Time-frequency decomposition

Theory and Practice

EEGLAB Workshop XXIII

JAIST, Tokyo, Japan

Day 1

- Signals – EEG
- Goals
  - Describe dynamic characteristics of brain activity
  - Describe relation between different regions of brain
- Approaches
  - Time domain
  - Frequency domain
  - Time/Frequency

Different meanings traditionally given to different frequency bands

Beta 15-30 Hz

Awake, normal alert consciousness

Alpha 9-14 Hz

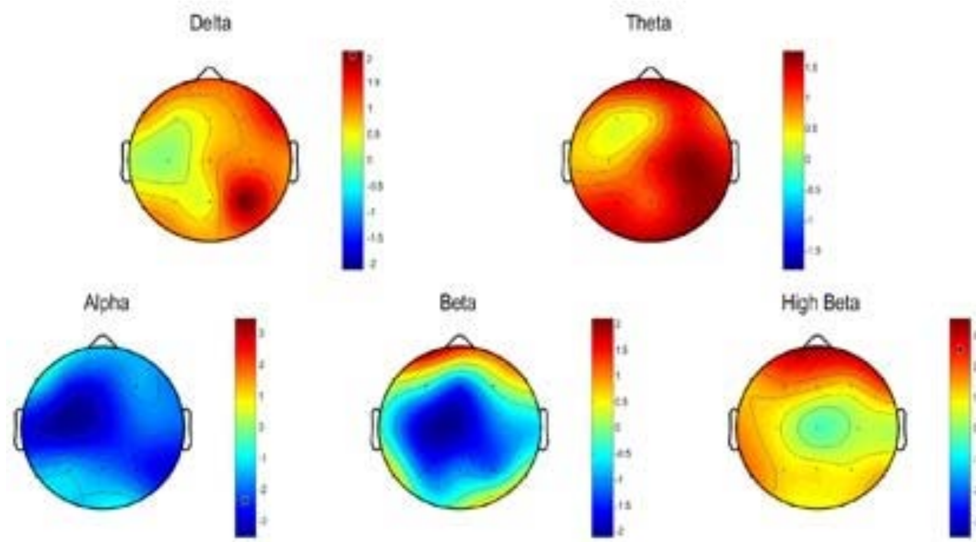
Relaxed, calm, meditation, creative visualisation

Theta 4-8 Hz

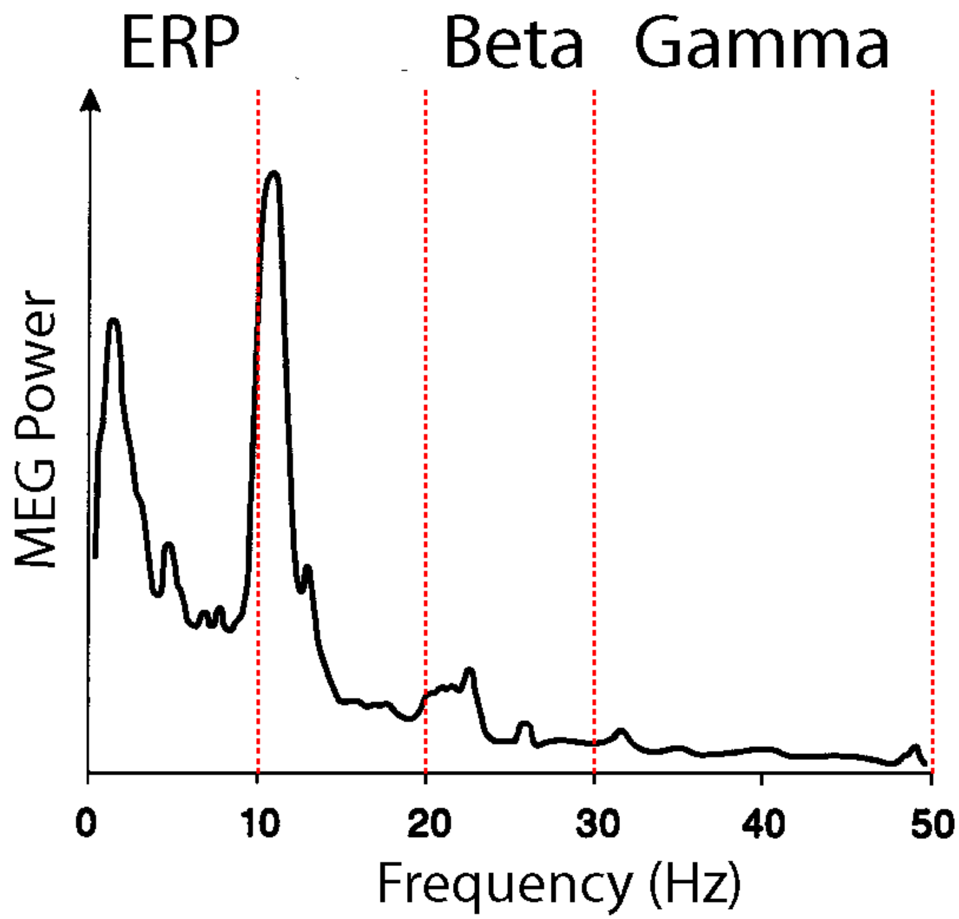
Deep relaxation and meditation, problem solving

