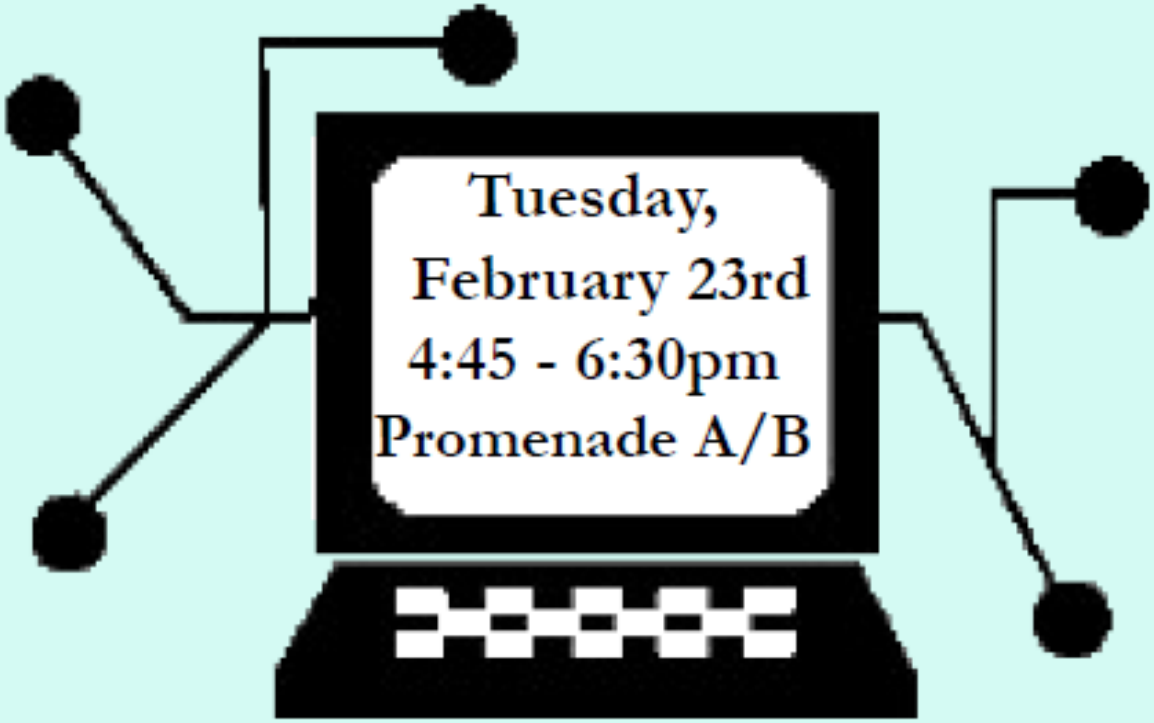
Research and Teaching Tools for MATLAB

A Community Discussion

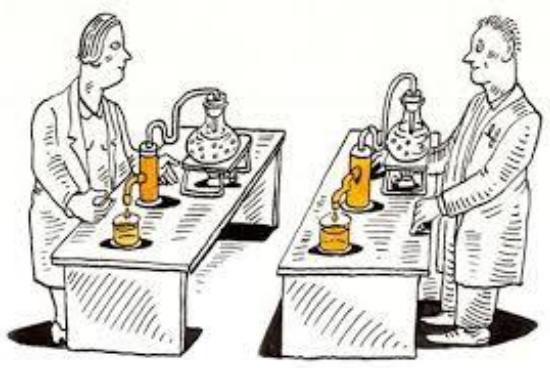


Purpose

What are the community needs for the efficient sharing of MATLAB tools and resources for research and teaching?

Why?

- Less duplication of effort across labs & teaching resources
- Rigor and reproducibility of research (new NIH requirement for grants)
- Improve learning experience for students
- Maybe have a more focused location for MATLAB research and teaching resources for the ARO community rather than Google



How?

- Talks to highlight how MATLAB is being used for research/teaching and what tools are available:
  - Michael Heinz
  - Ray Goldsworthy
  - Denis Drennan
  - Piotr Majdak
  - Laurel Carney

How?

- Lisa Kempler will share about what MATLAB has to offer for collaboration and promoting sharing of tools and resources
- Discussion on how to improve awareness/sharing of already available resources?

How?

Survey

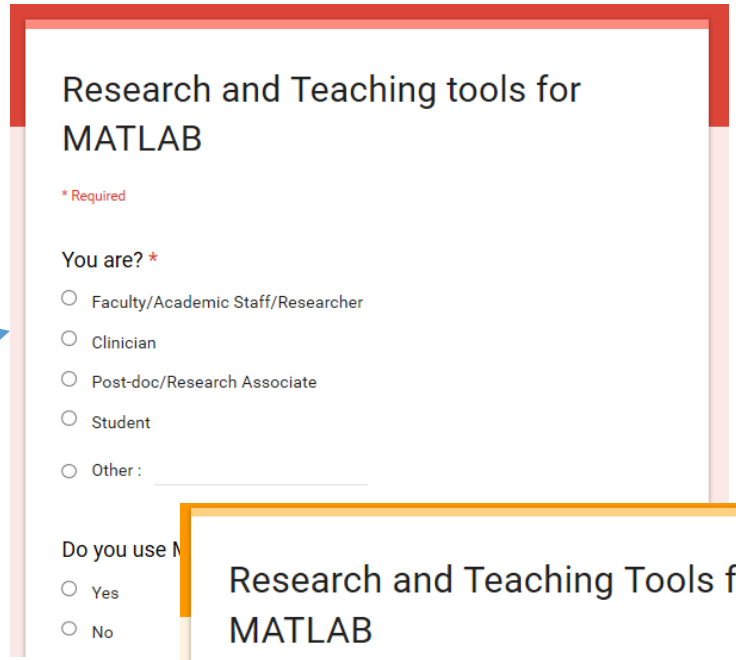
(Please fill in once)

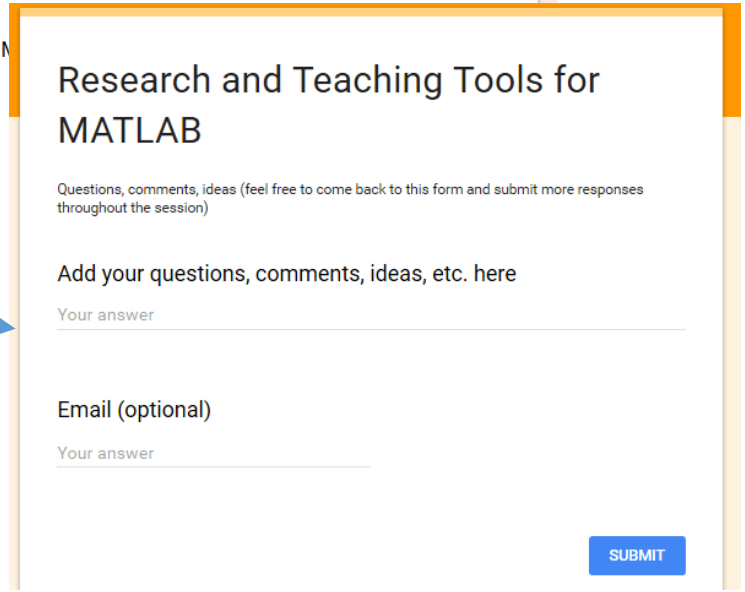
http://tinyurl.com/ARO2016MATLABSurvey

Comments/Questions/Ideas

(Fill in as many times as you like)

http://tinyurl.com/ARO2016MATLABComments





Uses of *MATLAB* for teaching and research in audiology and auditory neuroscience

***Michael G. Heinz*[1,2]**

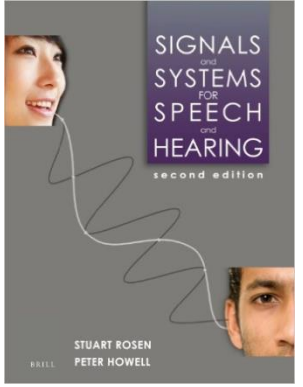
[1] Dept. of Speech, Language, and Hearing Sciences
&
[2] Weldon School of Biomedical Engineering,
Purdue University, West Lafayette, IN, USA

Outline

- ***Teaching***
  - ***Audiology (AUD) students***
    - ***Signal Processing for Hearing and Speech course***
  - Undergraduate SLHS students
    - Hearing Science course
  - Graduate BME students
    - Biomedical Signal Processing course

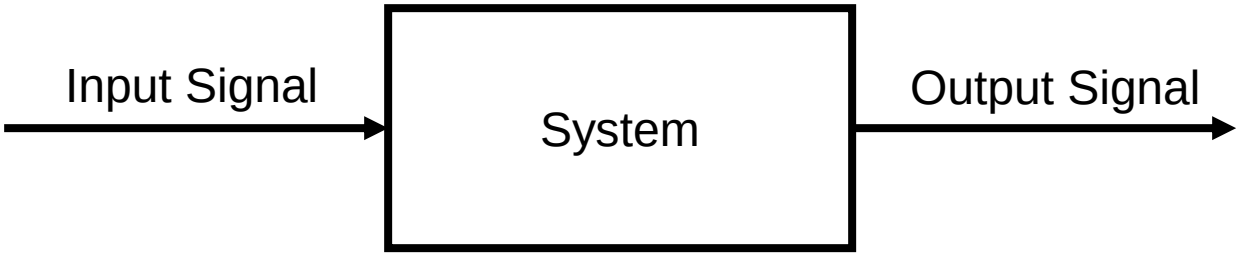
- ***Research***
  - AN models
  - ***Neural Spike train analyses – Correlograms***
  - Data collection – NEL system

Teaching: *Audiology (AUD) students, "Signal Processing for Hearing and Speech"*

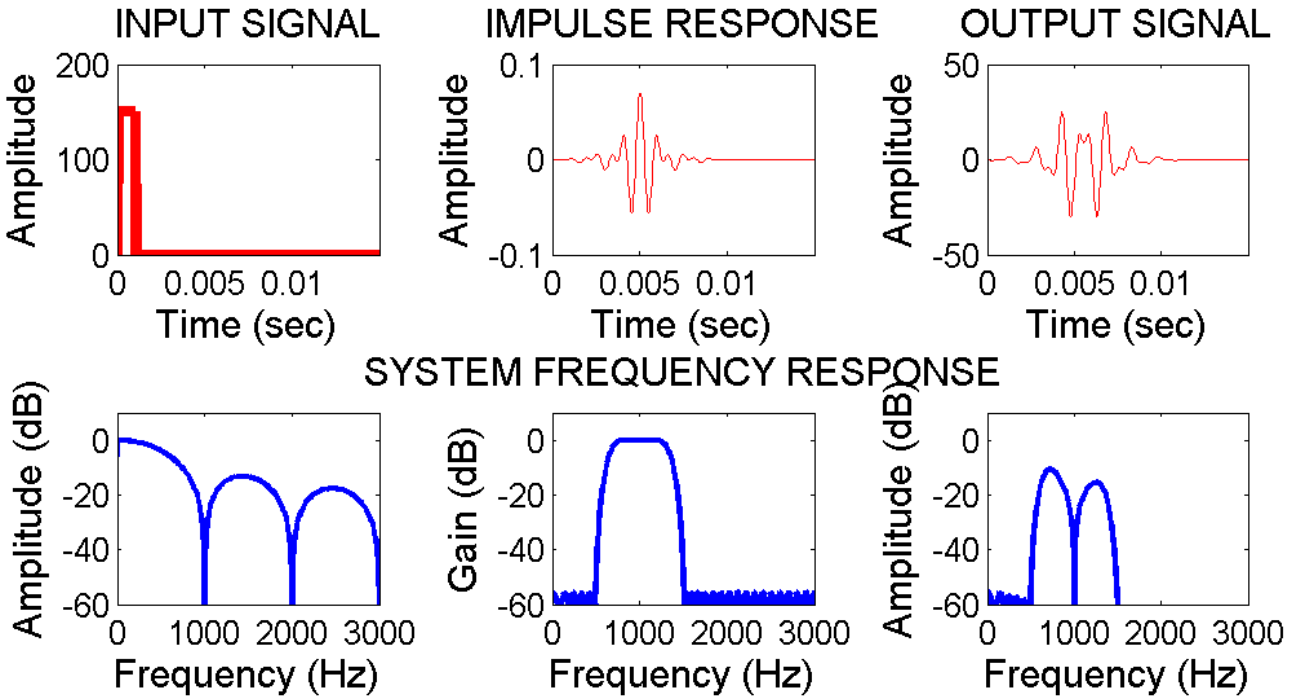


- ***Based on Rosen and Howell Textbook***
  - No equations, just figures conveying key concepts
  - I developed MATLAB code for many of the core concepts in parallel with text chapters/figures
  - Simple parameter changes allow students without MATLAB background to "explore" issues hands on and develop intuition beyond static text figures, e.g.,
    - Sinusoidal motion – unit circle
    - Time/frequency tradeoff (e.g., clicks, tones)
    - Linear vs log BWs,
    - Resolved/unresolved harmonics (effect of broadened BWs)
  - Course toolbox of core functions (signal generation, Fourier analysis, filtering, …)
- *ZIP file of course material shared with few colleagues*

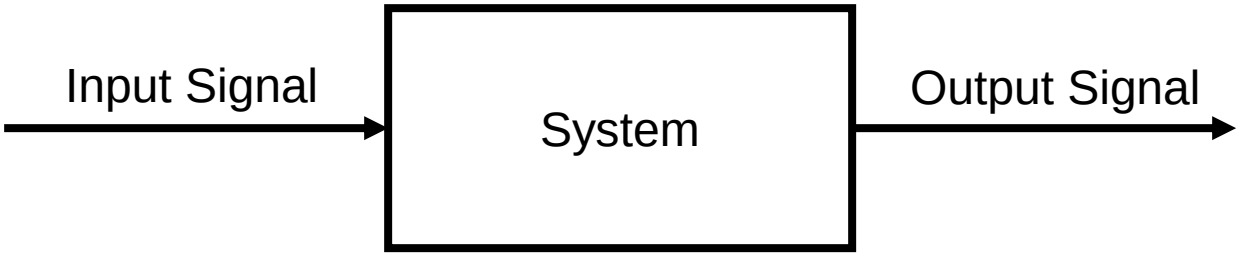
E.g., exploring click duration effects in characterizing a system



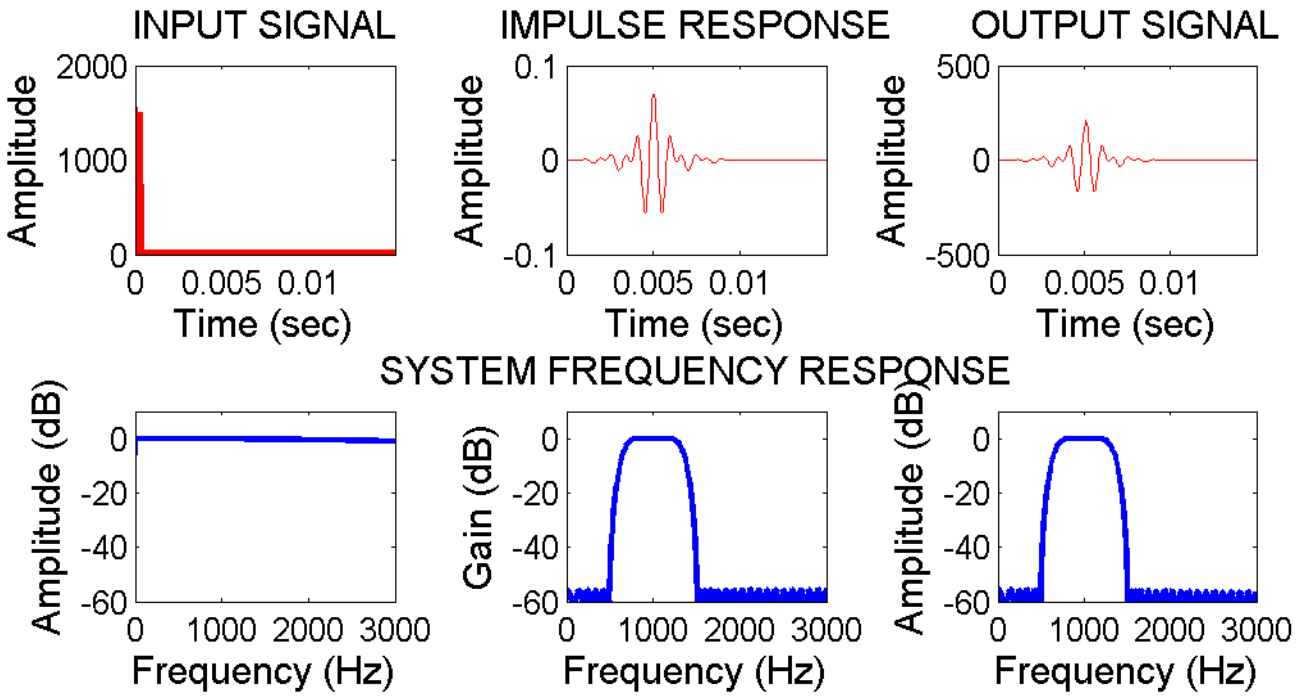
- *MATLAB code using core functions from "course toolbox" (signal generation, filtering, Fourier analysis ...) allows simple parameter changes (click duration) to explore fundamental concepts in real applications (e.g., time/BW tradeoff)*



E.g., exploring click duration effects in characterizing a system

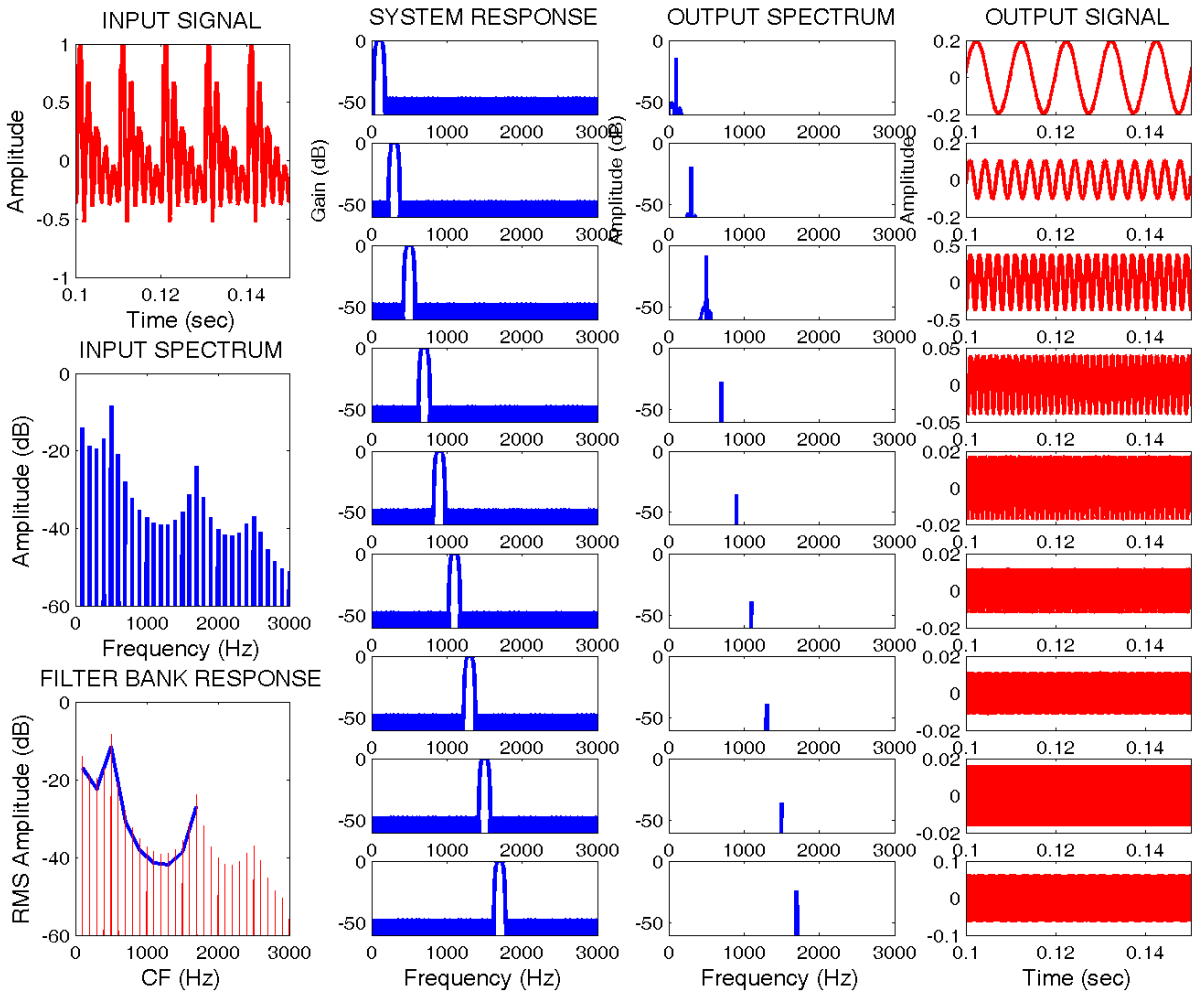


- *MATLAB code using core functions from "course toolbox" (signal generation, filtering, Fourier analysis ...) allows simple parameter changes (click duration) to explore fundamental concepts in real applications (e.g., time/BW tradeoff)*



