CHAPTER 1

THE PHYSIO TOOLBOX FOR LOCALIZED PHYSIOLOGICAL NOISE CORRECTION

1.1 Physiological Noise in fMRI Data

Physiological Noise is a major limitation for BOLD sensitivity. Originally, it was defined in contrast to thermal noise as any "signal-proportional" fluctuation (Krüger and Glover, 2001). In our framework, physiological noise refers to true magnetization changes in the brain, which stem from physiological, non-BOLD sources. Specifically two main generators of physiological noise have been identified besides subject bulk motion: the cardiac and the respiratory cycle (Figure 1.1).

For the cardiac cycle, the noise generating mechanism arises from the antagonistic flow dynamics during systole and diastole within a

heartbeat. During systole, the pulse pressure wave of the blood reaches the brain, leading to an increase in blood volume and expansion of the transporting arteries (Soellinger, 2008). Since total cranial volume is invariant due to the skull boundaries, this brain expansion has to be compensated by an outflow of cerebrospinal fluid (CSF) via the fourth ventricle into the spinal canal. Conversely, during diastole, blood volume is reduced, leading to a slight shrinkage of the brain and a back-flow of the CSF into the ventricles and subarachnoid space from the spinal canal. Thus, three effects on magnetization follow from the cardiac cycle: pulsatile flow of the CSF through ventricles and aqueduct, alterations in brain voxel composition due to local blood volume changes, and tissue displacement at brain/CSF and brain/vessel boundaries. The tissue displacement is maximal in inferior brain regions, e.g. amounting to about 1-2 mm shift in head-foot direction for the brainstem (Soellinger, 2008).

The respiratory cycle, as second physiological noise generator, injects image fluctuations via encoding field changes on the one hand, and changes in tissue oxygenation on the other hand (Windischberger et al., 2002). Changes in the encoding magnetic fields are typically of low spatial order in the brain, since they originate from distant magnetization changes through tissue displacement surrounding the lungs. Consequently, if uncorrected during image reconstruction, respiratory field fluctuations induce rather global effects in the image, such as shifts or scaling for EPI acquisitions. The changes in oxygenation level, on the other hand, lead to more local effects, since they alter the voxel composition and susceptibility distribution through the relative oxygen content in the tissue. Note that this oxygen fluctuation is independent of the energy consumption effects investigated by BOLD fMRI, varying purely with the phase of the respiratory cycle. Furthermore, cardiac and respiratory noise generators also interact by the respiratory-sinus arrhythmia. Lung expansion during inhalation leads to an increase in heart rate, while heart rate decreases during exhalation.

In summary, physiological noise is a major source of signal variability in voxel time series that accounts for up to 60 % of the observed fluctuations and occurs distributed at many sites all over the brain, including OFC, brainstem, and cortex adjacent to ventricles and subarachnoid space. Consequently, suitable correction methods for physiological noise are recommended for both task-based and resting-state fMRI studies, increasing the sensitivity for effects of interest and lowering the risk of spurious physiological correlations in functional connectivity analyses, respectively (Birn, 2012; Birn et al., 2006; Harvey et al., 2008; Hutton et al., 2011).

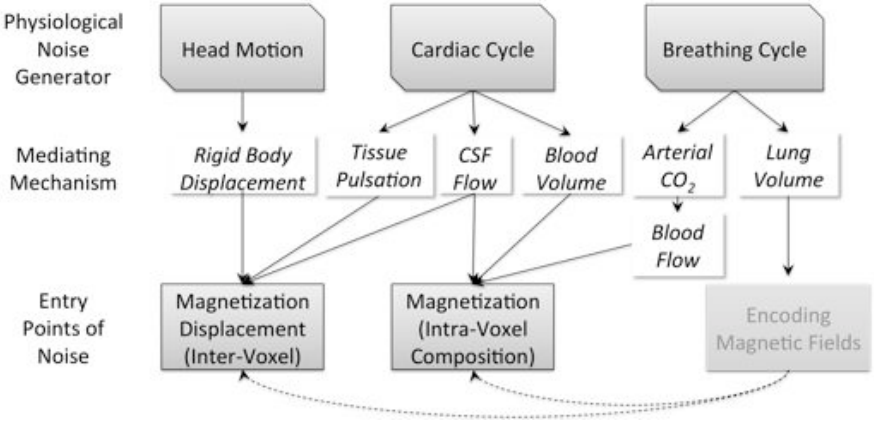

FIGURE 1.1 *Mechanisms of physiological noise generation in the brain.* ***(Top)*** *Source of physiological noise.* ***(Center)*** *Specific localization or property where physiological noise manifests.* ***(Bottom)*** *Entry point of noise into the MR image encoding (cf. Chapter 2, Figure 2.5). The schematic depicts an image-based perspective where uncorrected field fluctuations lead to changes in the estimated, i.e. apparent magnetization.*

1.2 Image-based Physiological Noise Correction

To identify and correct for physiological noise in image time series, exploratory techniques, such as independent component analysis (ICA(Beckmann and Smith, 2004; Perlbarg et al., 2007), as well as voxel-wise noise modeling have been suggested (Glover et al., 2000). Both approaches focus on isolating the typical periodicity of the cardiac and breathing cycle (Figure 1.2).

ICA methods, since they do not employ concurrent measures of physiology, have to rely on prior assumptions about the structure of physiological noise. For example, spatial ICA (sICA, (Thomas et al., 2002) classifies components post-hoc as physiological noise, if their representative time series is dominated by the characteristic breathing and heart rate, i.e. 0.2-0.3 Hz and about 1 Hz, respectively. An alternative approach, termed CORSICA (CORrection of Structured noise using spatial Independent Component Analysis(Perlbarg et al., 2007), instead utilizes the spatial characteristics of physiological noise to identify components. Specifically, the larger blood vessels (basilar artery, circle of Willis) are major sites of pulsatile displacement, while CSF reservoirs (ventricles, subarachnoid space) experience flow effects, and cortex/CSF boundaries are sensitive to global displacements.

Voxel-wise physiological noise modeling, on the other hand, relies on peripheral physiological time series, acquired from e.g. pneumatic breathing belts, electrocardiograms (ECG) or pulse oximetry units (attached to the finger/wrist). Different ways to utilize the information in this peripheral data have been proposed, most prevalently its frequency content via RETROICOR (RETROspective Image CORrection (Glover et al., 2000)). This approach is similar to sICA in that it focuses on the periodicity of the physiological noise as captured by the cardiac and respiratory phase (Figure 1.3). Herein, the cardiac phase at time t is expressed

as the time passed since the last heartbeat relative to the duration of the current cycle, i.e.

$$\varphi_{card}(t) = 2\pi \frac{t - t_1}{t_2 - t_1} \tag{1.1}$$

with t_1 being the time of the last heartbeat, and t_2 the time of the next one (Figure 1.3 A).

The respiratory phase, φ_{resp}, on the other hand, is computed using an equalized-histogram transfer function, accounting for the differing breathing amplitudes in each cycle:

$$\varphi_{resp}(t) = \pm\pi \frac{\int_{R_{min}}^{R(t)} H(R)\mathrm{d}R}{\int_{R_{min}}^{R_{max}} H(R)\mathrm{d}R} \tag{1.2}$$

Herein, $R(t)$ is the amplitude of the respiratory signal and H is the histogram capturing the frequency of each breathing amplitude over the course of the time series (Figure 1.3 BC). The use of the histogram-equalization ensures maximum sensitivity of the phase for the most frequently occurring amplitudes. Furthermore, a full breathing cycle is attained only if inhalation and exhalation are both complete. The sign of φ_{resp} is determined by the temporal derivative of R, i.e. positive for inhalation ($dR/dt > 0$), and negative for exhalation ($dR/dt < 0$).