Delta 1-3 Hz

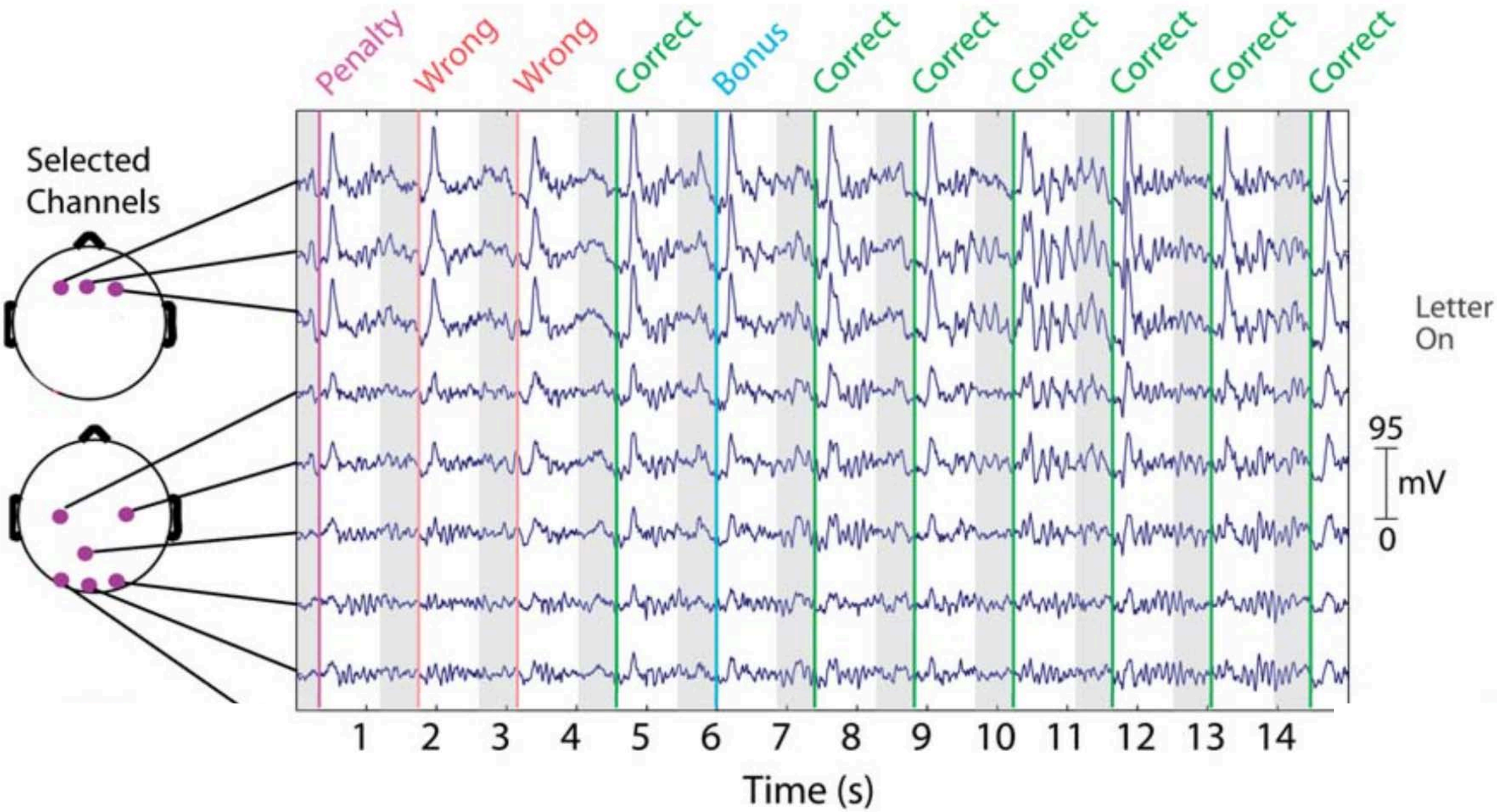
Deep, dreamless sleep



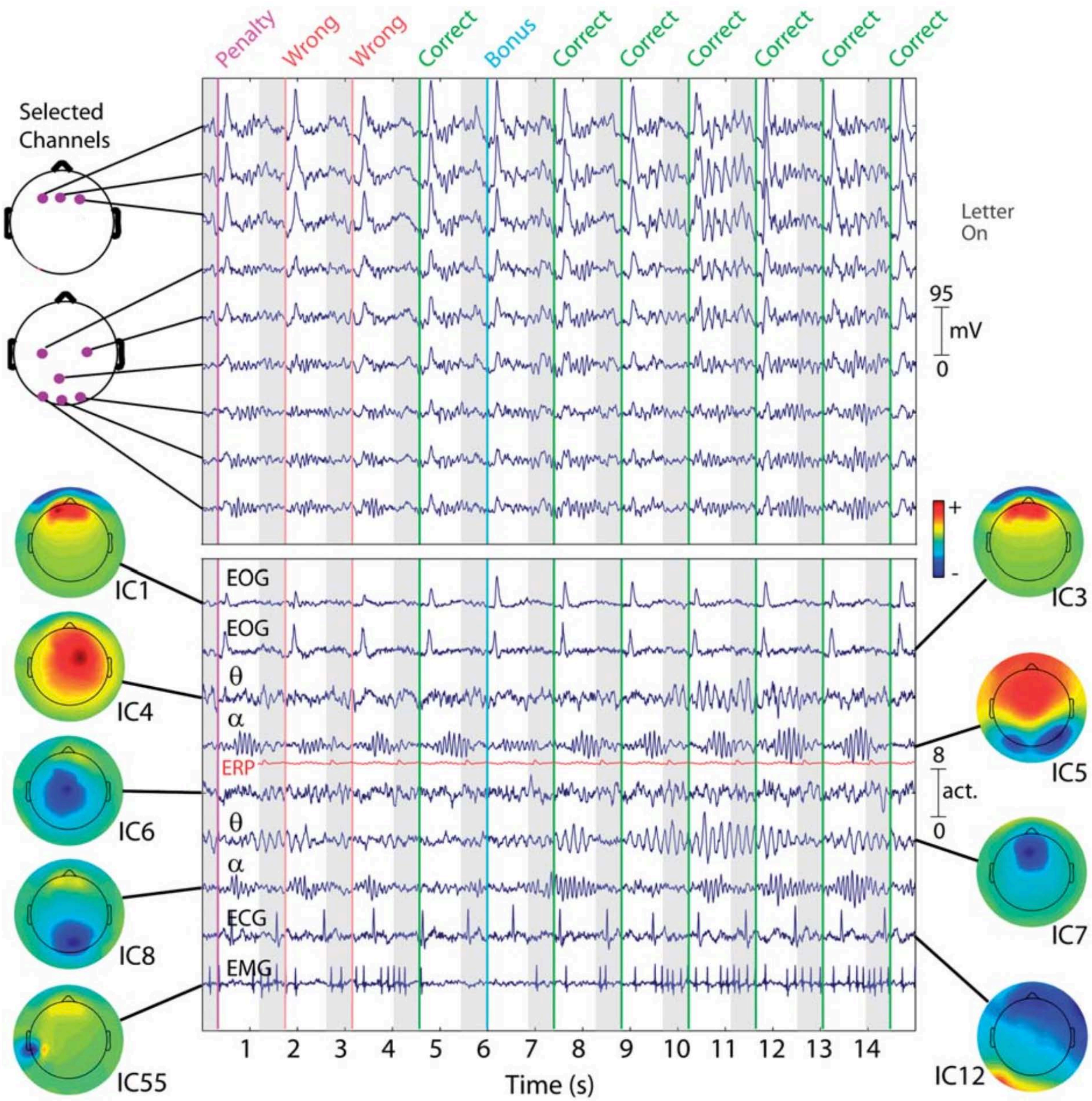
MEEG spectrum



Time varying frequency content

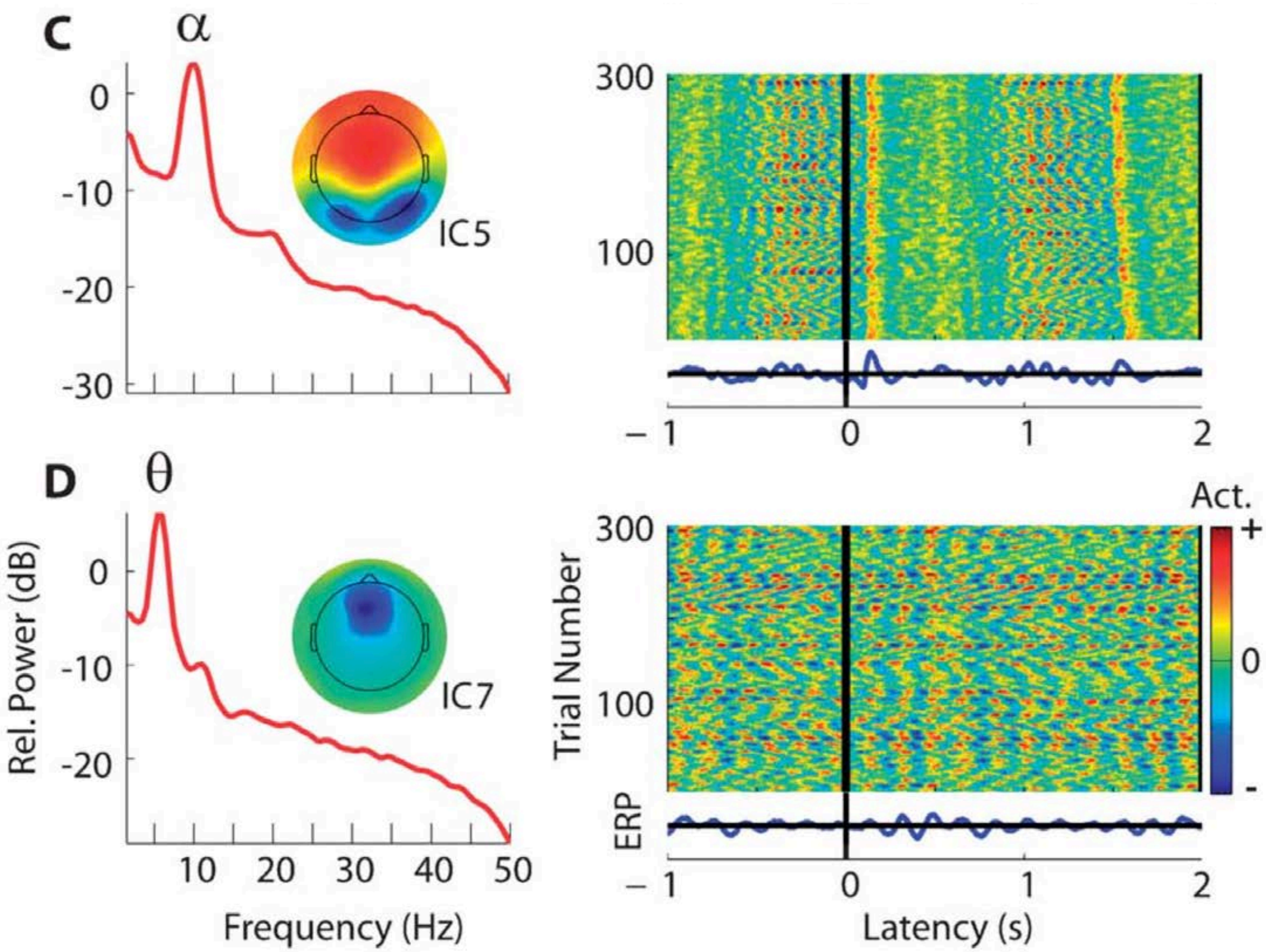


Time-varying frequency content

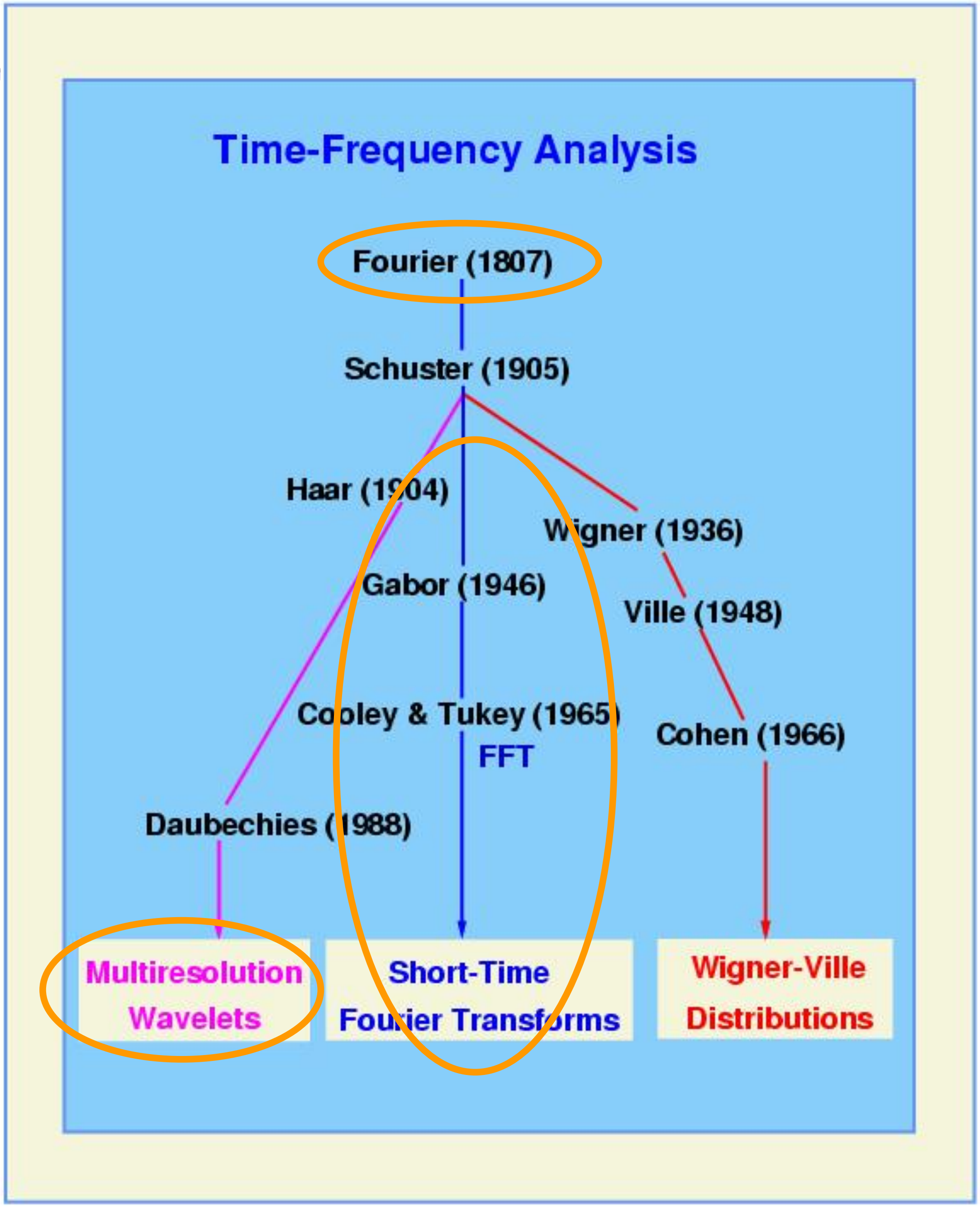


Onton & Makeig, 2006

Power Spectrum does not describe temporal variation



Onton & Makeig, 2006



S. Makeig, 2005

Plan

- **Part 1: Frequency Analysis**
  - Power Spectrum
    - Approaches
      - FFT
      - Welch's Method
    - Windowing

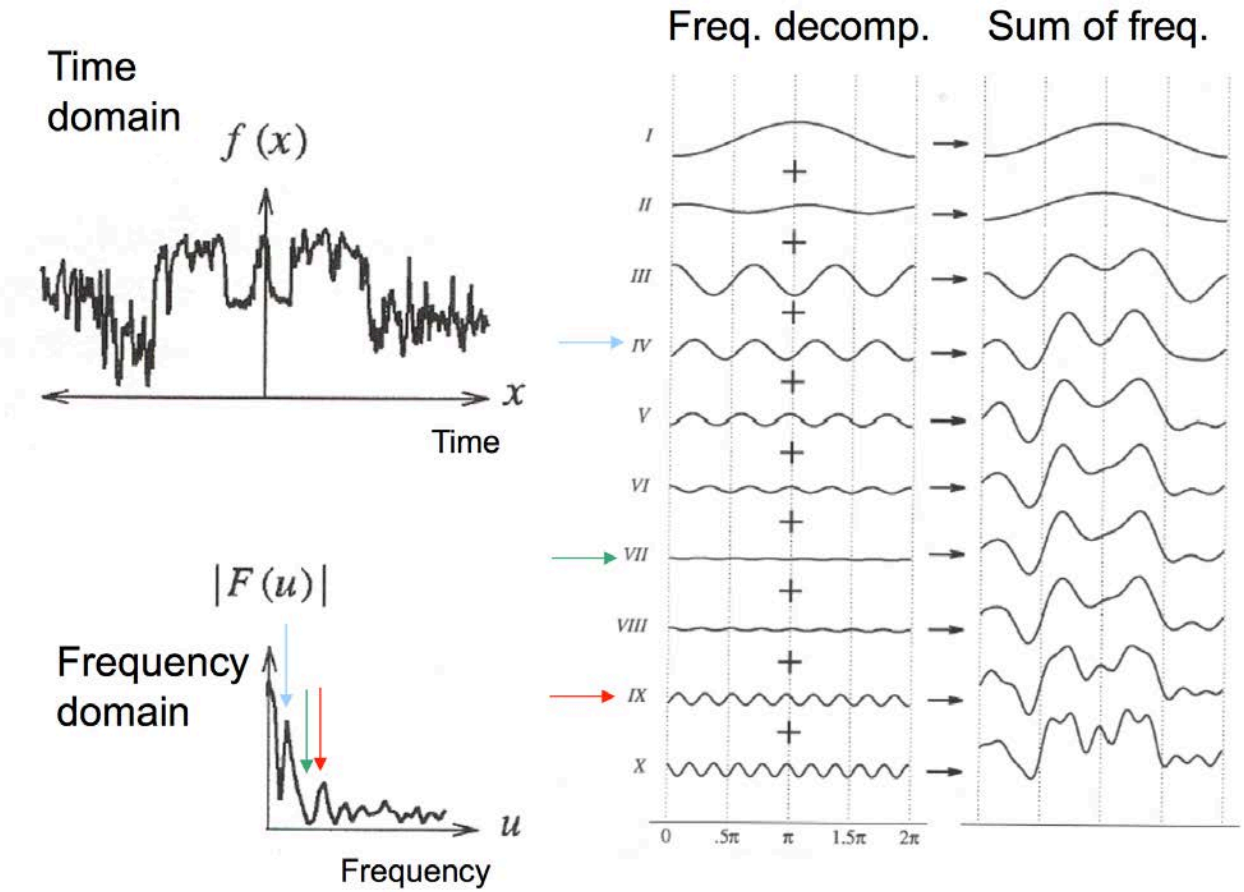
- **Part 2: Time-Frequency Analysis**
  - Short Time Fourier Transform
  - Wavelet Transform
  - ERSP

- **Part 3: Coherence Analysis**
  - Inter-Trial Coherence
  - Event-Related Coherence

Part 1: Frequency Analysis

- Goal: What frequencies are present in signal?
- What is power at each frequency?
- Principle: Fourier Analysis

Fourier Analysis



Forward transform $$F(u) = \int_{-\infty}^{+\infty} f(x) e^{-2\pi i u x} dx$$

Inverse transform $$f(x) = \int_{-\infty}^{+\infty} F(u) e^{2\pi i u x} du$$

Figure, courtesy of Ravi Ramamoorthi & Wolberg

Power Spectrum. Approach 1: FFT

- Why not just take FFT of our signal of interest?
- Advantage – fine frequency resolution
  - ΔF = 1 / signal duration (s)
  - E.g. 100s signal has 0.01 Hz resolution
  - But, do we really need this?

- Disadvantage 1 – high variance
  - Solution: e.g. Welch's method

- Disadvantage 2 – no temporal resolution
  - Solution 1: Short-Time Fourier Transform

Amplitude and phase

- Power spectra describe the *amount* of a given frequency present

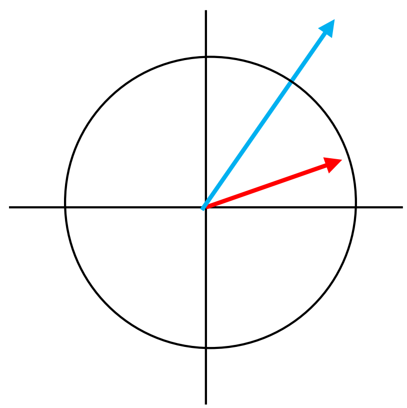
- NOT a complete description of a signal: We also must know the *phase* at each frequency

- FFT/STFT/Wavelet return an amplitude and phase at each time and frequency (represented as complex #).
- To find power, we compute the magnitude, which discards phase.

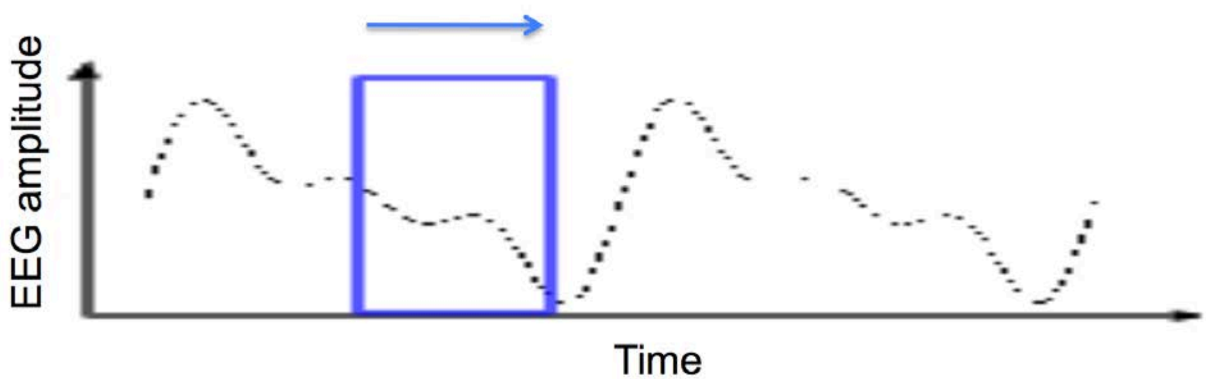
Phasor representation

- A complex number *x* + *yi* can be expressed in terms of amplitude and phase: $ae^{i\theta}$

```
amplitude*exp(i*phase)
amplitude = sqrt(x^2 + y^2); phase = atan(y/x);
```



Approach 2: Welch's Method



Calculate power spectrum of short windows, average.

Advantage: Smoother estimate of power spectrum

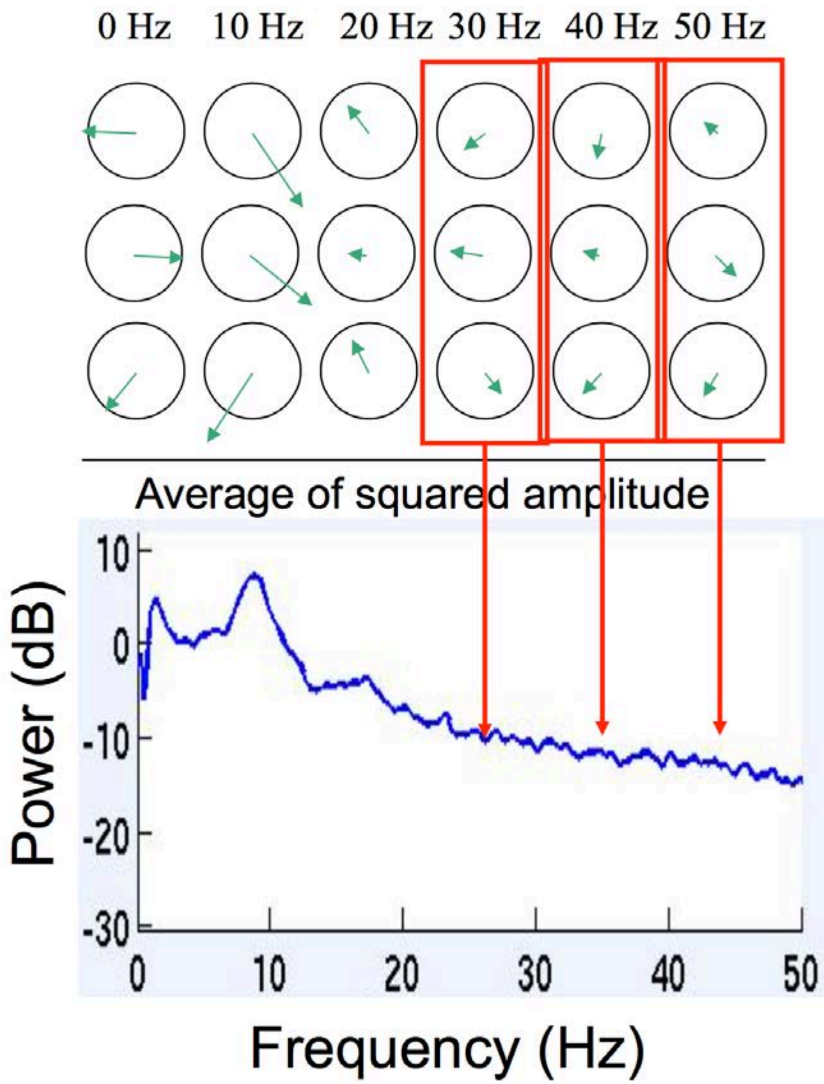
Frequency resolution set by window length

e.g. 1s window -> 1 Hz resolution

In practice: taper, don't use rectangular window

FFT of window 1

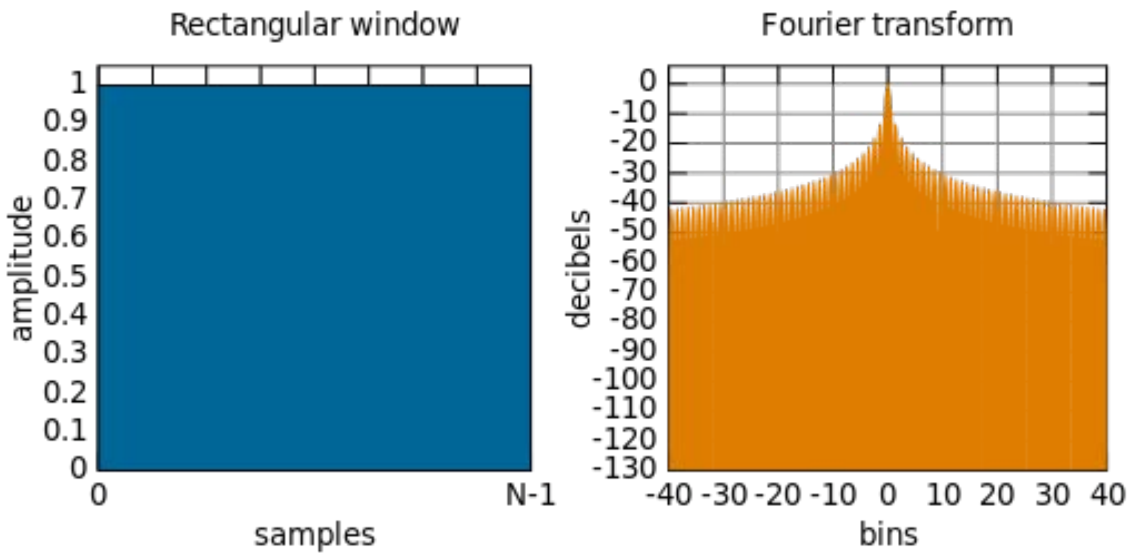
FFT of window 2



Windowing

- When we pick a short segment of signal, we typically window it with a smooth function.
- Windowing in time = convolving (filtering) the spectrum with the Fourier transform of the window
- No window (=rectangular window) results in the most smearing of the spectrum
- There are many other windows optimized for different purposes: Hamming, Gaussian…