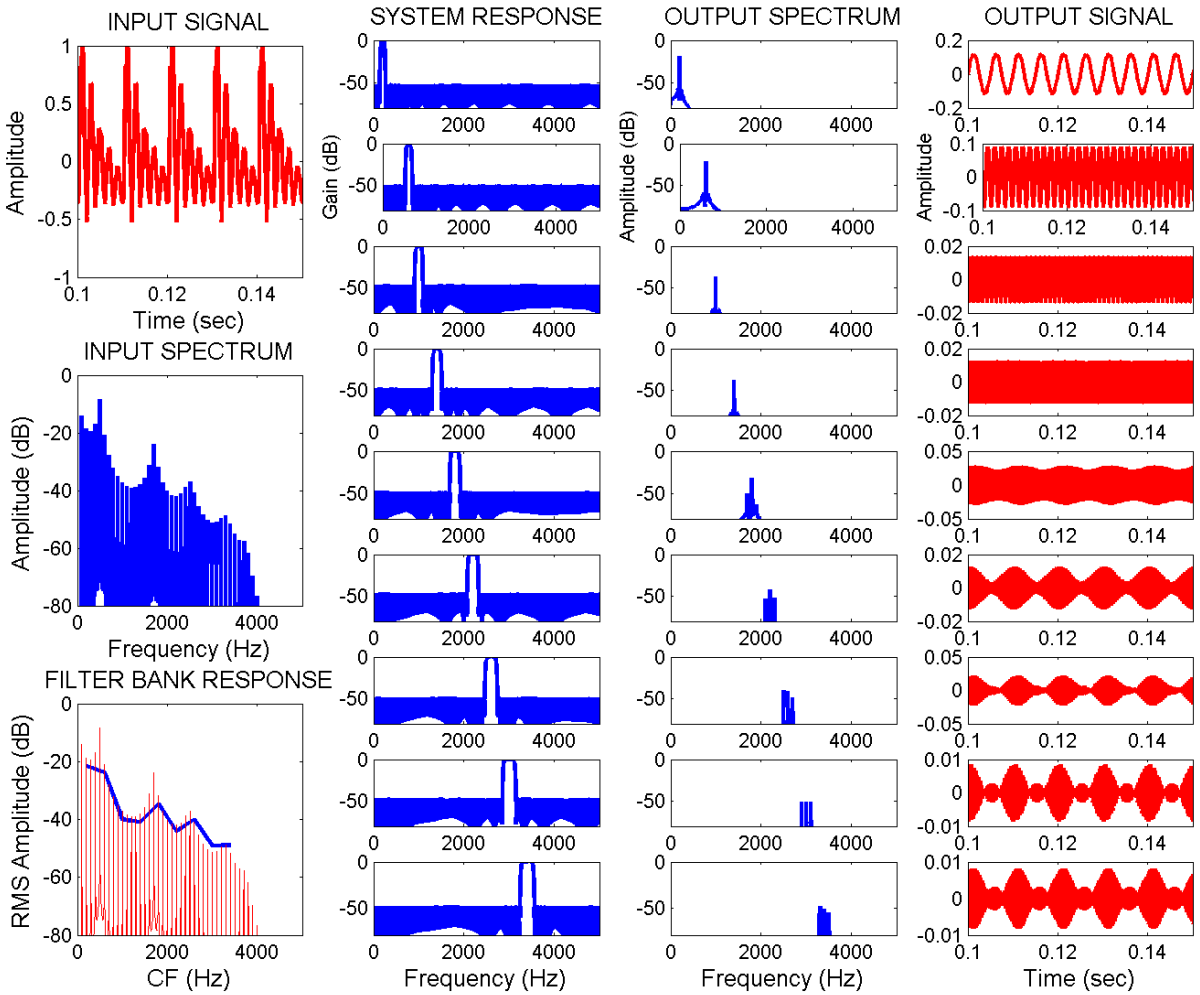
E.g., exploring effects of reduced frequency selectivity on harmonic resolvability

- *Students have simple control of filter BWs (using "course-toolbox" functions) to explore filtering issues – **linear vs logarithmic**; broadening with hearing loss*



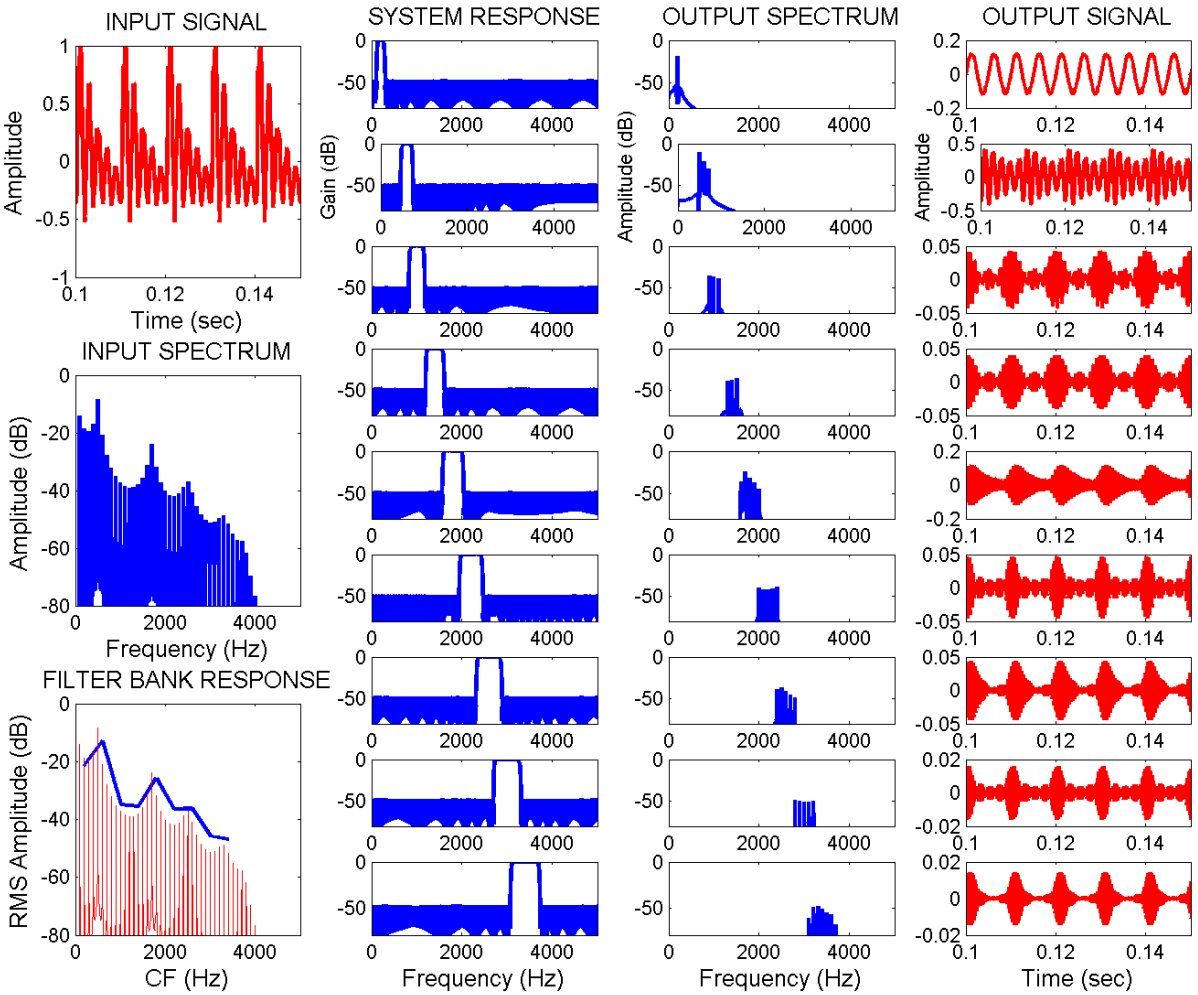
E.g., exploring effects of reduced frequency selectivity on harmonic resolvability

- *Students have simple control of filter BWs (using "course-toolbox" functions) to explore filtering issues – **linear vs logarithmic**; broadening with hearing loss*



E.g., exploring effects of reduced frequency selectivity on harmonic resolvability

- *Students have simple control of filter BWs (using "course-toolbox" functions) to explore filtering issues – linear vs logarithmic;* ***broadening with hearing loss***



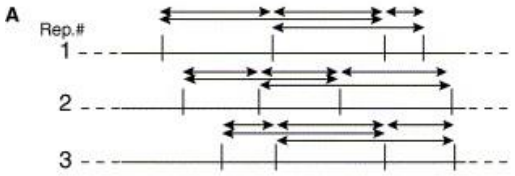
Outline

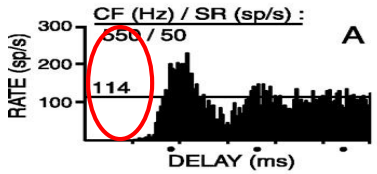
- ***Teaching***
  - ***Audiology (AUD) students***
    - ***Signal Processing for Hearing and Speech course***
  - Undergraduate SLHS students
    - Hearing Science course
  - Graduate BME students
    - Biomedical Signal Processing course

- ***Research***
  - AN models (… Laurel)
  - ***Neural Spike train analyses – Correlograms***
  - Data collection – NEL system

Shuffled auto-correlograms provide an improved quantification of temporal structure in neural responses to arbitrary stimuli

ALL-ORDER INTERSPIKE INTERVAL HISTOGRAMS (Ruggero, 1973)

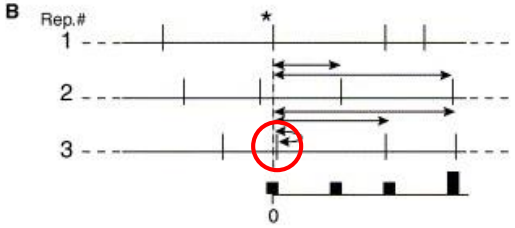


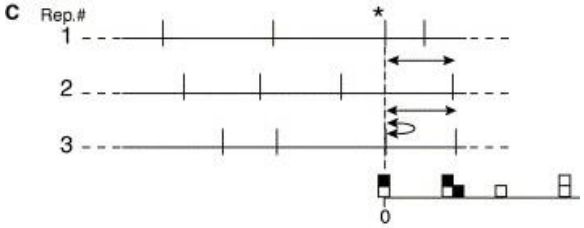


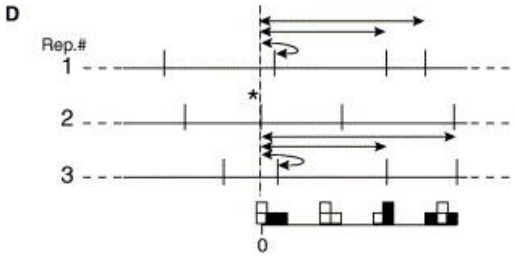
Response to broadband noise

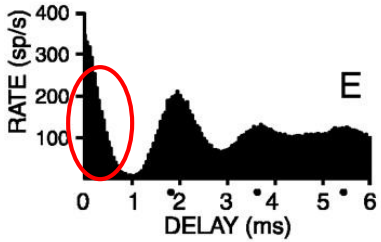
SHUFFLED AUTOCORRELOGRAMS (SACs)

(Joris, 2003; Louage et al, 2004; Joris et al., 2006, …)



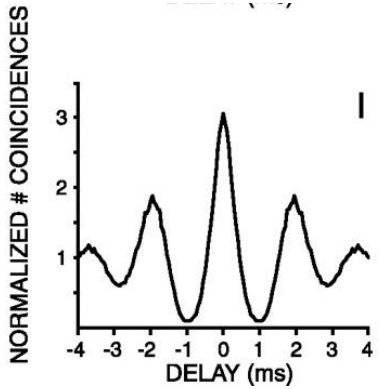






Shuffling avoids refractory period

N^2 Smoothing

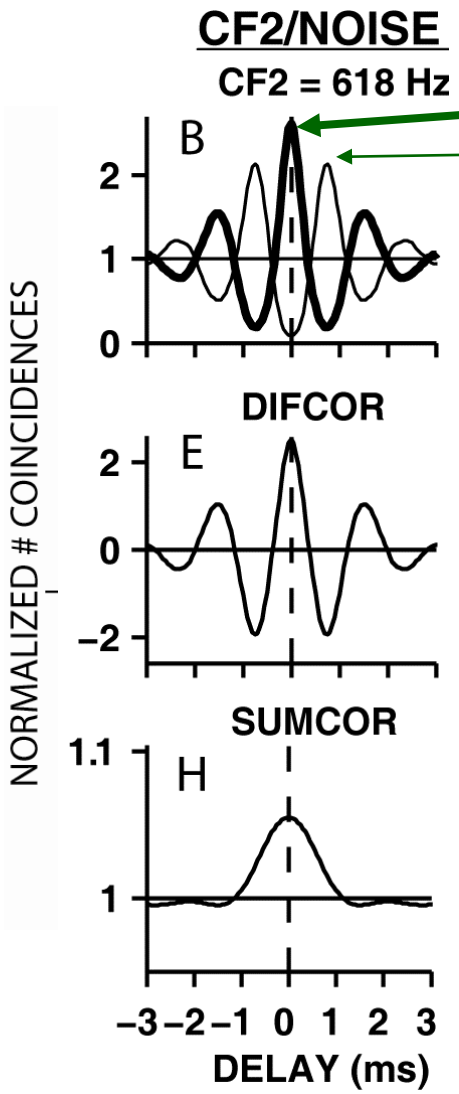


Normalized SAC:

- 1 => No correlation
- > 1 ~ positive correlation
- < 1 ~ negative correlation

- Oscillations ~ CF
- Peak decay ~ BW

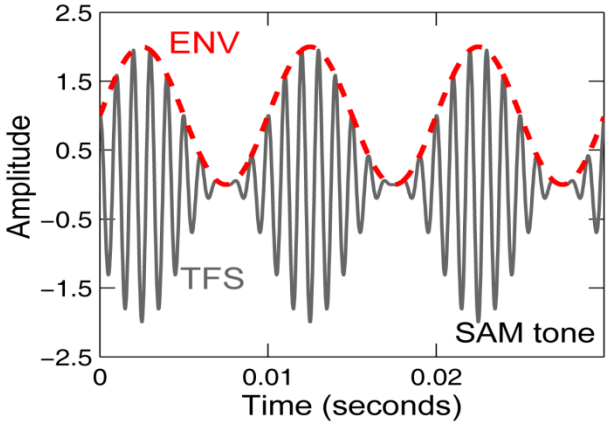
Temporal fine-structure (TFS) and envelope (ENV) coding can be separated via stimulus polarity inversion (Joris, 2003)



Shuffled correlograms

SAC (A+/A+);

XpAC (A+/A-)



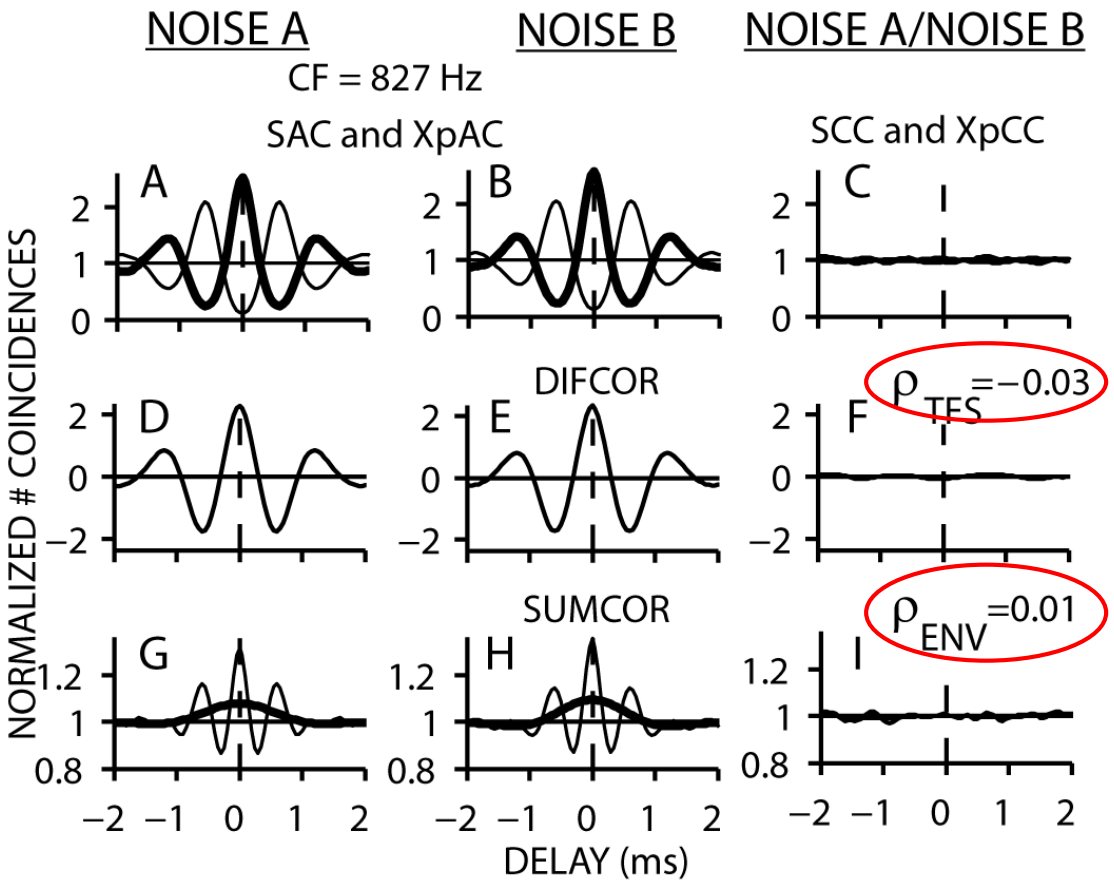
DIFCOR = SAC – XpAC (isolates fine structure)

- peak height quantifies TFS coding strength

SUMCOR = ½ (SAC + XpAC) [emphasizes envelope]

- peak height quantifies ENV coding strength

- Neural cross-correlation coefficients ($0 \le \rho_{TFS} \le 1$, Heinz and Swaminathan 2009
  - Computed from neural spike times (model or animal data)
  - Used for:
    - Cross-stimulus comparisons of TFS or ENV coding (chimaeric speech)
    - Cross-CF comparisons (traveling wave delay estimates)
  - Relies on correlograms computation function in C for efficiency, run as MEX
  - ZIP file with MATLB code shared with a number of colleagues:
    - basic AN model demo



