pressure measurement sites that needs to be determined individually for each subject. You can choose to low pass filter the signals prior to calculating PWV if the pressure waveforms appear noisy. The threshold and skip time parameters have the same meaning as described for *Derived Channels by Threshold* and are used to identify individual heart beats (see section 4.4 on page 27. Finally you can choose to output the time delay between the two pressure/flow waveforms (in ms) and the PWV (in m/s).

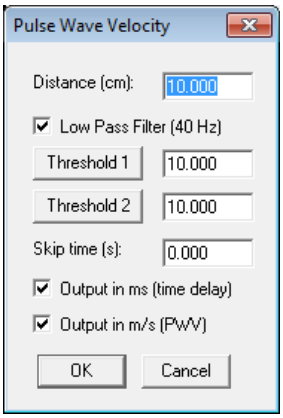

Figure 4.13: PWV from two waveforms (central and peripheral pressure) dialog box

4.9.8 PWV from LVP and a peripheral pressure waveform

PWV can also be calculated from cardiac left ventricular pressure and peripheral pressure waveforms. The calculation is a little different than for the case of two arterial pressure waveforms, because the corresponding time points in the left ventricular and peripheral pressure waveforms are the time points when the left ventricular pressure equals the diastolic blood pressure (opening of aortic valve) and the beginning of the peripheral pressure waveform (marked by the beginning of the systolic increase in pressure), respectively. The dialog box this function is shown in Fig. 4.14. The distance parameter is the distance from the aortic valve to the measurement site of the peripheral pressure waveform (in cm). You have to indicate if the first of the two activated channels is the left ventricular pressure signal (or if it is the second activated channel). You have the option to low-pass filter the signals if they appear noisy. The threshold and skip time parameters have the same meaning as described for *Derived Channels by Threshold* and are used to identify individual heart beats (see section 4.4 on page 27. Finally, you can choose to calculate the time delay between the left ventricular pressure and peripheral pressure (in ms) and the PWV (in m/s).

4.9.9 Left Ventricular Perfusion Index

The left ventricular perfusion index can be calculated as the area between the aortic pressure and left ventricular pressure waveforms during diastole. Before this area can be calculated, the two pressure waveforms need to be aligned with respect to the opening of the aortic valve (diastolic blood pressure). Analyzer does this alignment automatically. The principle is shown in Fig. 4.15.

To calculate the left ventricular perfusion index use *Pulse Wave Analysis - Left Ventricular Perfusion Index*. The dialog box shown in Fig. 4.16 will show

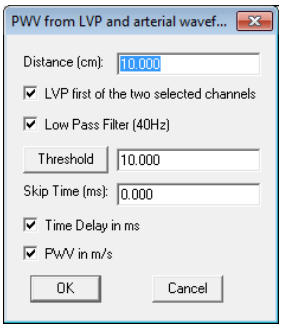

Figure 4.14: PWV from left ventricular pressure and peripheral pressure waveforms dialog box

up.

You have to indicate if the first of the two activated channels is the left ventricular pressure signal (or if it is the second activated channel). You have the option to low-pass filter the signals if they appear noisy. The threshold and skip time parameters have the same meaning as described for *Derived Channels by Threshold* and are used to identify individual heart beats (see section 4.4 on page 27. Analyzer calculates the left ventricular perfusion index on a beat-by-beat basis. The units are mmHg*seconds.

4.10 Generate Shorties

Let's say you have continuously recorded blood pressure waveforms for several days and you want to calculate the mean values and spectral powers every hour of your recording. The first step would be to extract a clean section of your recording (e.g., 10 min) for each hour of your recording. You would then derive heart rate, systolic, mean, and diastolic blood pressure for each of these 10 min sections. Finally, you would use the batch processor to calculate mean values and perhaps to perform spectral analysis. Extracting these 10 min sections can be very time consuming if you are dealing with long recordings (e.g., one week of continuous recordings results in 168 10 min sections).

Analyzer has a function that automatically extracts such "clean" sections (called "shorties") from your time series. Use *Calculate - Generate Shorties* und the dialog box shown in Fig. 4.17 will come up. The first parameter ("Every") is the duration (in minutes) of the repeated time intervals in which you want to identify "clean" segments. The second parameter ("Find the cleanest segment of length:") is the duration (in seconds) of the "clean" segments (also called "shorties"). There is a check box to mark if the shorty files shall be really generated or if the location of the shorties should be shown only. The next parameter ("Counter starts at:") is the number attached to the file name of the first shorty file. Subsequent shorties have incremented numbers attached to the