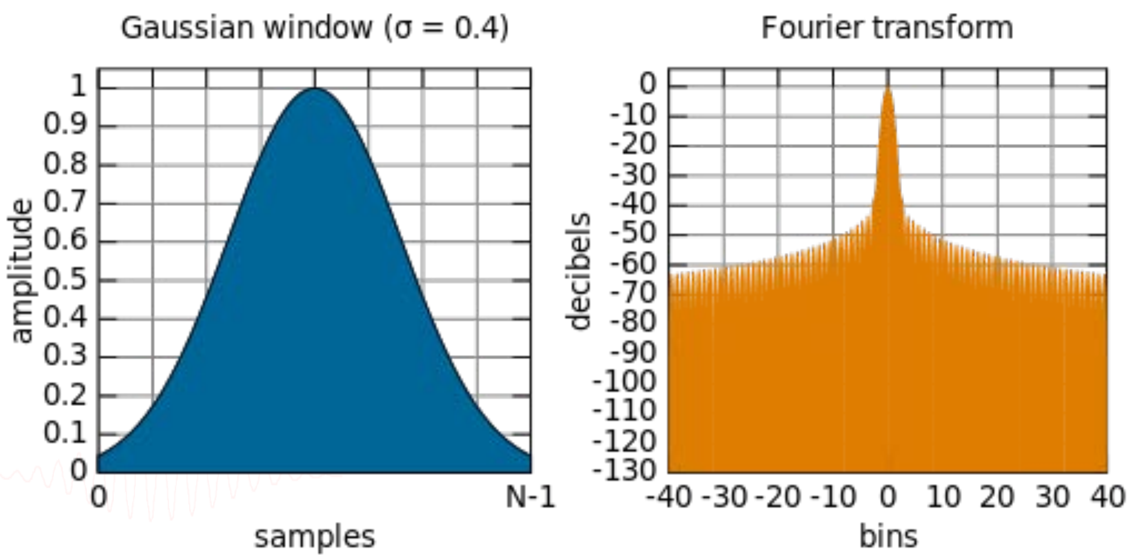
Windows and their Fourier transforms



Narrowest main peak, but

Highest side-lobes

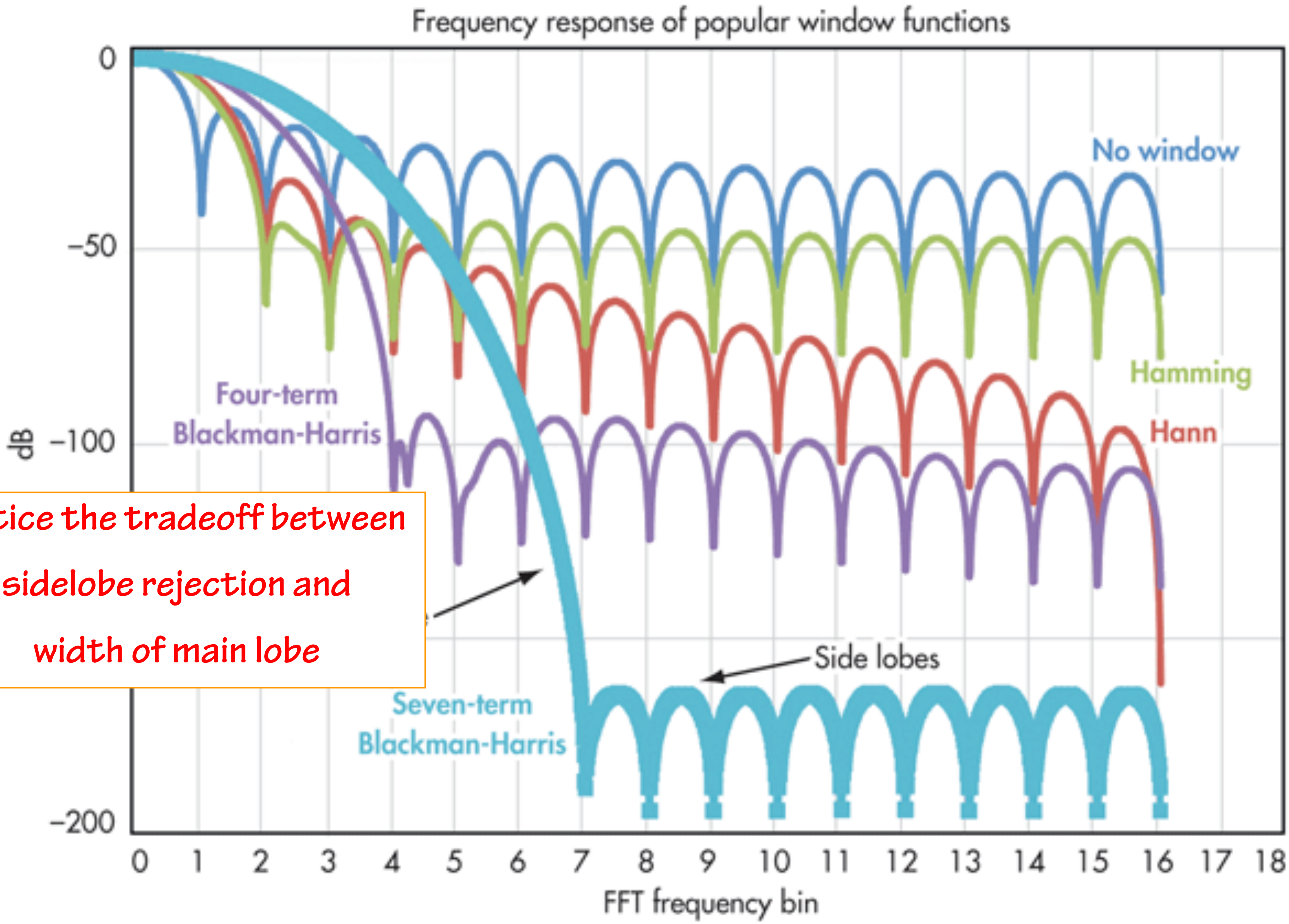
Most spectral 'smearing'



Wider main peak, but

much lower side-lobes

Close-up view



Notice the tradeoff between sidelobe rejection and width of main lobe

Part 2: Time-Frequency Analysis

- Short-Time Fourier Transform
  - Find power spectrum of short windows
  - “Spectrogram”

- Advantage: Can visualize time-varying frequency content

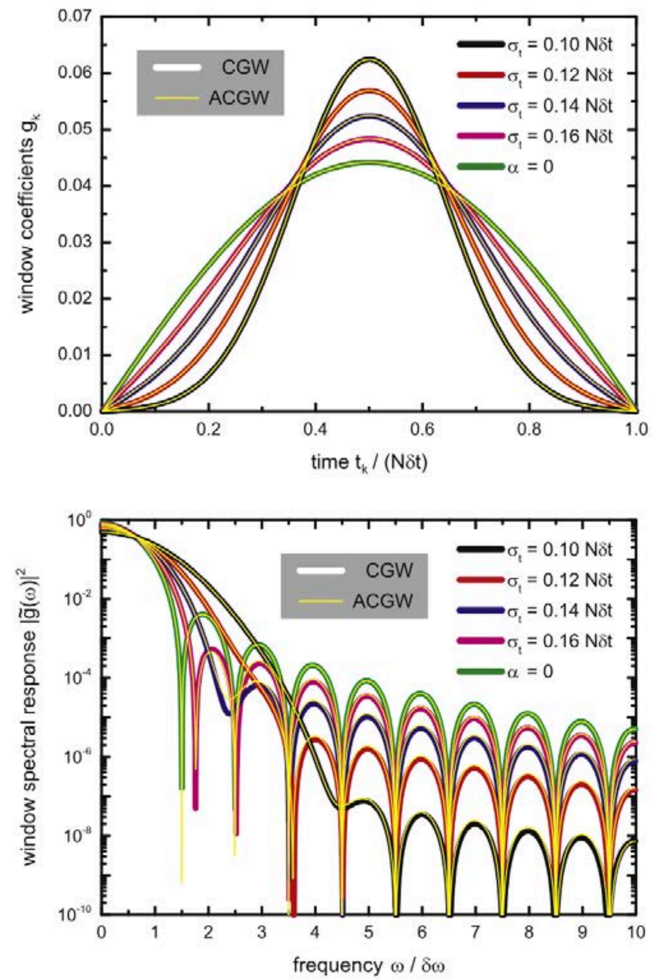
- Disadvantage: Fixed temporal resolution is not optimal

Time-Frequency Uncertainty

- You cannot have both arbitrarily good temporal and frequency resolution!
  - $\sigma_t * \sigma_f \geq 1/2$

- If you want sharper temporal resolution, you will sacrifice frequency resolution, and vice versa.

- (Optimal: Confined Gaussian)



Starosielec S, Hägele D (2014) Discrete-time windows with minimal RMS bandwidth for given RMS temporal width. Signal Processing 102:240–6.

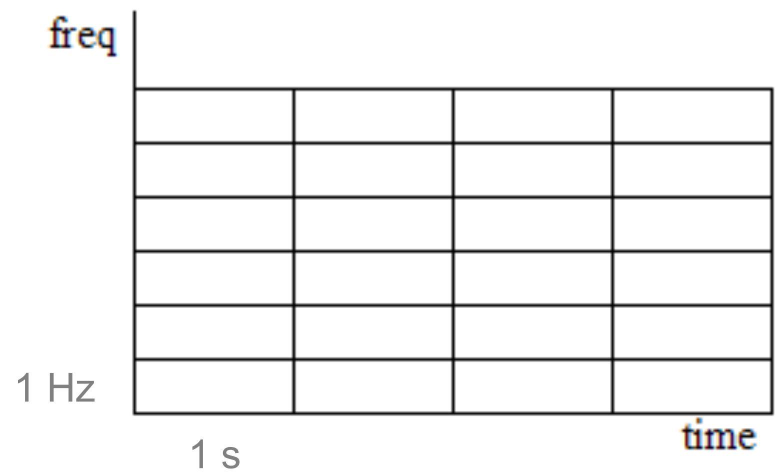
Consequence for STFT

Shorter Windows

poorer frequency resolution

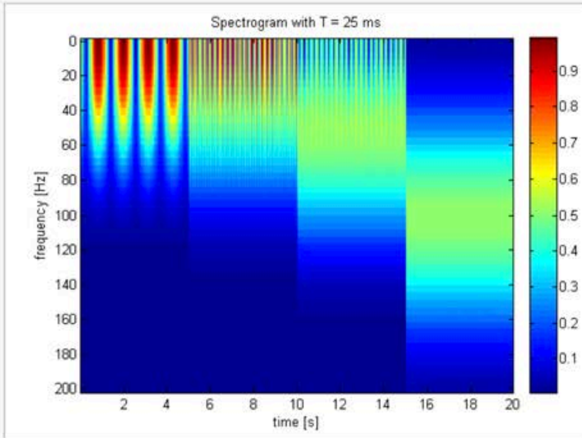
Longer Windows

finer frequency resolution

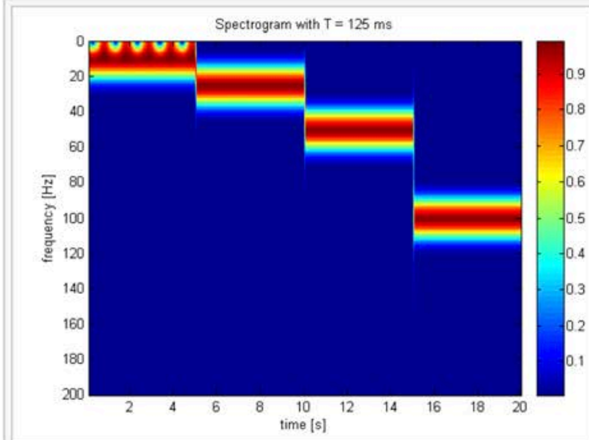


Time-Frequency Tradeoff

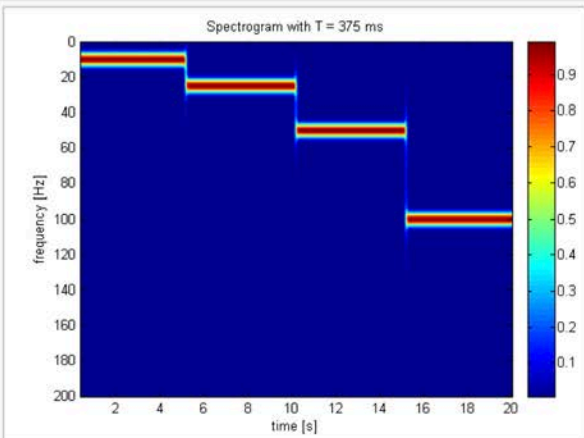
Signal: 10, 25, 50, 100 Hz



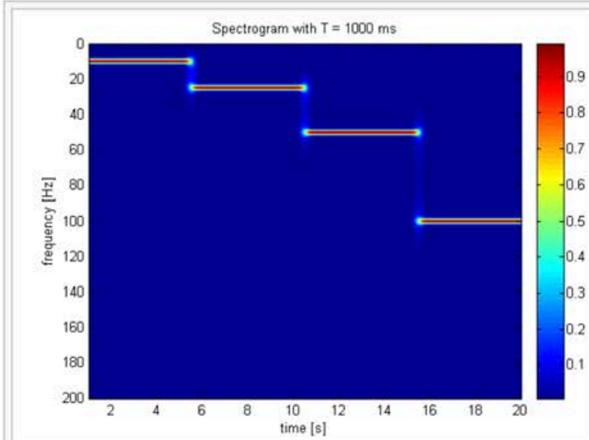
25 ms window



125 ms window



375 ms window



1000 ms window

A better way: Wavelet transform

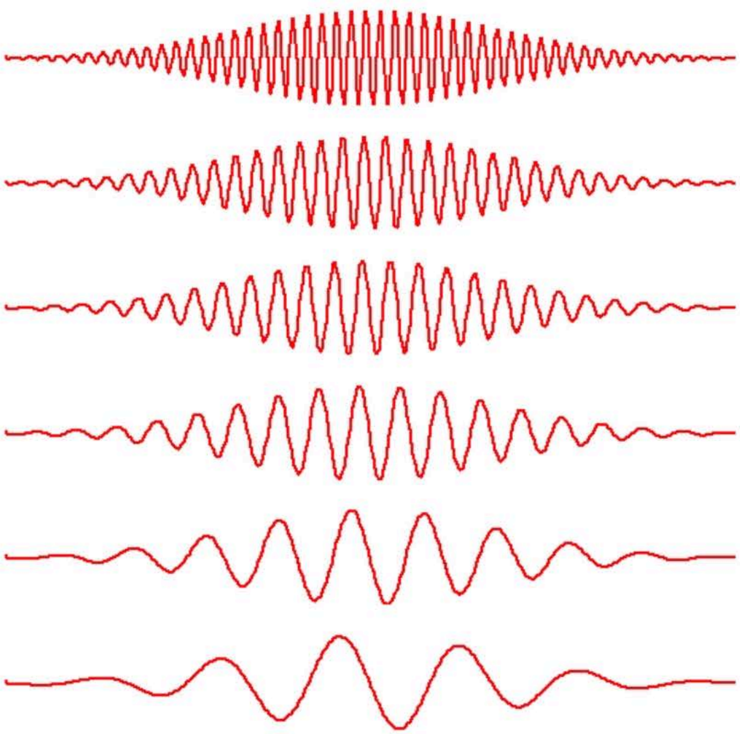
- Wavelet transform is a ‘multi-resolution’ time-frequency decomposition.

- Intuition: Higher frequency signals have a faster time scale

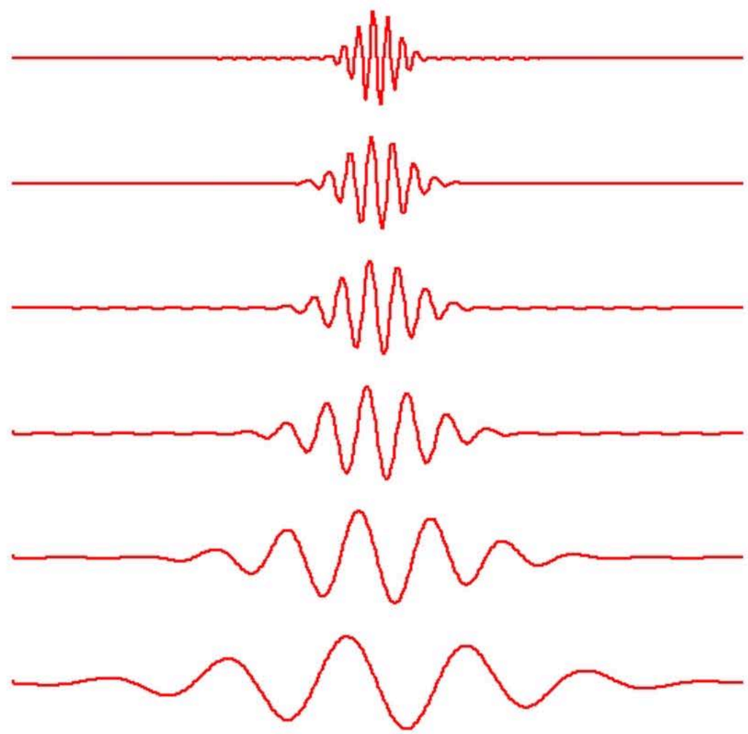
- So, vary window length with frequency!
  - longer window at lower frequencies
  - shorter window at higher frequencies