In RETROICOR, the periodic physiological noise time series is then modelled as a Fourier expansion of both cardiac and respiratory phase.

$$x_{card,resp}(t) = \sum_{m=1}^{N_m} A_m \cdot \cos(m\varphi_{card,resp}) + B_m \cdot \sin\left(m\varphi_{card,resp}\right), \tag{1.3}$$

where N_m is the order of the expansion, and A_m, B_m are the Fourier coefficients that have to be estimated for each voxel time series individually. Considering higher harmonics ($N_m > 1$) of the estimated physiological frequencies is a consequence of the low sampling rate of fMRI, typically 0.3-0.5 Hz (TR 2-3 seconds). Thus, aliase.ing occurs such that the under-sampled breathing and cardiac signals (0.25 Hz and 1 Hz) fold back into the spectrum of the sampled time series at different frequencies. To account for interaction effects between respiratory and cardiac cycle, e.g. via the respiratory-sinus arrhythmia, extensions to RETROICOR incorporating multiplicative Fourier terms have been proposed (Brooks et al., 2008; Harvey et al., 2008).

$$\begin{aligned} x_{cardXresp}(t) = \sum_{m=1}^{N_m} & A_m \\ & \cdot \cos(m\varphi_{card}) \cdot \cos\left(m\varphi_{resp}\right) + B_m \\ & \cdot \sin(m\varphi_{card}) \cdot \cos\left(m\varphi_{resp}\right) + C_m \\ & \cdot \cos(m\varphi_{card}) \cdot \sin\left(m\varphi_{resp}\right) + D_m \\ & \cdot \sin(m\varphi_{card}) \cdot \sin\left(m\varphi_{resp}\right). \end{aligned} \tag{1.4}$$

Beyond frequency content, other aspects of the physiological signal have been considered as independent noise sources, such as heart rate variability or respiratory volume per time (Birn et al., 2006; Chang et al., 2009). This follows the rationale that the arterial CO_2 level governs vasodilation and –constriction, thus altering blood flow (Shmueli et al., 2007), and is in itself modulated by heart rate and respiratory volume. Modeling their impact on the image voxel time series follows an approach analogous to BOLD modeling in the GLM (Chapter 2.2). Instead of a hemodynamic response function, which translates hypothesized neural activation into BOLD changes, cardiac and respiratory response functions were proposed to map heart rate and respiratory volume per time onto physiological noise of the fMRI time series. Assuming a linear, time-invariant (LTI) system, the noise model time series can then be retrieved as a convolution between the respective response

function and the physiological input time series. The concrete form of these response functions was determined from experimental calibration data.

Specifically, for the respiratory response function $RRF(t)$, the LTI was probed by single impulses, i.e. observing the BOLD response to single deep breaths (Birn et al., 2006). As functional form for the RRF, the difference of two gamma variate functions was proposed, which are typically used to describe bolus experiment dynamics, and the fit to the experimental data yielded:

$$RRF(t) = 0.6t^{2.1}e^{-\frac{t}{1.6}} - 0.0023t^{3.54}e^{-\frac{t}{4.25}} \quad (1.5)$$

This function is convolved with the estimated respiratory volume per time (RVT), i.e. the local integral of breathing belt amplitude, to yield the physiological noise time series.

To determine the cardiac response function $CRF(t)$, a free-form Gaussian-process deconvolution of BOLD data was performed with respect to the estimated current heart rate (Chang et al., 2009). Post-hoc, the resulting CRF was fitted to a combination of a gamma variate and Gaussian function, yielding

$$CRF(t) = 0.6t^{2.7}e^{-\frac{t}{1.6}} - \frac{1}{\sqrt{18\pi}}e^{-\frac{(t-12)^2}{4.5}} \quad (1.6)$$

As for the RRF, the resulting physiological noise time series can be retrieved by convolving the CRF with the running estimate of the heart rate (typically via a sliding-window average over 6 seconds).

Finally, the overall impact of physiological noise on the voxel time series is then modelled as a linear superposition of all the above-mentioned different Fourier and/or convolution terms, i.e. assuming that there is no further interaction between the different aspects of physiology. For actual noise correction, the weighting parameters (e.g. A_m, B_m) of the different noise time series have to be fitted to the voxel time series and projected out of the data.

Conveniently, for the mass-univariate analysis approach prevalent in fMRI (Chapter 2.2), this amounts merely to an inclusion of these different noise mode time series into the design matrix of the GLM (Josephs et al., 1997). Thus, physiological noise time series, sampled at the volume acquisition times, become confound regressors, analogous to movement parameters or session means of the BOLD time series. Finally, this inclusion into the fMRI analysis enables to evaluate the significance of physiological noise removal via F-tests on the estimated weighting parameters (Chapter 2.3). Furthermore, since the F-test reports the extra-sum-of-squares explained by the physiological noise regressors, this also provides a quantitative measure of the efficacy of physiological noise removal.

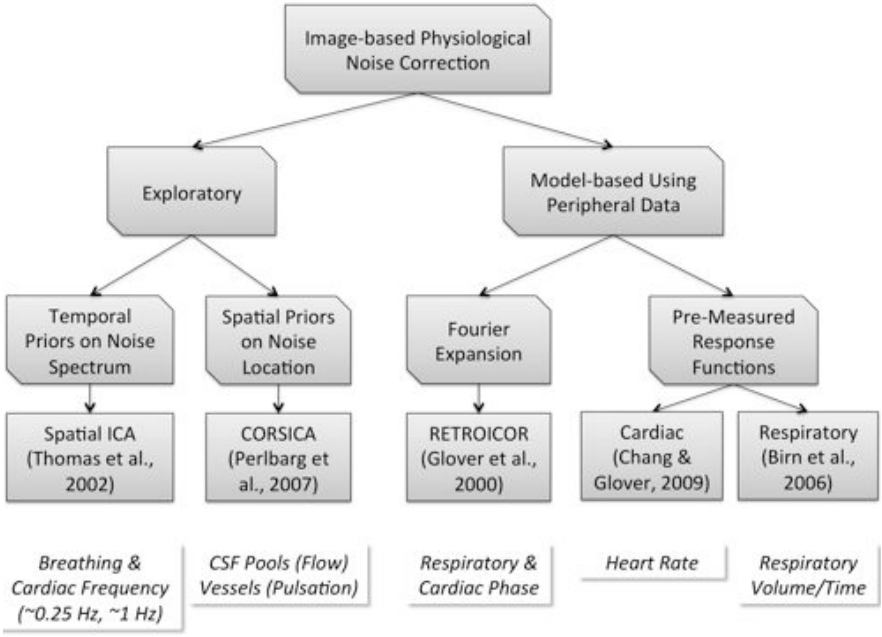

FIGURE 1.2 *Different image-based physiological noise correction methods. Individual methods are given in boxes with reference of first occurrence. White boxes indicate specific input data or priors of each method.*

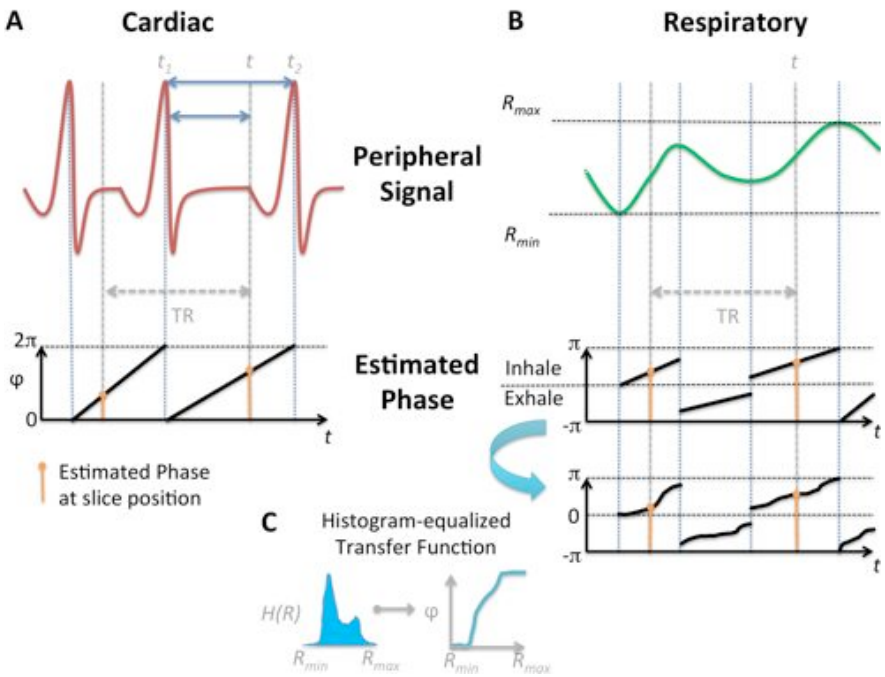

FIGURE 1.3 *Cardiac and Respiratory Phase Estimation in RETROICOR.* ***(A)*** *For the cardiac phase, the R-peaks of the ECG-wave are detected and the phase at slice acquisition time is estimated by the relative position within the current heartbeat duration.* ***(B)*** *For the respiratory phase, the amplitude of the breathing signal R is transformed into a positive phase (inhalation) or negative phase (exhalation). The peak absolute phase of π is only reached for maximum amplitude over the whole time-course.* ***(C)*** *The mapping from respiratory volume to respiratory phase is non-linear, realized by a histogram-equalized transfer function that allocates sensitivity for the most frequently occurring respiration amplitudes.*

1.3 Implementation of Noise Correction in the PhysIO Toolbox

1.3.1 Overview

The physIO Toolbox was developed within this work to provide state-of-the-art model-based physiological noise correction for fMRI using peripheral measurements. The toolbox models voxel-wise physiological noise components via RETROICOR (Glover et al., 2000; Harvey et al., 2008) as well as cardiac and respiratory response functions (Birn et al., 2006; Chang et al., 2009). A particular focus of the implementation of this software was robustness and ease-of-use for large-scale studies. Hence, considerable effort went into standardizing and automatizing the processing stream for various kinds of MR systems and peripheral measurement devices. The physIO toolbox is aimed at both researchers and clinicians, and provides seamless integration with existing packages used in the neuroscientific community, in particular Statistical Parametric Mapping (SPM, www.fil.ion.ucl.ac.uk/spm/). Furthermore, it was implemented platform-independent in Matlab and as part of the software suite TAPAS (TNU Algorithms for Psychiatry-Advancing Science, http://www.translationalneuromodeling.org/tapas/) of the Translational Neuromodeling Unit (TNU), that offers long-term support and development.