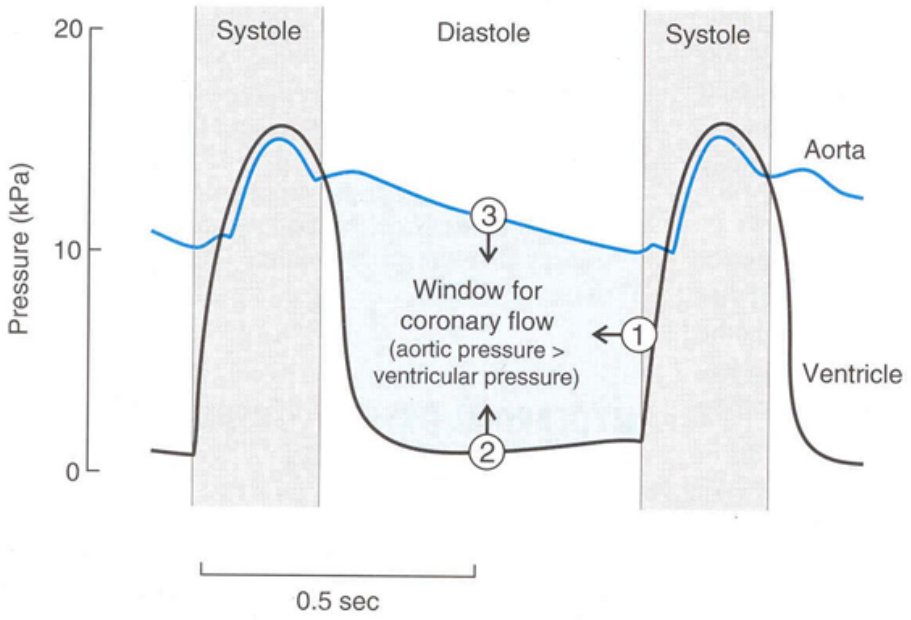

Figure 4.15: Left ventricular perfusion index or the "window for coronary flow".

file name. This allows to label the first shorty with the hour of the day when the recording was started. Finally the output directory needs to be selected by clicking on the push button labeled "Output Directory".

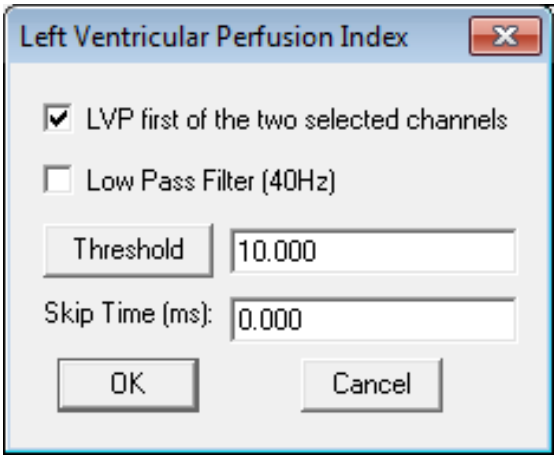

Figure 4.16: Left ventricular perfusion index dialog box.

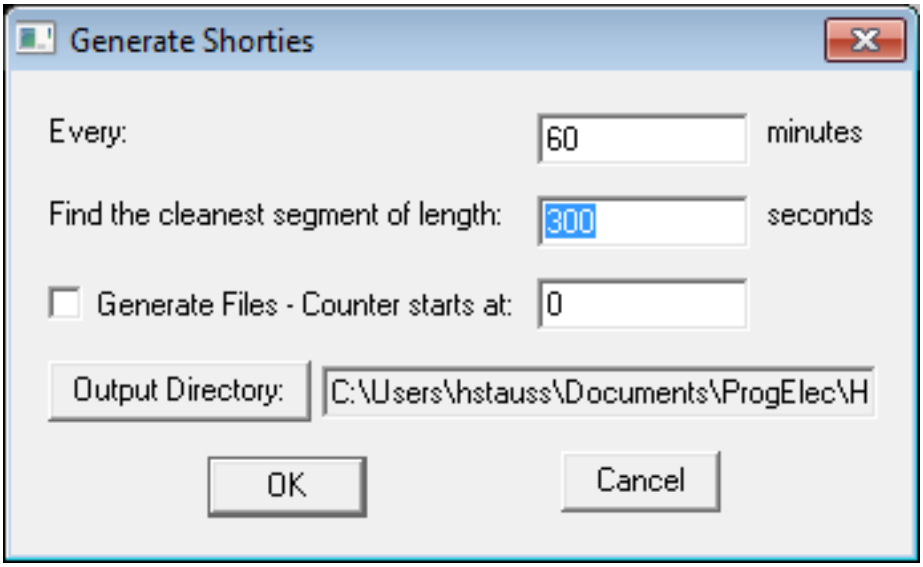

Figure 4.17: Generate Shorties

Chapter 5

Batch Processor

5.1 What is Batch Processor

Often a large number of similar data files need to be analyzed. Batch processor was designed to allow analysis of a large number of data files automatically. This makes data analysis of large numbers of similar data files highly time efficient. The main window of the Batch Processor software is shown in Fig. 5.1.

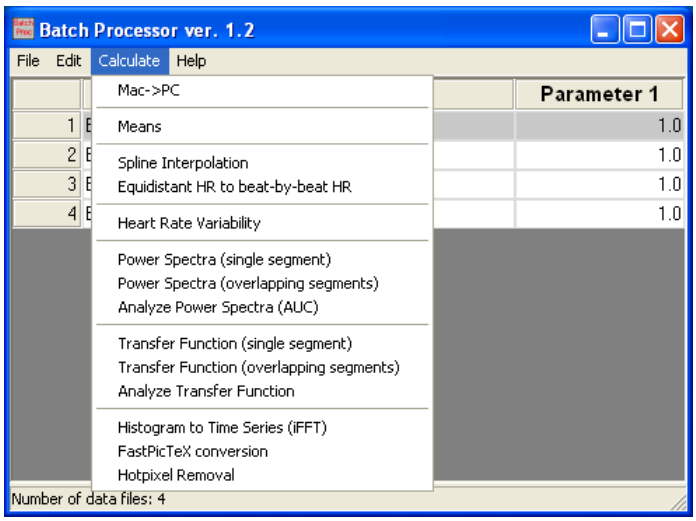

Figure 5.1: Batch Processor, main window

Batch processor currently includes the following functions:

- Conversion of text files from MacIntosh computers to Windows computers
- Calculation of mean values for all channels in all files

- Spline Interpolation (also used to convert beat-by-beat sampled data files into equidistant data files).
- Conversion of equidistant (at fixed sampling rate) sampled HR time series to beat-by-beat sampled HR time series.
- Heart Rate Variability (time domain analysis)
- Powerspectral analysis
- Transfer function analysis.
- A function that converts frequency distributions (histograms) to a time series that can then be used with the autoregressive modeling function of Analyzer to identify and analyze multi-modal frequency distributions.
- Conversion of FastPicTeX files (e.g., from WinStat) to other output formats (TIF, JPG, GIF, PS, and PDF).
- Hotpixel removal. This is an imaging processing function that removes dead pixels from images taken with digital cameras. Of course, this is not a true hemodynamic analysis function.