Two uncorrelated noises

$$\rho_{AB} = \frac{Cov(A,B)}{\sqrt{Var(A) * Var(B)}}$$

$$\rho_{TFS} = \frac{DC(AB)}{\sqrt{DC(A) * DC(B)}}$$

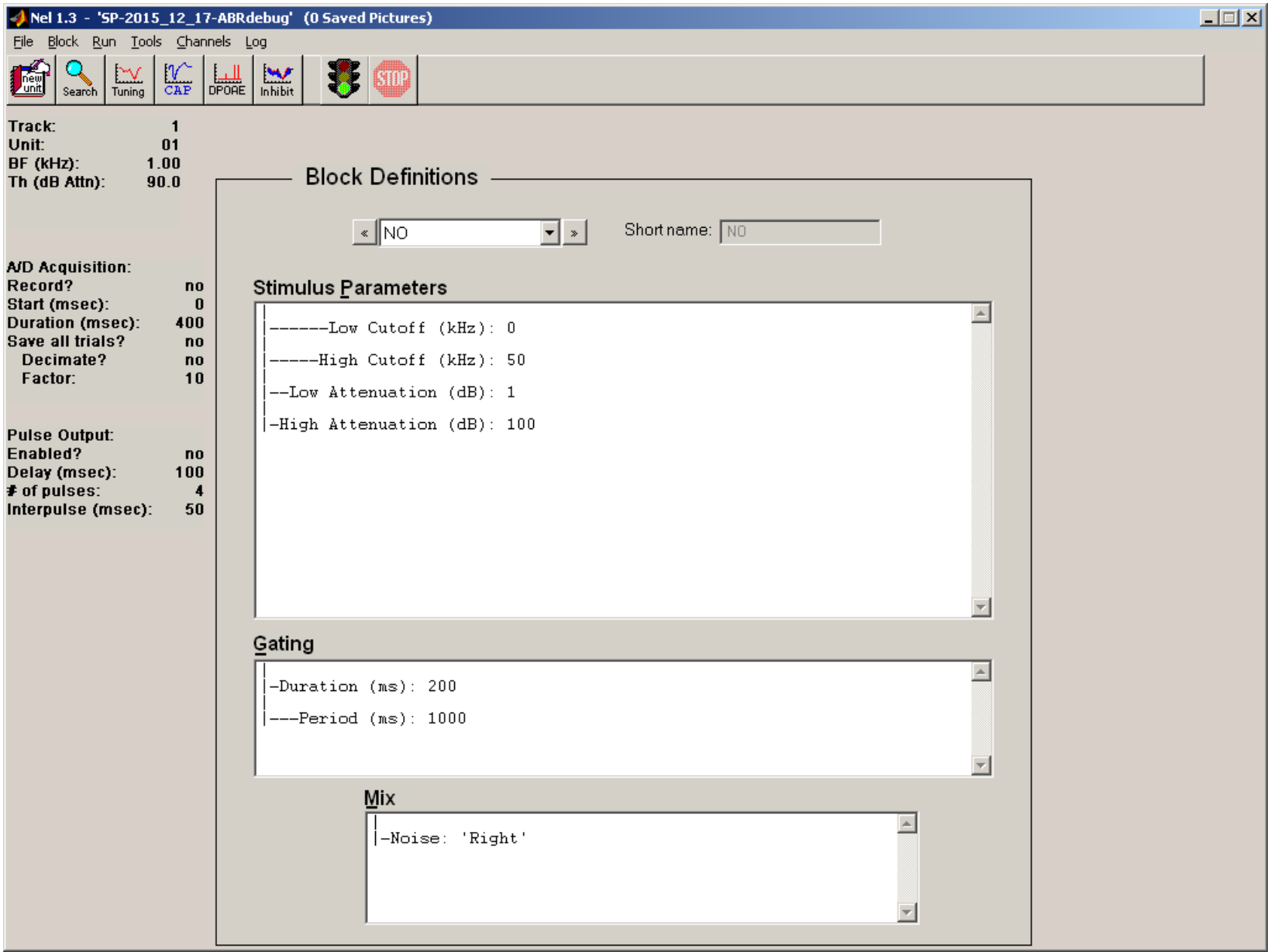
$$\rho_{ENV} = \frac{SC(AB)}{\sqrt{SC(A) * SC(B)}}$$

Outline

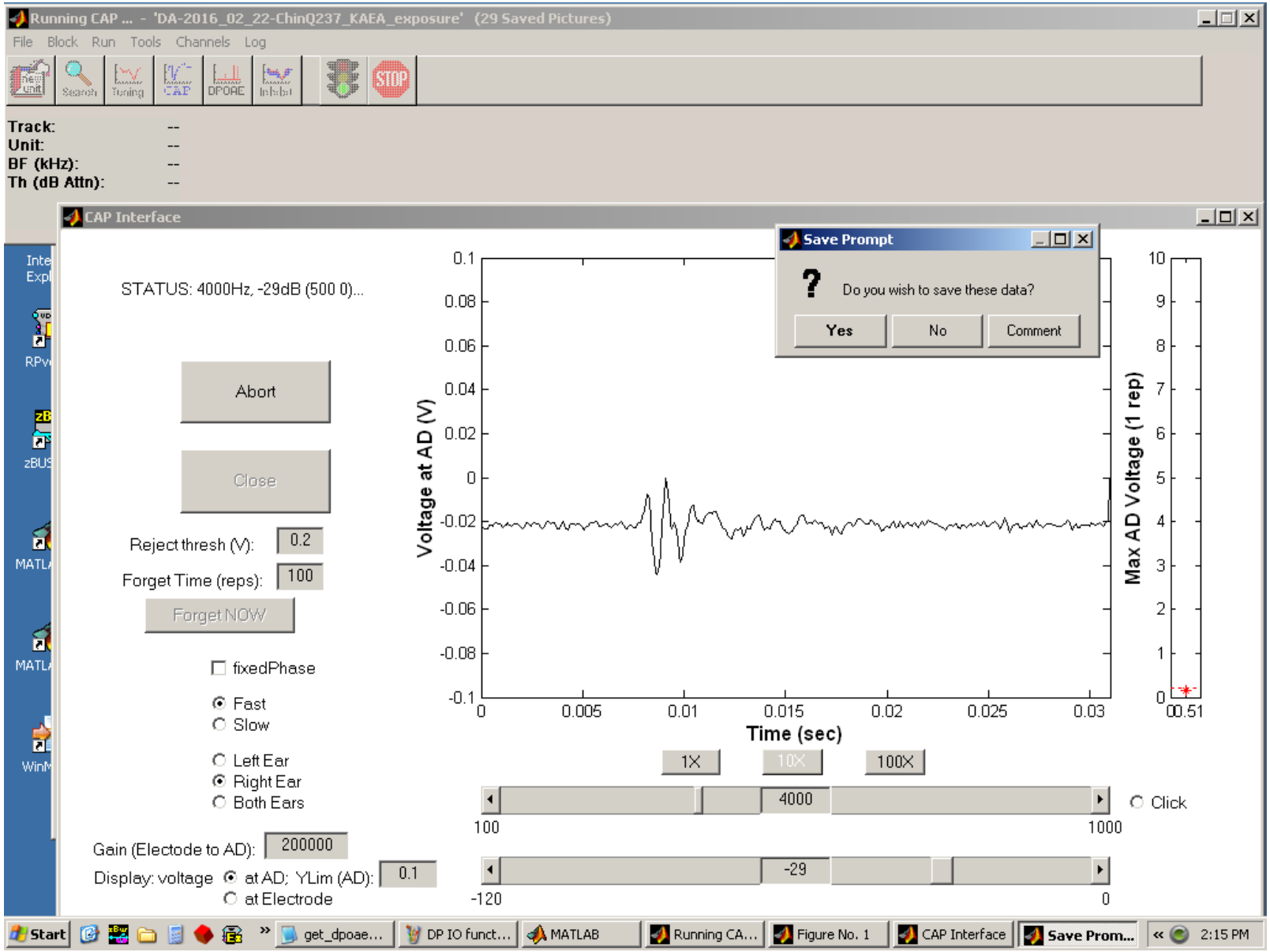
- ***Teaching***
  - ***Audiology (AUD) students***
    - ***Signal Processing for Hearing and Speech course***
  - Undergraduate SLHS students
    - Hearing Science course
  - Graduate BME students
    - Biomedical Signal Processing course

- ***Research***
  - AN models (… Laurel)
  - *Neural Spike train analyses – Correlograms*
  - Data collection – NEL system

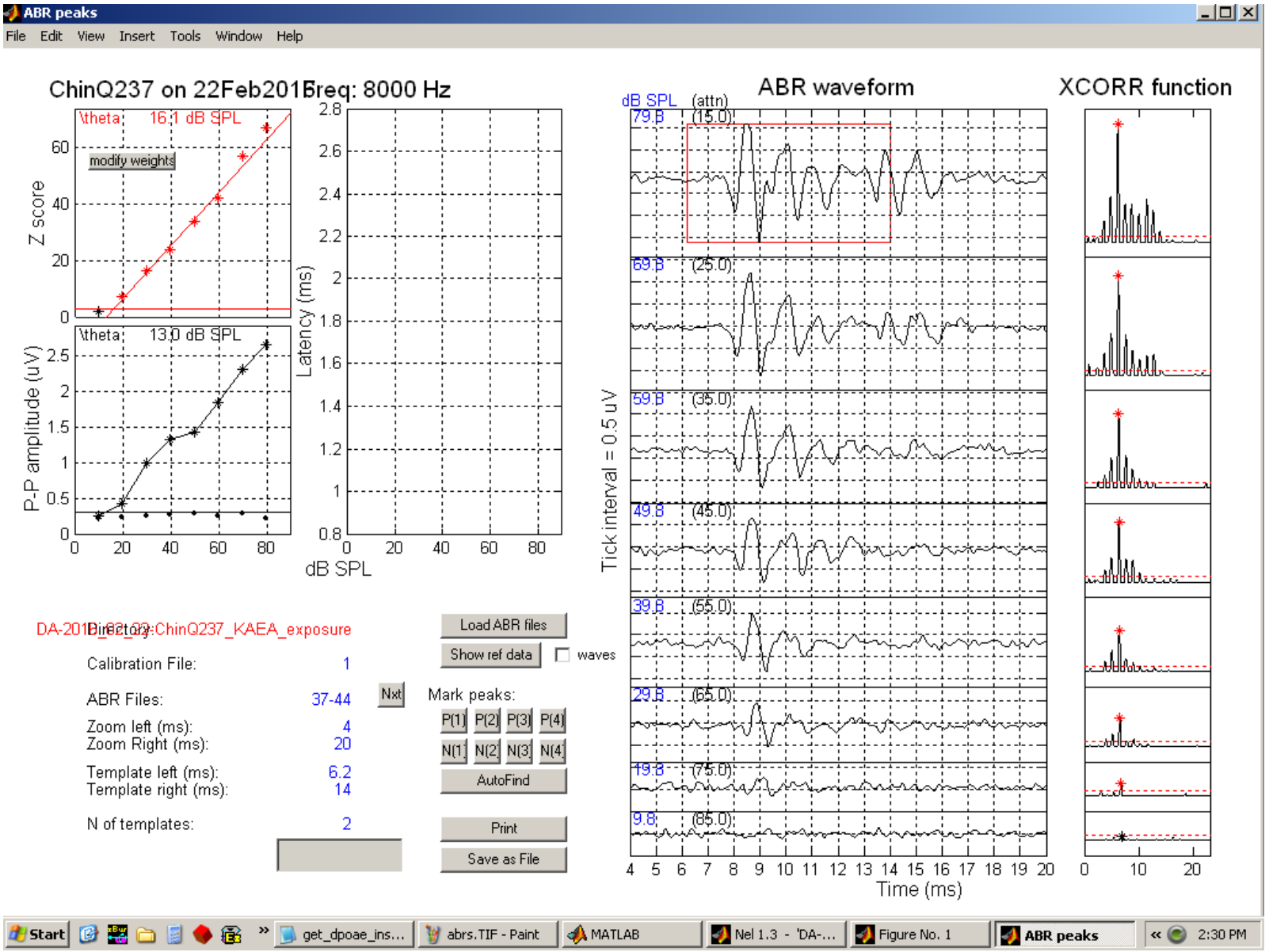
Neurophysiological Data Collection – "NEL" system (from Eric Young's Lab)



Neurophysiological Data Collection – "NEL" system (from Eric Young's Lab)



Neurophysiological Data Collection – "NEL" system (from Eric Young's Lab)



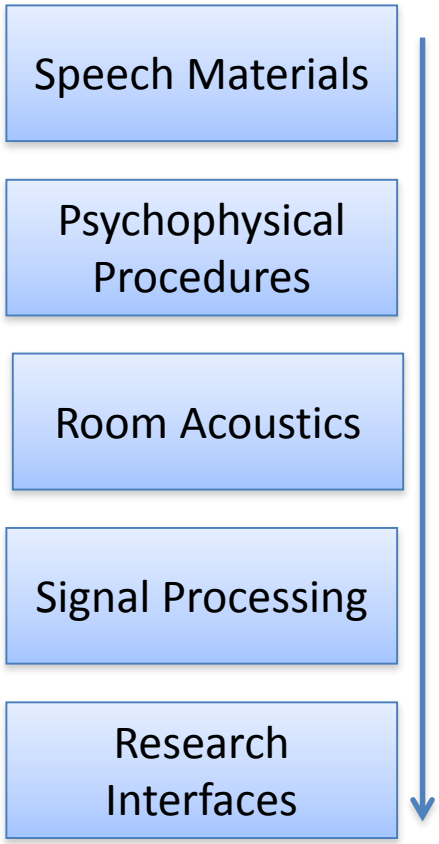
Software Tools
for Hearing Research

Ray Goldsworthy

USC University of Southern California

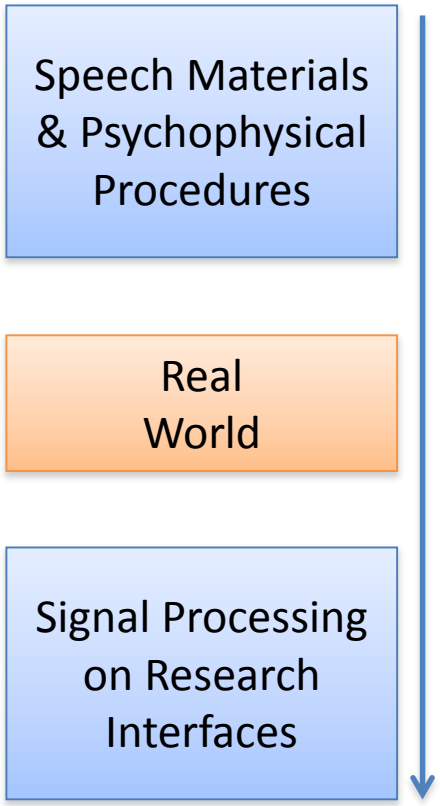
Why Matlab?

- Strikes the right balance
  - Complete acoustic control
  - Community resources
  - Signal processing tools
  - Research interfaces

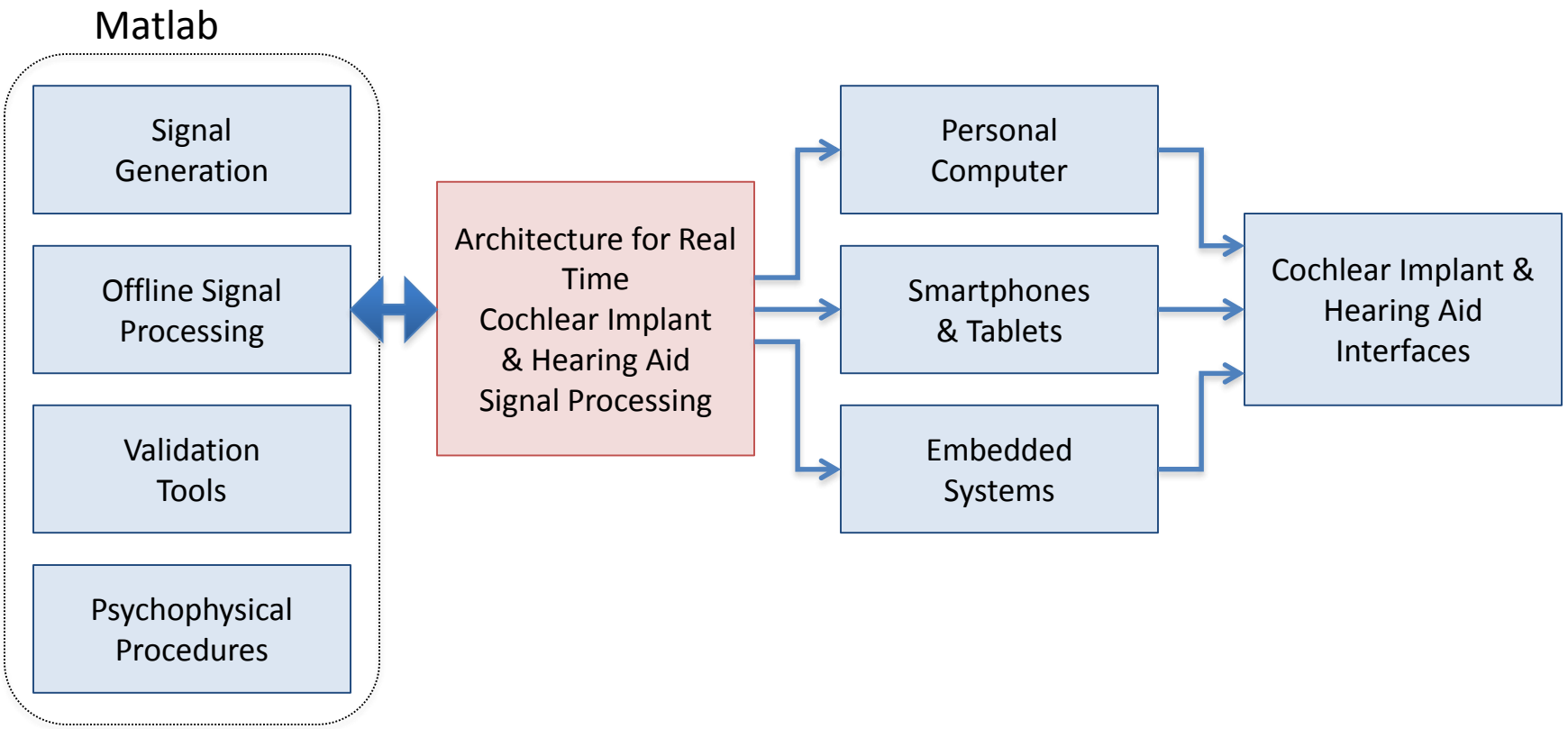


Why ~Matlab?