The workflow of the toolbox consists of five major modules that are explained in more detail in the following sections (Figure 1.4): (1) read-in of (vendor-specific) physiological log-files, (2) pre-processing of noisy peripheral physiological data, (3) physiological noise modeling using RETROICOR and cardiac/respiratory response functions, (4) physiological noise correction by providing confound regressors for the mass-univariate GLM analysis and (5) assessment of noise correction efficacy using automated F-contrast generation in SPM.

The overall workflow is executed by running the main function `tapas_physio_main_create_regressors`. This function takes one input argument, the `physIO`-structure, which holds all parameter and processing choices for the workflow. `PhysIO` is created by the constructor-function `tapas_physio_new`.

The generation and modification of the `physIO`-structure, as well as the execution of the toolbox functions, can be inspected for various example datasets in the `examples/`-folder provided with the toolbox. For example, the standard use case using ECG and a pneumatic breathing belt is illustrated in `examples/Philips/ECG3T/main_ECG3T.m`

In the following, we will dissect the workflow of the physIO toolbox along by a commented step-wise walk-through of `tapas_physio_main_create_regressors`.

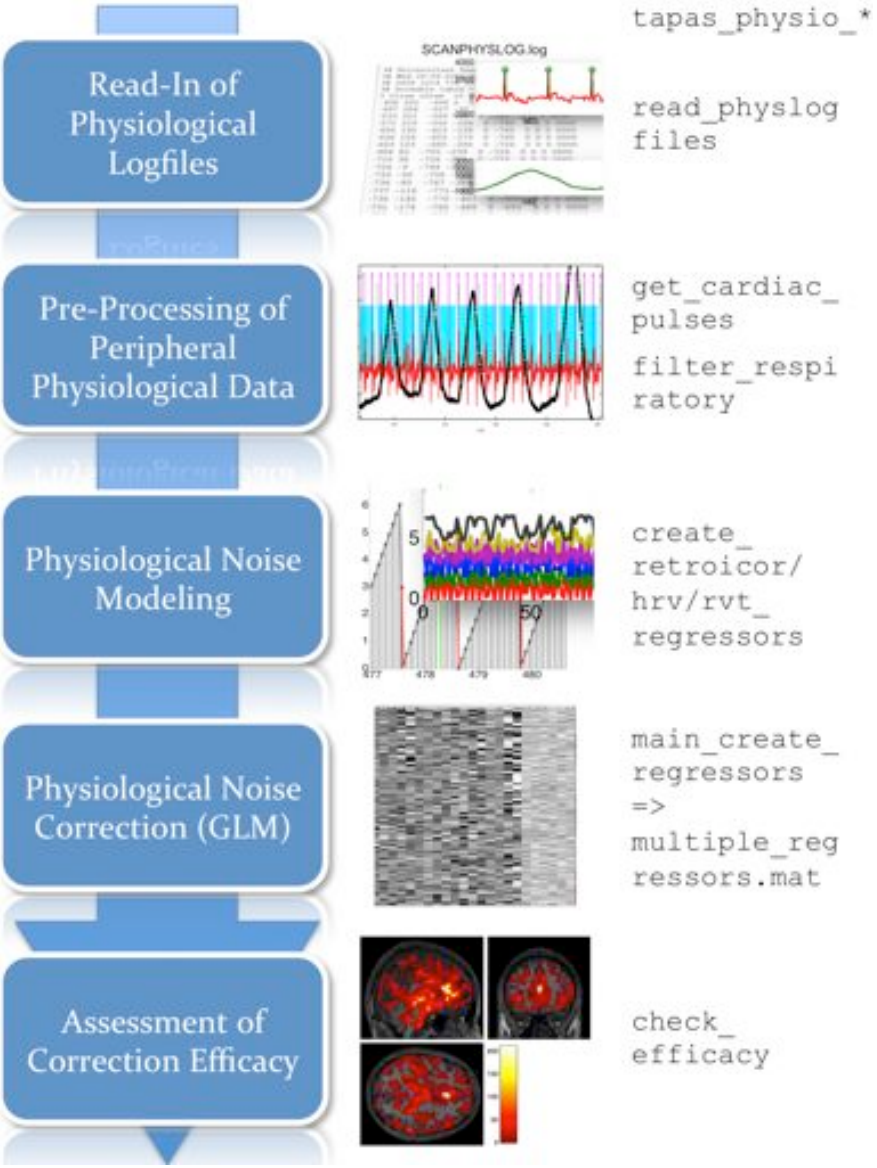

FIGURE 1.4 *The Workflow of the PhysIO Toolbox.* ***(Left)*** *Modules of the PhysIO Toolbox.* ***(Center)*** *Typical graphical output in respective modules.* ***(Right)*** *Most relevant functions within module (function name prefixed with* `tapas_physio_`*).*

1.3.2 Read-in of Physiological Log files

Currently, physiological log-files from Philips and General Electric (GE) MR systems are natively supported by the physIO Toolbox. For both vendors, measurements of electrocardiograms (ECG), pulse plethismograph/oximetry units (PPU) and pneumatic respiration belts are accepted inputs. This first module of the physIO Toolbox reads in the physiological log files and synchronizes the scan timing to the physiological measurements.

To specify the input files for read-in, the physIO-structure has to be initialized first by a call of the constructor:

```
physIO = tapas_physio_new();
```

Then, the following parameters in the `physIO.log_files` and `physio.thresh` sub-structures have to be set:

```
thresh.cardiac.modality   'PPU' or 'ECG'
log_files.vendor          'Philips' or 'GE'
log_files.cardiac         'SCANPHYSLOG*.log' or
                          'ECGData_epiRT*'
log_files.respiration     'SCANPHYSLOG*.log' or
                          'RespData_epiRT*'
log_files.
sampling_interval         in seconds, e.g. 2e-3,
                          40e-3
```

Note that for Philips, both cardiac and respiratory information are saved in the same log file, `SCANPHYSLOG_<timestamp>.log`., and the sampling interval of the physiological log file is fixed (2 ms, 500 Hz sampling rate).

In the main function of the toolbox, `tapas_physio_main_create_regressors`, this input information is used to generate the raw vectors of sampling time and physiological information (PPU or ECG time course, breathing belt amplitude signal) by calling

```
[ons_secs.c, ons_secs.r, ons_secs.t,
ons_secs.cpulse] = ...
    tapas_physio_read_physlogfiles(log_files,
    thresh.cardiac.modality);
```

where the output is saved in the `physIO.ons_secs` substructure ("onsets in seconds").

```
ons_secs.c          [nSamples, 1] vector of
                    cardiac signal time course
                    (ECG/PPU)
ons_secs.r          [nSamples, 1] vector of
                    respiratory amplitude signal
ons_secs.t          [nSamples,1] Vector of
                    sampling times (in seconds)
ons_secs.cpulse     [nBeats,1] vector of
                    heartbeat onsets (in
                    seconds) - if provided by the
                    log file
```

For other vendors, the physIO toolbox can be used as well, if `physIO.ons_secs` is created manually with the aforementioned fields. The `tapas_physio_main_create_regressors`-function has to be slightly adapted then by replacing the `tapas_physio_read_logfiles` with a custom-made read-in function.