5.2 Calculate Mean Values

To calculate descriptive statistics on the hemodynamic parameters derived from an experiment, it is often necessary to derive the mean values of all time series obtained from the experiment. Batch processor can calculate the mean values of many files (time series) in one single step.

1. Add all files to be analyzed in Batch Processor (*File* - *Add Files*).
2. Use *Calculate* - *Means* from the menu of the Batch Processor.
3. A new window pops up that allows to select an output file with the mean values of all channels of all files added to the Batch Processor main window.
4. A new window (Command Prompt) with black background color will show up that shows the batch process that calculates the mean values for all files listed in the main window of the Batch Processor. Once finished, you are prompted to "Press any key to continue ...". After pressing a key the Command Prompt window disappears and the new output file is generated.
5. Repeating this step (selecting *Calculate* - *Means* again) with the same output file will append the new results to the end of the existing file. The file will not be overwritten.
6. The resulting output file is a so-called "tab-delimited" text file that can be directly opened in MS-Excel or WinStat and other software that can read "tab-delimited" ASCII or text files.

5.3 Spline Interpolation

If necessary, this function will perform a low pass filter (in order to prevent aliasing) and then use a cubic spline interpolation to resample the data. If beat-by-beat data files are to be converted to equidistant files, the beat-by-beat data must be provided in the `>>> *.tsa <<<` format.

1. Add all files to be spline interpolated into Batch Processor (*File - Add Files*).
2. Select *Calculate - Spline Interpolation* from the menu of the Batch Processor.
3. Select the channel number of the channel that should be interpolated. This channel number starts at 1 and will apply to all files added to the main window of the Batch Processor.
4. A channel number of "0" means that all channels in the files are spline interpolated.
5. Type in the output sampling rate.
6. If the input files are text (or ASCII) files, enter the sampling rate for the input files. If the input files are TSA files, this parameter is ignored.
7. Push the "Select Output Directory" pushbutton to select the output directory. Make sure the output directory is different from the input directory (where your input files are saved). The output files will have the same file names as the input files. **If the input and output directory are the same, all input files will be overwritten without further notice.**
8. Finally hit "OK". A new (command prompt) window with black background will come up that shows the progress of the spline interpolation of the files listed in the main window of the Batch Processor.
9. At the end you need to push any button to return to Batch Processor. The spline interpolated output files will be in the selected output directory.

5.4 Butterworth Filter

Use this function to apply *low pass* or *high pass* filters to multiple time series. Select *Calculate - Butterworth Filter* to bring up the Butterworth Filter dialog box as shown in Fig. 5.2.

First, select the channel number to which the filter should be applied. A channel number of 0 means that all channels in the file should be filtered. Select "Low Pass Filter" or "High Pass Filter". The filter order describes how sharp the filter cuts off high or low frequencies at the corner frequency. Higher numbers make the filter "look more like" a rectangular filter. Select the corner frequency

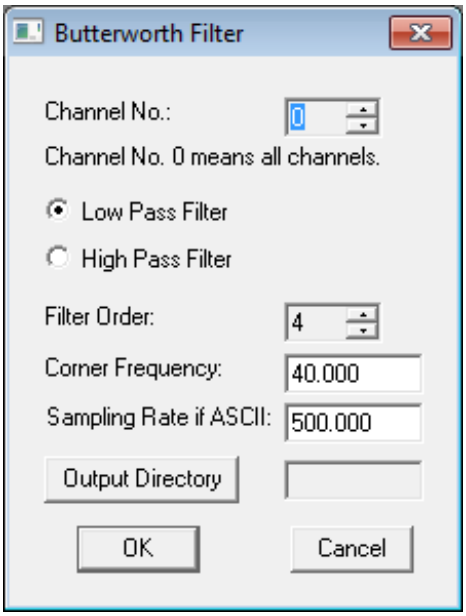

Figure 5.2: Batch Processor, Butterworth Filter Dialog Box

of the filter. In case the input files are ASCII (or Text) files, the sampling rate for the input files need to be entered. Finally, select an output directory for the filtered files. The same file names will be used as the input file names. **Therefore, if you don't select a different output directory, the input files will be overwritten.**

Chapter 6

HR Variability Time Domain Analysis

6.1 General Remarks

1. It is highly recommended to become familiar with the Paper of the Task Force of The European Society of Cardiology and The North American Society of Pacing and Electrophysiology [8]. In addition, for a review on the mechanisms contributing to heart rate variability see [12].

2. Time Domain Heart Rate (HR) Variability (HRV) parameters include:

   SDNN (ms) Standard deviation of all NN intervals.

   SDANN (ms) Standard deviation of the averages of NN intervals in all 5 min segments of the entire recording.

   RMSSD (ms) The square root of the mean of the sum of the squares of differences between adjacent NN intervals.

   SDNN index (ms) Mean of the standard deviations of all NN intervals for all 5 min segments of the entire recording.

   SDSD (ms) Standard deviation of differences between adjacent NN intervals.