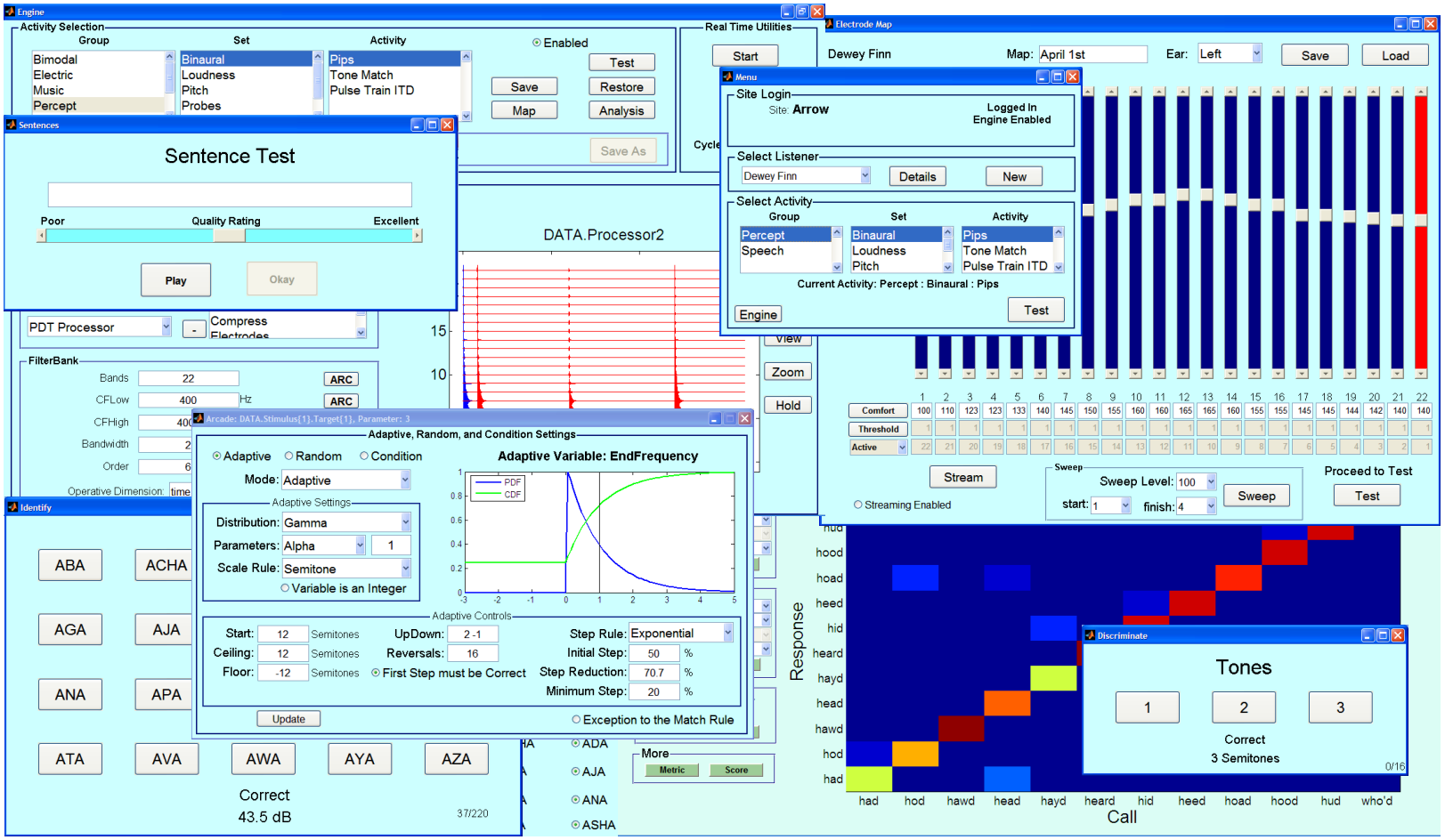
- Wearable algorithms
  - Longitudinal studies
  - Real environments
  - At-home training



Comprehensive System

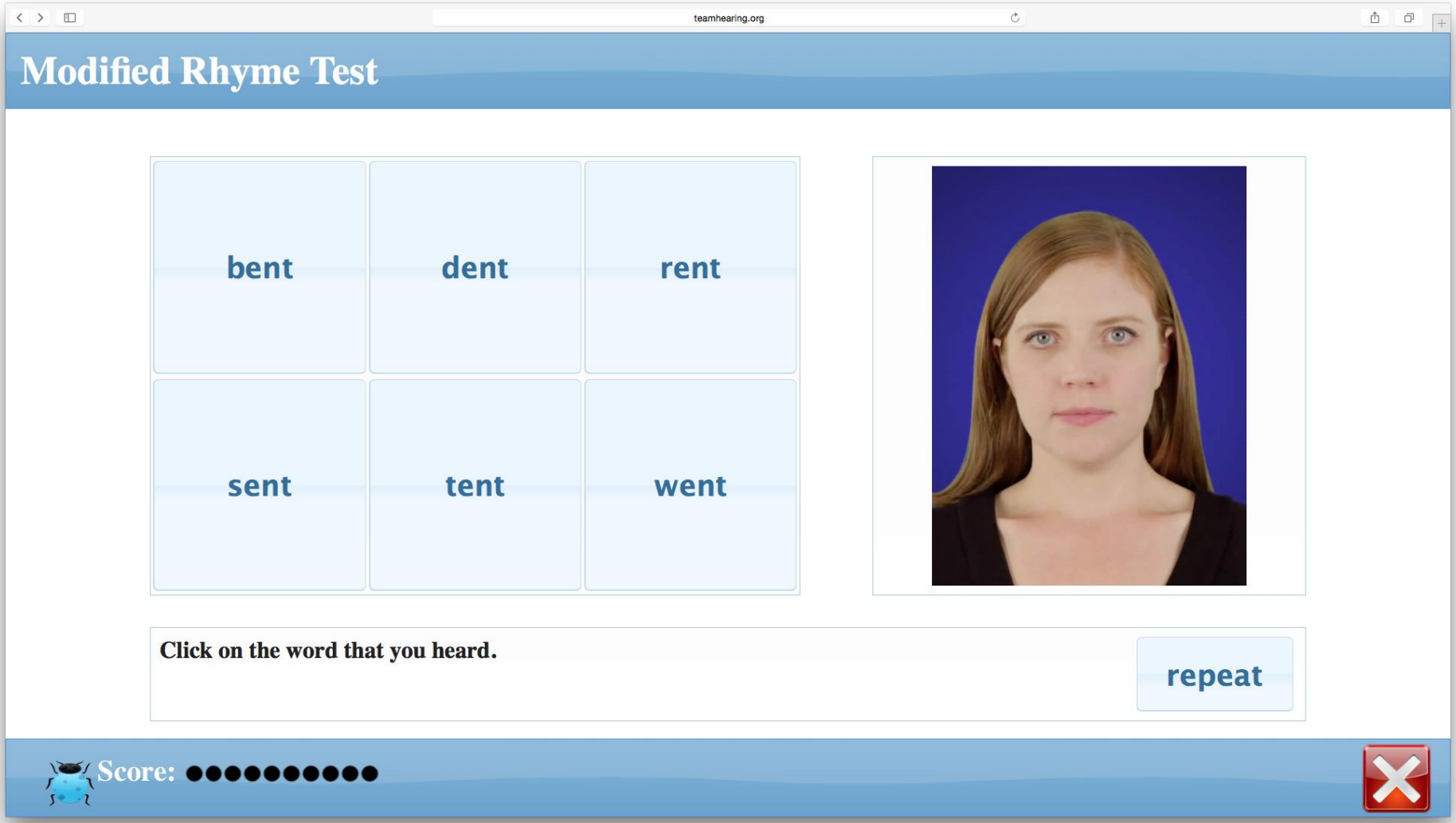


Open Design Matlab Tools



Percept, a MATLAB toolbox for cochlear implant and hearing aid research.

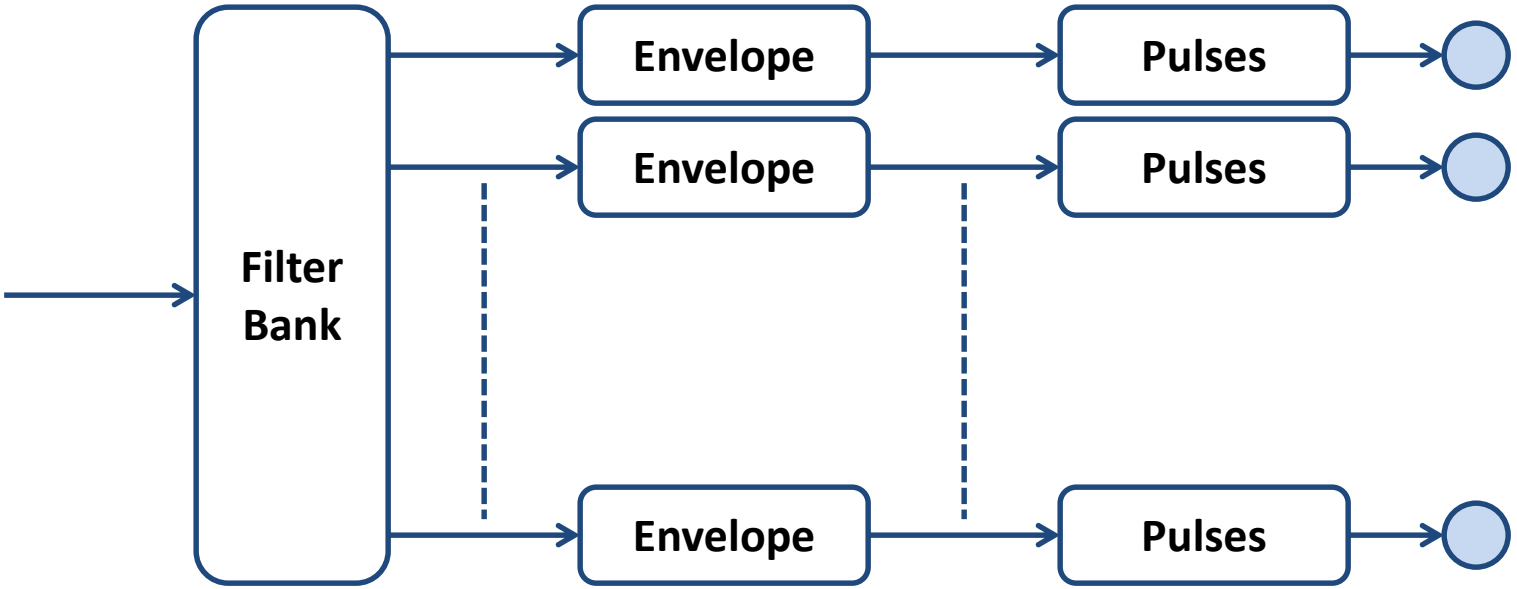
Open Design Web Tools



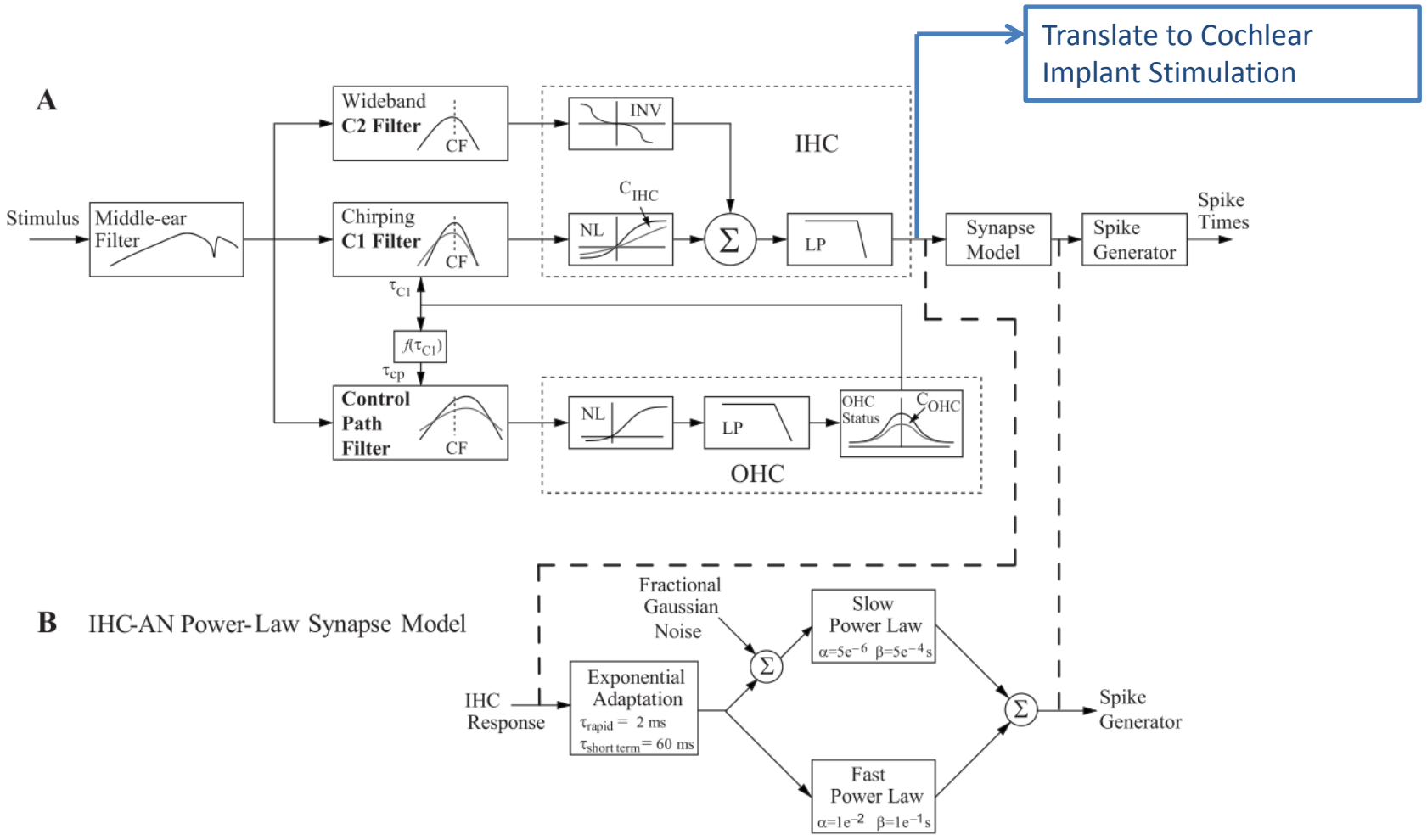
Example Usage

Adaptive Filtering Based on Auditory Models

Basic Signal Processing for Cochlear Implants



Auditory Periphery Model

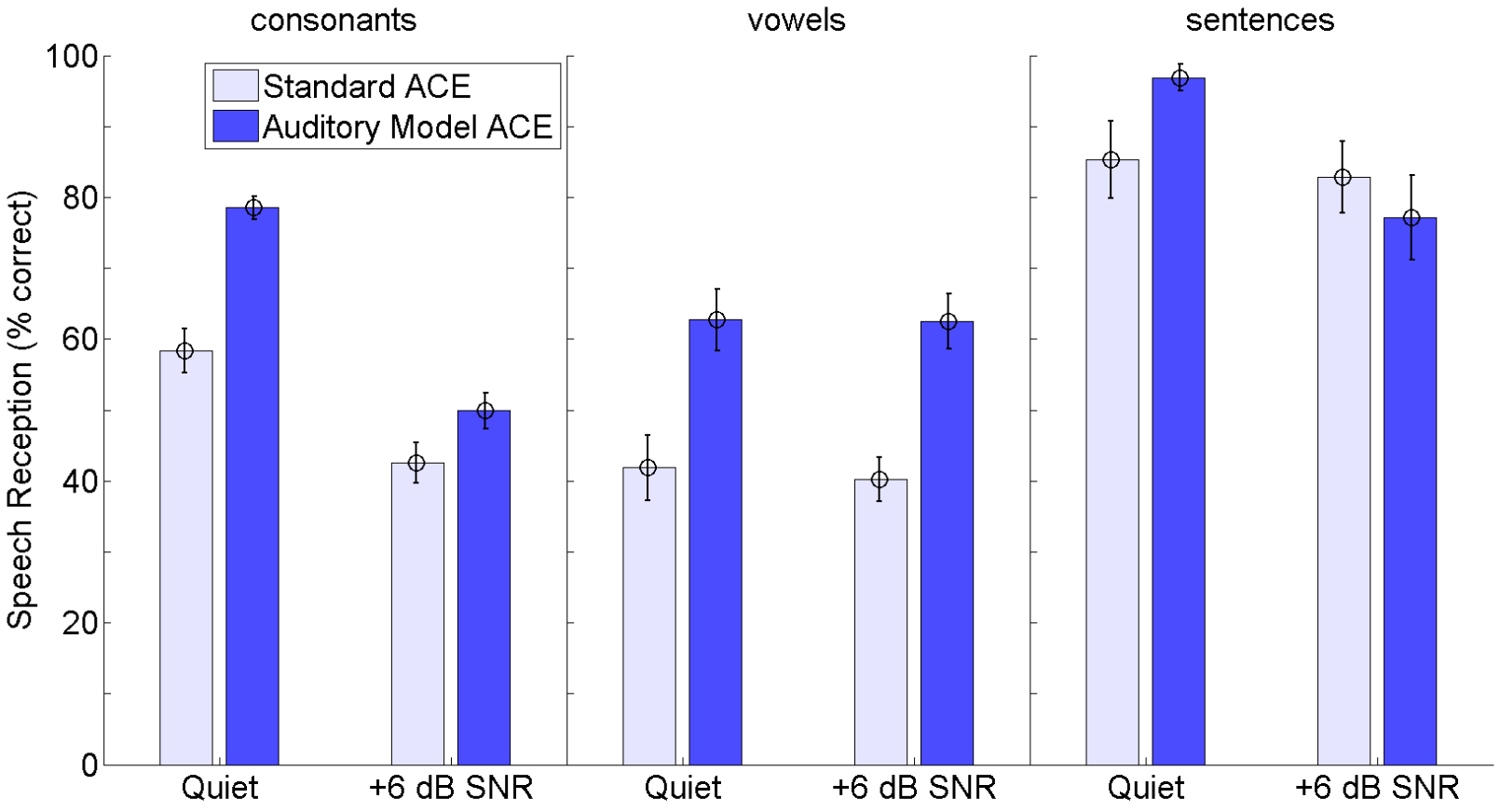


Zilany *et al.*, 2009

Study: Initial Assessment

- 5 adult cochlear implant users tested
- Comparison of Standard and Auditory-Model implant processing
- Strategies delivered via research interface in laboratory setting
- Speech reception in quiet and noise measured using consonant, vowel, and sentence materials

Preliminary Results



Conclusions

- Growing body of academic toolboxes for hearing research
- Clear articulation of academic and research goals is an important next step
- Need for real-time signal processing architecture for wearable devices

The multivariate temporal response function (mTRF) toolbox: a MATLAB toolbox for relating neural signals to continuous stimuli