The second important functionality of this module is the synchronization of the physiological data with the scan timing, which is crucial for the correction of the fMRI voxel time series later on. The nominal scan timing is defined in the `physIO.sqpar`-substructure ("sequence parameters"), holding the following parameters:

```
sqpar.Nslices           number of slice per
                        volume
sqpar.NslicesPerBeat    default: Nslices; for
```

```
                          triggered sequences:
                          number of slices per
                          heartbeat
sqpar.TR                  repetition time
                          (seconds)
sqpar.Ndummies            number of dummy volumes
                          before first scan
sqpar.Nscans              number of scans
                          (volumes) in session
sqpar.Nprep               number of preparation
                          scans; leave [] or set
                          to integer value
sqpar.TimeSliceToSlice    time between acquisition
                          of subsequent slices; if
                          [], TR/Nslices is
                          assumed, enter a
                          duration for non-
                          equidistant slice
                          spacing
sqpar.onset_slice         default: Nslices/2;
                          reference slice for
                          timing within a volume
```

From these inputs, the physIO toolbox generates the nominal scan timing of each slice and volume acquisition (`LOCS`, `VOLLOCS` given in sample indices) during the run via the function

```
[VOLLOCS, LOCS] = tapas_physio_create_nominal_
    scan_timing(ons_secs.t, sqpar);
```

with

```
VOLLOCS             [nPrep+nDummies+nScans,1]
                    vector of volume scan onsets
                    (in sample indices of
                    ons_secs.t)
LOCS                [nSlices*nVolLocs, 1] vector
                    of slice onsets (in sample
```

```
                    indices of ons_secs.t)
```

If `sqpar.Nprep` is empty (`[]`), the toolbox assumes that the physiological log files end exactly when the last slice of the last volume of the fMRI run has been acquired and counts slices and volumes from the end of the files. Conversely, if `Nprep` has an integer value ≥ 0, the toolbox ignores the start of the log file up to time `(Nprep+Ndummies)* TR`, and then creates slice and scans timestamps in a forward direction.

For Philips SCANPHYSLOG-files, given appropriate software keys, a more accurate way to determine the exact scan timing relative to the physiological time courses is available. Specifically, the gradient time course during the scan is also logged in the SCANPHYSLOG-file. The PhysIO toolbox provides functions to extract the onset of every slice and volume acquisition that is accurate up to 10 ms, i.e. on the order of one slice. To activate this functionality, the `physio.thresh.scan_timing` substructure, that is empty per default, has to be defined with the following fields

```
scan_timing.grad_   'x', 'y' or 'z'; specifies
direction           which physical gradient time
                    course shall be used
scan_timing.zero    e.g. 1500; amplitudes below
                    this value are not considered
                    for slice detection
scan_timing.slice   e.g. 1700; minimum peak
                    height of a slice-encoding
                    gradient (arbitrary units)
vol                 e.g. 1800; minimum peak
                    height of a preparation
                    gradient at the start of a
                    volume
vol_spacing         e.g. 20e-3 (in seconds);
                    instead of the vol amplitude
                    threshold, vol_spacing can be
```

```
                    specified as the time gap
                    between the last slice and
                    first slice of the next
                    volume
```

Then, the slice and volume acquisition time-stamps are retrieved by the following call:

```
[VOLLOCS, LOCS] =
    tapas_physio_create_scan_timing_from_
    gradients_philips( log_files,
    thresh.scan_timing, sqpar, verbose);
```

The values of the thresholds have to be determined for each fMRI sequence individually (but can be kept between different subjects and sessions). By setting `verbose` ≥ 2, the physIO toolbox provides informative plots about the gradient time course and the current thresholds, thus enabling their effective adjustment to the sequence, in order to detect all slice/volume acquisitions.

Finally, for the workflow realized in `tapas_physio_main_create_regressors`, the complete scan timing is converted into seconds and inserted into `physIO.ons_secs` via

```
[ons_secs.svolpulse, ons_secs.spulse,
ons_secs.spulse_per_vol, verbose] =
    tapas_physio_get_onsets_from_locs(
    ons_secs.t, VOLLOCS, LOCS, sqpar, verbose);
```

1.3.3 Pre-Processing of Peripheral Physiological Data

After read-in of the raw physiological data, both the respiratory and cardiac signal time courses have to be pre-processed to improve SNR for subsequent noise modeling.

Typically, the respiratory amplitude signal recovered from pneumatic breathing belts is rather robust, if the belt was properly attached to the subject. Hence, simple band-pass filtering and outlier removal suffice for respiratory pre-processing. In the physIO toolbox, a 2nd order Butterworth bandpass filter (0.1-5 Hz) is implemented that accounts for long-term drifts (e.g. through loosening of the belt) and high-frequency noise (micro-movement, digitization noise). Furthermore, extreme apparent breathing amplitudes, e.g. through subject movement, exceeding more than 3 amplitude standard deviations are removed as outliers. Both, filtering and outlier removal are performed via a call to

```
ons_secs.fr =
    tapas_physio_filter_respiratory(ons_secs.r,
    log_files.sampling_interval);
```

where `physIO.ons_secs.fr` holds the filtered respiratory signal.

The cardiac signal, on the other hand, requires more pre-processing for both ECG and PPU data, since the amplitude signal of these devices has to be transformed into the phase information of the cardiac cycle. Specifically, cardiac pulses, i.e. the R-wave peaks of the ECG signal or maxima of the pressure wave of the PPU have to be detected. However, especially at high field or with badly attached electrodes, the ECG signal acquired in the MR environment is typically very noisy due to magneto-hydrodynamic effects. Similarly, hand movement affects pulse oximetry units attached to the finger. Thus, the default prospective R-wave or plethysmograph peak detection of the MR system, which is used for cardiac triggering, frequently fails in this situation, and cardiac pulse time stamps in the log-files are incomplete.

To enable reliable cardiac pulse detection, the physIO toolbox pursues a two-step procedure for retrospective detection of the cardiac pulse onsets. The options for this pulse detection are set in `physio.thresh.cardiac`. In the first pass, pulses can be either

loaded from the prospective algorithm logged by the MR-system or re-estimated via autocorrelation with a representative time-course during a single heartbeat. For ECG-data at and below 3 Tesla, the prospective detection by the system is usually sufficient, and data can be adopted from the Philips SCANPHYSLOG-file by setting

```
thresh.cardiac.modality   'ECG'
thresh.cardiac.initial_
cpulse_select.method      'load_from_logfile'.
```

However, at high field (7 Tesla and above), it is recommended to determine the R-peaks of the ECG by auto-correlating it with a single representative QRS-wave for the subject. This is accomplished by setting

```
thresh.cardiac.modality   'ECG'
thresh.cardiac.initial_
cpulse_select.method       'manual'
thresh.cardiac.initial_
cpulse_select.file        <filenameECGWave>
```

and subsequently running

```
[ons_secs.cpulse, verbose] =
    tapas_physio_get_cardiac_pulses(
    ons_secs.t, ons_secs.c,
    thresh.cardiac.initial_cpulse_select,
    thresh.cardiac.modality, [], verbose);
```

The selected QRS-wave is stored in `<filenameECGWave>`, and repeated runs (e.g. for other sessions) of the cardiac pulse detection can afterwards be performed by setting

```
thresh.cardiac.modality   'ECG'
```

```
thresh.cardiac.initial_
cpulse_select.method       'load'
thresh.cardiac.initial_
cpulse_select.file        <filenameECGWave>.
```

For PPU data, the manual waveform selection can be omitted thanks to a self-calibrating technique for cardiac pulse estimation developed by Steffen Bollmann (Children's Hospital Zurich). This method has been optimized for very noisy peripheral data in a patient population (children with ADHD). Specifically, expected heart rate and the representative time-course for autocorrelation are estimated iteratively from the data. To use this automatic first pass pulse detection for PPU data, the only required setting in the `thresh.cardiac` structure is

```
thresh.cardiac.modality   'OXY',
```

followed by the above call to

```
[ons_secs.cpulse, verbose] =
    tapas_physio_get_cardiac_pulses(
    ons_secs.t, ons_secs.c,
    thresh.cardiac.initial_cpulse_select,
    thresh.cardiac.modality, [], verbose);
```

The second pass is an optional manual R-wave/pulse oximetry peak selection implemented by Jakob Heinzle (Translational Neuromodeling Unit, University of Zurich and ETH Zurich). Herein, a graphical user interface is presented in Matlab that shows the cardiac time course with the detected pulses and asks for manual addition and removal of pulses via mouse clicks. This functionality is activated by setting

```
thresh.cardiac.posthoc_   'manual'
cpulse_select.method
thresh.cardiac.posthoc_
```

```
cpulse_select.file        <filenameManualPulses>
```

before calling

```
[ons_secs, outliersHigh, outliersLow] =
    tapas_physio_correct_cardiac_pulses_
    manually(ons_secs,thresh.cardiac.posthoc_
    cpulse_select);
```

The close-up region chosen for display and manual pulse selection/removal can be modified by adjusting the outlier detection parameters in `cardiac.posthoc_cpulse_select`:

`posthoc_cpulse_select.percentile`	e.g. 80; percentile of beat-to-beat interval histogram that constitutes the "average heart beat duration" in the session
`posthoc_cpulse_select.upperThresh`	e.g. 80; minimum exceedance (in %) from average heartbeat duration to be classified as missing heartbeat
`posthoc_cpulse_select.lowerThresh`	e.g. 60; minimum reduction (in %) from average heartbeat duration to be classified an abundant heartbeat

As for the QRS-wave, the manually detected and removed pulses are stored in `<filenameManualPulses>` and can be loaded by setting `posthoc_cpulse_select.method='load'`. If no post-hoc manual pulse selection is desired, set `cardiac.posthoc_cpulse_select.method='off'`.