   NN50 count (no units) Number of pairs of adjacent NN intervals differing by more then 50 ms in the entire recording.

   pNN50 (%) NN50 count divided by the total number of all NN intervals.

   HRV triangular index (no units) Total number of all NN intervals divided by the height of the histogram of all NN intervals measured on a discrete scale with bins of 7.8125 ms (1/128 s).

   TINN (ms) Baseline width of the minimum square difference triangular interpolation of the highest peak of the histogram of all NN intervals.

Differential index (ms) Difference between the widths of the histogram of differences between adjacent NN intervals measured at selected heights.

Logarithmic index (no units) Coefficient φ of the negative exponential curve $k \cdot e^{\varphi t}$ which is the best approximation of the histogram of absolute differences between adjacent NN intervals.

3. From these parameters, the following are implemented in HemoLab: SDNN, RMSSD, SDSD, NN50, pNN50, and HRV triangular index.

4. Time domain HRV parameters are calculated from beat-by-beat sampled NN intervals. These are time series that contain one value for the inter-beat interval (or RR-interval or NN-interval) in milliseconds (ms) per heart beat. **Nevertheless, HemoLab calculates time domain HRV parameters from beat-by-beat HR time series in beats per minute (bpm).**

5. These beat-by-beat time series are typically of 5 minutes duration (in humans). The time domain HRV parameters actually depend on the duration of the time series. Typically, the values of the time domain HRV parameters become larger if the time series are longer. Therefore, it is important, that a fixed duration (e.g., 5 min) is used for time domain HRV analysis.

6. Beat-by-beat time series can be derived (using the Analyzer software) from the EKG or other signals that contain a heart beat synchronous periodicity, such as a blood pressure or flow signal.

6.2 Use Analyzer to derive beat-by-beat HR values

1. Load heart beat synchronous time series (e.g., EKG, blood pressure, or blood flow signal) into Analyzer using *File - Load Data*. Typically, these are equidistant sampled time series (e.g., sampling rate = 1000 Hz).

2. Select *Calculate - Derived Channels by Threshold*. Select "Frequency (bpm)" (to derive the HR in bpm) and "Beat by beat" (to obtain one value for every heart beat). Depending on whether the peak of the waveform (e.g., tip of R wave in EKG) or the lower values of the waveform (e.g., diastolic blood pressure) has a more distinct (or sharp) feature, select or unselect "Frequency/IBI from Max". If this check box is checked, the HR (frequency) will be calculated as the inverse of the time difference between two consecutive maxima. If this check box is not checked, the HR (frequency) will be calculated from the minima of the waveform. For EKG the maxima typically work better, while for blood pressure signals the minima (diastolic values) seem to work better.

3. The threshold value is the amount by which the signal must continuously increase from a local minimum (e.g., diastolic blood pressure value) in order to detect a heart beat. Typically, half of the pulse pressure amplitude (for blood pressure signals) or half of the height of the R wave (for EKG signals) works well. The software suggests a "Threshold" value based on the local minima and maxima in the time series. If this suggested threshold does not work well (too many artifacts), you can zoom out on a section of the time series, so that just a few individual heart beats are visible. Zoom on a section that appears to have the smallest amplitude (e.g., lowest R-wave). You can then determine a threshold value based on the pulse pressure or hight of the R-wave. One may have to play a little around with the threshold on a file by file basis to obtain the best results (least number of artifacts).

4. If the signal appears "noisy" you may want to apply a low-pass Butterworth filter with a corner frequency of ∼20-40 Hz before deriving HR. Activate the channel to be used to derive the HR time series by clicking in the channel with the middle mouse button (some mice use the left and right button simultaneously to simulate the middle mouse button, on some mice the middle mouse button is integrated in the wheel). The red "P" (for passive) should change to a blue "A" (for active). Then select *Calculate - Butterworth Filter*. Select "Low Pass Filter" and enter a "Corner Frequency" of 20 Hz. Leave the "Filter order" at 4. Then click "OK". The low pass filter will be applied and the noise should be mostly gone.

5. Enter a "Skip time" in seconds. From the time point at which a periodic event has been triggered (e.g., beginning of systole in BP signal or R-wave in EKG) the next periodic event (e.g., next systole or next R-wave) is only searched starting the "Skip time" after the beginning of the previous periodic event. This feature can be used if there are marked dicrotic waves (in BP signals) or marked T-waves (in EKGs) that may be mistaken as heart beats. By introducing a "Skip time" the dicrotic wave or the T-wave can be skipped before the next heart beat is identified.

6. Click "OK".

7. This procedure will generate two new channels that contain (1) the time points of each HR value and (2) the HR values in Hz (not in bpm). The HR signal may contain some artifacts at positions, where heart beat detection did not work properly. This is typically due to artifacts in the original file. To remove these artifacts, use *Calculate - Artifact Removal*. Only activate the new HR channel (middle mouse button, or context menu with right mouse button). Zoom out on an artifact. Mark the artifact by dragging a red rectangle around the artifact (hold left mouse button and drag mouse). Select "Linear Interpolation" and "Real" from the Artifact Removal window (make sure "Automatic" is off) and click "OK". This should remove the artifact. Repeat this for all artifacts. You can also try

"Automatic" artifact removal. However, make sure you do this in "Test" mode first because the automatic function can easily destroy your time series.

8. To save the beat-by-beat HR time series, hide all channels except the HR time series channel. Activate (blue "A") all channels except the HR channel and use *View - Hide active channels*. Then use *File - Save Data*. The HR time series can be saved in the ASCII (text) or TSA (binary) file format. For ASCII select "Save as type:" as ASCII (*.asc), for TSA select "Save as type:" as TSA (*.tsa). Enter a file name and save the file. It is a good idea to use an extension ".asc" or ".tsa" for the filename, because the extension is used by the Batch Processor to identify the file format.