Denis Drennan

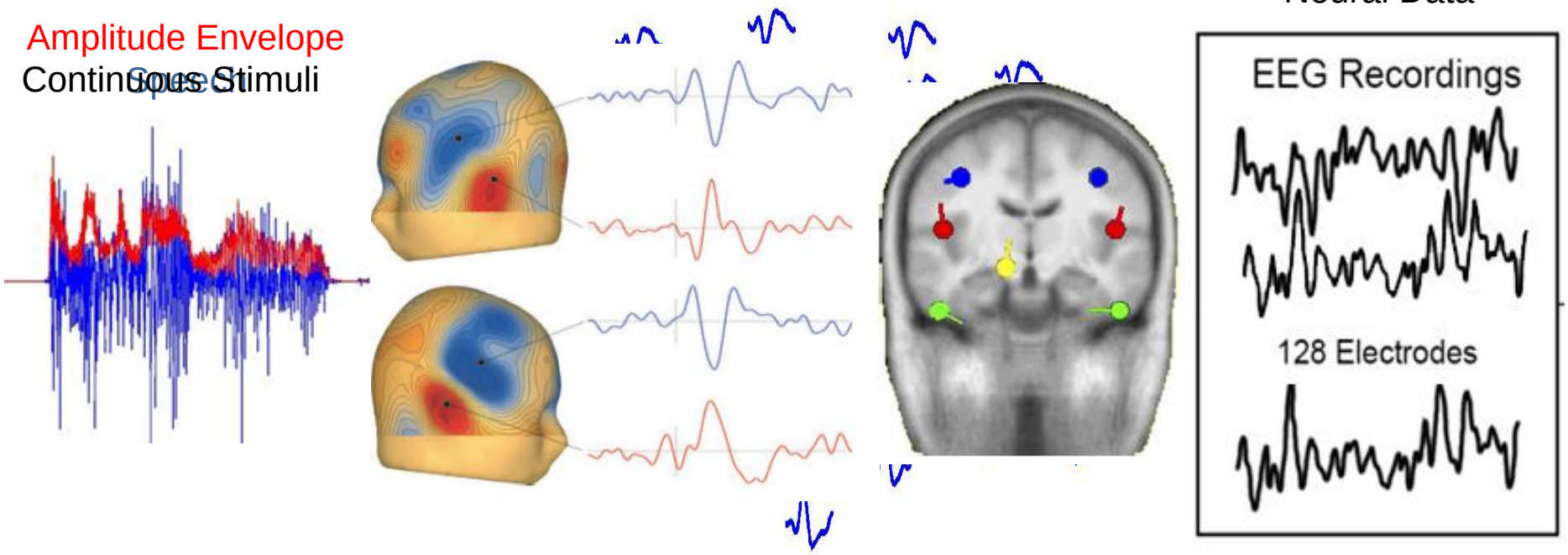
Lalor Lab, Trinity College Dublin

Forward Encoding

Amplitude Envelope

Continuous Stimuli

Neural Data

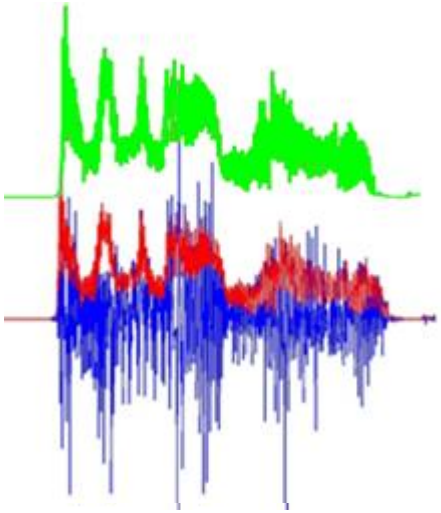


$$x(t) \quad * \quad \omega(\tau) \quad = y(t)$$

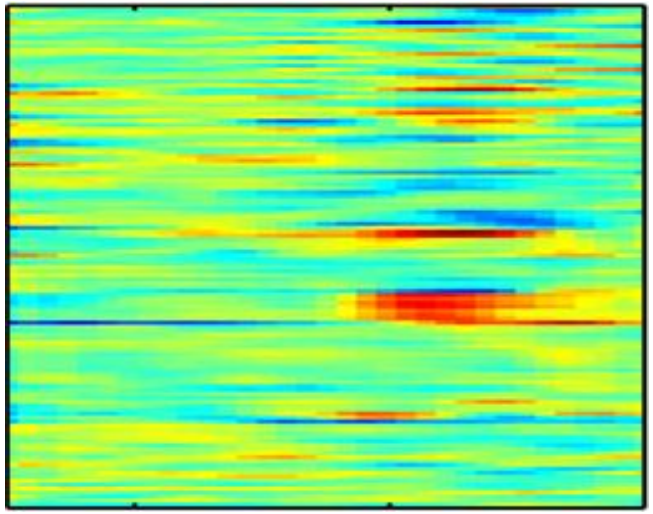
TRF

Backward Decoding

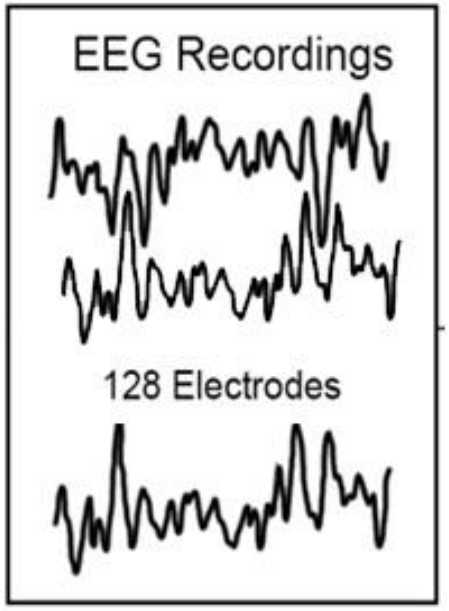
Stimulus Reconstruction



Decoder



Neural Data

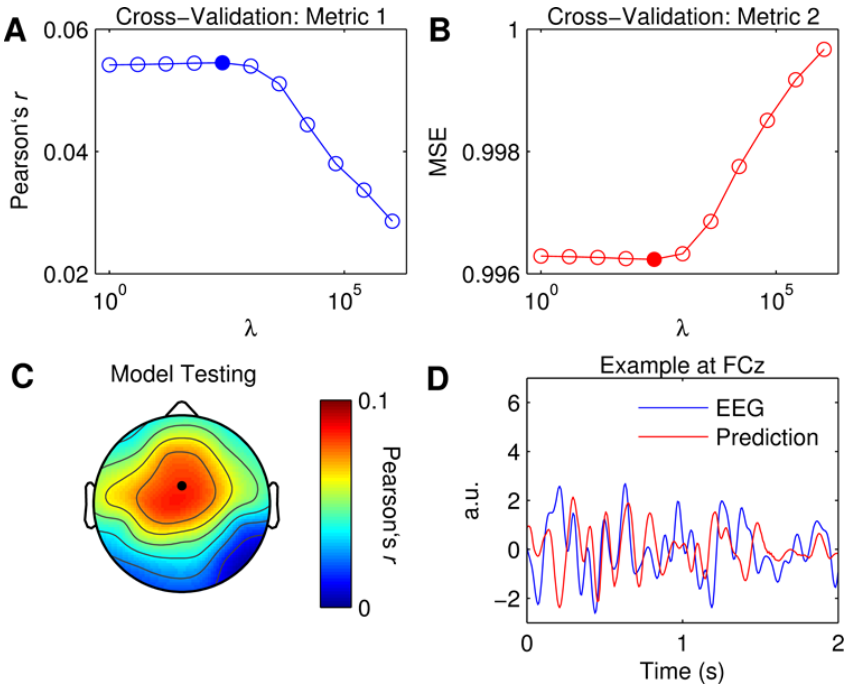


$$x(t) = g(\tau, n) * y(t)$$

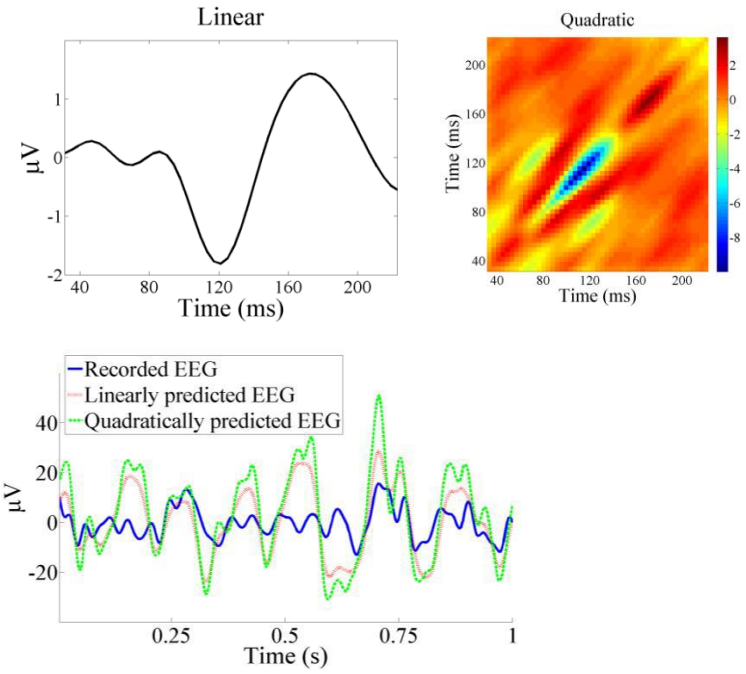
mTRF

Optimization and Prediction

Examples

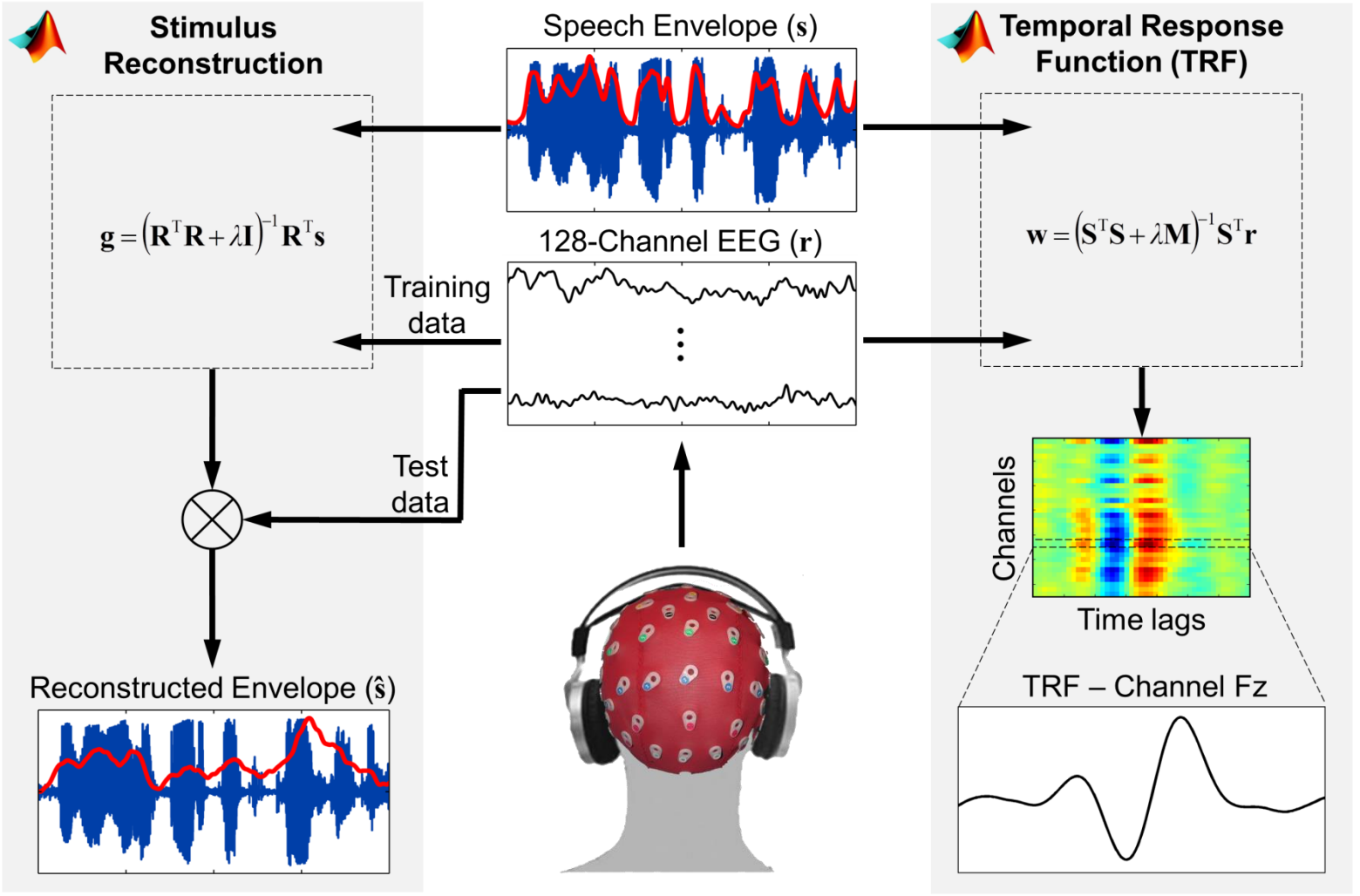


Previous Work



Power et al., *EMBC*, 2011

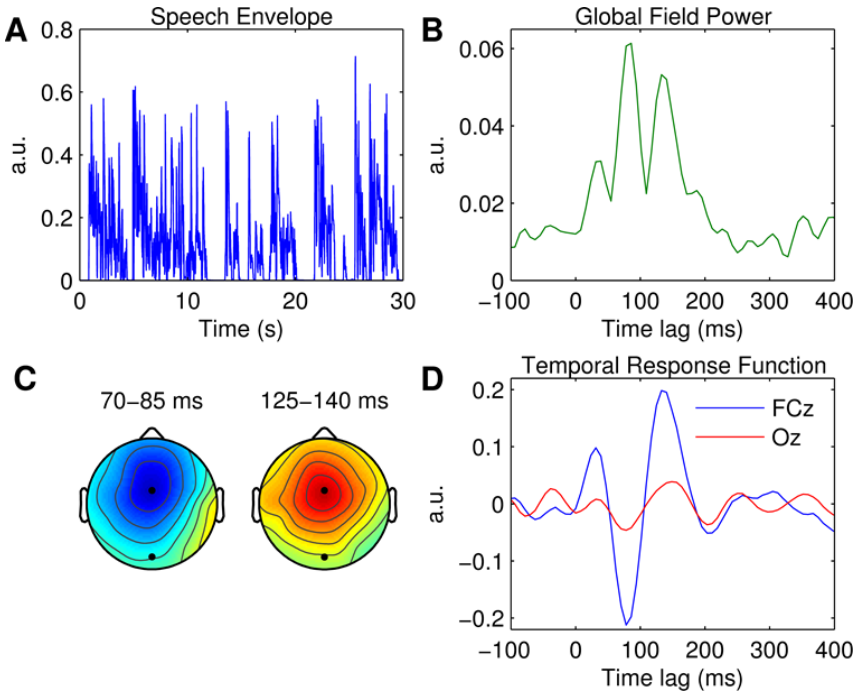
Experimental Procedure & Analysis

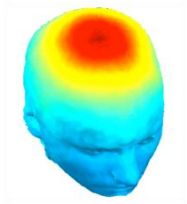


Univariate TRF Estimation

(Forward Encoding)

Examples

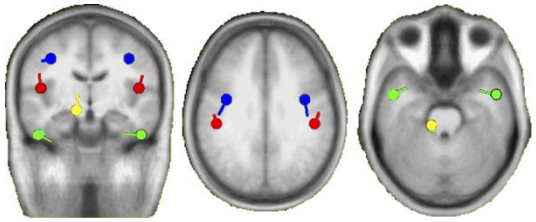




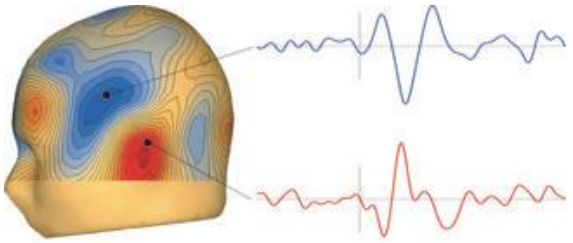
EEGLAB

http://sccn.ucsd.edu/eeglab/

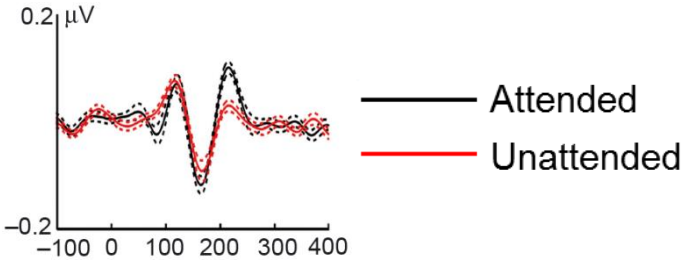
Previous Work



Lalor et al., *J Neurophysiol*, 2009



Lalor & Foxe, *Eur J Neurosci*, 2010

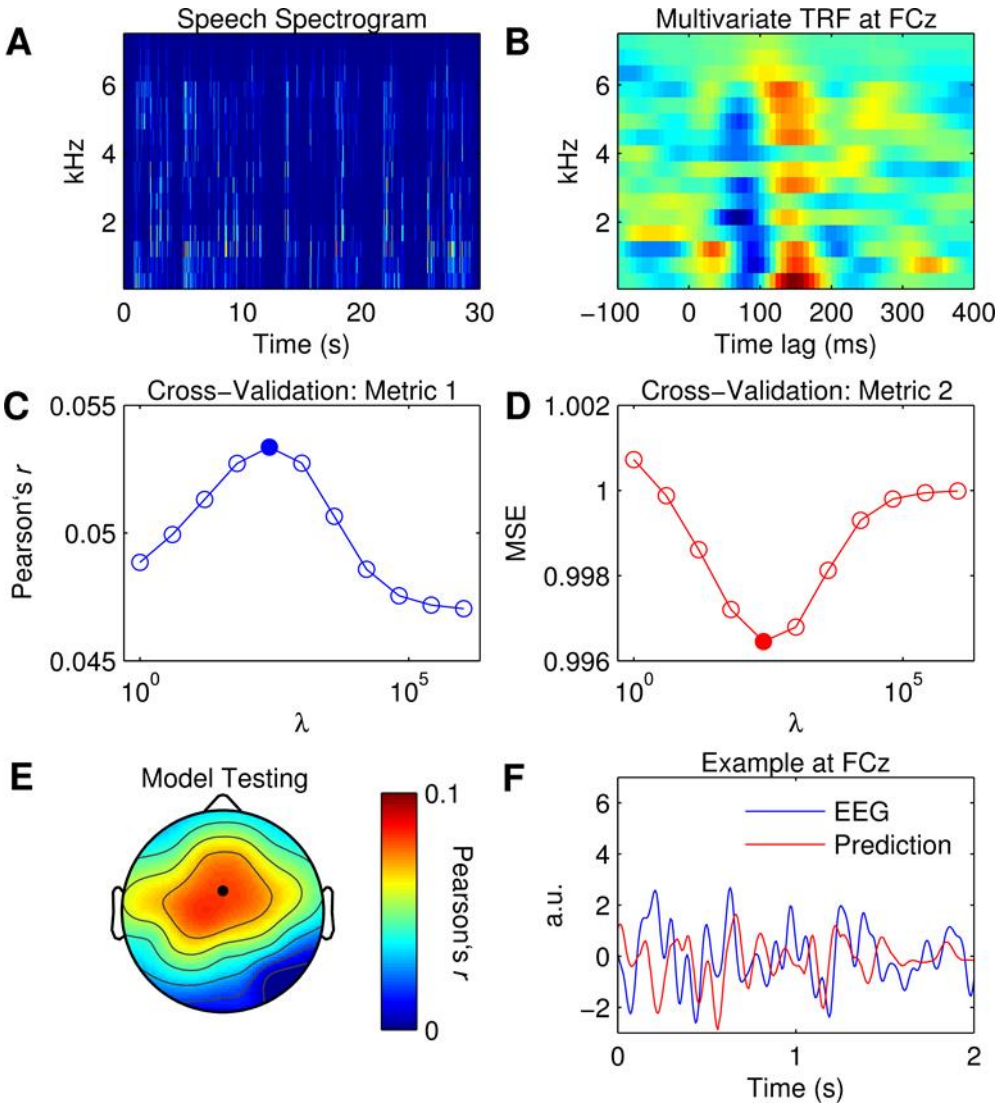


Power et al., *Eur J Neurosci*, 2012

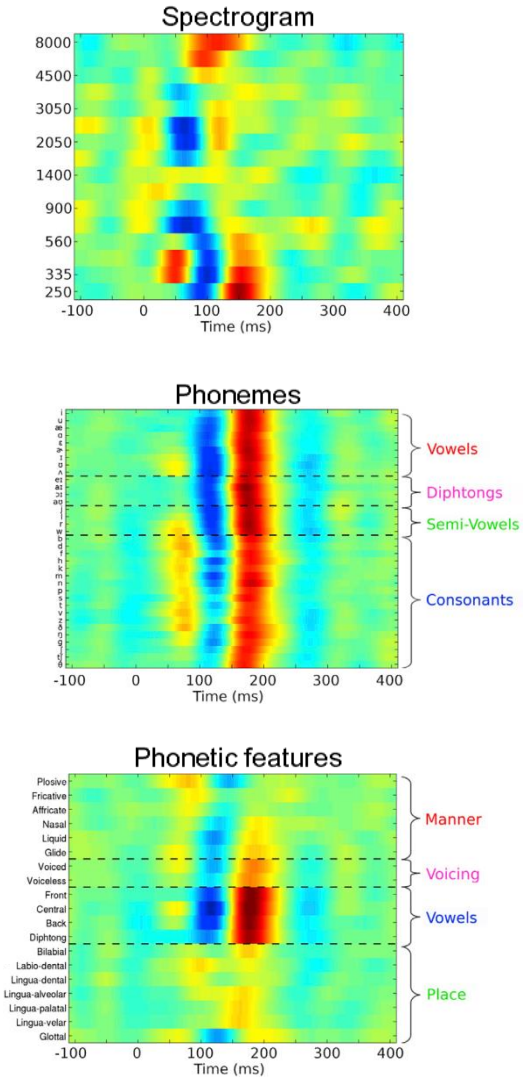
Multivariate TRF Estimation

(Forward Encoding)

Examples



Previous Work

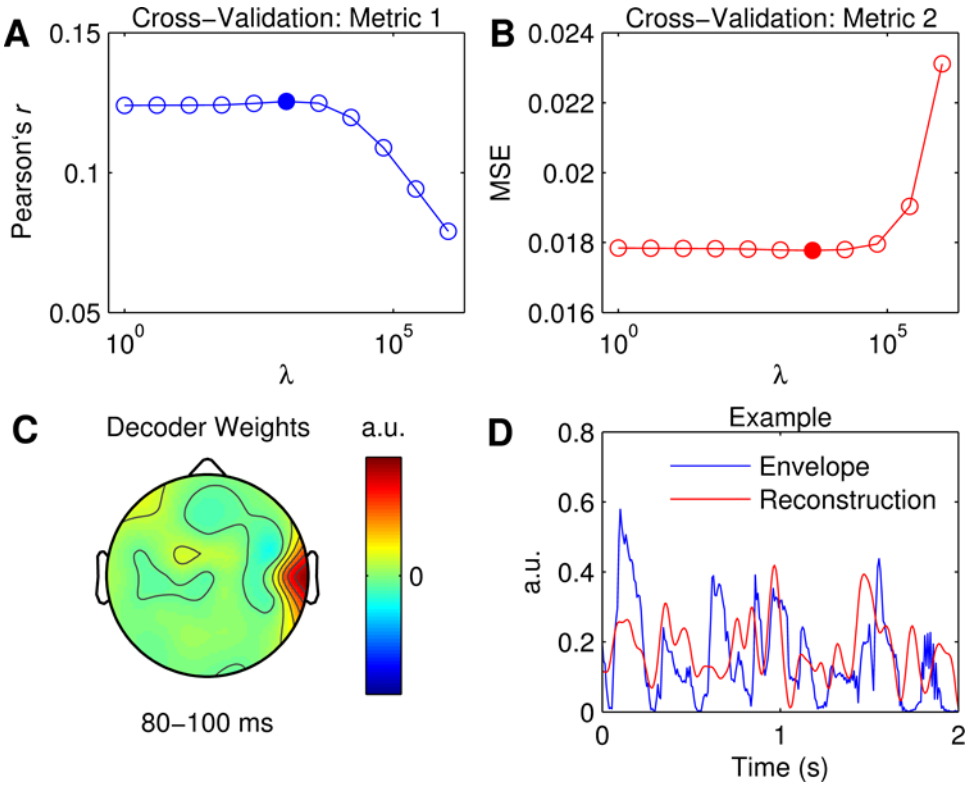


Di Liberto et al., *Current Biol*, 2015

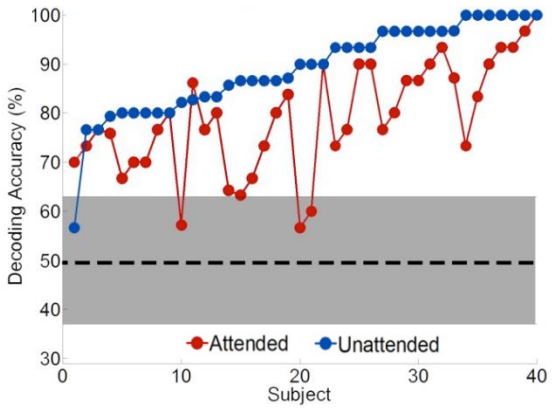
Multivariate Stimulus Reconstruction

(Backward Decoding)