Finally, preprocessing of the physiological time courses, i.e. the filtered respiratory time course `ons_secs.fr` and the vector of cardiac pulse occurrences, `ons_secs.cpulse`, is concluded by cropping them to the time interval relevant for scanning, i.e. from

the start of the first scan (after dummy scans) until the end of the last scan of the session:

```
[ons_secs, sqpar] =
    tapas_physio_crop_scanphysevents_to_
    acq_window(ons_secs, sqpar);
```

The uncropped time series are preserved in a substructure `ons_secs.raw`.

1.3.4 Physiological Noise Modeling

Given the cropped physiological time series of filtered breathing belt amplitude and onset times of heartbeats, the modeling of the physiological noise time series can take place.

The physIO Toolbox offers to model Fourier expansions of cardiac and respiratory phase according to RETROICOR (Glover et al., 2000; Harvey et al., 2008), as well as noise modeling of heart rate variability (HRV) and respiratory volume per time (RVT) utilizing the cardiac and respiratory response function, respectively (Birn et al., 2008; Chang et al., 2009).

Modeling options for the physiological noise can be set in the `physio.model`-substructure:

```
model.type        'RETROICOR' , 'HRV', 'RVT' or
                  any combination of them, e.g.
                  'RETROICOR_HRV',
                  'RETROICOR_HRV_RVT', 'HRV_RVT'
model.order.c     e.g. 3; order of cardiac phase
                  Fourier expansion
model.order.r     e.g. 4; order of respiratory
                  phase Fourier expansion
model.order.cr    e.g. 1; order of sum/difference
                  of cardiac/respiratory phase
                  expansion (phase interaction)
```

For both the cardiac and respiratory RETROICOR regressors, the regressor generation consists of three steps: (1) phase estimation, (2) downsampling to the acquisition time-points (at a reference slice of each scan volume defined by `sqpar.onset_slice`) and (3) generation of different noise time series via Fourier expansion. Finally, an interaction between cardiac and respiratory phases can be modelled via an expansion of their phase sum and differences (Brooks et al., 2008; Harvey et al., 2008).

These three steps are comprised in one function for cardiac, respiratory and interaction terms

```
[cardiac_sess, respire_sess, mult_sess,
ons_secs] =
    tapas_physio_create_retroicor_regressors(
    ons_secs, sqpar, model.order, verbose);
```

with the following output parameters:

```
cardiac_sess      [nScans, 2*model.c] Fourier
                  expansion (cosine/sine columns)
                  of cardiac phase at reference
                  slice for each scan
respire_sess      [nScans, 2*model.r] Fourier
                  expansion of respiratory phase
                  for each scan
mult_sess         [nScans, 4*model.cr] Fourier
                  expansion of sum and difference
                  of cardiac and respiratory
                  phase for each scan
ons_secs.         [nScans, 1] cardiac phase
c_sample_phase    sampled at reference slice for
                  each scan
ons_secs.         [nScans, 1] respiratory phase
c_sample_phase    sampled at reference slice for
                  each scan
```

Within `tapas_physio_create_retroicor_regressors`, step (1), the phase estimations, are performed via

```
c_phase =
tapas_physio_get_cardiac_phase(ons_secs.cpulse,
    sqpar.spulse);
r_phase =
    tapas_physio_get_respiratory_phase(
    ons_secs.fr, log_files.sampling_interval);
```

for cardiac and respiratory data, respectively. These functions implement equations (1.1) and (1.2) (cf. Figure 1.3), following the original RETROICOR publication (Glover et al., 2000). Step (2), the downsampling to one reference time-point per scan, is accomplished by two functions:

```
sample_points =
    tapas_physio_get_sample_points(ons_secs,
    sqpar);
c_sample_phase =
    tapas_physio_downsample_phase(spulse,
    c_phase, sample_points,
    log_files.sampling_interval);
r_sample_phase =
    tapas_physio_downsample_phase(spulse,
    r_phase, sample_points,
    log_files.sampling_interval);
```

Finally, step (3), the Fourier expansion, is performed via `tapas_physio_get_fourier_expansion` for cardiac, respiratory and interaction regressors:

```
cardiac_sess =
    tapas_physio_get_fourier_expansion(
    c_sample_phase,order.c);
respire_sess =
    tapas_physio_get_fourier_expansion(
    r_sample_phase,order.r);
crplus_sess =
```

```
    tapas_physio_get_fourier_expansion(
    c_sample_phase+r_sample_phase,order.cr);
crdiff_sess =
    tapas_physio_get_fourier_expansion(
    c_sample_phase-r_sample_phase,order.cr);
mult_sess = [crplus_sess crdiff_sess];
```

For the impulse-response function based noise models, the confound regressors are generated in three different steps by (1) estimation of the time series of the respective physiological signal component, i.e. heart rate and respiratory volume per time, (2) convolution with the corresponding response functions at a high temporal resolution (slice TR) and (3) extraction of the reference time-points within each scan volume.

These three steps are summarized in one function for each heart rate variability (HRV) and respiratory volume per time (RVT) regressors:

```
[convHRV, ons_secs.hr, verbose] =
    tapas_physio_create_hrv_regressor(ons_secs,
    sqpar, verbose);
[convRVT, ons_secs.rvt, verbose] =
    tapas_physio_create_rvt_regressor(ons_secs,
sqpar, verbose);
```

The relevant output parameters of these functions are:

```
convHRV          [nScans, 1] heart-rate
                 variability regressor;
                 convolved with cardiac response
                 function
convRVT          [nScans, 1] respiratory volume
                 per time regressor; convolved
                 with respiratory response
                 function
ons_secs.hr      [nScans,1] estimated heart rate
                 at reference slice during each
```

	scan
ons_secs.rvt	[nScans,1] estimated respiratory volume per time at reference slice during each scan

Within `tapas_physio_create_hrv_regressor`, the heart rate is (1) computed from the cardiac pulse data *for all slices* by calling

```
hr = tapas_physio_hr(ons_secs.cpulse,
    sample_points);
```

This vector `hr` is (2) convolved with the cardiac response function `tapas_physio_crf` realizing equation (1.6) before (3) being reduced to the values at the reference slice.

Similarly, `tapas_physio_create_rvt_regressor` computes (1) the respiratory volume per time for all slices via

```
rvt = tapas_physio_rvt(ons_secs.fr, ons_secs.t,
    sample_points);
```

and then (2) convolves the sampled `rvt` with the respiratory response function `tapas_physio_rrf`, which implements equation (1.5). The resulting vector is (3) reduced to the final `convRVT` by extracting the values at the reference slice time for all scans.

1.3.5 Physiological Noise Correction

The estimated noise time series are summarized as a design matrix of nuisance or confound regressors by the physIO Toolbox. Specifically, RETROICOR, HRV and RVT regressors are concatenated – if existing – as subsequent columns to yield

```
physIO.model.R = [cardiac_sess, resp_sess,
```

```
    mult_sess, convHRV, convRVT];
```

This design matrix can be appended to the specified design matrix of a mass-univariate GLM employed in fMRI analysis.

Optionally, some of the physiological regressors can be orthogonalized to each other by specifying

```
model.order.orthogonalise   'none', 'cardiac',
                            'resp', 'mult', 'all'
```

We have found this to be beneficial for fMRI sessions acquired with cardiac triggering, since cardiac regressors tend to be nearly constant there due to recurring cardiac phases for the same slice over scans (Kasper et al., 2009).