Comparison of FFT & Wavelet

FFT



Wavelet

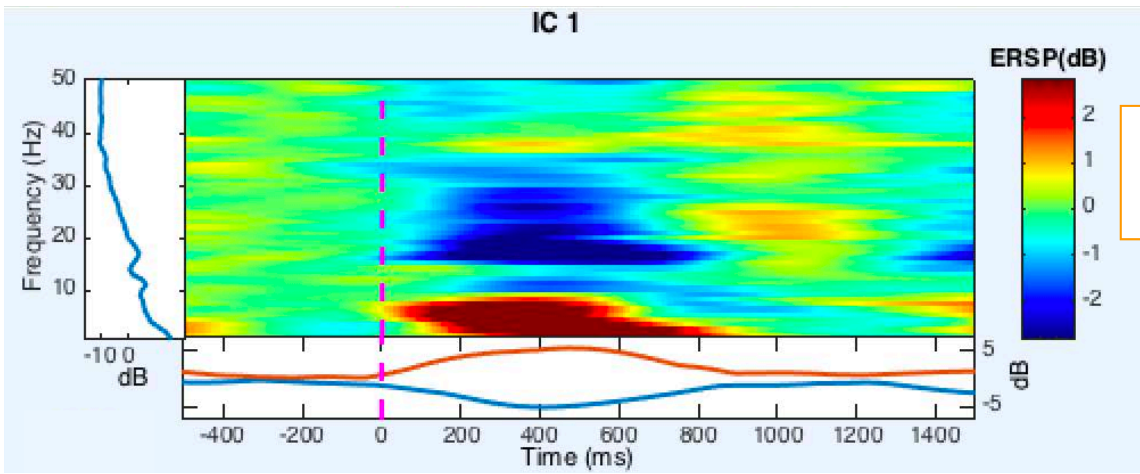


Scaled versions of one shape

Constant number of cycles

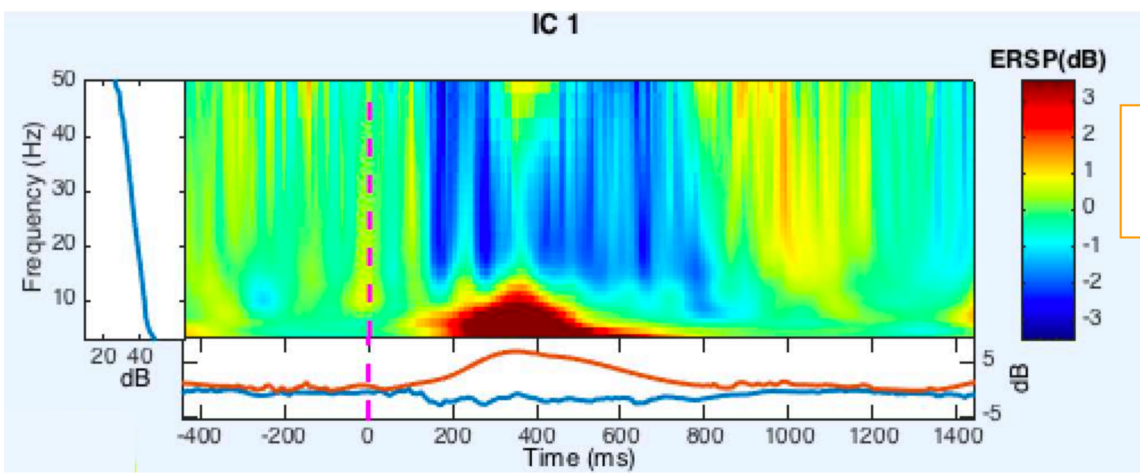
Comparison of FFT & Wavelet

FFT

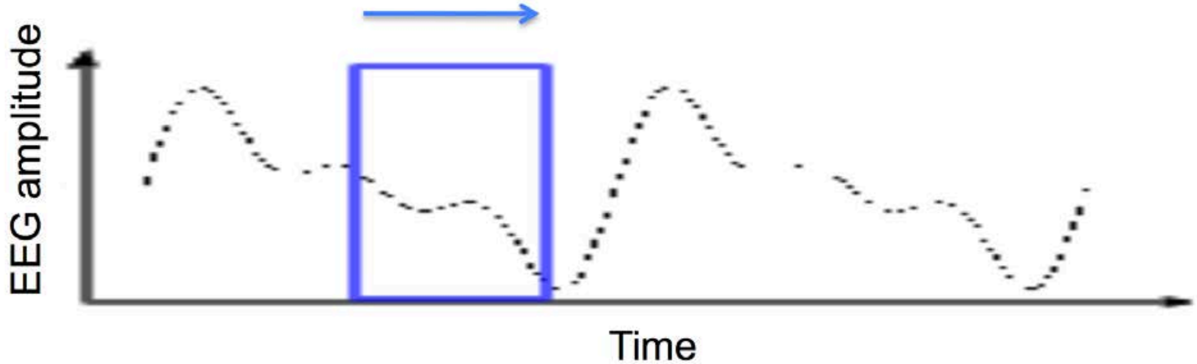


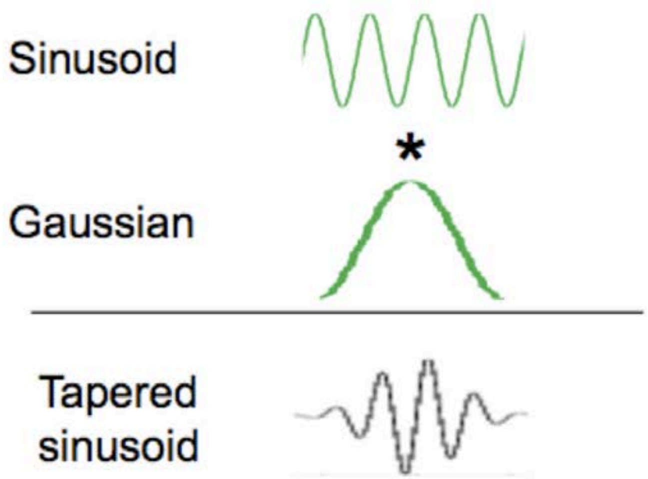
Similar time resolution across frequencies

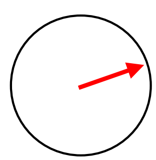
Wavelet



Finer time resolution at high frequencies







For each time point

Analyze signal using the wavelets

for different frequencies.

Exercise

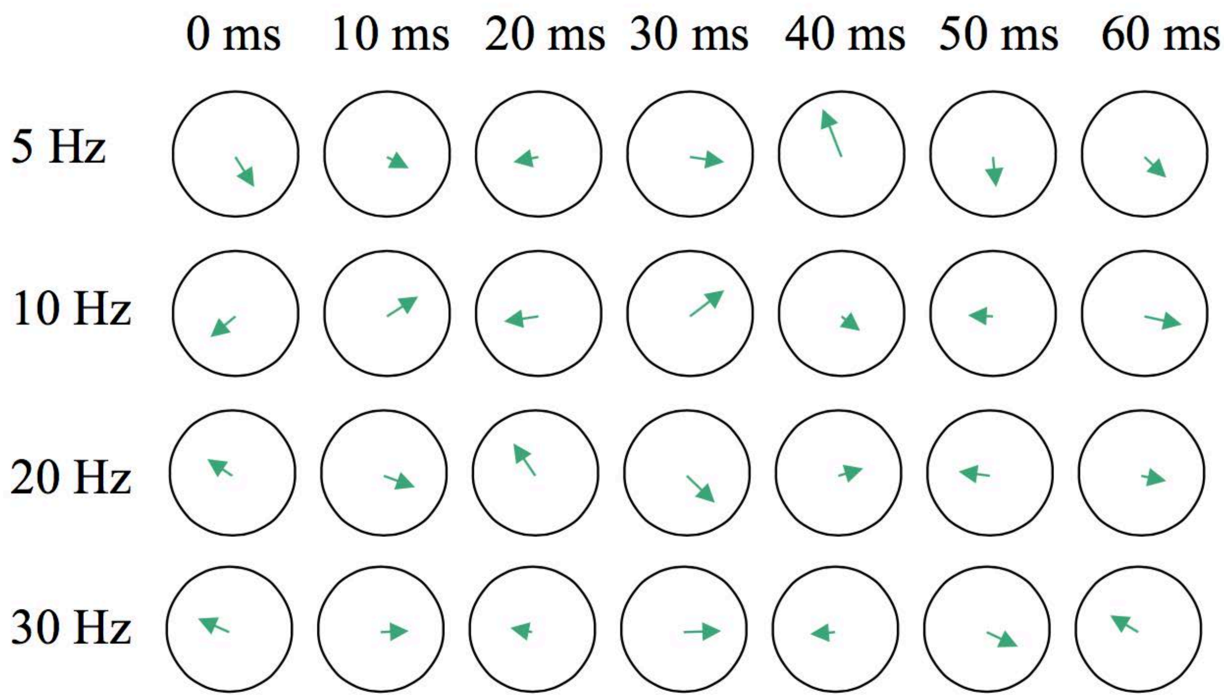
- Create a signal

```
>> t = 0:0.01:100;
>> x = sin(2*pi*10*t); plot(t,x)
```

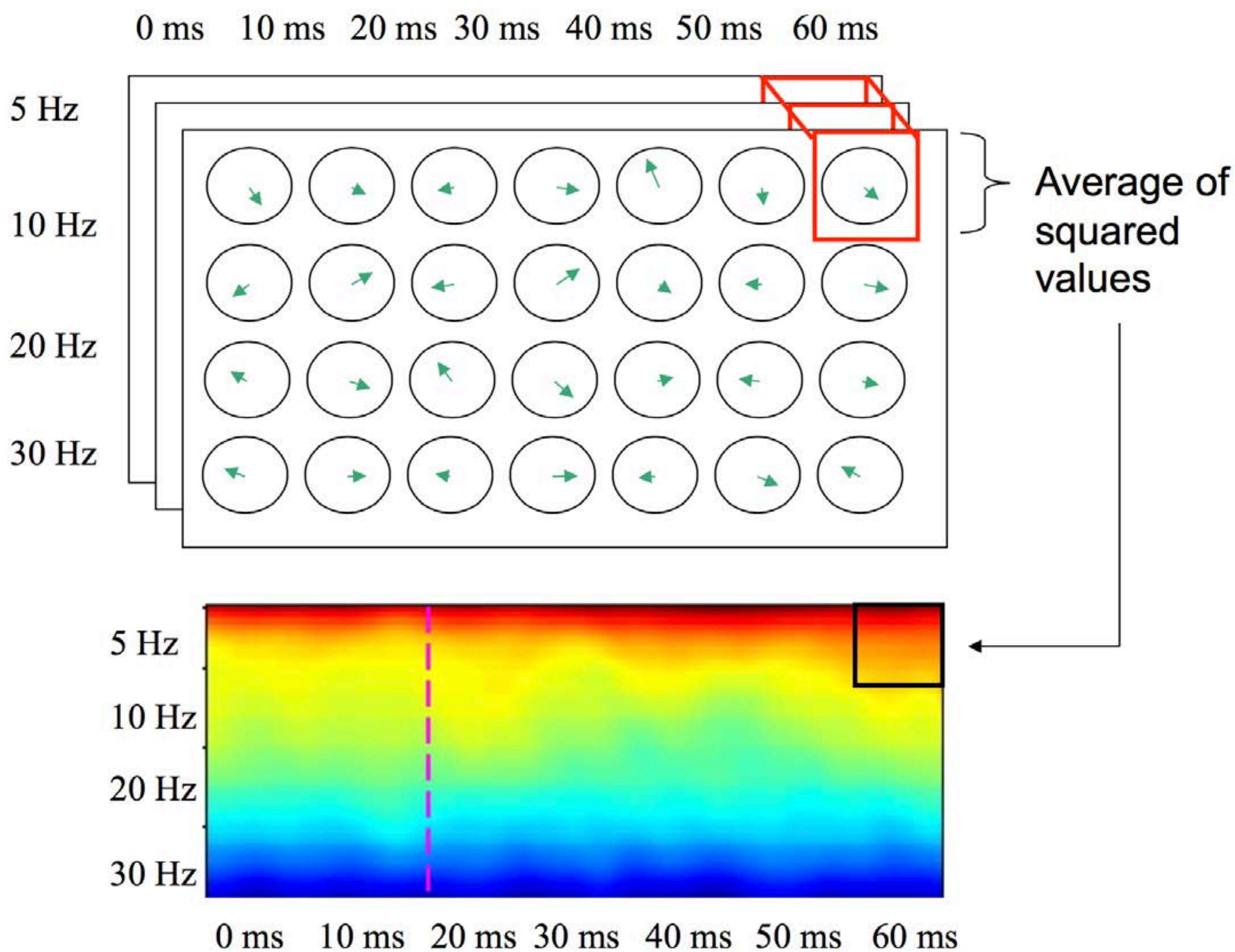
- Find FFT

```
>> F = fft(x);
>> F(1:3) %complex
>> power = F.*conj(F);
```

Spectrogram of one epoch of data



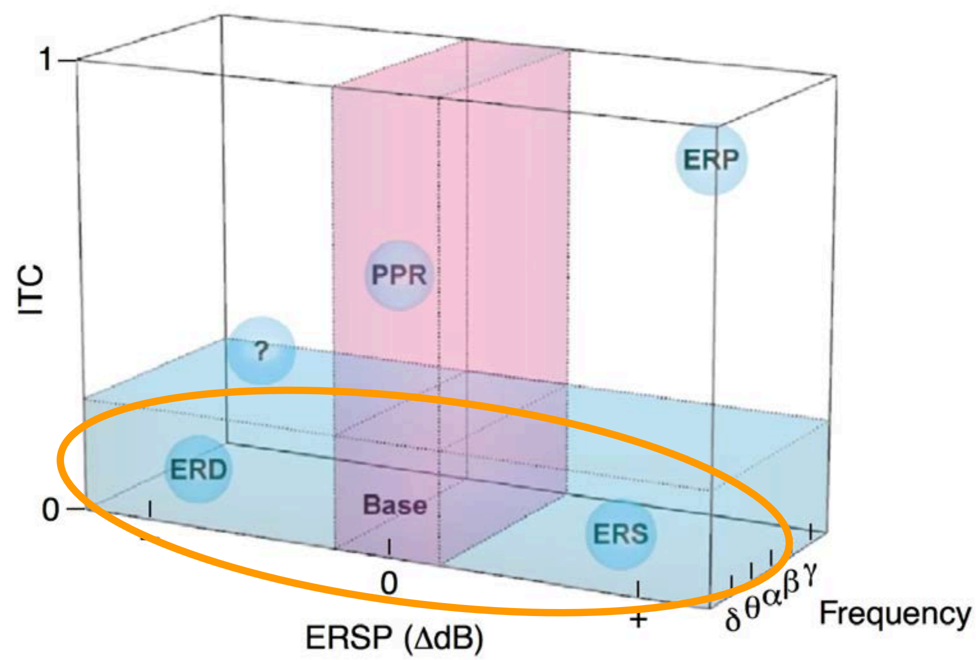
Computing Spectrogram Power



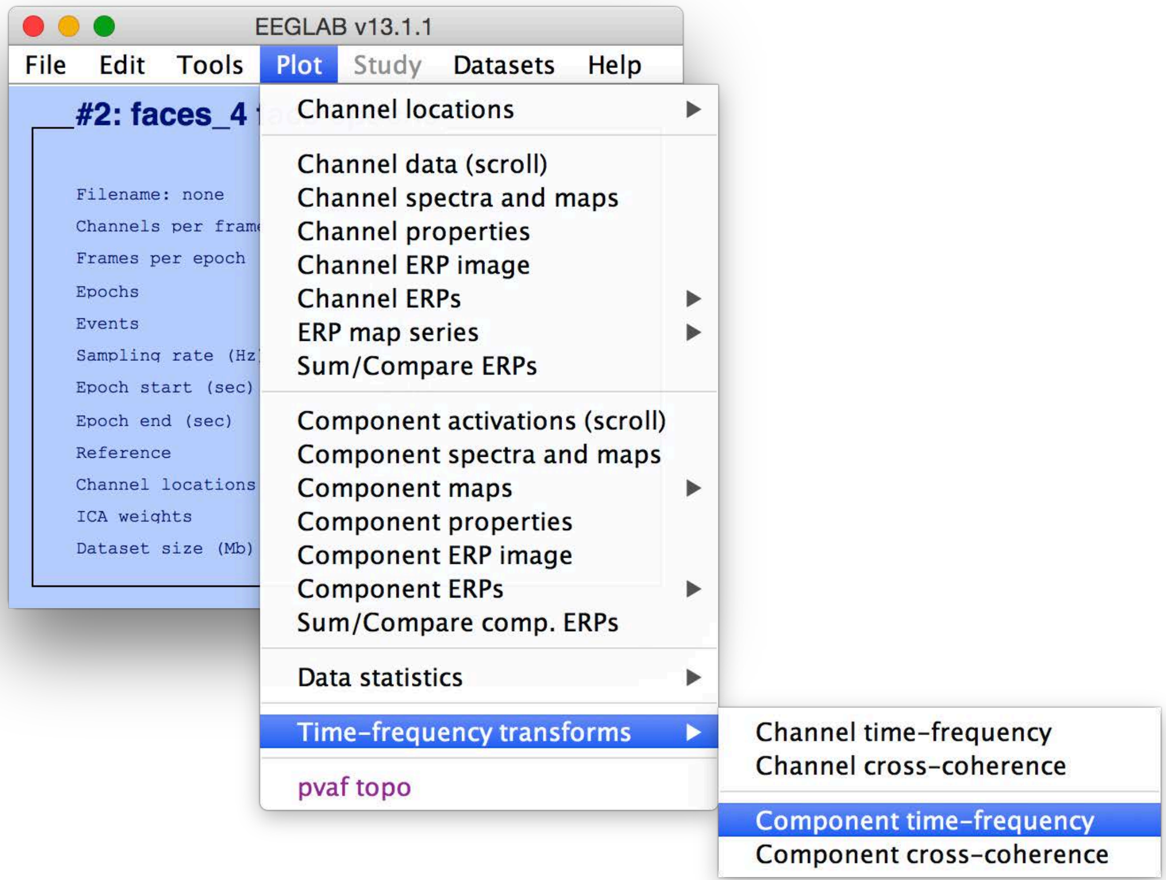
Definition: ERSP

- Event Related Spectral Perturbation

- Change in power in different frequency bands relative to a baseline. ERS (Event-Related *Synchronization*), ERD (Event-Related *Desynchronization*)