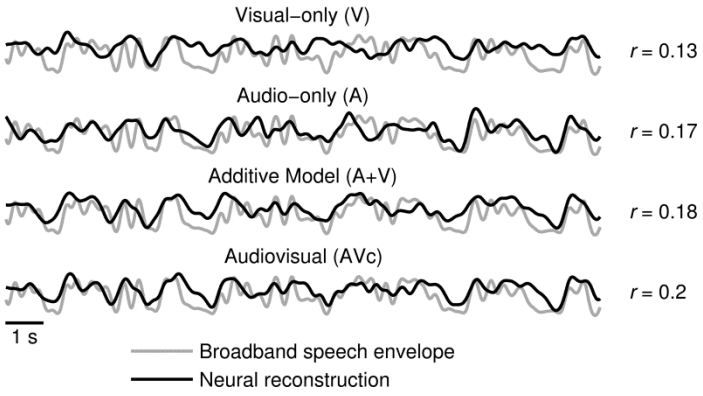
Examples



Previous Work



O'Sullivan et al., *Cereb Cortex*, 2015



Crosse et al., *J Neurosci*, 2015

mTRF Toolbox Functions

Function	Inputs	Outputs	Details
mTRFtrain	STIM– stimulus (time × feats) RESP–response (time × chans) FS– sampling rate (Hz) DIR–mapping direction (forward==1, backward==–1) START– start time lag (ms) FIN–finish time lag (ms) LAMBDA–ridge parameter	MODEL–stimulus-response mapping (forward = feats × lags × chans, backward = chans × lags × feats) T– time lags (ms) CONST–regression constant	Performs multivariate ridge regression on stimulus STIM and response RESP to solve for stimulus-response mapping function MODEL. Time lags should be set in milliseconds between START and FIN and sampling rate FS should be in Hertz. Pass in DIR==1 to map forwards or DIR==–1 to map backwards. Regularisation is controlled using the ridge parameter LAMBDA.
mTRFpredict	STIM– stimulus (time × feats) RESP–response (time × chans) MODEL–stimulus-response mapping (forward: feats × lags × chans, backward: chans × lags × feats) FS– sampling rate (Hz) DIR–mapping direction (forward==1, backward==–1) START– start time lag (ms) FIN–finish time lag (ms) CONST–regression constant	PRED–prediction (forward: time × chans, backward: time × feats) RHO– correlation coefficient PVAL–correlation *p*-value MSE–mean squared error	Predicts outcome PRED of convolving stimulus STIM (or response RESP) with stimulus-response mapping function MODEL. Pass in DIR==1 to predict RESP or DIR==–1 to predict STIM. Returns correlation coefficient RHO between PRED and STIM (or RESP), corresponding *p*-value PVAL and mean squared error MSE.
mTRFcrossval	STIM– stimulus trials [cell{1,trials}(time × feats)] RESP–response trials [cell{1,trials}(time × chans)] FS– sampling rate (Hz) DIR–mapping direction (forward==1, backward==–1) START– start time lag (ms) FIN–finish time lag (ms) LAMBDA–ridge parameter values	RHO– correlation coefficient PVAL–correlation *p*-value MSE–mean squared error PRED–prediction [forward: cell{1,trials}(time × chans), backward: cell{1,trials}(time × feats)] MODEL–stimulus-response mapping (forward = feats × lags × chans, backward = chans × lags × feats)	Performs leave-one-out cross-validation on stimulus trials STIM and response trials RESP for range of ridge values LAMBDA. Validation measures returned include correlation coefficient RHO, corresponding *p*-value PVAL and mean squared error MSE. Time lags T should be set in milliseconds between START and FIN and sampling rate FS should be in Hertz. Pass in DIR==0 to map forwards or DIR==1 to map backwards. Returns predicted signals PRED and stimulus-response mapping function MODEL.
lagGen	X–vector or matrix of time series data (forward: time × feats, backward: time × chans) LAGS–vector of time lags (samples)	XLAG–matrix of lagged time series data (forward: time × lags*feats, backward: time × lags*chans)	Returns matrix XLAG containing lagged time series of X for range of time lags given by vector LAGS. If X is multivariate, LAGGEN will concatenate features for each lag along the columns of XLAG.

History of mTRF Toolbox

- This toolbox had its beginnings at the Telluride Neuromorphic Cognition Workshop in 2012

- ***Applications for this year now open!!*** (https://neuromorphs.net/nm/wiki/2016)

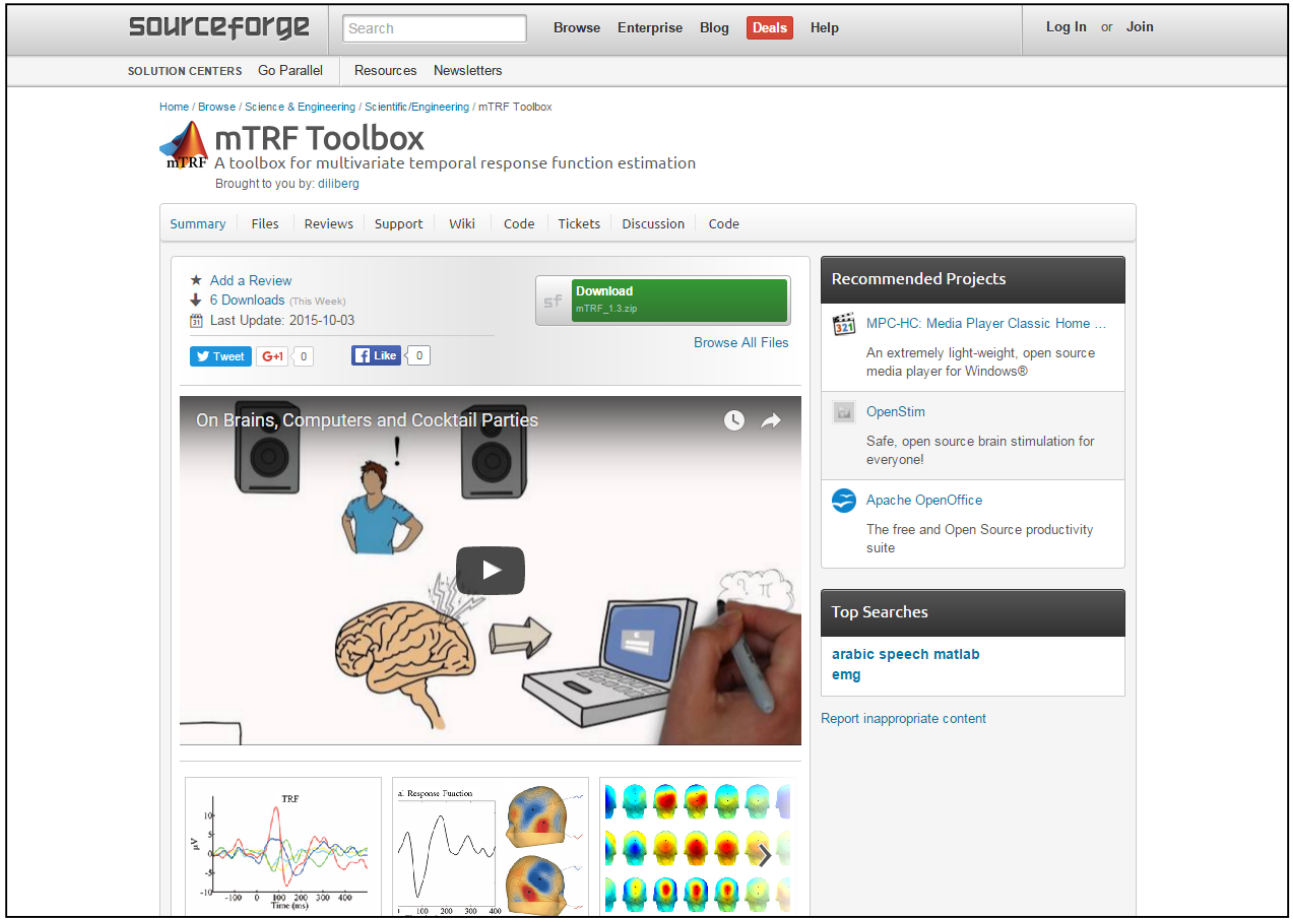
- This workshop continues to produce novel approaches and progress is being made towards a broader toolbox incorporating other machine learning approaches for relating naturalistic stimuli to multivariate neural data

Where can you find mTRF Toolbox?

Sourceforge: https://sourceforge.net/projects/aespa

or

Lab website: http://lalorlab.net



mickcrosse@gmail.com

edlalor@tcd.ie

Piotr Majdak

Teaching and Toolboxes

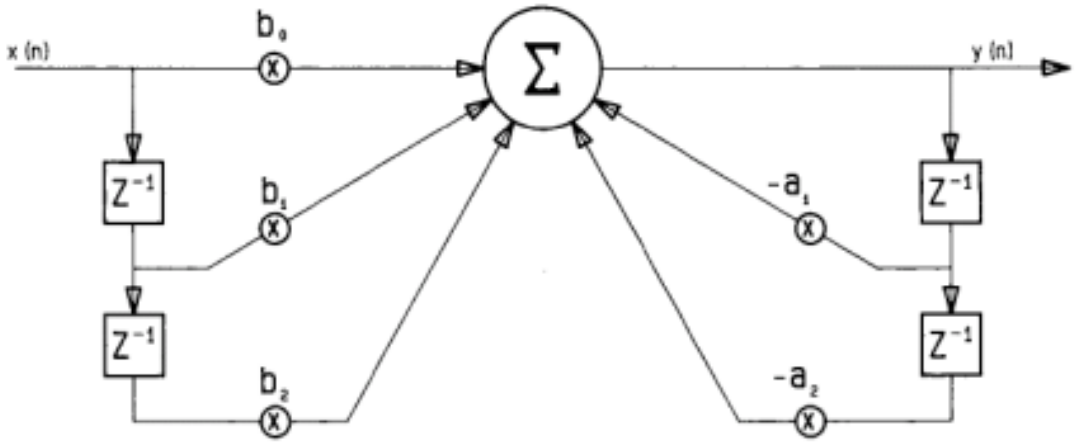
ARO 2016, San Diego, USA

Background

- Working with Matlab since 1999
- Teaching:
  - Algorithms in computer music and acoustics (since 2002)
  - Audio engineering (since 2014)
- Research:
  - Spatial hearing (NH and CI listeners)
  - Acoustic measurements

Background

- Coding, from the scratch



$$y(n)=b_0 x(n)+b_1 x(n-1)+b_2 x(n-2)$$
$$-a_1 y(n-1)-a_2 y(n-2)$$

Background

- Coding, from the scratch

```
function out=biquad_lp(in,fc,d,fs)

% biquad_lp          - low pass filter, canonical bi-quad form
% out=biquad_lp(in,fc,d,[fs])
%
% Implementation of low pass biquad filter in canonical form.
%
% input parameters:
% in:   input signal
% fc:   cut-off frequency [Hz]
% d:    damping factor
% fs:   sampling frequency
```

Background

- Coding, from the scratch

```
function out=biquad_lp(in,fc,d,fs)

  ...

    % calculate all coefficients
F=1/tan(pi*fc/fs);
b0=1/(1+2*d*F+F*F);
b1=2*b0;
b2=b0;
a1=2*b0*(1-F*F);
a2=b0*(1-2*d*F+F*F);
    % initialize vectors
len=length(in)+2;    % get the length + overhead
y=zeros(len,1);      % define y and x
x=zeros(len,1);
x(3:len)=in;        % merge to: zeros . in . zeros
    % main loop
for n=3:len
    y(n)=b0*x(n)+b1*x(n-1)+b2*x(n-2)-a1*y(n-1)-a2*y(n-2);
end
    % set the output
out=y(3:end);
```

Background

- Coding, from the scratch
- Goal: Students being able to code new stuff

- Advantage: students being able to code
- Disadvantage: takes time
- Work around:
  - basics in exercises
  - more interesting things in theory lectures with many examples

Background

- **Teaching**: Tools to simplify often used processes:
  - **spect**: plot a spectrum in context of audio engineering:
    - `spect(fft(my_time_signal));`
    - `spect(my_complex_valued_spectrum);`
  - **etc**: calculate the logarithmic energy-time curve
  - **expsweep**: create an exponential frequency sweep
  - **sysid_basic**: basic system identification for audio devices
  - Download: http://piotr.majdak.com

Background

- **SOFA API**: handling of HRTFs and BRIRs
  - implements the spatially oriented format for acoustics (**SOFA**, AES standard since last May)
  - load/save SOFA files
  - plot/analyze SOFA objects
  - spatialize sounds
  - relies on netcdf (standard library in Matlab)
  - Resources: http://www.sofaconventions.org/

Background

- Auditory modeling toolbox (**AMT**):
  - Collection of auditory models (currently 22 models)
    - Auditory periphery: Outer/middle/inner ear, auditory nerve
    - Monaural and binaural masking effects
    - Spatial effects like lateralization, localization, distance estimation, speaker recognition
    - Speech perception and speech reception in rooms

Background

- Auditory modeling toolbox (AMT):
  - Collection of auditory models (currently 22 models)
  - Focus on reproducibility of research data and algorithms

Background

Figure 9 in original publication (Langendijk and Bronkhorst, 2002):

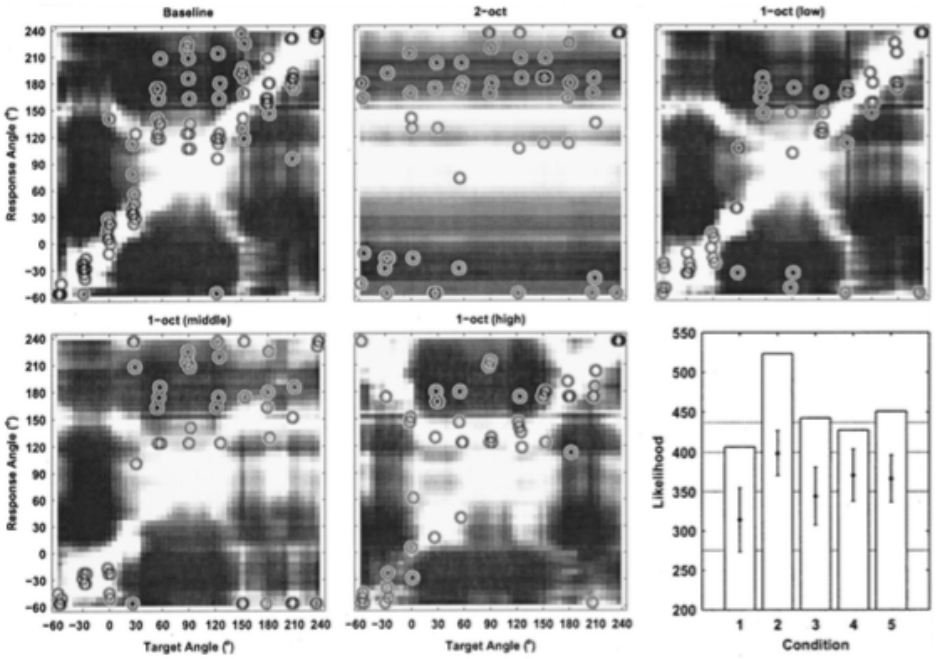
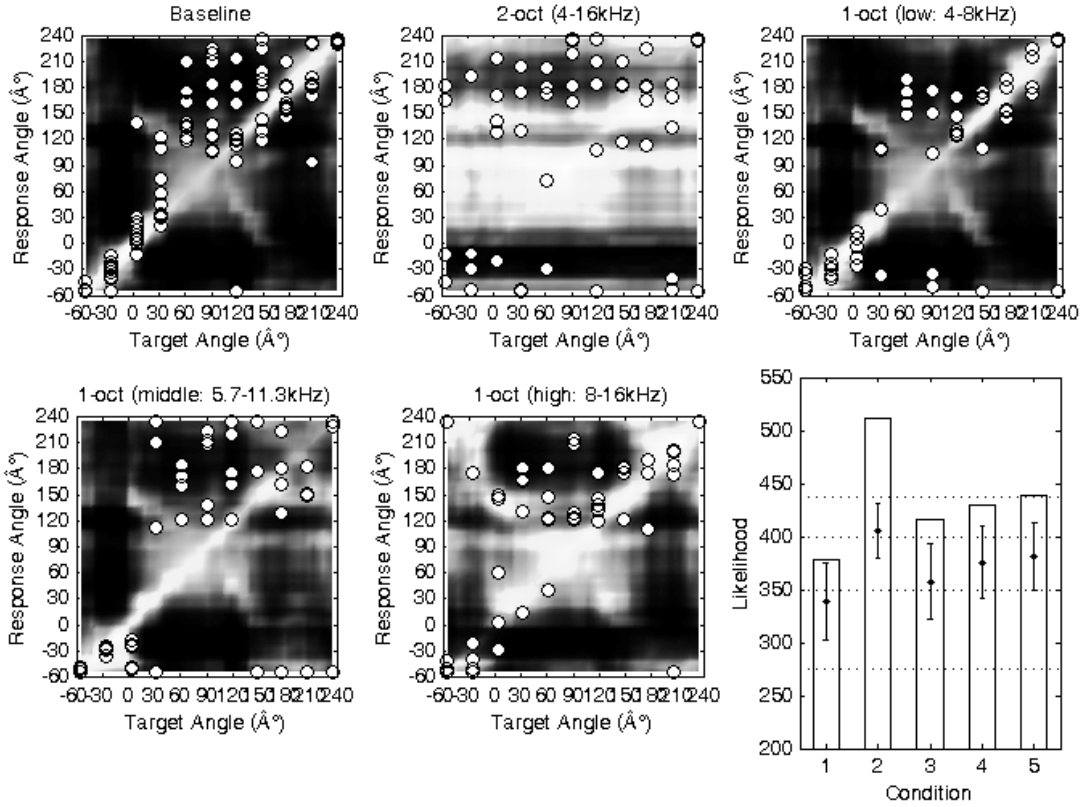


FIG. 9. Same as Fig. 7, but now showing the data of listener P3.