Furthermore, other confound regressors, e.g. motion parameters, can be appended to `model.R` by specifying the name of an ASCII-file with the same `[nScans, nRegressors]` matrix structure:

```
model.input_other_multiple  e.g.'rp_fMRI_001.txt'
_regressors
```

The final nuisance matrix will be saved to the file specified in

```
model.output_multiple_regressors   filename, e.g. 'multiple_regressors.txt'
```

If the filename contains an extension other than `.mat` (e.g. `.txt`), the matrix is saved as ASCII-file, in which each column represents one physiological regressor, and the rows contain regressor entries for each scan in ascending order. For SPM in particular, the output file can also be specified as `.mat`-file, holding a variable `R`. In both cases, this output file can serve as "multiple regressors" entry for the first level GLM specification in SPM. Thus, the physIO Toolbox provides a direct interface of noise correction using the established GLM framework (Josephs et al., 1997). The voxel-wise estimation of the physiological noise component itself is performed by the fMRI analysis software of choice, e.g. SPM, independent of the physIO Toolbox.

1.3.6 Assessment of Noise Correction Efficacy

The efficacy of voxel-wise physiological noise correction can be assessed using F-contrasts, as mentioned in section 1.2. Therefore, the physIO Toolbox provides scripts to automatically create the relevant contrasts for physiological regressors in SPM and subsequently report the statistical maps of physiological noise distribution in a PostScript-file. Specifically, family-wise error corrected contrasts are displayed on a structural overlay, centered on the global maximum F-value, for each subject and all existing sets of physiological noise regressors, including movement parameters.

The script to run this automated assessment of noise efficacy is `tapas_physio_check_efficacy.m`. Herein, all paths at the beginning of the script (flagged by `#MOD`) have to be modified to match the actual study/subject/GLM/physIO-code folders. All relevant auxiliary functions to this script are also prepended with `tapas_physio_check_` and include SPM-jobs for contrast creation and results reporting, as well as functions to extract regressor names and columns from the SPM.mat-file itself.

A typical output page of the script can be found in Figure 1.5.

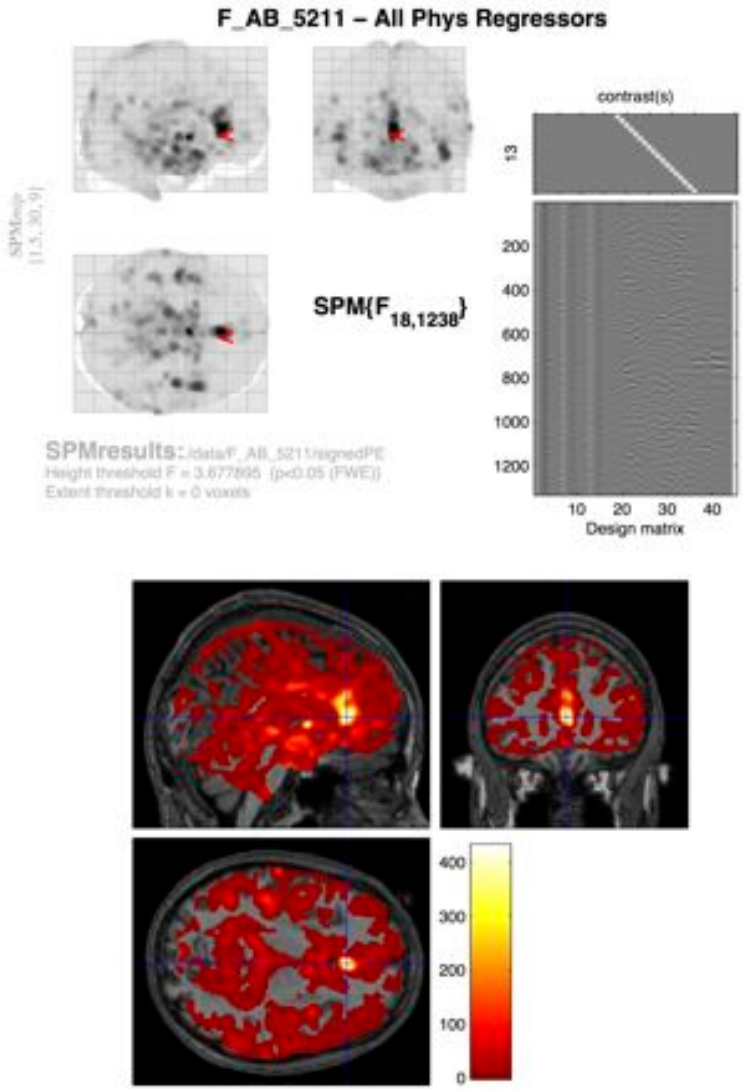

FIGURE 1.5 *Typical output page of PostScript-file created by* `tapas_physio_check_efficacy`*-script using SPM12b.*

1.4 Applications of the PhysIO Toolbox

To date, the PhysIO Toolbox has been successfully used in a number of fMRI studies covering sensory-reward-learning paradigms (Iglesias et al., 2013), social learning experiments (Diaconescu et al., 2013), real-time feedback fMRI (Sulzer et al., 2013) and high-field fMRI (Kasper et al., 2009). Furthermore, its complementary value for noise reduction in combination with magnetic field monitoring was demonstrated utilizing basic visuo-motor tasks and resting-state fMRI (Kasper et al., 2014). In fact, for any kind of fMRI experiment, physiological noise correction is highly recommended. Benefits of model-based physiological noise correction have been reported for both task-based (Hutton et al., 2011) and resting-state fMRI (Birn, 2012; Birn et al., 2006).

In particular for the classical mass-univariate GLM analysis, as e.g. implemented in SPM, a three-fold improvement due to physiological noise correction is anticipated on the single-subject level. First and foremost, the reduction in unexplained variance, i.e. residual noise, directly enhances the sensitivity for relevant task effects in statistical parametric maps. Both t- and F-values contain an expression of residual noise in the denominator (cf. Chapter 2.3). Thus, t- and F-based statistics will improve through proper physiological noise correction, since unexplained variance decreases. Secondly, physiological noise modeling will reduce the risk of false positives and negatives for the contrasts of interest. Specifically, false positives are avoided, because task regressors that are partially correlated with physiology contain shared variance with physIO regressors and thus lower parameter estimates after noise correction. False negatives, on the other hand, are mitigated by reducing effects of partial anticorrelation between task and physiological regressors. Thirdly, physiological noise modeling removes the long-lasting autocorrelation in fMRI voxel time series due to the approximate periodicity of cardiac and respiratory fluctuations. Hence, the AR(1)-assumptions of the hyperparameter estimations are better fulfilled for the empirical Bayesian model

inversion procedure of the GLM in SPM, presumably providing more accurate parameter and contrast estimates (Kiebel and Holmes, 2007).

In principle, all these improvements of first-level analyses through the physIO Toolbox propagate to the second-level, i.e. group effects, directly, if a full random-effects analysis is performed (Penny and Holmes, 2007). However, practically, the summary statistics approach is typically employed in second-level analysis (Holmes and Friston, 1998). Herein, only contrast estimates (i.e. numerators of t-contrasts) are taken to the second level to perform a t-test, assuming an equal noise distribution between subjects; Thus, the above-mentioned first improvement of first-level statistics through residual error reduction does not translate into improved second-level sensitivity. Nevertheless, both the more accurate parameter estimation for regressors correlated with physiology as well as AR(1) hyperparameters will render the second-level results more accurate.

Beyond mass-univariate GLM analyses, physiological noise correction using the physIO toolbox can both impact on functional connectivity analysis of resting-state data (Beckmann and Smith, 2004; Biswal et al., 1995) and DCM-based effective connectivity measures (Friston, 2007). For functional connectivity analyses, it is essential to remove physiological noise, because it is often correlated over brain regions, and can thus be misinterpreted as the neuronally-induced BOLD coupling targeted in resting-state fMRI (Birn, 2012; Cole et al., 2010). For effective connectivity analyses, on the other hand, the modelled time series extracted from brain regions of interest can be adjusted for the physIO contrasts. Thereby, unexplained variance of the time series due to physiological fluctuations is reduced and the estimates of relevant connectivity parameters become more robust.

The aforementioned conceptual benefits of applying the physIO toolbox have been validated on empirical data, as well as the robustness of the employed noise modeling approach. To illustrate

the effect of physIO-based noise correction, we conclude with some examples from a recent cognitive neuroscience fMRI study focusing on social learning (Diaconescu et al., 2013). In this experiment, participants had to make decisions based on the advice of another agent. Hence, they had to learn the advice reliability to perform optimally. In the study, the observable learning signals in the fMRI data were hypothesized to be hierarchical prediction errors that contrasted the predicted advice reliability with the actual advice validity on every single trial.