Try it out

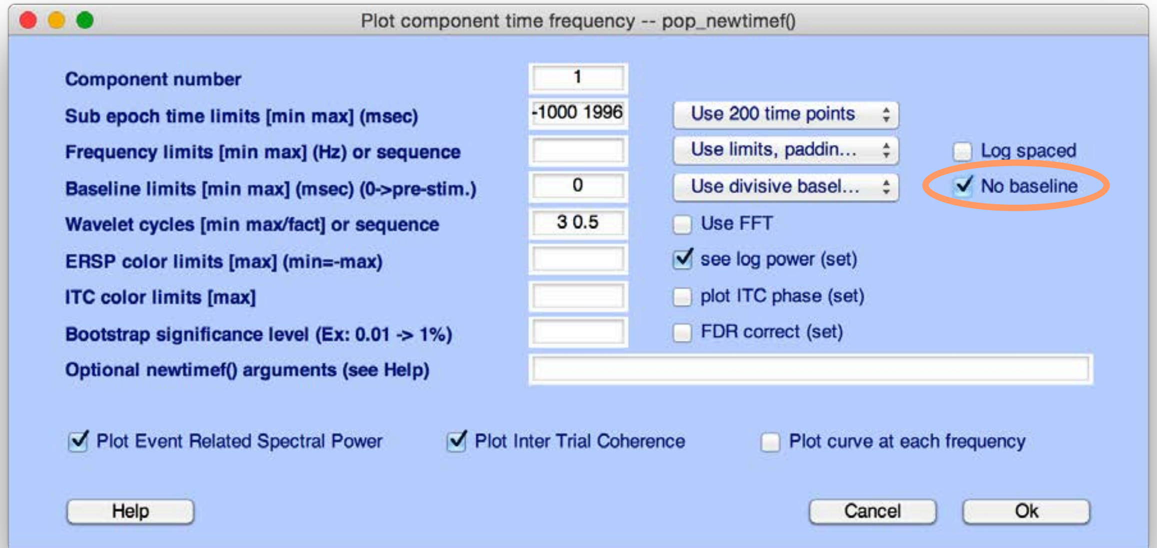


(Load faces_4.set
Epoch on 'face' event)

Display ERS vs. ERSP

Event-related

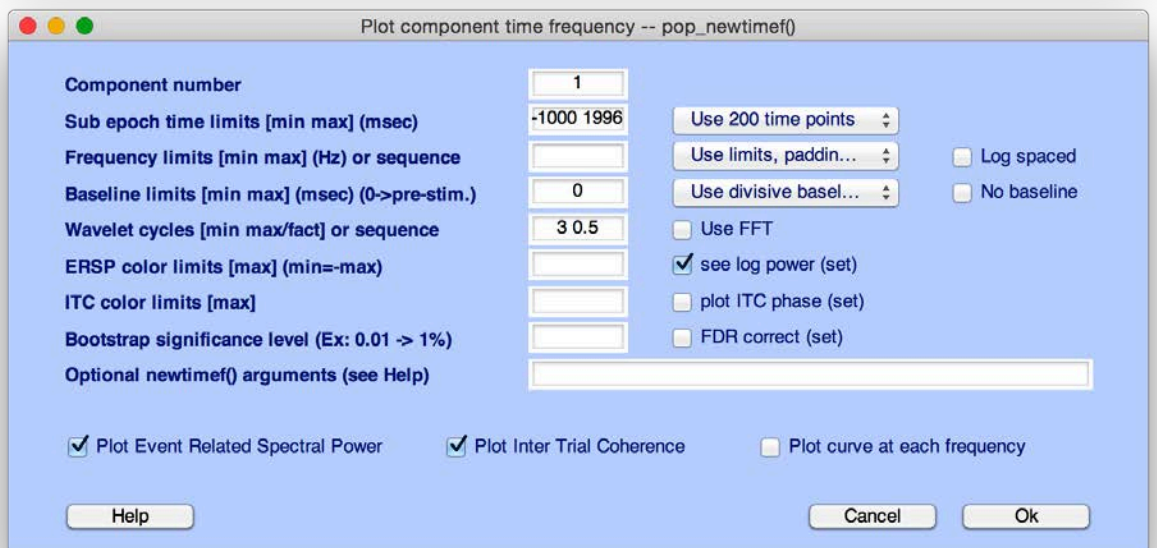
Spectrogram



Event-Related

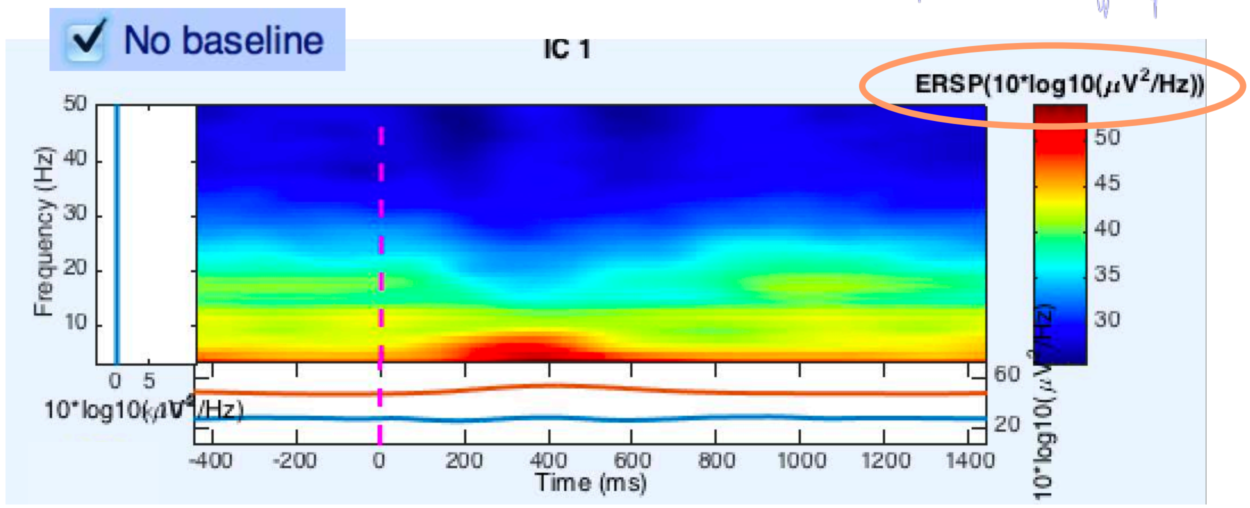
Spectral Perturbation

(ERSP)



Event-related

Spectrogram

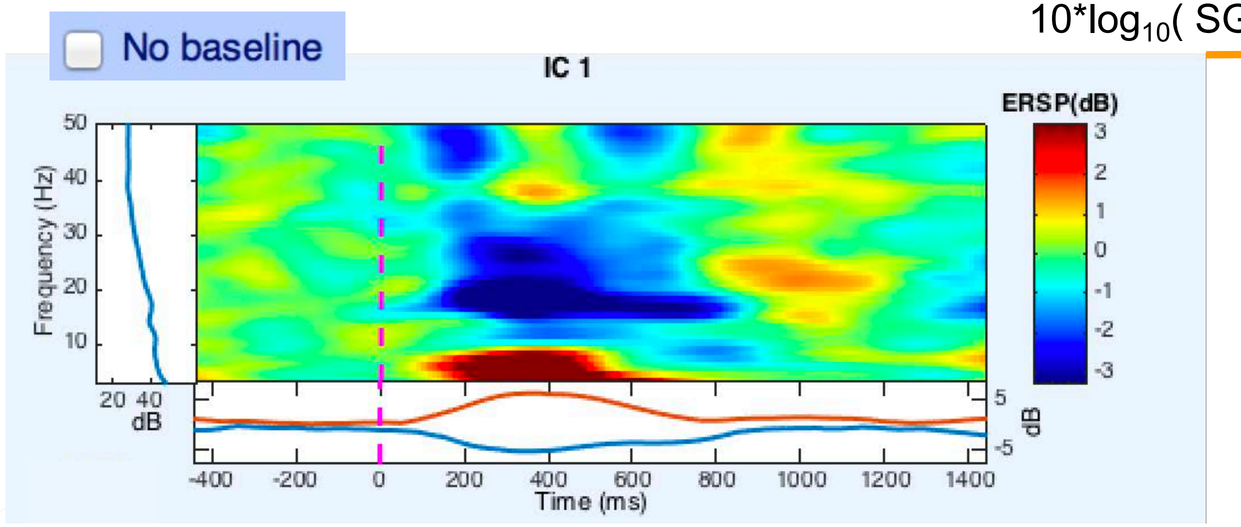


$10*\log_{10}(SG(t,f) / baseline(f))$

Event-Related

Spectral Perturbation

(ERSP)

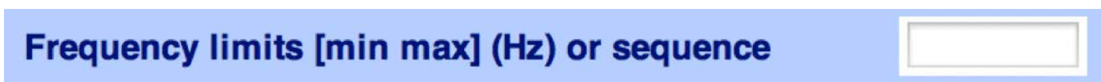


Exercises

- Try different wavelet specifications

  – Default: 3 0.5
    - 3 cycles. Try 2. How do the time limits of the plot change?
    - What is the 0.5? Try 0. Try 1…what do you observe?

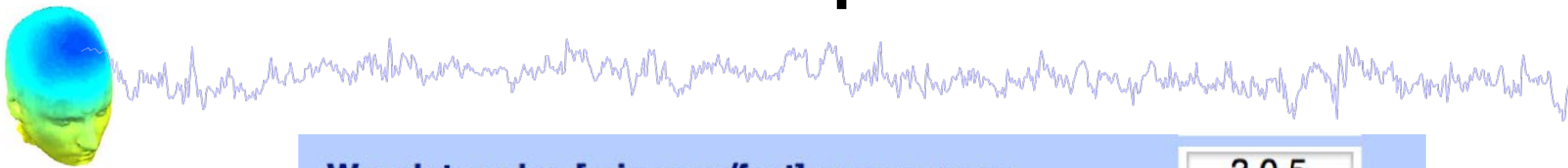
- Try different low-frequency limit



  – what is the effect on the time limits of the ERSP?

- Try different baseline methods
  – divisive
  – standard deviation (express spectral perturbations in #sd relative to baseline sd)

Wavelet Specification



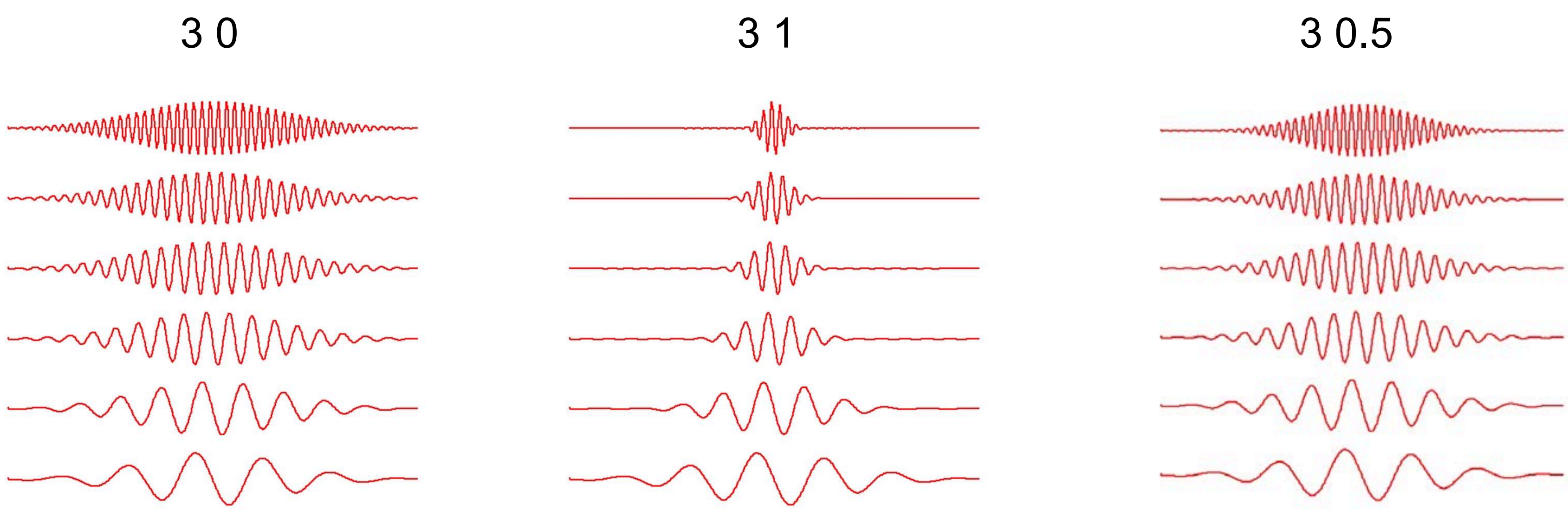
Answer: The first #cycles controls the basic duration of the wavelet in cycles.

The second factor controls the degree of shortening of time windows as frequency increases

0 = no shortening = FFT (duration remains constant with frequency)

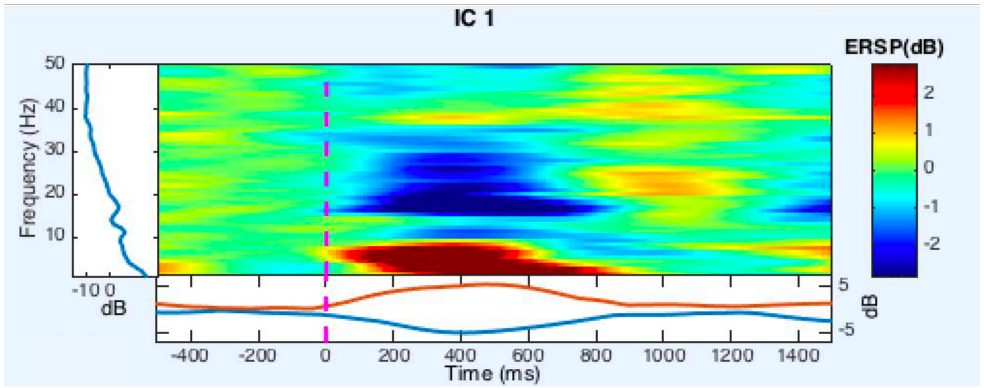
1 = pure wavelet (#cycles remains constant with frequency)

0.5 = intermediate, a compromise that reduces HF time resolution to gain more frequency resolution.