AMT: `exp_langendijk2002('fig9')`



Background

- Auditory modeling toolbox (AMT):
  - Collection of auditory models (currently 22 models)
  - Focus on reproducibility of research data and algorithms
  - Includes a repository of data and signals (>6 GB)
    - Download of auxiliary data on demand
    - Download of pre-calculated results
    - Caching calculated results

Background

- Auditory modeling toolbox (AMT):
  - Collection of auditory models (currently 22 models)
  - Focus on reproducibility of research data and algorithms
  - Includes a repository of data and signals (>6 GB)
  - Integrates/relies on other toolboxes/systems:
    - the **SOFA** API for HRTFs
    - the sound field synthesis toolbox (**SFS**, Wierstorf and Spors)
    - the large time-frequency analysis toolbox (**LTFAT**, Soendergaard et al.) for time-frequency signal processing
    - MEX files (compiling C and C++ files)
    - Python

Background

- Auditory modeling toolbox (AMT):
  - Collection of auditory models (currently 22 models)
  - Focus on reproducibility of research data and algorithms
  - Includes a repository with data and signals (>6 GB)
  - Integrates/relies on other toolboxes/systems
  - Implements documentation directly in the header
    - Compiles to PHP for a website
    - Compiles to clean .M files for Matlab's help system

Background

- Auditory modeling toolbox (AMT):
  - Collection of auditory models (currently 22 models)
  - Focus on reproducibility of research data and algorithms
  - Includes a repository with data and signals (>6 GB)
  - Integrates/relies on other toolboxes/systems
  - Implements documentation directly in the header
  - Still in development (alpha status; changes may affect backward compatibility):
    - Download version: 0.9.7
    - Plan for 2016: release of version 1.0

Background

- Teaching material: http://piotr.majdak.com
- SOFA API (HRTFs):
  - Download: http://sourceforge.net/projects/sofacoustics
  - Sources: https://github.com/sofacoustics/sofa
  - Documentation and files: http://www.sofaconventions.org/
- The AMT:
  - Download: http://sourceforge.net/projects/amtoolbox
  - Documentation: http://amtoolbox.sourceforge.net/

Supported by the FWF (P 24124) and the EU (ALT, #691229)

Experiences Sharing Auditory Models in Matlab

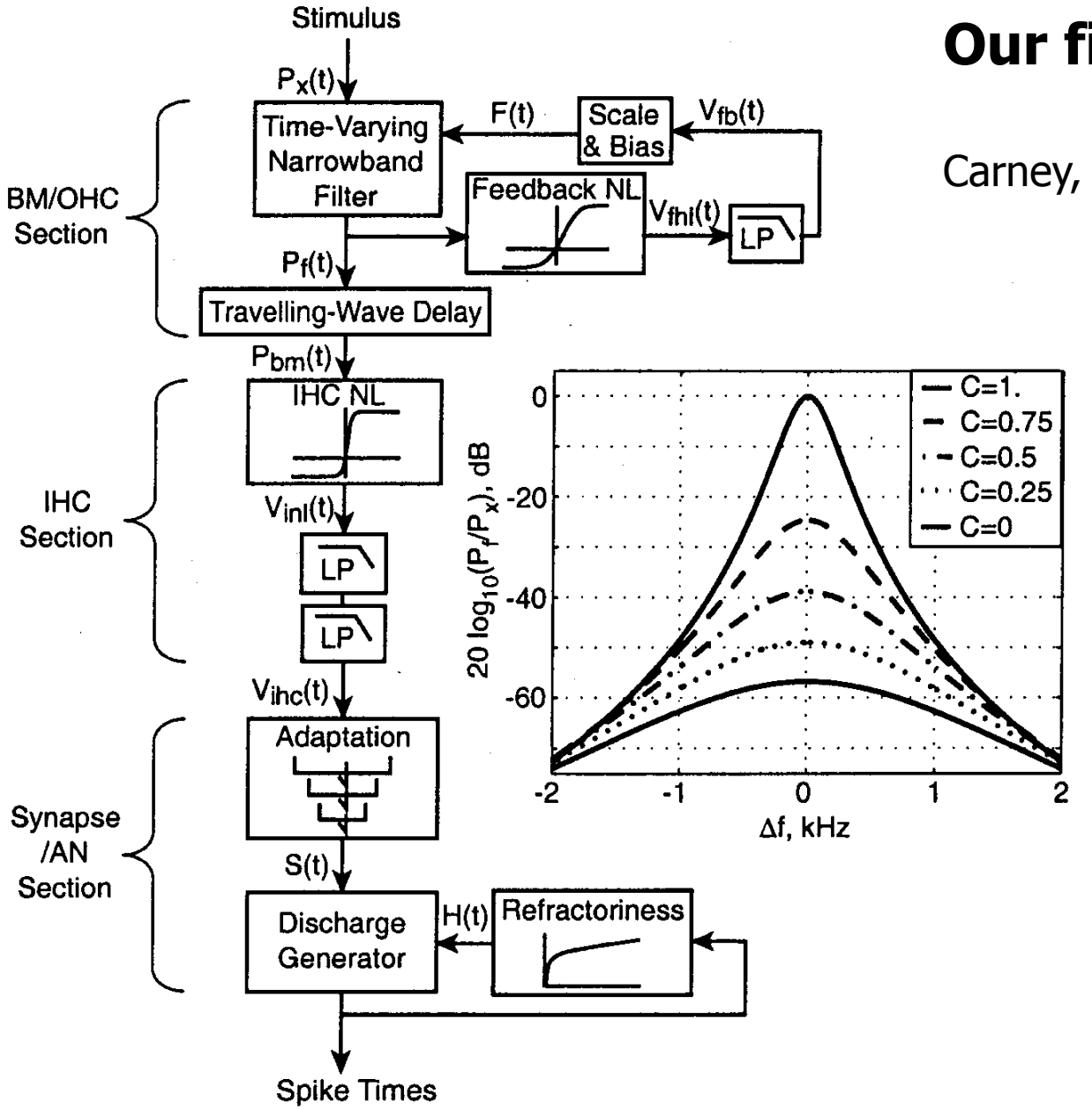
Laurel Carney

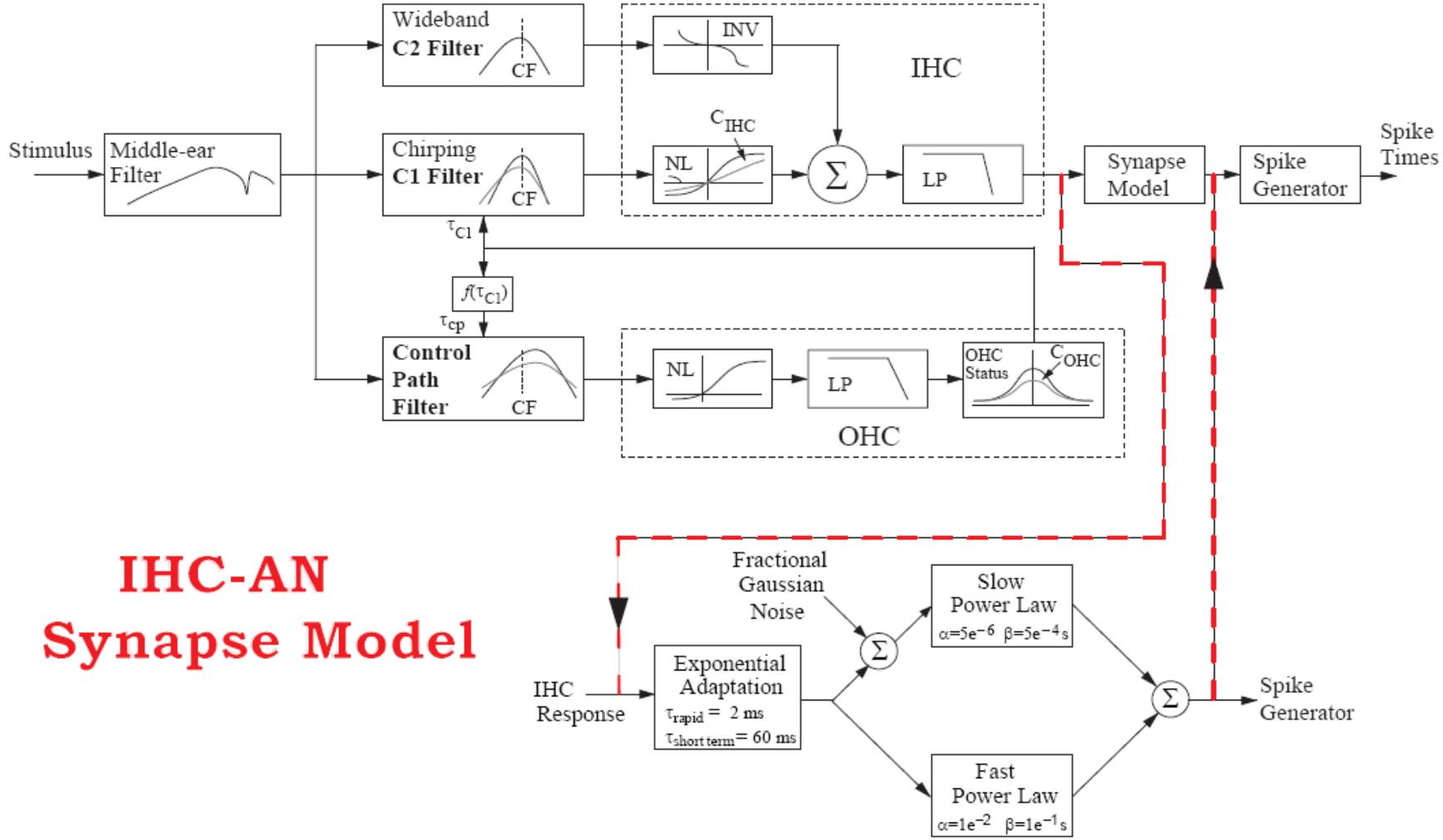
University of Rochester

https://www.urmc.rochester.edu/labs/Carney-Lab/publications/auditory-models.cfm

Our first effort to share code:

Carney, 1993 - Written in C (from Fortran)
1st shared via floppy disk





Zilany et al, 2009, 2014 – most recent version of AN model

What has worked?

- Make code freely and easily available –
  - Post it (rather than having users request it personally)
  - Readme file
  - Readable code (no professional programmers!)
  - “anmodel_test.m” – simple sample m-file that runs simulation & plots response
- (Try to) Share only Published models
  - Post models & papers together
- Have faith in evolution!
  - (If it were up to me alone, the code would be in BASIC!)
  - Matlab is a good solution today, but our students will decide on the next gen….

What's New? ***UR EAR***

Visualizing Population responses of AN fibers and Amplitude-Modulation Tuned Cells in Inferior Colliculus

Students:
Natalia Galant
Braden Maxwell
Danika Teverovsky
Thomas Varner

Existing AN & IC models
Matlab Tool, to be posted soon

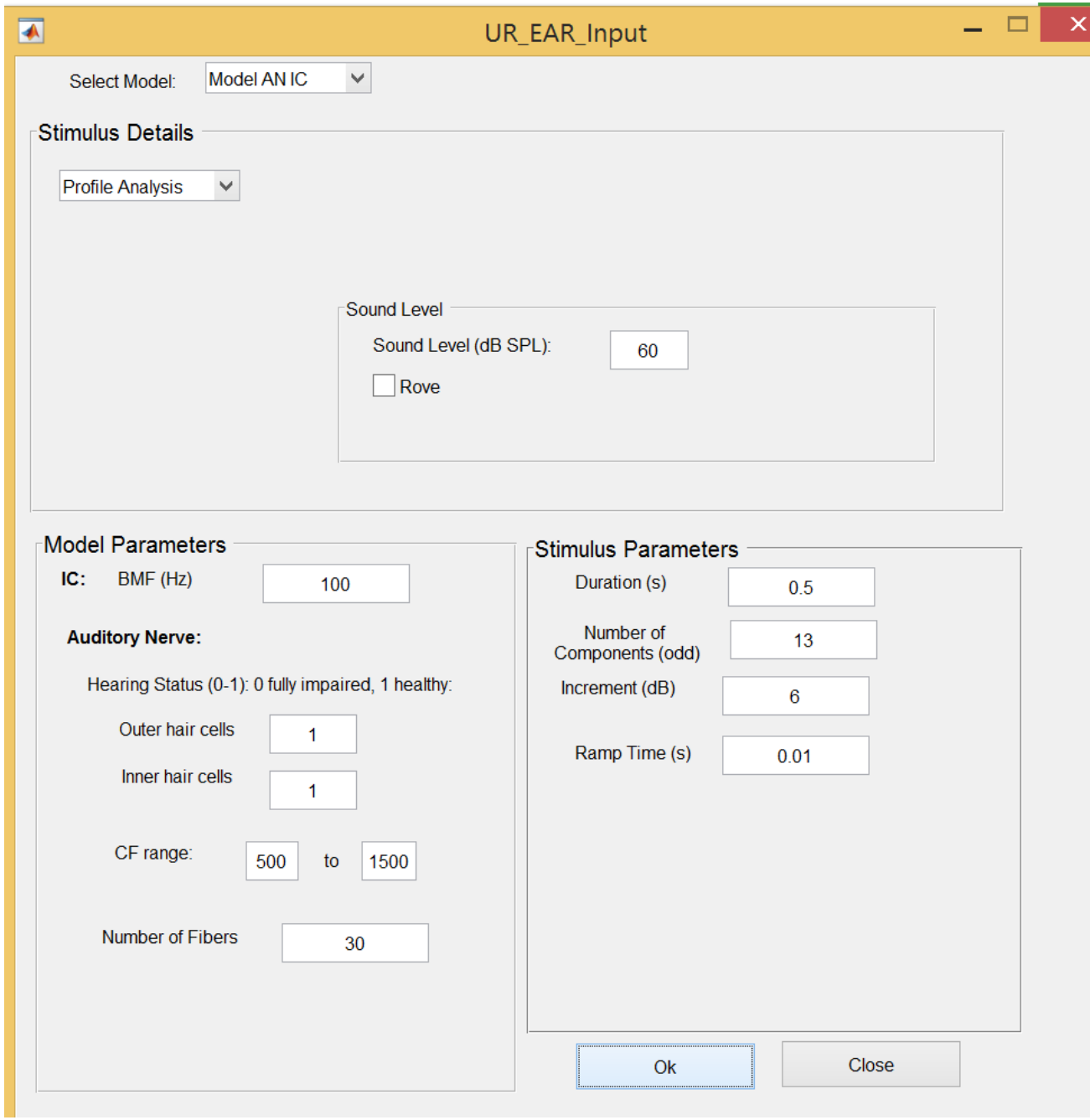
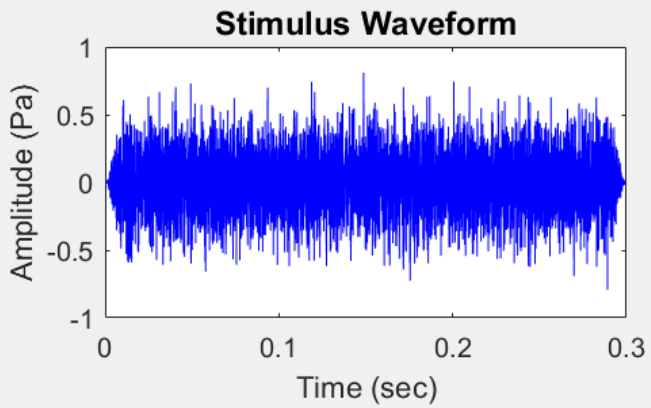
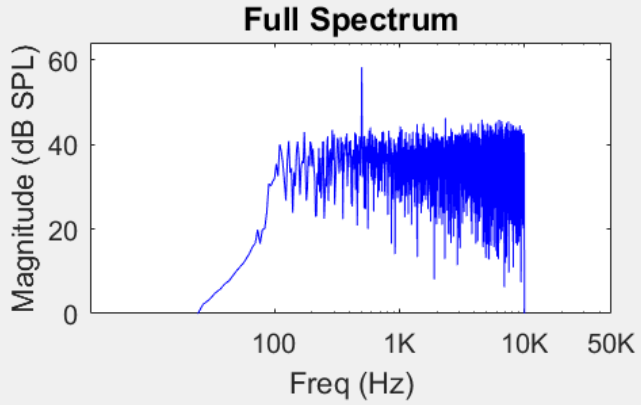
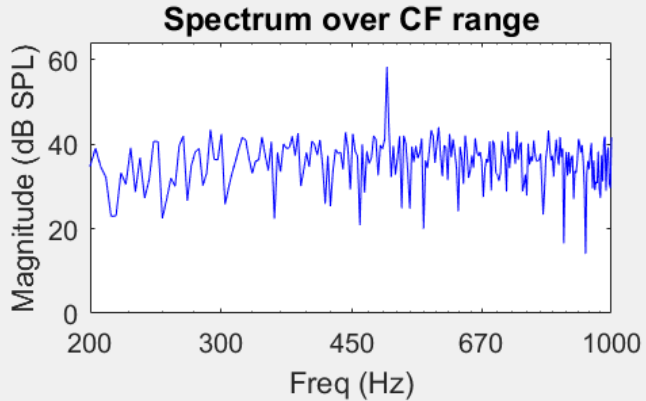