With regard to the robustness and reproducibility of the noise correction, we find the same spatial noise distribution patterns for physIO regressors in every subject (Figure 1.6 A). Fluctuations correlated to the cardiac cycle manifest around major vessels (basilar artery, anterior communicating arteries, internal carotid arteries, sagittal sinus) and in pulsatile CSF regions (especially surrounding the brainstem), but also more distributed in gray matter cortical areas. The breathing cycle, on the other hand, particularly induces fluctuations close to tissue and brain boundaries as well as inferior brainstem areas (pons). Interactions between cardiac and respiratory cycle generate additional noise foci in the aqueduct, temporal horn of the lateral ventricle, and the inferior part of the pons, close to the basilar artery.

These fluctuations, if unaccounted for, increase the voxel variance by up to 70 % (Figure 1.6 A). Consequently, on the single-subject level, we found that the application of the physIO-based noise correction lead to widespread variance reduction in the brain, significant after whole-brain multiple comparison correction in every subject (family-wise error rate p=0.05). The spatial extent of the clusters affected by physiological noise varied between subjects, but not the qualitative localization mentioned in the last paragraph (Figure 1.6 B).

Furthermore, we found that physiological noise correction does not only improve BOLD sensitivity for physiology-related tasks, but also in subtle cognitive paradigms, like learning from social

information and reward. We observed that, for the relevant learning-related contrasts, a higher sensitivity at the group level could be accomplished through physiological noise correction (Figure 1.7). Specifically, predictions of advice reliability could be localized in the bilateral fusiform-face area (FFA), the right posterior superior temporal sulcus (STS), anterior temporal-parietal junction (TPJ), and dorsomedial prefrontal cortex (dmPFC) in GLMs including physIO regressors. These areas have been implicated in social learning and learning from facial expressions in previous studies (Diaconescu et al., 2013). In contrast to that, without physiological noise correction, only the left FFA was robustly recovered. Similarly, social prediction error signals could be captured in the bilateral superior occipital cortex (SOC) and right dmPFC after physiological noise correction, but only in the SOC without noise modeling. Interestingly, false positives were also reduced through the physIO-based noise correction for the prediction error contrast. Part of the substantia nigra (SN) that seemed to correlate with prediction error in the absence of physiological noise modeling, did not exhibit significant activation after applying the physIO correction. This emphasizes the need for a rigorous physiological noise correction, since a priori the SN is both a region connected to prediction error learning and a site of particularly high fluctuations.

In summary, the correction for physiological fluctuations in fMRI data is an essential component of fMRI analysis, regardless of the experimental paradigm (resting or task-based), research area (e.g. cognitive or sensory-motor processes) or analysis approach (GLM, functional/effective connectivity). Explicit physiological noise modeling, as offered by the physIO Toolbox, improves the robustness of statistical findings, enables the detection of subtle cognitive effects, and prevents from interpreting false positives arising from non-BOLD physiology.

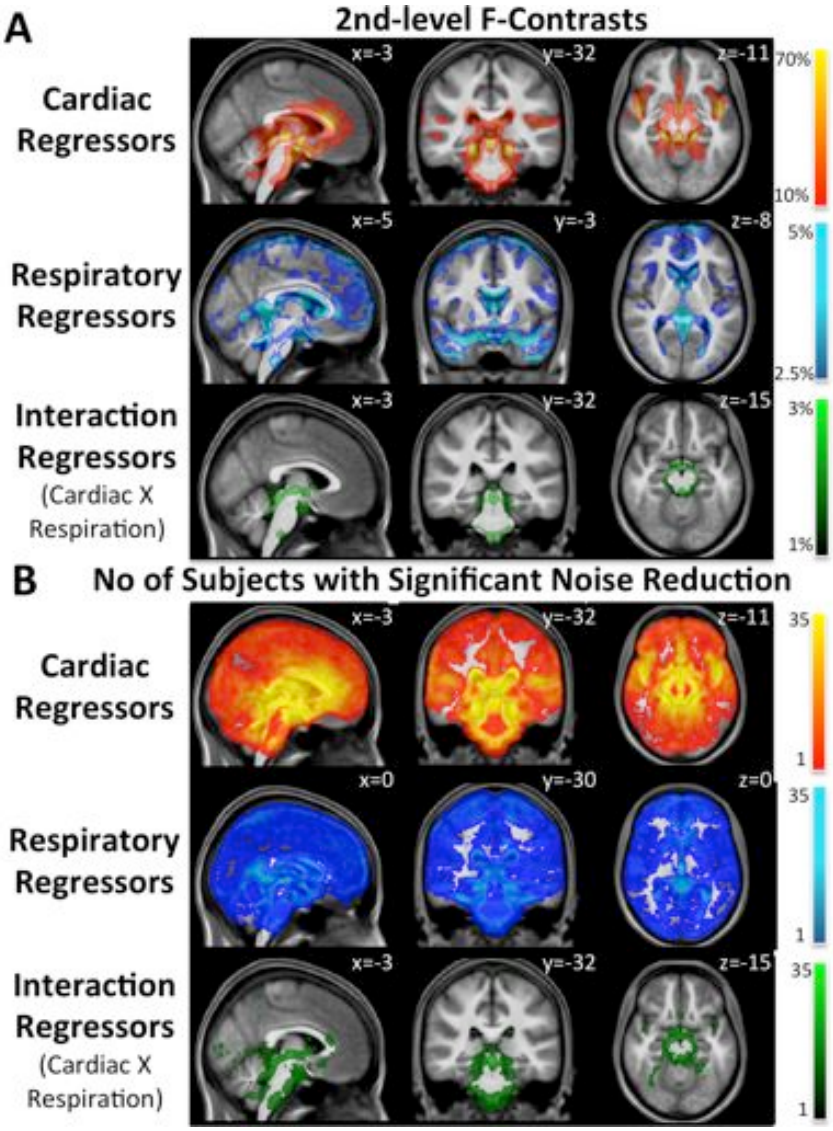

Figure 1.6 *Robust Noise Reduction through physIO Regressors* ***(A)*** *Average additional noise variance explained by physIO regressors (relative to residual after correction) over subjects (N=35); F-value corrected for degrees of freedom* ***(B)*** *Subject count for significant physiological noise detected in each voxel (whole brain FWE-corrected p=0.05)*

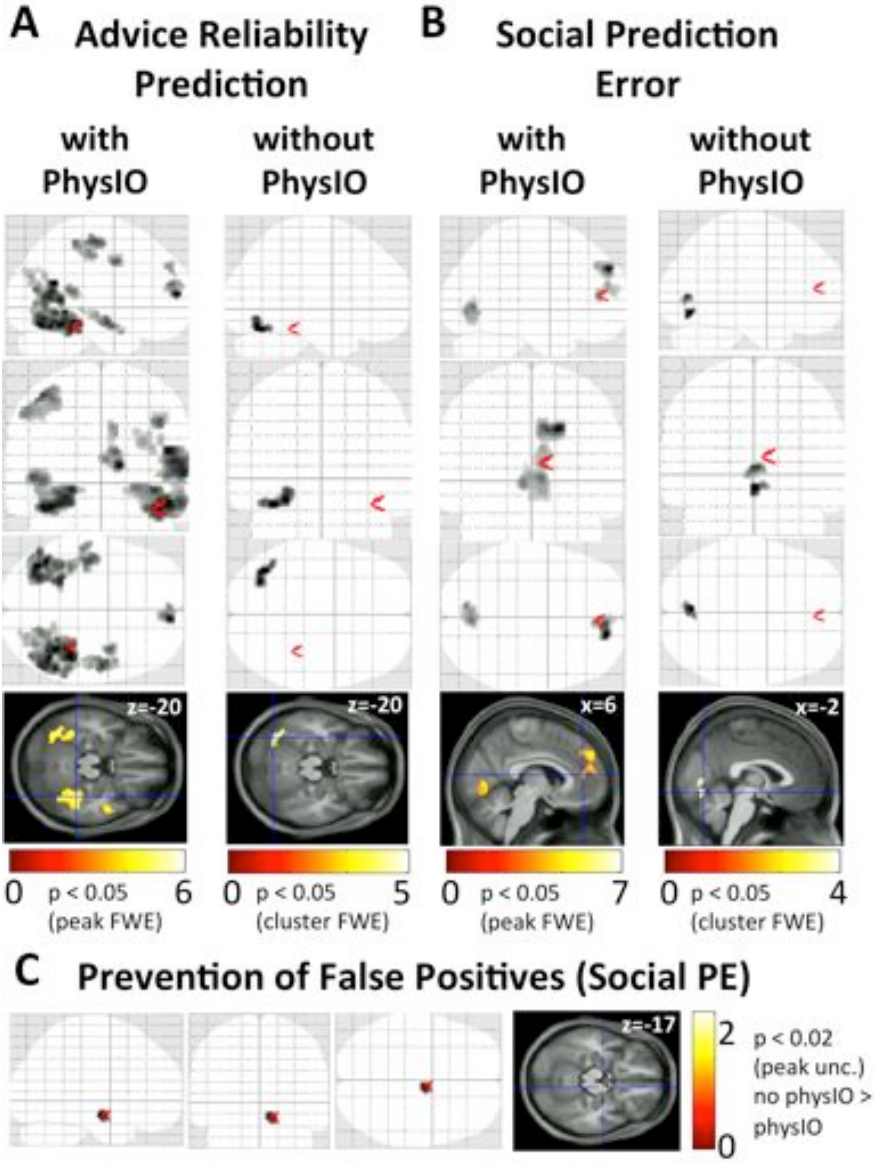
A
Advice Reliability Prediction
B
Social Prediction Error
with PhysIO
without PhysIO
with PhysIO
without PhysIO
z=-20
z=-20
x=6
x=-2
0 p < 0.05 6
(peak FWE)
0 p < 0.05 5
(cluster FWE)
0 p < 0.05 7
(peak FWE)
0 p < 0.05 4
(cluster FWE)
C
Prevention of False Positives (Social PE)
z=-17
2
0
p < 0.02
(peak unc.)
no physIO >
physIO