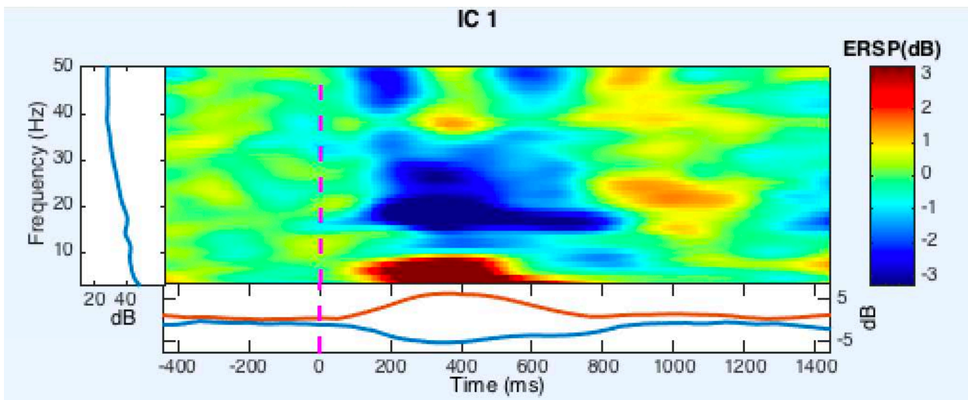


Comparison of FFT & Wavelet

[3 0] (FFT)



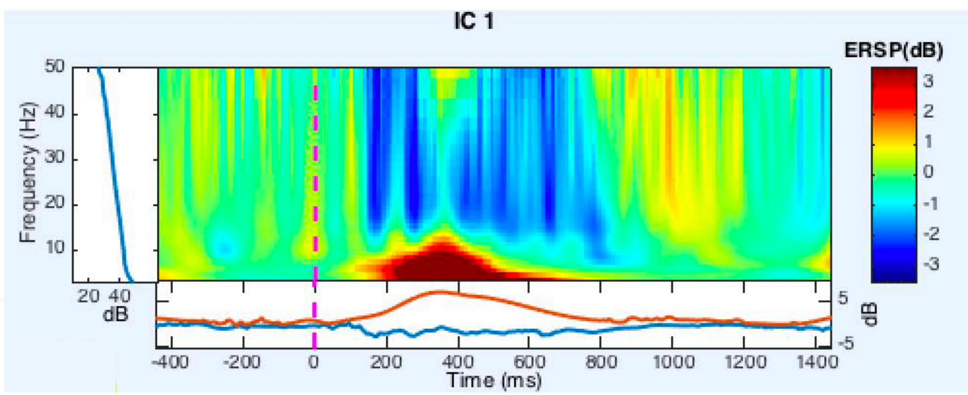
[3 0.5] Wavelet



This is the EEGLAB default

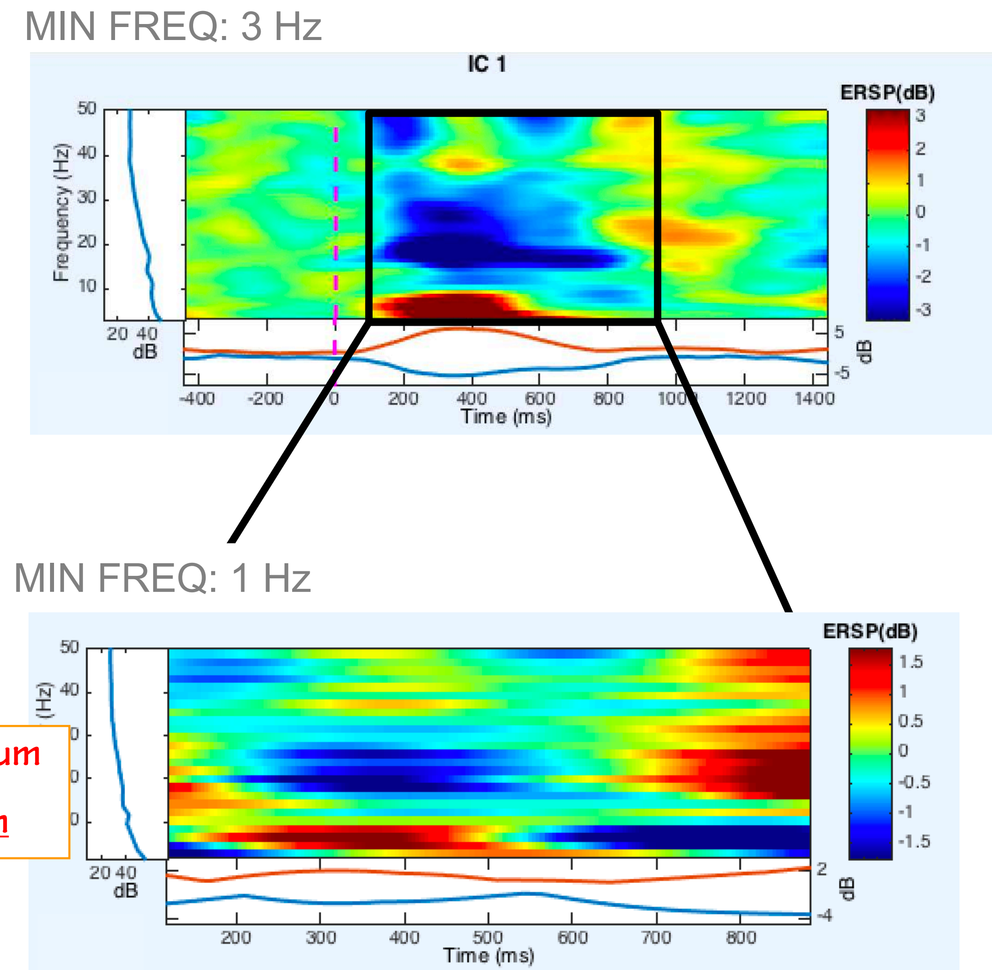
Notice: features have similar time and frequency resolution

[3 1] Wavelet



Time loss at edge of ERSP

- Settings for wavelet cycles and lowest frequency impact the time limits of analysis



*more wavelet cycles, or a lower minimum frequency loses time at edges of epoch

Solution: If you need low frequencies, be sure to extract longer epochs to counteract this. Barring this, try reducing the number of wavelet cycles.

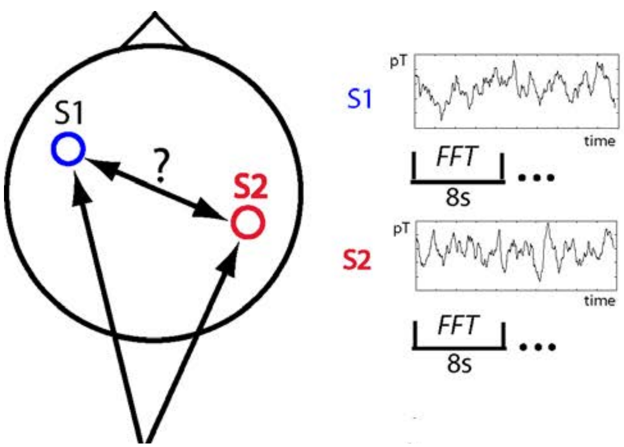
Part 3: Coherence Analysis

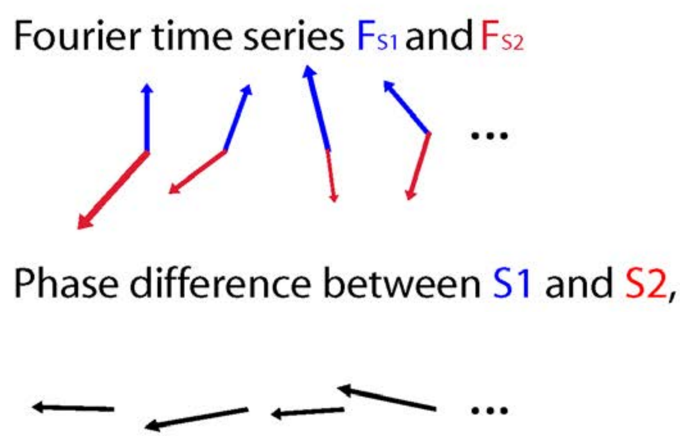
- Goal: How much do two signals resemble each other?
- Coherence = complex version of correlation: how similar are power and phase at each frequency?
- Variant: phase coherence (phase locking, etc.) considers only phase similarity, ignoring power
  - Regular coherence is simply a power-weighted phase coherence
  - Inter-trial coherence is useful!
- NOTE: For **understanding** connectivity between regions, *channel* coherence is a poor choice due to volume conduction. For IC connectivity, directional, 'causal' measures of connectivity have been developed (See SIFT lecture).

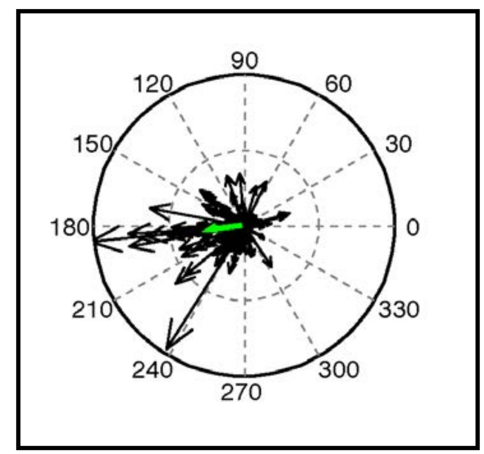
Coherence

$$C(f,t) \propto \sum_{k=trials} F1_k(f,t)\overline{F2_k(f,t)}$$

$$a_1 e^{i\theta_1} a_2 e^{-i\theta_2} \propto e^{i(\theta_1 - \theta_2)}$$







Part 3a: Inter-Trial Coherence

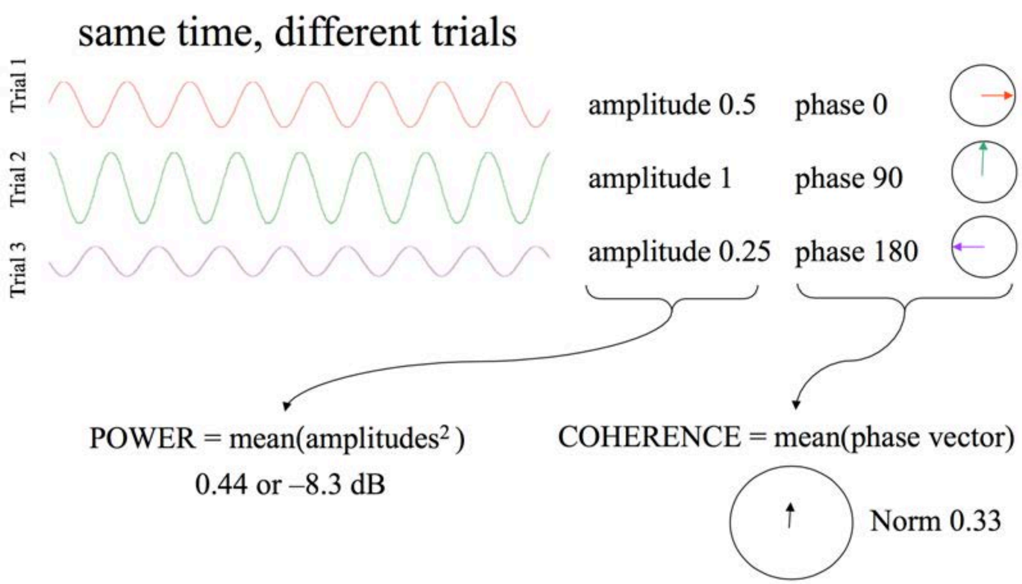
- Goal: How much do different trials resemble each other?
- Phase coherence not between two processes, but between multiple trials of the same process
- Defined over a (generally) narrow frequency range

EEGLAB's Inter-Trial Coherence is *phase* ITC

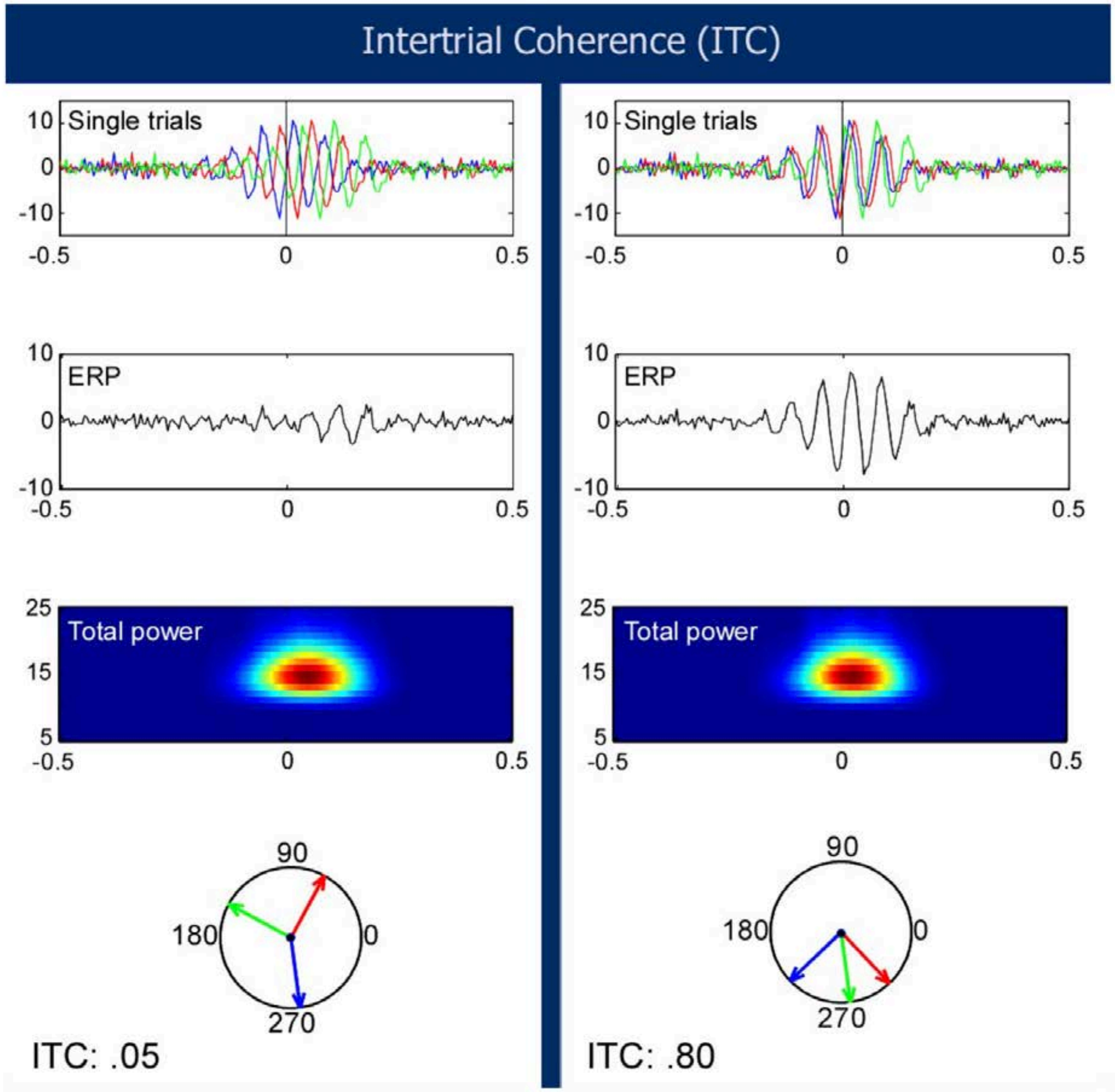
Phase ITC

$$ITPC(f,t) = \frac{1}{n}\sum_{k=1}^{n}\frac{F_k(f,t)}{|F_k(f,t)|}$$

Normalized
(no amplitude information)



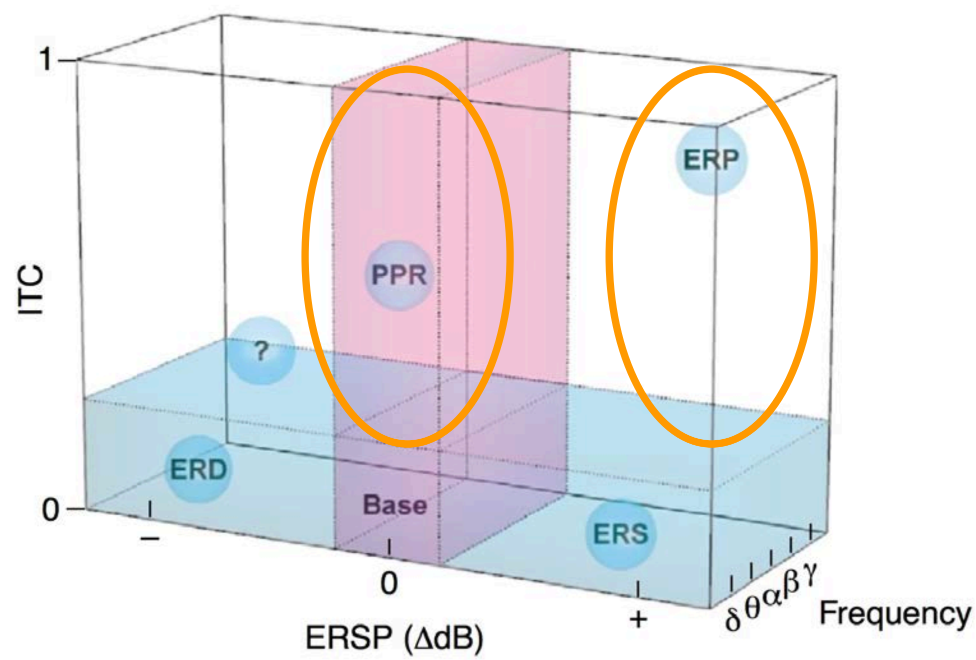
ITC Example (3 trials)