9. Since the HR is sampled on a beat-by-beat basis, the output file will contain 2 channels. The first channel is a time channel and contains the time point of every heart beat. the second channel is the heart rate channel.

10. Repeat these steps for all files from which you want to calculate time domain HRV parameters.

6.3 What if HR files are sampled at a fixed sampling rate?

HR files that are sampled at a fixed (equidistant) sampling rate can be converted to beat-by-beat sampled time series using the Batch Processor.

1. If the file is a text (or ASCII) file you need to know the sampling rate. If it is a TSA file, the sampling rate is encoded in the file. You also need to know in which channel the HR is. In text (ASCII) files, this is the column number of the HR time series. If you have just one column with the HR values, the channel number is simply 1.

2. Text files must have the file extension `>>> *.asc <<<`. TSA files must have the extension `>>> *.tsa <<<`.

3. Add all files that should be converted to beat-by-beat files to Batch Processor (*File - Add Files*).

4. Select *Calculate - Equidistant HR to beat-by-beat HR* from the menu.

5. Enter the HR Channel No. using the up- and down-arrows.

6. Enter the sampling rate if your files are text (or ASCII) files. If the input files are in the TSA format, the sampling rate is ignored (because the sampling rate encoded in the TSA file is used).

7. Select a directory for the output files. **Make sure this is a different directory than your input directory. Otherwise all input files will be overwritten without warning!**

8. Click OK. A new window (with black background) pops up. You are required to hit any key to continue.

9. The output directory will contain the output files that have the HR as beat-by-beat values added as a new channel.

6.4 Use Batch Processor to calculate HRV parameters

1. Start Batch Processor.

2. Use *File - Add Files* and open all HR files (beat-by-beat HR values in bpm) that should be analyzed (hold the "Ctrl" button on the keyboard while selecting all individual files with the mouse). The files should show up in the main window of Batch Processor.

3. Use *Calculate - Heart Rate Variability* and select the channel that contains the beat-by-beat sampled HR time series in beats per minute (bpm). Since the Analyzer software also saves the time of each heart rate in a separate channel, the HR channel is likely to be the second channel. Thus, you need to enter 2 for "HR channel (beat-by-beat, bpm)".

4. Select a time period for which HRV will be calculated. Typically, 5 minutes are used in human studies.

5. After clicking "OK" a new window pops up that allows to select an Output File for the results of the HRV analysis. Select a directory, enter a file name, and click on "Save". It is a good idea to use a file name extension of ".txt", because other software, such as MS-Excel or WinStat identifies such files as text (tab-delimited) files.

6. The resulting output file can be opened in MS-Excel, WinStat, or any other software that can read tab-delimited text (ASCII) files. For each input file (in Batch Processor) the output file contains one line with the values for: number of heart beats analyzed, time duration analyzed, average NN interval length in ms, variance of NN intervals, SDNN, RMSSD, SDSD, NN50, pNN50, and HRV triangular index.

Chapter 7

Spectral Analysis

7.1 General Remarks

For a review of spectral analysis of arterial blood pressure see [11] and for a review on the mechanisms contributing to heart rate variability see [12].

7.1.1 What signals to use for Spectral Analysis?

First of all, it is important to note that the FFT technique requirs equidistant time series. While the autoregressive technique (in general) does not require equidistant time series, the implementation in the Analyzer software does require equidistant time series. Thus, beat-by-beat sampled HR or BP data cannot be used directly. Beat-by-beat sampled data can be converted to equidistant time series using the Analyzer software using the *Calculate - Spline Interpolation* function.

It is not recommended or not even useful to perform a spectral analysis directly from an EKG or from a pulsatile arterial blood pressure waveform. Instead, spectral analysis should be performed from equidistant HR (or RR-interval) time series, or from equidistant systolic, mean, or diastolic blood pressure time series.

7.1.2 Recording Duration

The duration of the time series for spectral analysis should be 5-10 times the wavelength of the slowest frequency component of interest (f_{min}). The minimum duration of the recording can then be calcuated as $5/f_{min}$.

Applying this equation will ensure that at least 5 full oscillatory cycles of the lowest frequency of interest are included in the time series. For example, if the lowest frequency range of interest is the very low frequency range (VLF, 0.02-0.2 Hz), then the duration of the time series should be at least 5/0.02 Hz = 250 s or 4 min and 10 sec. For the *optimal frequency ranges* in mice please refer to Baudrie et al. [1].

7.1.3 Number of data values for spectral analysis

The number of data values for spectral analysis depends on the sampling rate (SR) of the time series and on the lowest frequency component of interest (f_{min}). To determine the number of data values for spectral analysis apply the equation: 5*SR/f_{min}. If the FFT technique is used, the next highest power of 2 must be used, because the FFT algorithm requires a number of data values equal to a power of 2 (e.g., 512, 1024, 2048, 4096, ...). The autoregressive approach does not require a power of 2 for the number of data values.

For example, to detect the very low frequency range (VLF, 0.02-0.2 Hz) in a time series that was recorded at 15 Hz, the number of data values for spectral analysis is 5*15/0.02 Hz = 3,750. If the FFT technique is used, 4096 data values (next highest power of 2) must be used.