FIGURE 1.7 *Improvement in Sensitivity through Physiological Noise Correction in a Cognitive Paradigm.* ***(A)*** *Group statistics for t-contrast modeling the prediction of advice reliability. Only by using PhysIO noise correction, bilateral FFA, TPJ and dmPFC can be identified as task-relevant regions.* ***(B)*** *Group statistics for t-contrast modeling the prediction error learning signal. Involvement of dmPFC is implicated only after physiological noise correction.* ***(C)*** *Prevention of false positives through physIO noise correction. The correlation of midbrain (substantia nigra) activity with prediction error can be explained away by shared variance with physiological fluctuations, preventing false conclusions of the involvement of the midbrain in this task.*

1.5 References

Beckmann, C.F., Smith, S.M., 2004. Probabilistic independent component analysis for functional magnetic resonance imaging. IEEE Trans. Med. Imaging 23, 137 –152.

Birn, R.M., 2012. The role of physiological noise in resting-state functional connectivity. NeuroImage 62, 864–870.

Birn, R.M., Diamond, J.B., Smith, M.A., Bandettini, P.A., 2006. Separating respiratory-variation-related fluctuations from neuronal-activity-related fluctuations in fMRI. NeuroImage 31, 1536–1548.

Birn, R.M., Smith, M.A., Jones, T.B., Bandettini, P.A., 2008. The respiration response function: The temporal dynamics of fMRI signal fluctuations related to changes in respiration. NeuroImage 40, 644–654.

Biswal, B., Zerrin Yetkin, F., Haughton, V.M., Hyde, J.S., 1995. Functional connectivity in the motor cortex of resting human brain using echo-planar mri. Magn. Reson. Med. 34, 537–541.

Brooks, J.C.W., Beckmann, C.F., Miller, K.L., Wise, R.G., Porro, C.A., Tracey, I., Jenkinson, M., 2008. Physiological noise modelling for spinal functional magnetic resonance imaging studies. NeuroImage 39, 680–692.

Chang, C., Cunningham, J.P., Glover, G.H., 2009. Influence of heart rate on the BOLD signal: The cardiac response function. NeuroImage 44, 857–869.

Cole, D.M., Smith, S.M., Beckmann, C.F., 2010. Advances and pitfalls in the analysis and interpretation of resting-state FMRI data. Front. Syst. Neurosci. 4, 1–15.

Diaconescu, A.O., Mathys, C., Weber, L.A.E., Kasper, L., Stephan, K.E., 2013. Deciphering the best intentions; genetic and model-based mapping of individual variability in social inference and decision making. Prep.

Friston, K., 2007. Chapter 41 - Dynamic Causal Models for fMRI, in: Karl Friston, John Ashburner, Stefan Kiebel, Thomas

Nichols and William PennyA2 - Karl Friston, J.A., William Penny (Eds.), Statistical Parametric Mapping. Academic Press, London, pp. 541–560.

Glover, G.H., Li, T.Q., Ress, D., 2000. Image-based method for retrospective correction of physiological motion effects in fMRI: RETROICOR. Magn. Reson. Med. 44, 162–7.

Harvey, A.K., Pattinson, K.T.S., Brooks, J.C.W., Mayhew, S.D., Jenkinson, M., Wise, R.G., 2008. Brainstem functional magnetic resonance imaging: Disentangling signal from physiological noise. J. Magn. Reson. Imaging 28, 1337–1344.

Holmes, A., Friston, K., 1998. Generalisability, random effects & population inference. NeuroImage 7, S754.

Hutton, C., Josephs, O., Stadler, J., Featherstone, E., Reid, A., Speck, O., Bernarding, J., Weiskopf, N., 2011. The impact of physiological noise correction on fMRI at 7 T. NeuroImage 57, 101–112.

Iglesias, S., Mathys, C., Brodersen, K.H., Kasper, L., Piccirelli, M., den Ouden, H.E.M., Stephan, K.E., 2013. Hierarchical Prediction Errors in Midbrain and Basal Forebrain during Sensory Learning. Neuron 80, 519–530.

Josephs, O., Howseman, A.M., Friston, K.J., Turner, R., 1997. Physiological noise modelling for multi-slice EPI fMRI using SPM, in: Proceedings of the 5th Annual Meeting of ISMRM. Presented at the ISMRM, Vancouver, Canada, p. 1682.

Kasper, L., Haeberlin, M., Vannesjo, S.J., Gross, S., Dietrich, B.E., Brunner, D.O., Ruff, C.C., Stephan, K.E., Pruessmann, K.P., 2014. Matched-filter Acquisition for BOLD fMRI. submitted.

Kasper, L., Marti, S., Vannesjö, S.J., Hutton, C., Dolan, R., Weiskopf, N., Stephan, K.E., Prüssmann, K.P., 2009. Cardiac Artefact Correction for Human Brainstem fMRI at 7 Tesla, in: Proceedings of the Organization for Human Brain Mapping 15. Presented at the HBM, San Francisco, p. 395.

Kiebel, S.J., Holmes, A.P., 2007. Chapter 8 - The General Linear Model, in: Karl Friston, John Ashburner, Stefan Kiebel, Thomas Nichols and William PennyA2 - Karl Friston, J.A.,

William Penny (Eds.), Statistical Parametric Mapping. Academic Press, London, pp. 101–125.

Krüger, G., Glover, G.H., 2001. Physiological noise in oxygenation-sensitive magnetic resonance imaging. Magn. Reson. Med. 46, 631–637.

Penny, W.D., Holmes, A.J., 2007. Chapter 12 - Random Effects Analysis, in: Karl Friston, John Ashburner, Stefan Kiebel, Thomas Nichols and William PennyA2 - Karl Friston, J.A., William Penny (Eds.), Statistical Parametric Mapping. Academic Press, London, pp. 156–165.

Perlbarg, V., Bellec, P., Anton, J.-L., Pélégrini-Issac, M., Doyon, J., Benali, H., 2007. CORSICA: correction of structured noise in fMRI by automatic identification of ICA components. Magn. Reson. Imaging 25, 35–46.

Shmueli, K., van Gelderen, P., de Zwart, J.A., Horovitz, S.G., Fukunaga, M., Martijn Jansma, J., Duyn, J.H., 2007. Low Frequency Fluctuations in the Cardiac Rate as a Source of Variance in the Resting-State fMRI BOLD Signal. NeuroImage 38, 306–320.

Soellinger, M., 2008. Assessment of intracranial dynamics using MRI. Zurich.

Sulzer, J., Sitaram, R., Blefari, M.L., Kollias, S., Birbaumer, N., Stephan, K.E., Luft, A., Gassert, R., 2013. Neurofeedback-mediated self-regulation of the dopaminergic midbrain. NeuroImage 83, 817–825.

Thomas, C.G., Harshman, R.A., Menon, R.S., 2002. Noise Reduction in BOLD-Based fMRI Using Component Analysis. NeuroImage 17, 1521–1537.

Windischberger, C., Langenberger, H., Sycha, T., Tschernko, E.M., Fuchsjäger-Mayerl, G., Schmetterer, L., Moser, E., 2002. On the origin of respiratory artifacts in BOLD-EPI of the human brain. Magn. Reson. Imaging 20, 575–582.