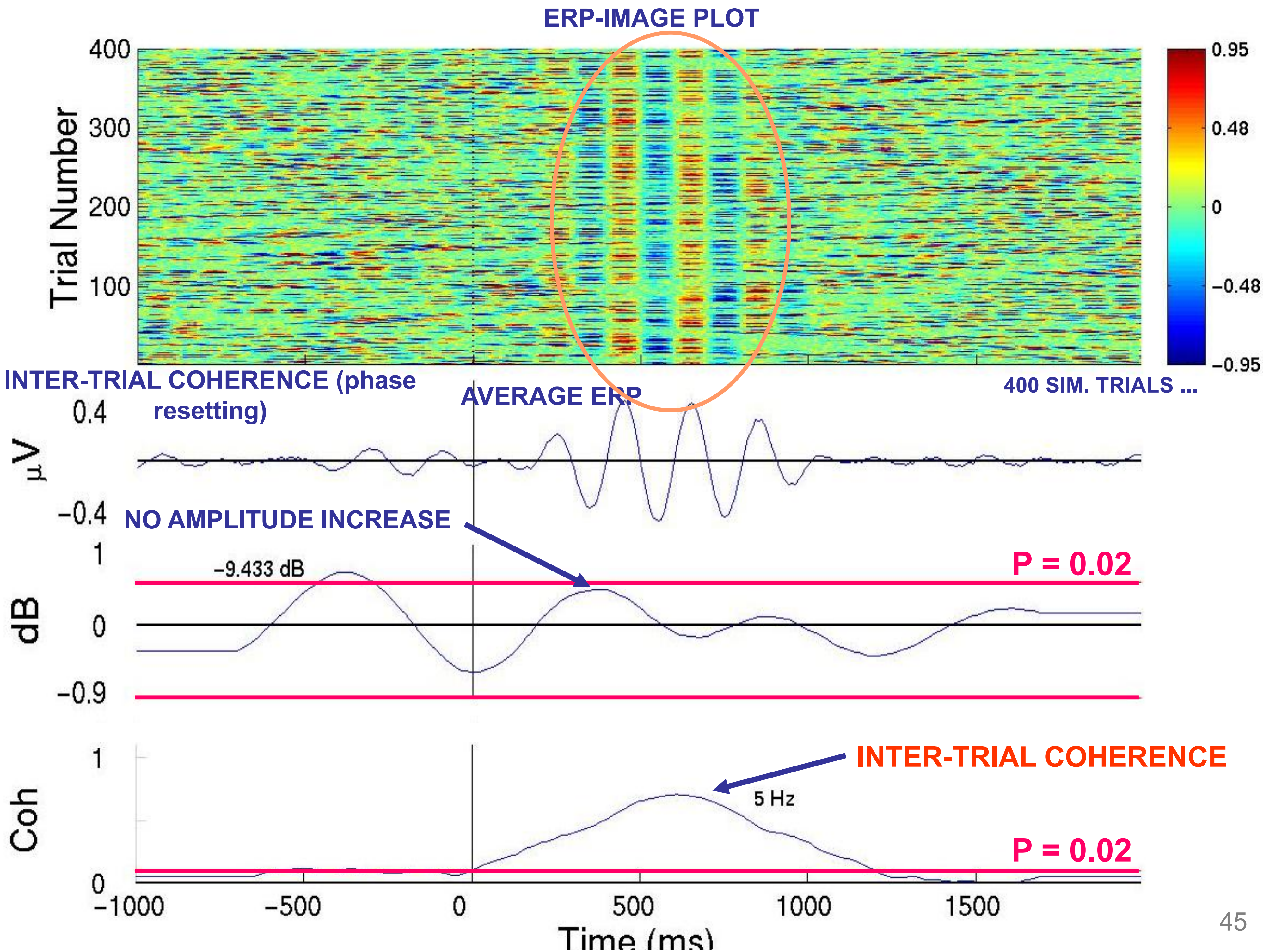


Slide courtesy of Stefan Debener

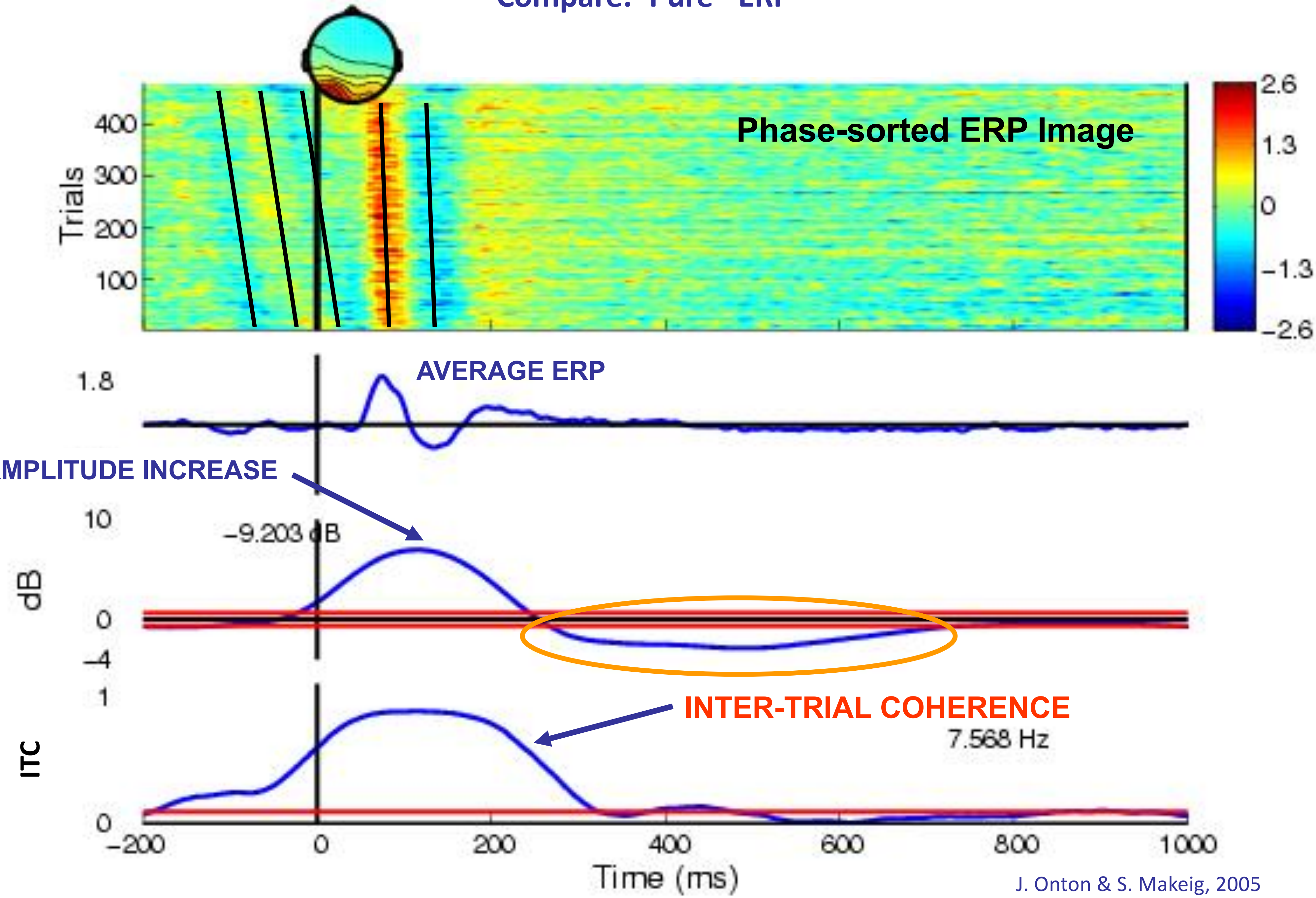
Several possible origins of an ERP

- Event Related Potential can result from
  - ITC increase (with no change in power)
  - ITC & Power change



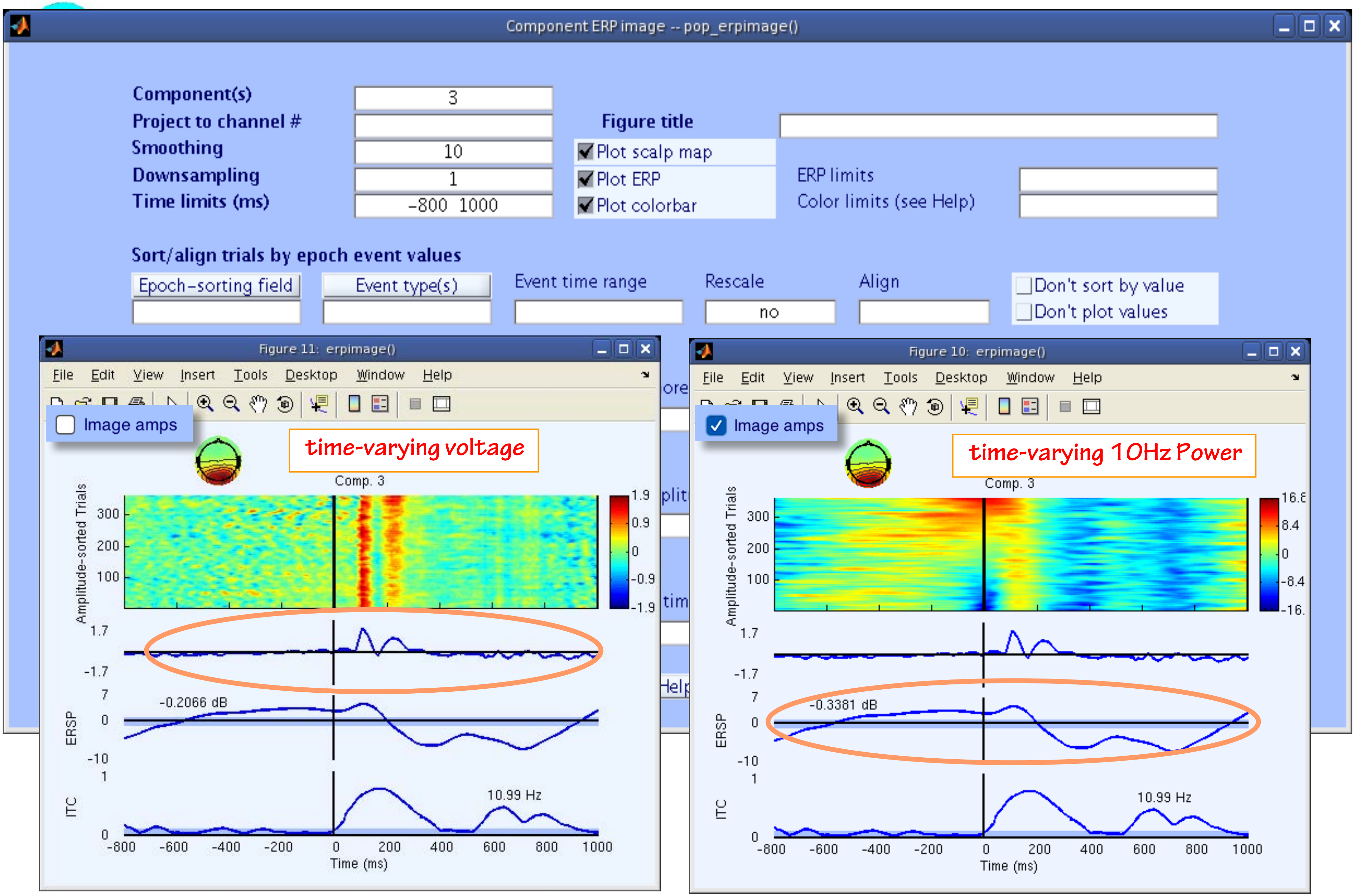

ERP-IMAGE PLOT
Trial Number
400
300
200
100
0.95
0.48
0
−0.48
−0.95
INTER-TRIAL COHERENCE (phase resetting)
AVERAGE ERP
400 SIM. TRIALS ...
0.4
μV
−0.4
NO AMPLITUDE INCREASE
1
−9.433 dB
P = 0.02
dB
0
−0.9
1
INTER-TRIAL COHERENCE
5 Hz
Coh
P = 0.02
0
−1000
−500
0
500
1000
1500
Time (ms)

Compare: ‘Pure’ ERP

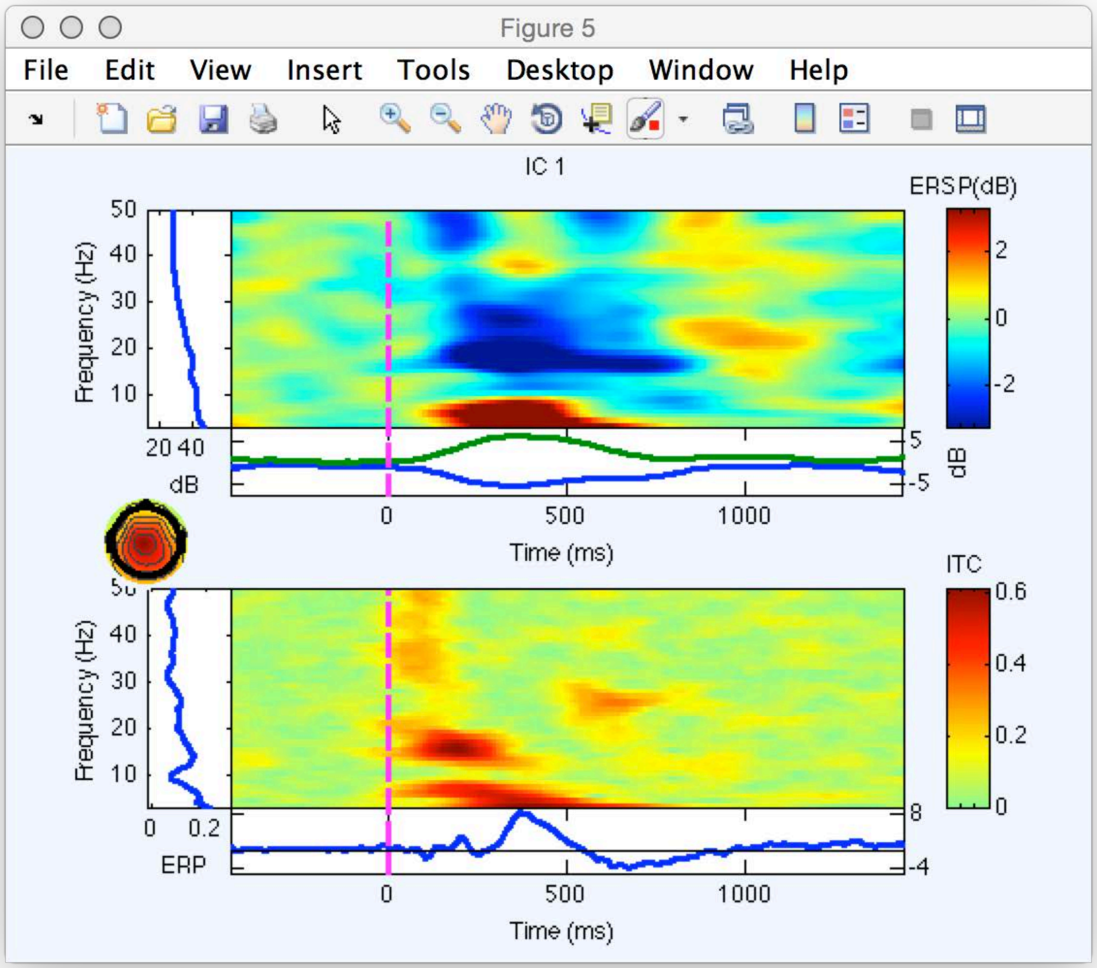


J. Onton & S. Makeig, 2005

Component ERP Image: Activation vs. Amplitude



Putting it all together



Exercise

All: Compute ERSP/ITC for a component of your choice

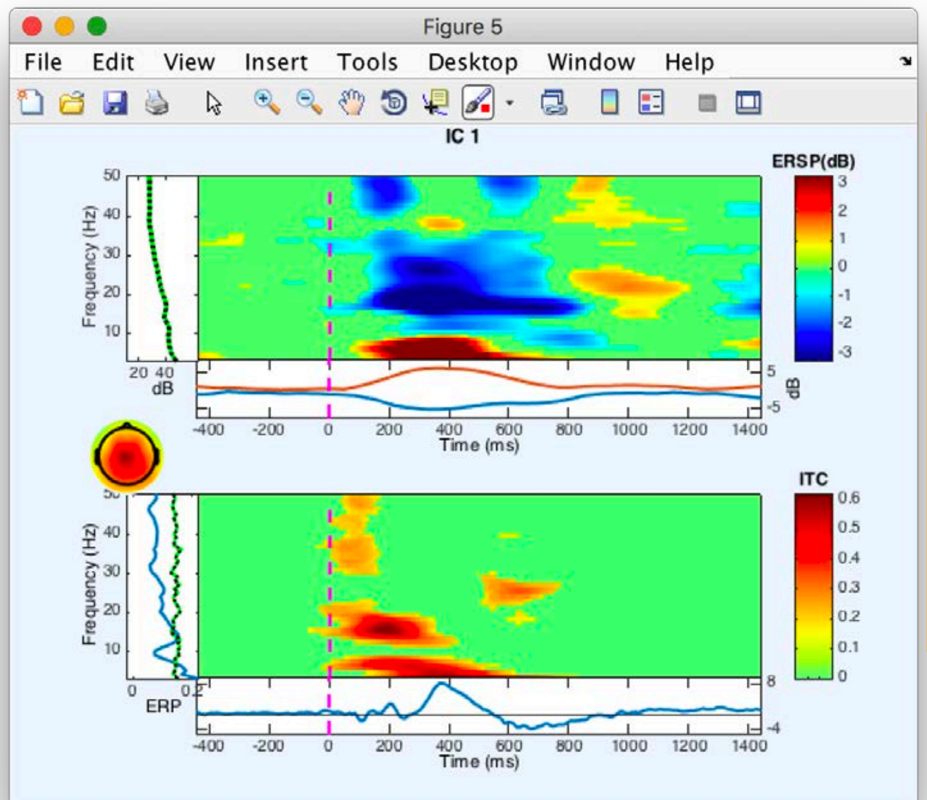
Compute ERP Image (with ERSP and ITC displayed*)

Use all of this information to explain the origin of the Evoked Response

Question: Which changes are significant? Use the options in ERP Image and ERSP dialogs to set significance threshold e.g. 0.01. Do the results survive?

Significance Testing

- Keep in mind: "is this significant?"



Method: Bootstrap

Green areas are not significant.