7.2 Spectral Analysis using the Autoregressive Technique

1. Load the time series from which you want to perform spectral analysis in Analyzer (*File - Load Data*).

2. If your file is an EKG or a pulsatile blood pressure waveform, derive HR or systolic, mean, or diastolic blood pressure using *Calculate - Derived channels by Threshold.*

3. Activate the channels, from which spectral anaylsis should be performed using the middle mouse button or the context menu (right mouse button). The red "P" (passive) should switch to a blue "A" (for active).

4. Select the time period from which spectral analysis should be performed by dragging a red rectangle around the time series (left mouse button). The spectrum is only calculated from the selected time period.

5. Use *Calculate - Autoregressive Spectral Analysis.* In the Autoregressive Spectral Analysis windows select the number of poles (= number of parameters for the autoregressive model). This number can be entered manually ("Manual, No of Poles:") or it can be determined automatically by one of three criterions (AIC, AICC, or BIC). The AIC criterion often results in a large number of parameters, the BIC typically uses less parameters, and the AICC is in between. Also indicate if a linear regression line should be subtracted from the time series prior to spectral analysis to reduce the so-called "DC-component" of the spectrum. Then click on "OK".

6. Often the results of the automatic determination of the number of poles is not satisfactory. In this case the number of poles should be entered manually.

7. The result window contains a table with several parameters for each frequency component of the spectrum identified by the autoregressive model. (1) the center frequency (Freq.) of the frequency component, (2) the width of the peak of the frequency component (a measure of how constant the frequency of the frequency component is), (3) the absolute (abs_Pow) and (4) relative (rel_Pow) spectral powers. The total power is identical with the variance of the data values of the input time series.

8. The table of the results is automatically copied into the clipboard of MS-Windows.

9. Paste the results from the clipboard in a different software (e.g., MS-Excel) for later statistical analysis.

10. The spectrum will also be added to Analyzer as a new channel.

7.3 Spectral Analysis using the FFT Technique

1. It is assumed that a number of artifact free data files with equidistant sampled time series are available. These files can consist of HR data, systolic, diastolic, or mean blood pressure data or any other signals.

2. However, the input files need to be in the TSA (`>>> *.tsa <<<`) file format. If the files are in ASCII (or text) format they need to be converted to TSA by opening them in Analyzer and then saving them as TSA files.

3. Spectral analysis by the FFT technique is done using the Batch Processor software. Add all files that shall be analyzed in the Batch Processor (*File - Add Files*). **Again, make sure these files are in the TSA format.**

4. Select either *Calculate - Power Spectra (single segment)* or *Calculate - Power Spectra (overlapping segments)*. A new window pops up. There, the "Number of data values for FFT" must be entered. See the section on "Number of data values for spectral analysis" to determine this number. If the "single segment" menu option was selected, one spectrum is calculated based on this number of data points. However, if the "overlapping segments" menu option was used, multiple spectra will be calculated on this number of data points for consecutive segments of the entire recording that overlap by as much percent as indicated by the "% overlap of segments:" option in the "Overlapping Segments PSD window". The individual spectra calculated from the overlapping segments are averaged and the averaged spectrum is written as the output file. Finally, you need to select an output directory by clicking on the "Browse for Output Directory" button. It is highly recommended to use a new directory (or folder) for the output files. **The output files will have the same file names as the input files (the ones that were initally added to the Batch Processor main window). Thus, if the directory (or folder) is**

selected that contains the input files, then the input files will be overwritten (without notice!) by the newly calculated spectra!.

5. Once the new files for the power spectra are generated, you can use the Batch Processor to analyze these spectra and calculate the areas under the curve for the VLF, LF, and HF frequency bands. Select *File - New* to empty the main window of the Batch Processor and clear all files. Then add the files containing the power spectra via *File - Add Files*.

6. Select *Calculate - Analyze Power Spectra (AUC)*.

7. In the Analyze Power Spectra window select the frequency boundaries for the very low frequency (VLF) and low frequency (LF) ranges. Then decide if you want to use fixed frequency boundaries for the high frequency range (i.e., same frequency range for all files added to Batch Processor) or if you want to use different frequency ranges for the different files for the high frequency (HF) range. It is best to use individual frequency ranges for the individual files because theoretically, the HF range should be centered around the respiration frequency, which can vary from file to file. However, if many files are to be analyzed, it is often not practical to identify the respiration rate for each file individually. In this case a fixed frequency range (same for all files) may be selected.

8. For fixed frequency range enter the center frequency and the width of the frequency band (enter a value for half of the bandwidth because the software used the width value in both directions from the center frequency).

9. For individual frequency ranges, the center frequencies must be provided as Parameter 1 in the main window of the Batch Processor. The width value has the same meaning as for fixed frequency ranges.

10. Finally, enter the "End frequency for total power:". The software calculates the area under the entire power spectrum from 0 Hz up to this end frequency. This total power is then used to calculate relative specral powers.

11. Click on "OK".

12. You are requested to enter an output file for the results of the spectral analysis. It is recommended to use the file extension `>>> *.txt <<<`, because *.txt files are recognized by other software (e.g., MS-Excel or WinStat) as tab-delimited text (or ASCII) files.

13. The output file contains a table with the results of the spectral analysis. For each channel of the input files (e.g., HR, systolic BP, etc.) seven values are provided: (1) absolute VLF power (chX_VLFa), (2) relative VLF power (chX_VLFr), (3) absolute LF power (chX_LFa), (4) relative LF power (chX_LFr), (5) absolute HF power (chX_HFa), (6) relative HF power (chX_HFr), (7) total spectral power (chX_total).

14. The output files can be opened in MS-Excel or WinStat.

Chapter 8

Transfer Function Analysis

8.1 General Remarks

The transfer function between two signals describes the relationship between the two signals in the "frequency domain". **Specifically, it described how changes in one signal (the input function) translate into corresponding changes in the other signal (the output function).** Three parameters are typically derived from the transfer function: (1) squared coherence ($\gamma^2(q)$); (2) gain ($|H(q)|$), and (3) phase ($\phi(q)$).

The squared coherence function $\gamma^2(f)$, the phase $\phi(f)$, and the gain $|H(f)|$ of the transfer function can be calculated based on the autospectral density functions of the input ($S_{xx}(q)$) and output ($S_{yy}(q)$) functions, the cross-spectrum of the input and output functions ($S_{xy}(q)$), and the imaginary ($H_I(q)$) and real ($H_R(q)$) parts of the complex transfer function $H(q)$ according to:

$$\gamma^2(q) = \frac{|S_{xy}(q)|^2}{S_{xx}(q)S_{yy}(q)} \qquad |H(q)| = \left|\frac{S_{xy}(q)}{S_{xx}(q)}\right| \qquad \phi(q) = tan^{-1}\frac{H_I(q)}{H_R(q)}$$

8.1.1 Squared Coherence

The squared coherence $\gamma^2(q)$ function is unitless and its values can range from 0 to 1. It describes the degree to which the input and output functions are correlated with each other. A value of 0 indicates no correlation, while a value of 1 is consistent with perfect correlation. A value of >0.5 is generally accepted as significant correlation because a squared coherence value of 0.5 suggests a relation between two signals based on 50% shared variance [14].

8.1.2 Transfer Function Gain

The gain of the transfer function $|H(q)|$ is a measure of how much the output function changes for a given change in the input function. Thus, the unit of the gain is the unit of the output function divided by the unit of the input function.

The gain of the transfer function can also be normalized, so that the normalized gain is unitless. To achieve normalization, the absolute gain is multiplied with the mean value of the input function, divided by the mean value of the output function.

8.1.3 Transfer Function Phase

The phase of the transfer function $\phi(q)$ is a measure of the time delay between a change in the input function and the resulting change in the output function. The unit of the phase is radian. A value of 0 rad indicates that the output changes immediately (without delay) in response to the input function. A phase value of $\phi(q) = 2\pi$ rad corresponds to the wavelength of an oscillation at the respective frequency q or $1/q$ seconds. Conversion of phase values in radian to time values in seconds can be done according to:

$$time\ (s)\ =\ \frac{\phi(q)}{2\pi\ q}$$

8.2 Applications of the Transfer Function

8.2.1 Baroreflex Sensitivity

The gain of the transfer function between blood pressure (input function) and heart rate or pulse interval (output function) has been used as measure of baroreflex sensitivity. The rationale for this application is that the transfer function gain is defined as "how much does the output function (i.e., heart rate or pulse interval) change for a given change in the input function (i.e., blood pressure). This is exactly the definition of the baroreflex sensitivity. The gain of the transfer function in the LF frequency band has been considered as being more related to sympathetic baroreflex modulation of heart rate, whereas the gain of the transfer function in the HF frequency band has been suggested to reflect more the parasympathetic portion of the baroreflex.

8.2.2 Autoregulation of Blood Flow

The gain of the transfer function between blood pressure as input function and local vascular blood flow (e.g., renal blood flow) as output function has been used as a measure of autoregulation of blood flow. If the gain of this transfer function is 0, perfect autoregulation is assumed, because a change in blood pressure does not elicit any change in local blood flow. On the other hand, a high gain is interpreted as less potent autoregulation. A high gain in combination with a phase of 0 indicates passive vascular responses because flow changes immediately to changes in pressure.

8.3 Transfer Function Analysis using the Batch Processor

8.3.1 Calculate Transfer Functions

1. Add all files to be analyzed into the Batch Processor (*File - Add Files*). Each file need to contain the input and output function of the transfer function in separate channels. The files must be in the `>>> *.tsa <<<` format.

2. Select either *Calculate - Transfer Function (single segment)* or *Calculate - Transfer Function (overlapping segments)* from the Batch Processor menu.

3. Select the channels for the input and output functions of the transfer function ("Input Channel:" and "Output Channel:").

4. The transfer function is calculated using the fast Fourier transform (FFT) algorithm. Thus, the "Number of data values for FFT" must be entered. See the section on "Number of data values for spectral analysis" to learn how to determine this number. If the "single segment" menu option was selected, one transfer function is calculated based on this number of data points. However, if the "overlapping segments" menu option was used, multiple transfer functions will be calculated on this number of data points for consecutive segments of the entire recording that overlap by as much percent as indicated by the "% overlap of segments:" option in the "Overlapping Segments Transfer Function" window. The individual transfer functions calculated from the overlapping segments are averaged and the averaged transfer function is written as the output file.

5. Next, enter a "Smoothing factor:". This smoothing factor is used to smooth the input and output functions prior to transfer function analysis. Without this smoothing the squared coherence would always be 1. The greater the smoothing factor, the smoother the transfer function and the lower the calculated squared coherence values. A smoothing factor of $\sim$0.25 % of the "Number of data values for FFT" can be a good starting point in selecting the smoothing factor. It is highly recommended to visually inspect the transfer functions to check if the selected smoothing factor results in "reasonable transfer functions".

6. Finally, you need to select an output directory by clicking on the "Browse for Output Directory" button. It is highly recommended to use a new directory (or folder) for the output files. **The output files will have the same file names as the input files (the ones that were initially added to the Batch Processor main window). Thus, if the directory (or folder) is selected that contains the input files, then the input files will be overwritten (without notice!) by the newly calculated spectra!**.

7. After clicking on "OK" all the transfer functions will be calculated automatically and saved in the directory selected using the "Browse for Output Directory" push button.