Scale of ERSP & ITC vales also give a clue:

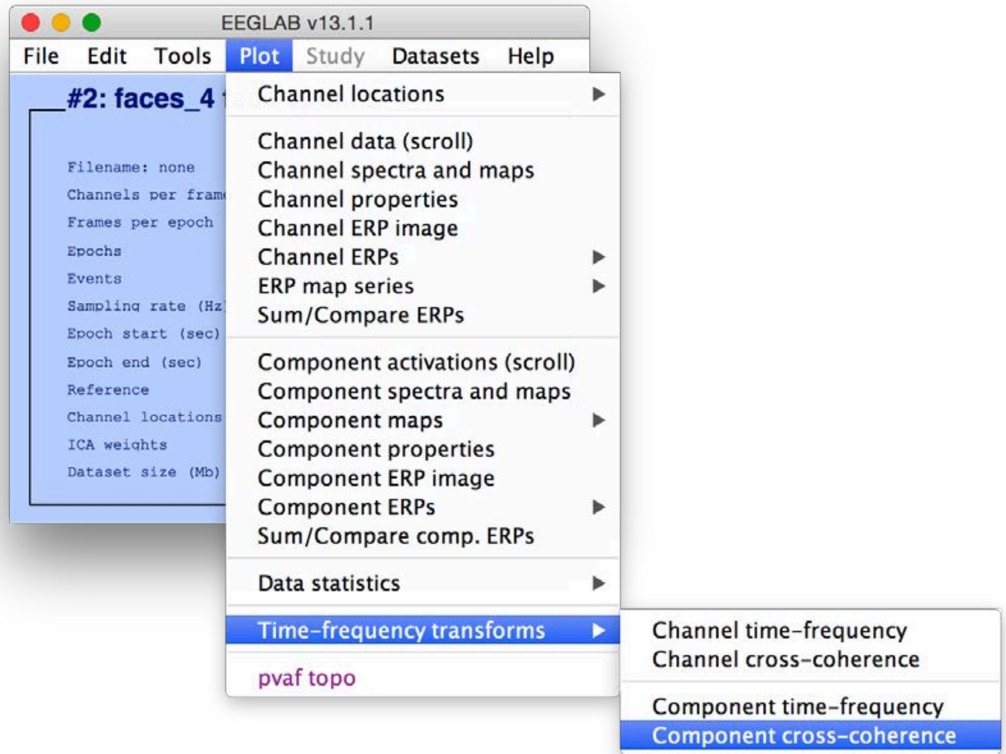
Large values are often encouraging of a significant effect

(Large ≈ > 1dB for ERSP; > 0.5 for ITC)

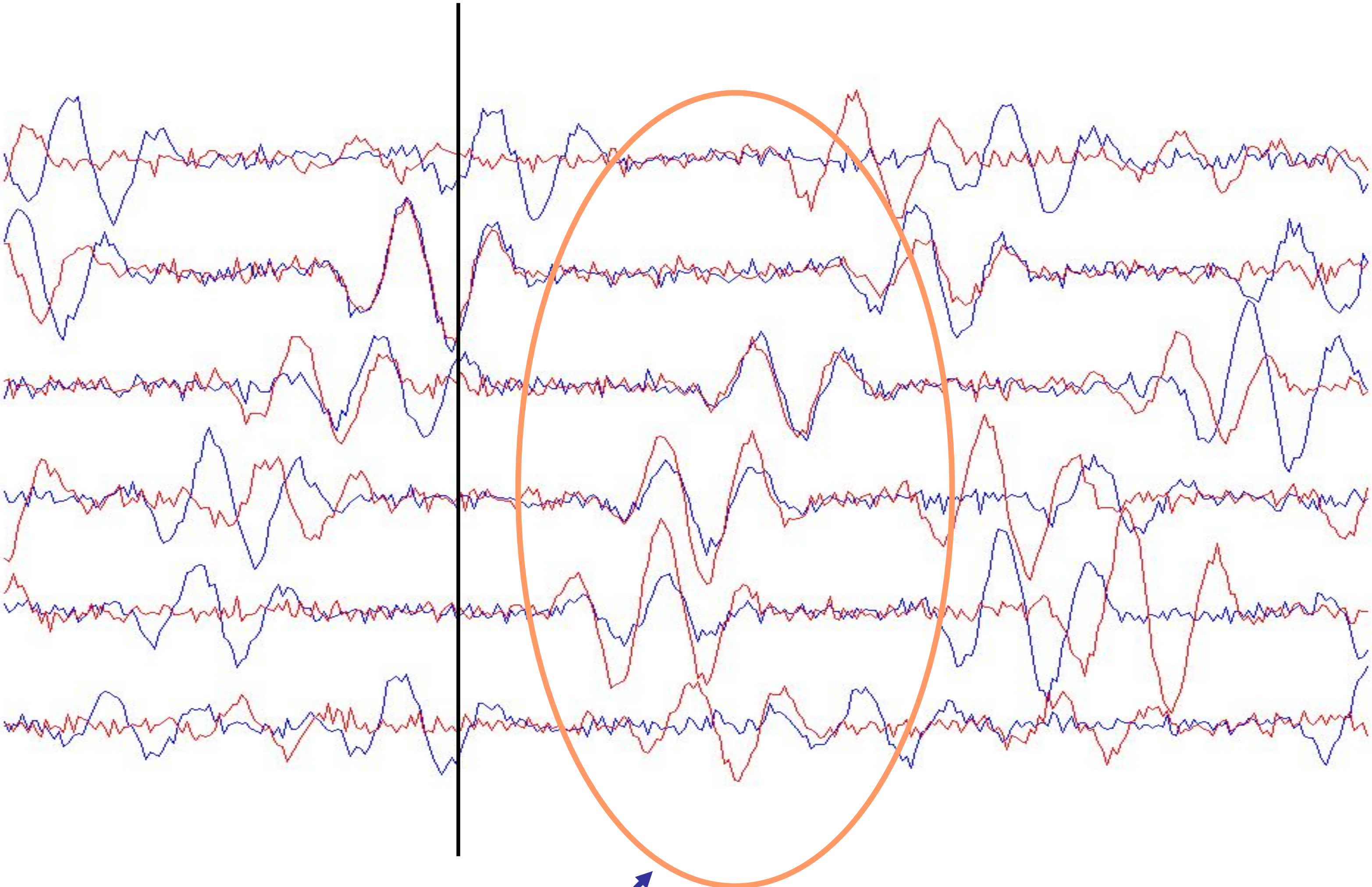
For exploratory purposes, can try 0.01 without FDR correction

Part 3b: Event Related Coherence

- Goal: How similar is the event-related response of two signals?
  - Between channels (problematic due to volume conduction)
  - Between ICs
  - Useful to quickly begin to understand relationships between components
  - SIFT provides more complete solution

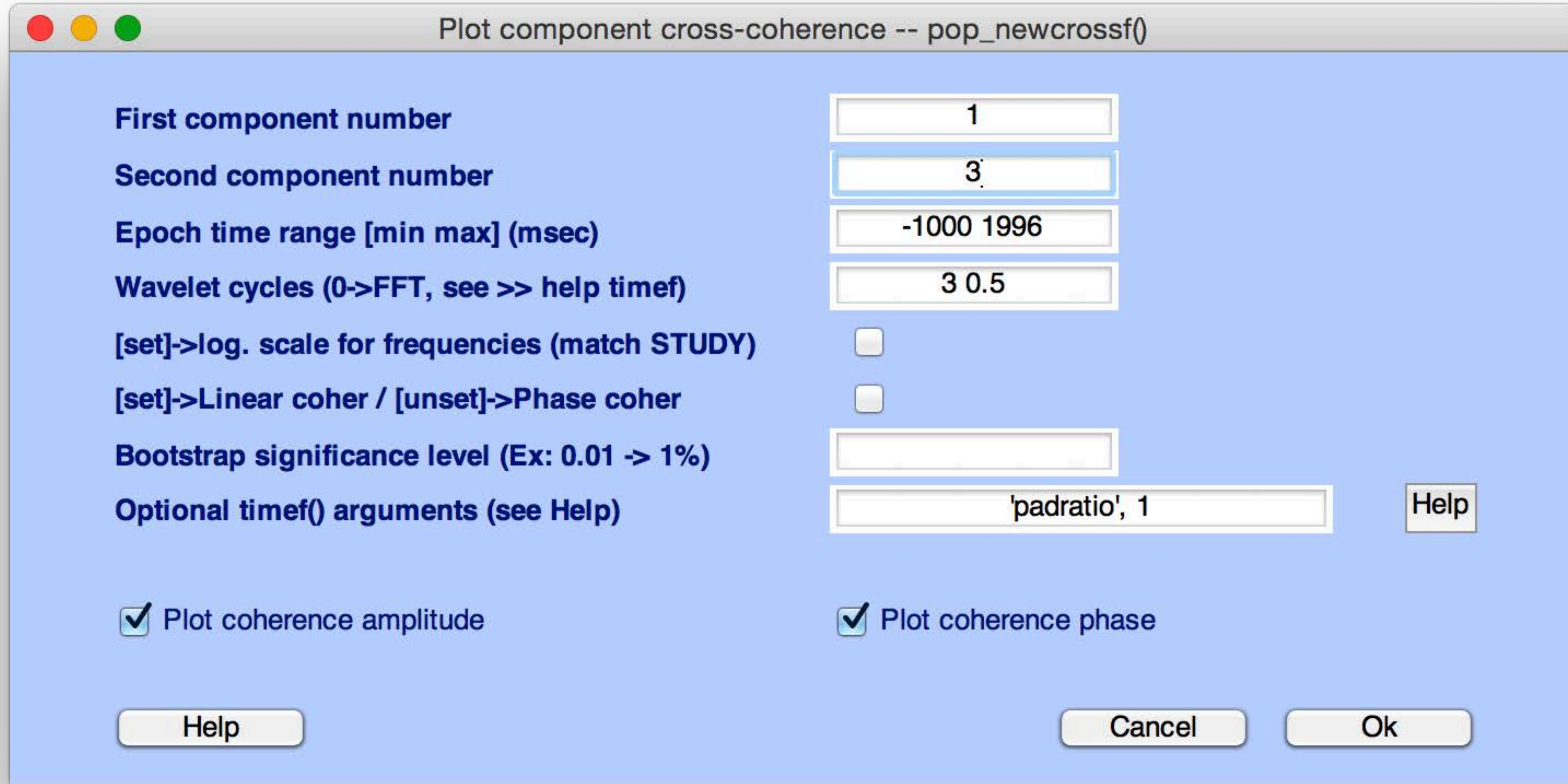


TWO SIMULATED THETA PROCESSES



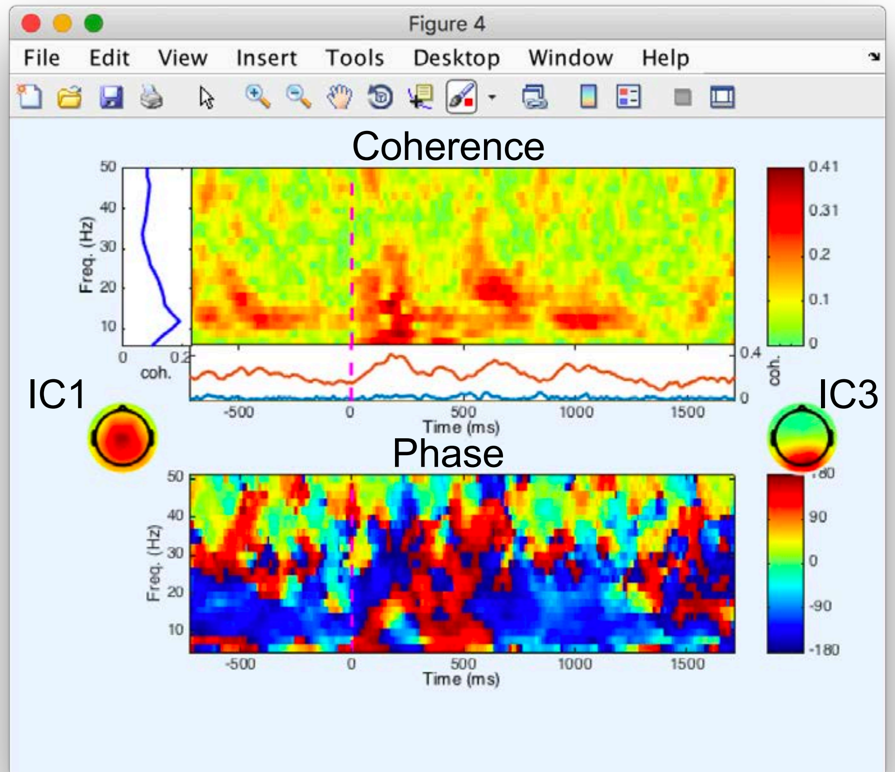
Event-related Coherence

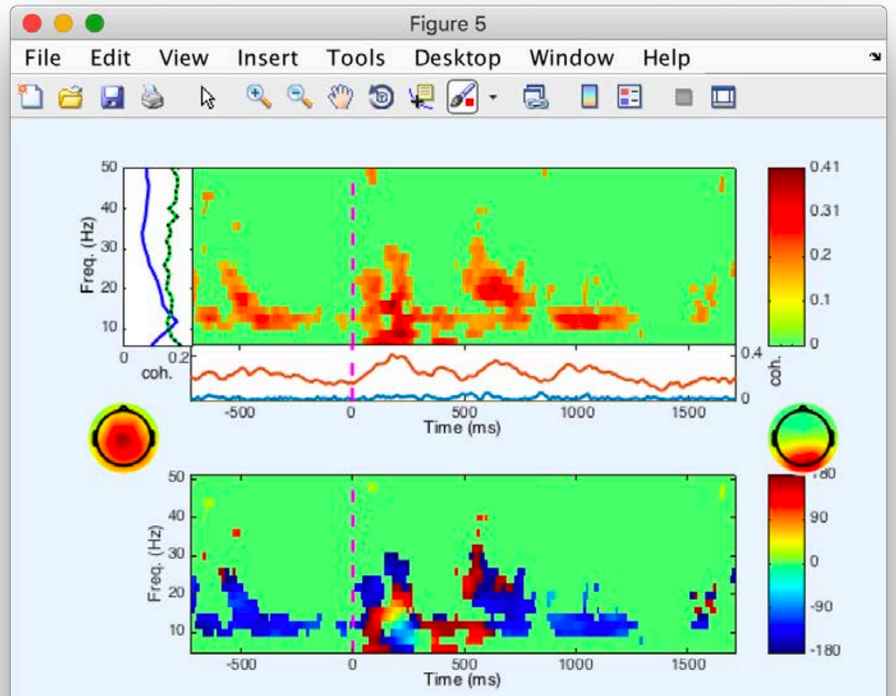
Try it!


Plot component cross-coherence -- pop_newcrossf()
First component number
1
Second component number
3
Epoch time range [min max] (msec)
-1000 1996
Wavelet cycles (0->FFT, see >> help timef)
3 0.5
[set]->log. scale for frequencies (match STUDY)
[set]->Linear coher / [unset]->Phase coher
Bootstrap significance level (Ex: 0.01 -> 1%)
Optional timef() arguments (see Help)
'padratio', 1
Help
Plot coherence amplitude
Plot coherence phase
Help
Cancel
Ok

Cross coherence between IC 1 and IC 3

$\alpha = 0.01$





Significant event-related coherence (as well as tonic coherence) in alpha/beta bands

IC 1 tonically leads IC 3 (negative phase), but phase relationships are changed post-stimulus

More advanced, directional, measures of effective connectivity are present in the SIFT toolbox (a later lecture).

Event-Related Coherence Exercise

- Examine event-related coherence between two ICs
  - Which pair did you pick, and why? What do you predict?
  - What did you learn?

- Explore other options:
  - Significance threshold
  - Figure out how to subtract a baseline
  - Phase vs. Linear Coherence

Possible fix to enable significance testing

topoplot.m | pop_newcrossf.m | newcrossf.m

```
end;

% compute epoch limits
% --------------------
if isempty(tlimits)
    tlimits = [EEG.xmin, EEG.xmax];
end;
pointrange1 = round(max((tlimits(1)/1000-EEG.xmin)*EEG.srate, 1));
pointrange2 = round(min((tlimits(2)/1000-EEG.xmin)*EEG.srate, EEG.pnts));
pointrange = [pointrange1:pointrange2];

% call function sample either on raw data or ICA data
% ---------------------------------------------------
if typeproc == 1
    tmpsig1 = EEG.data(num1,pointrange,:);
    tmpsig2 = EEG.data(num2,pointrange,:);
else
    if ~isempty( EEG.icasphere )
        eeglab_options; % changed from eeglaboptions 3/30/02 -sm
        tmpsig1 = eeg_getdatact(EEG, 'component', num1, 'samples',pointrange);
        tmpsig2 = eeg_getdatact(EEG, 'component', num2, 'samples',pointrange);
    else
        error('You must run ICA first');
    end;
end;

% JRI 1/15/17 Needed to comment these to be able to do significance testing.
% tmpsig1 = reshape( tmpsig1, 1, size(tmpsig1,2)*size(tmpsig1,3));
% tmpsig2 = reshape( tmpsig2, 1, size(tmpsig2,2)*size(tmpsig2,3));

% outputs
% -------
outstr = '';
if ~popup
    for io = 1:nargout, outstr = [outstr 'varargout{' int2str(io) '},' ]; end;
    if ~isempty(outstr), outstr = [ '[' outstr(1:end-1) '] =' ]; end;
end;

% plot the datas and generate output command
```