8. The output files of the transfer function analysis contain 5 channels. From top to bottom: (1) frequency in Hz, (2) Square Coherence, (3) absolute gain of the transfer function, (4) normalized gain of the transfer function, (5) phase of the transfer function in radian.

9. The output files of the transfer function can be analyzed using the Analyzer software or more automatically using the Batch Processor.

8.3.2 Analyze Transfer Functions in Batch Processor

1. Clear the Main Window of the Batch Processor by using *File - New*.

2. Add the files for the transfer functions to be analyzed (*File - Add Files*).

3. Select *Calculate - Analyze Transfer Function* from the Batch Processor menu.

4. Enter the frequency boundaries for the very low frequency (VLF) and low frequency (LF) ranges.

5. Then decide if you want to use fixed frequency boundaries for the high frequency range (i.e., same frequency range for all files added to Batch Processor) or if you want to use different frequency ranges for the different files for the high frequency (HF) range. It is best to use individual frequency ranges for the individual files because theoretically, the HF range should be centered around the respiration frequency, which can vary from file to file. However, if many files are to be analyzed, it is often not practical to identify the respiration rate for each file individually. In this case a fixed frequency range (same for all files) may be selected.

6. For fixed frequency range enter the center frequency and the width of the frequency band (enter a value for half of the bandwidth because the software used the width value in both directions from the center frequency).

7. For individual frequency ranges, the center frequencies must be provided as Parameter 1 in the main window of the Batch Processor for each transfer function file. The width value has the same meaning as for fixed frequency ranges.

8. After clicking on "OK", a new windows pops up that allows to select an output file. The output file will be a "tab-delimited" text (or ASCII) file. Thus, the file extension *.txt is best, because MS-Excel, WinStat, and other software recognize this file extension as tab-delimited text files.

9. The output file contains: (1) the center frequency of the frequency band in Hz (freq), (2) the squared coherence (coh), (3) the absolute gain of the transfer function (gain), (4) the normalized gain of the transfer function (n_gain), and (5) the phase of the transfer function in radian for each of the three frequency ranges (VLF, LF, and HF).

Bibliography

[1] Véronique Baudrie, Dominique Laude, and Jean-Luc Elghozi. Optimal frequency ranges for extracting information on cardiovascular autonomic control from the blood pressure and pulse interval spectrograms in mice. *Am. J. Physiol. Regul. Integr. Comp. Physiol.*, 292:R904–R912, 2006.

[2] J. Baulmann, R. Homsi, S. Uen, R. Dusing, R. Fimmers, H. Vetter, and T. Mengden. Pulse wave velocity is increased in patients with transient myocardial ischemia. *J Hypertens*, 24(10):2085–90, 2006.

[3] M Baumert, V Baier, S Truebner, A Schirdewan, and A Voss. Short- and long-term joint symbolic dynamics of heart rate and blood pressure in dilated cardiomyopathy. *IEEE Trans Biomed Eng.*, 52(12):2112–2115, 2005.

[4] G Bertinieri, M Di Rienzo, A Cavallazzi, AU Ferrari, A Pedotti, and G Mancia. A new approach to analysis of the arterial baroreflex. *J Hypertens*, 3(Suppl. 3):S79–S81, 1985.

[5] M. Di Rienzo, G. Bertinieri, G. Mancia, and A. Pedotti. A new method for evaluating the baroreflex role by a joint pattern analysis of pulse interval and systolic blood pressure series. *Med. Biol. Eng. & Computing*, suppl. part1:313–314, 1985.

[6] D Laude, V Baudrie, and JL Elghozi. Applicability of recent methods used to estimate spontaneous baroreflex sensitivity to resting mice. *Am J Physiol Regul Integr Comp Physiol*, 294(1):R142–R150, 2008.

[7] G. F. Mitchell, N. Wang, J. N. Palmisano, M. G. Larson, N. M. Hamburg, J. A. Vita, D. Levy, E. J. Benjamin, and R. S. Vasan. Hemodynamic correlates of blood pressure across the adult age spectrum: noninvasive evaluation in the framingham heart study. *Circulation*, 122(14):1379–1386, 2010.

[8] Task Force of the European Society of Cardiology, the North American Society of Pacing, and Electrophysiology. Heart rate variability: standards of measurement, physiological interpretation and clinical use. *Circulation*, 93:1043–1065, 1996.

[9] CK Peng, SV Buldyrev, S Havlin, M Simons, HE Stanley, and AL Goldberger. Mosaic organization of DNA nucleotides. *Phys. Rev. E*, 49(2):1685–1689, 1994.

[10] A. Qasem and A. Avolio. Determination of aortic pulse wave velocity from waveform decomposition of the central aortic pressure pulse. *Hypertension*, 51(2):188–195, 2008.

[11] HM Stauss. Identification of blood pressure control mechanisms by power spectral analysis. *Clin and Exper Pharmacol and Physiol*, 34:362–368, 2007.

[12] HM Stauss. Physiologic mechanisms of heart rate variability. *Rev Bras Hipertens*, 14:8–15, 2007.

[13] M. Sugimachi, T. Kawada, T. Shishido, N. Matsumoto, Jr. Alexander, J., and K. Sunagawa. Estimation of arterial mechanical properties from aortic and tonometric arterial pressure waveforms. *Methods Inf Med*, 36(4-5):250–253, 1997.

[14] JA Taylor, DL Carr, CW Myers, and DL Eckberg. Mechanisms underlying very-lowfrequency rr-interval oscillations in humans. *Circulation*, 98:547–555, 1998.