Figure 9

File Edit View Insert Tools Desktop Window Help



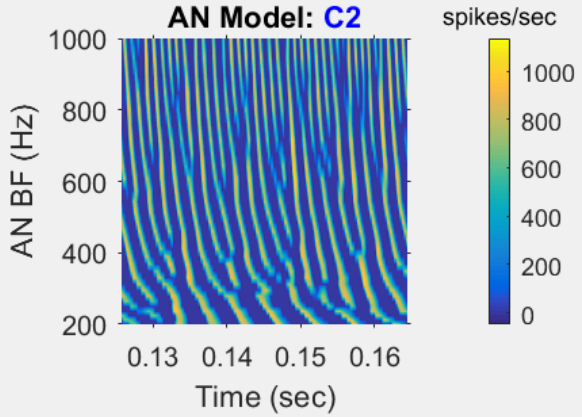


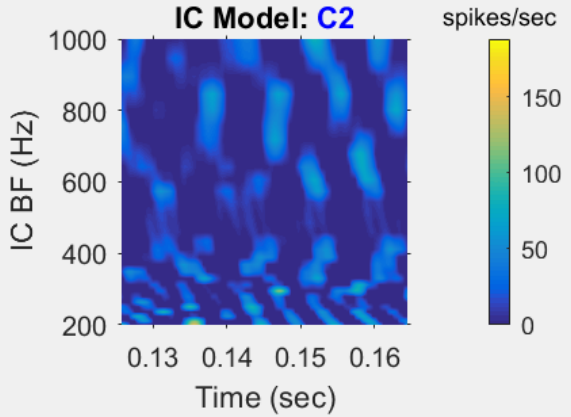


Tone in Noise

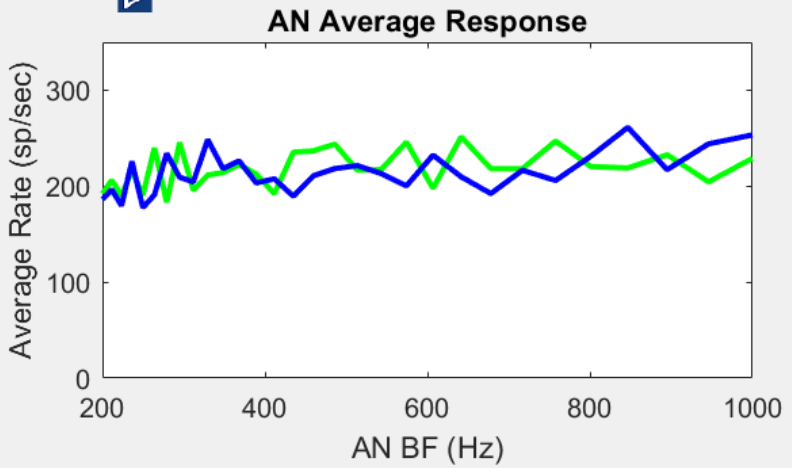
C1: Without Tone

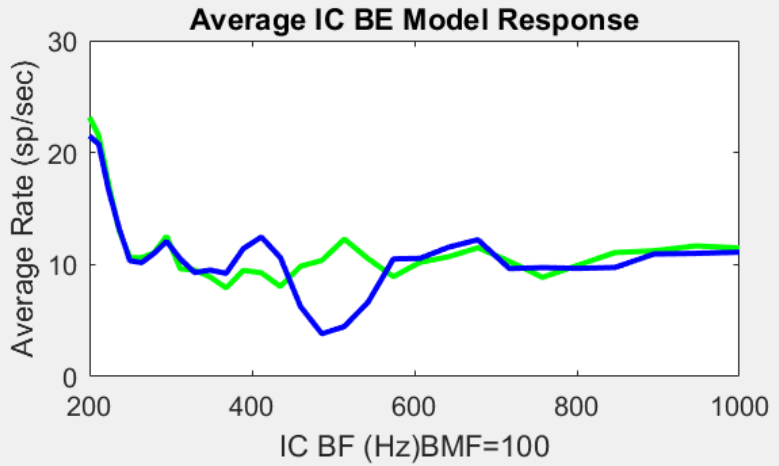
C2: With Tone





UR EAR 1.0

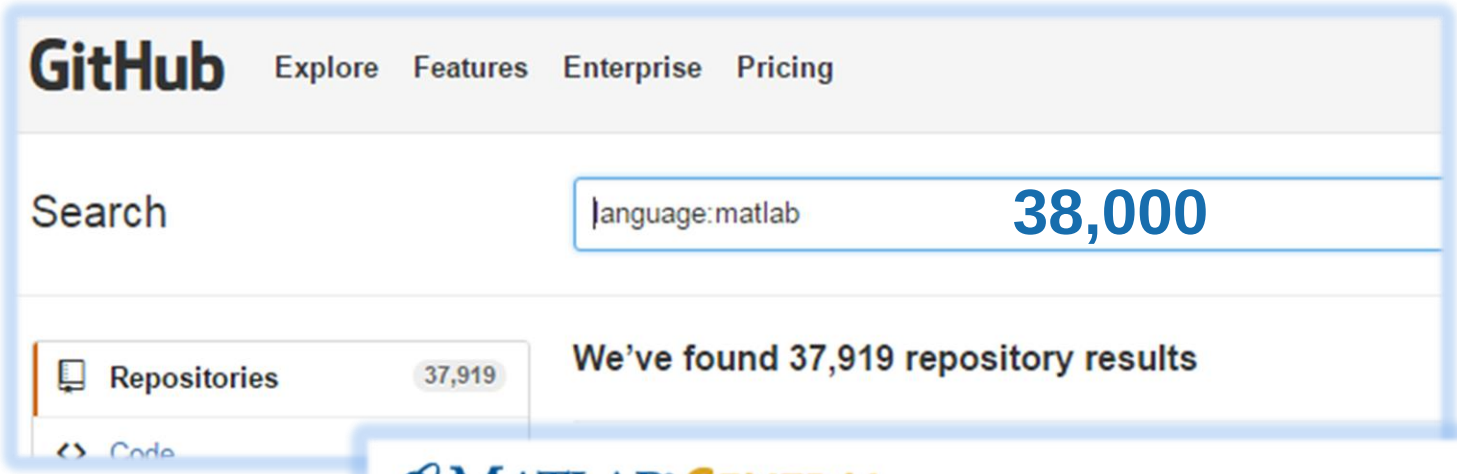


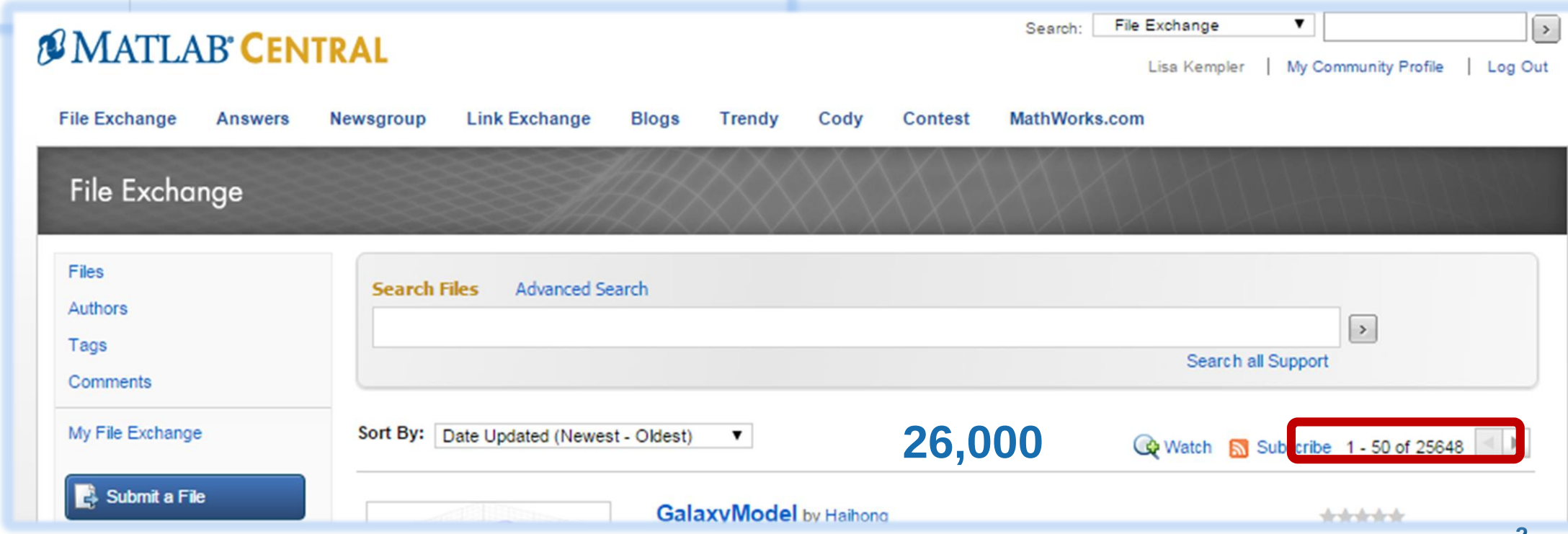


Speaking MATLAB – Implanting Community Best Practices

Lisa Kempler, MATLAB Community Strategist
lisak@mathworks.com
ARO Winter Meeting
February 23, 2016

Lots of Code!!! Multiple Repositories . . . .





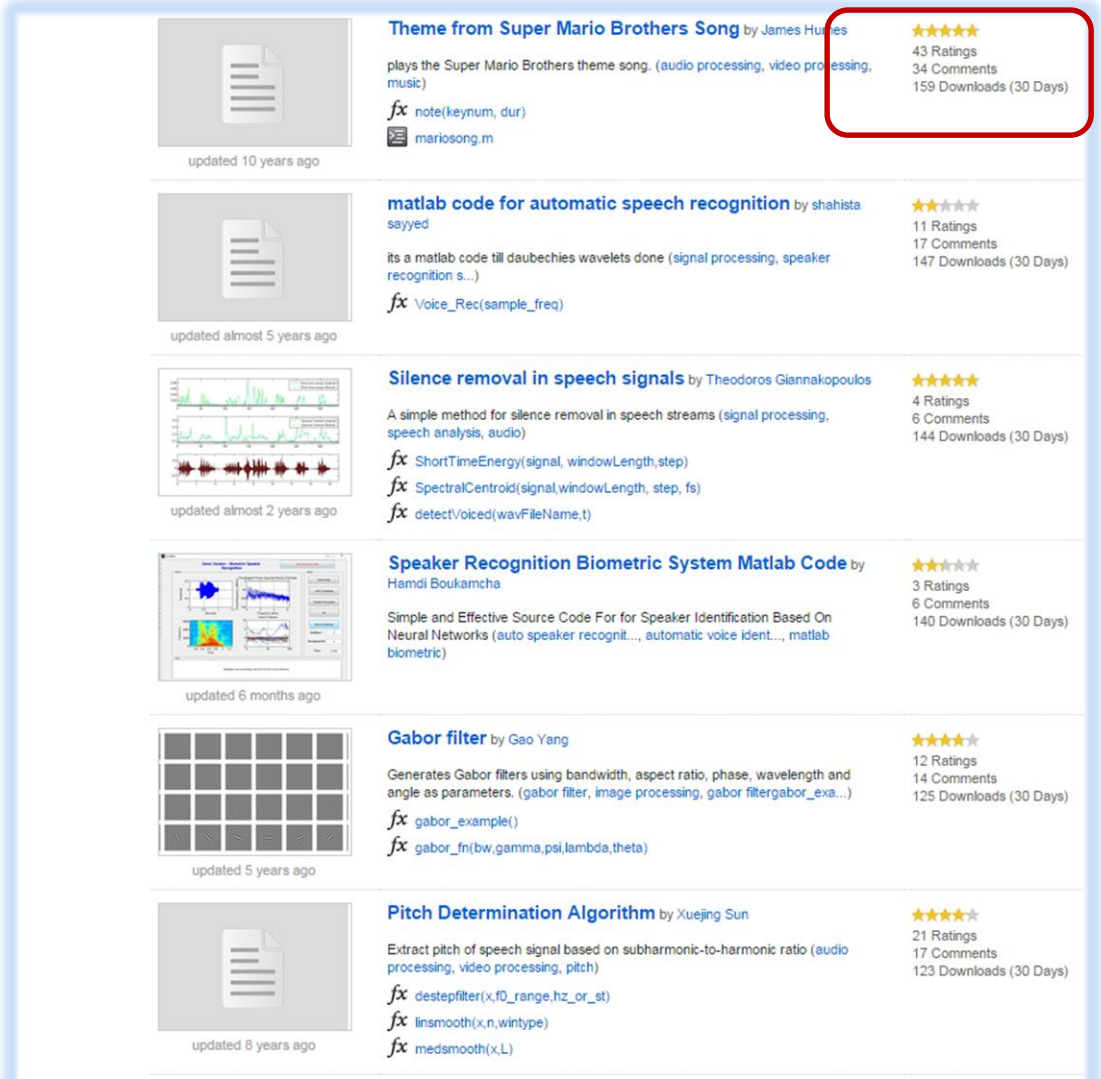
The Grad Student Legacy: Unfinished Code

Source:
Photo, concept
Ian Foster's
EC AHM talk

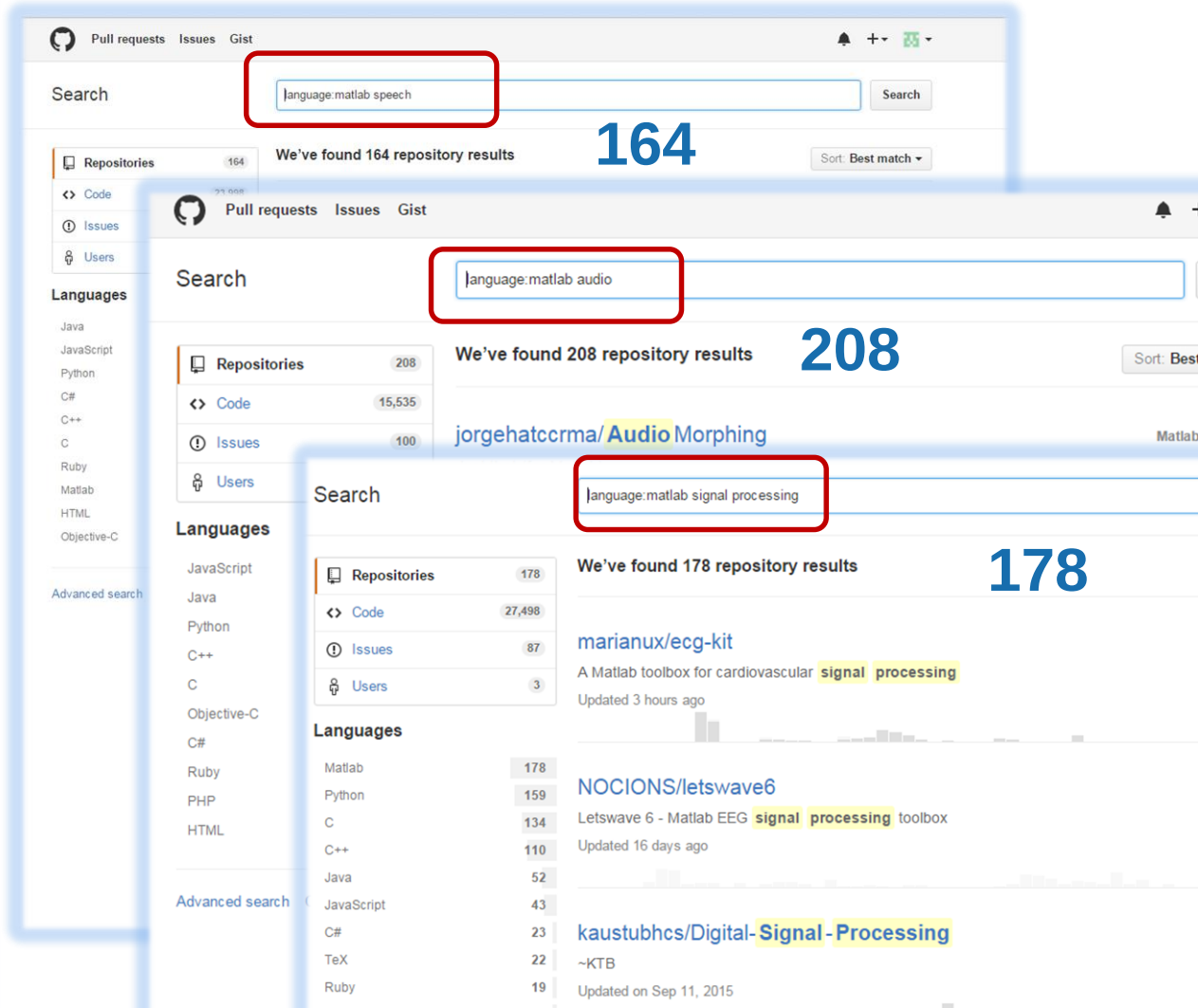
Lots of Audiology, Speech Code: Is it Vetted? Discoverable?

“Speech” Code on File Exchange

331



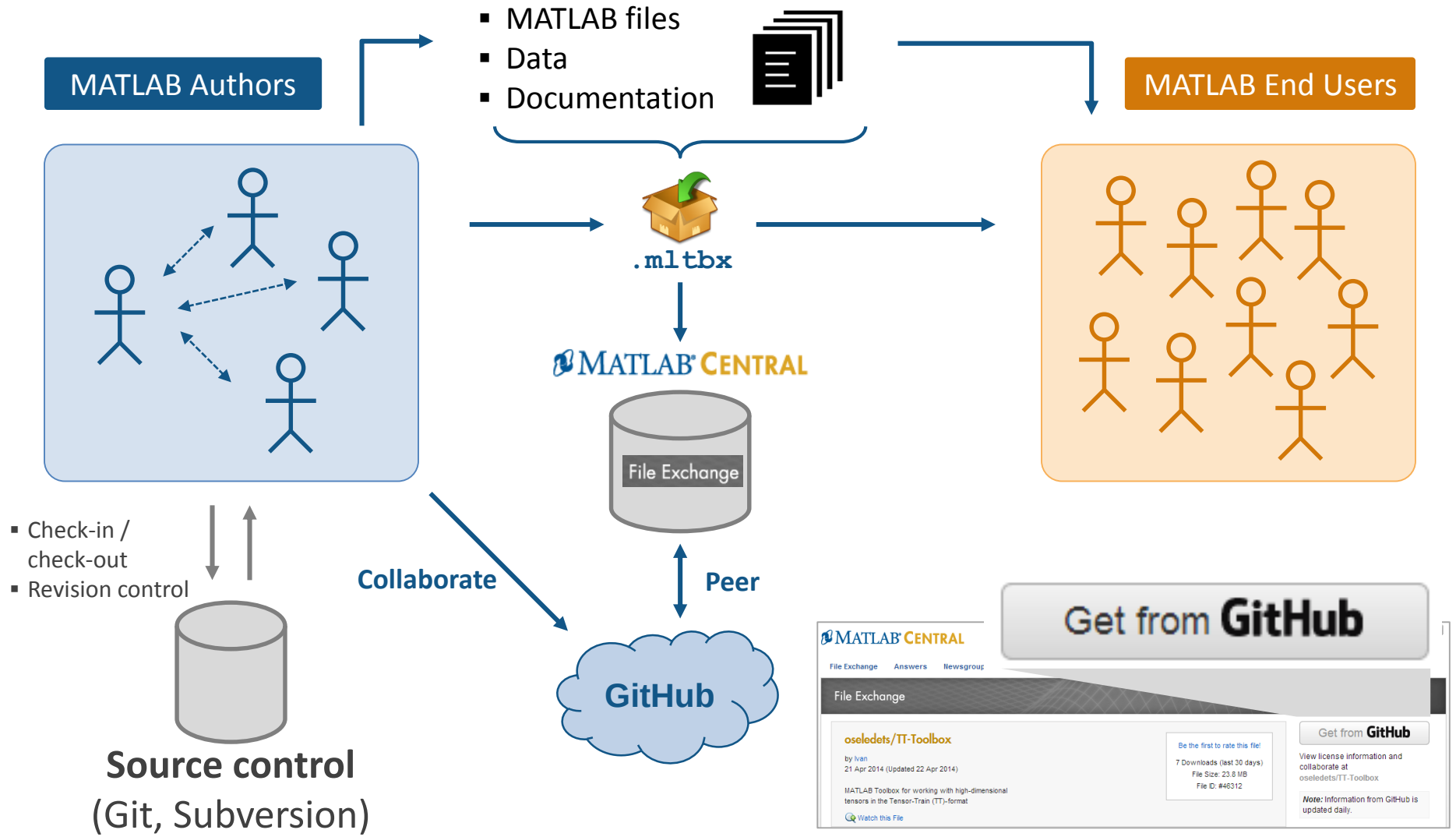
GitHub: “Speech”, Audio, Signal Processing Code



Why Share Tools in Public Repositories?

- Conservation of Resources (Reuse)
- Collaboration
- Measure Value (Downloads, Rating)
- Helps you be organized and accountable!

Managing & Sharing MATLAB Code



SPM

By members & collaborators of the Wellcome Trust Centre for Neuroimaging

Introduction | Software | Documentation | Courses | Email list | Data | Extensions

Software

Introduction

SPM is made freely available to the [neuro]imaging community, to promote collaboration and a common analysis scheme across laboratories. The software represents the implementation of the theoretical concepts of Statistical Parametric Mapping in a complete analysis package.

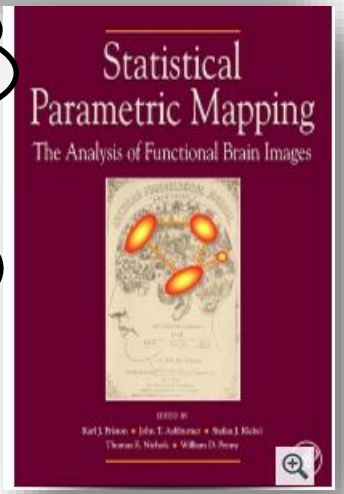
The SPM software is a suite of MATLAB (The MathWorks, Inc) functions and subroutines with some externally compiled C routines. SPM was written to organise and interpret our functional neuroimaging data. The distributed version is the same as that we use ourselves.

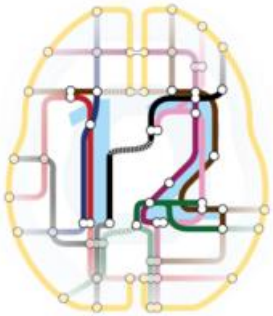
NRIAlign *- Motion correction with reduced paradigm-correlation bias*

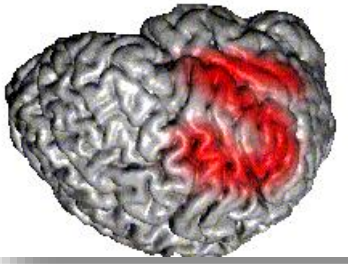
ALI *- Automated Lesion Identification*

SAMIT *- Small Animal Molecular Imaging Toolbox*

at4fmri - adaptive thresholding of fMRI maps







Sharing platforms	**SPM computing tools**
Datasets	**Computing infrastructure**

PET
fMRI

MATLAB®
C

[SPM main site]
[SPM add-on MATLAB toolboxes]

Reproducible Research *and* Teaching (through Sharing)

Scientists Do Reproducible Research with MATLAB. (Also known as open science).

Loren on the Art of MATLAB

< Run Code Faster With the...

Recent Posts Archive

15 FEB Reproducibility Musings - Hey, do that again!

12 FEB Run Code Faster With the New MATLAB Execution Engine

3 FEB Visualizing Facebook Networks with MATLAB

20 JAN Mapping Uber Pickups in New York City

6 JAN Generating an Optimal Employee Work Schedule Using Integer Linear Programming

Categories

Best Practice	56
Robustness	31
Big Data	7
Common Errors	42
Less Used Functionality	60

more »

Recent Comments

Chad Bernier on Mapping Uber Pickups in New York City

Loren Shure on Run Code Faster With the New MATLAB Execution Engine

Brad Stiritz on Run Code Faster With the New MATLAB Execution Engine

Reproducibility Musings – Hey, do that again!

Posted by **Loren Shure**, February 15, 2016

Today I have a guest post from Lisa Kempler, MATLAB Community Strategist, MathWorks. Lisa works with MATLAB communities in domains such as geophysics, oceanography, audiology, and more, helping users developing MATLAB based tools, and creating resources for teaching and research with MATLAB. A primary goal is to have MATLAB users share their tools and best practices more widely within their communities, enhancing their use of MATLAB, and, in turn, accelerating their work. This blog talks about some efforts in research that support that sharing objective.

Contents

- What is Reproducibility?
- Why is it important?
- Getting Consensus on Reproducible Research Needs and Approaches
- Individual Researcher Efforts and MATLAB Related Tools and Capabilities
- Your Turn
- Kevin Moerman's Affiliations

What is Reproducibility?

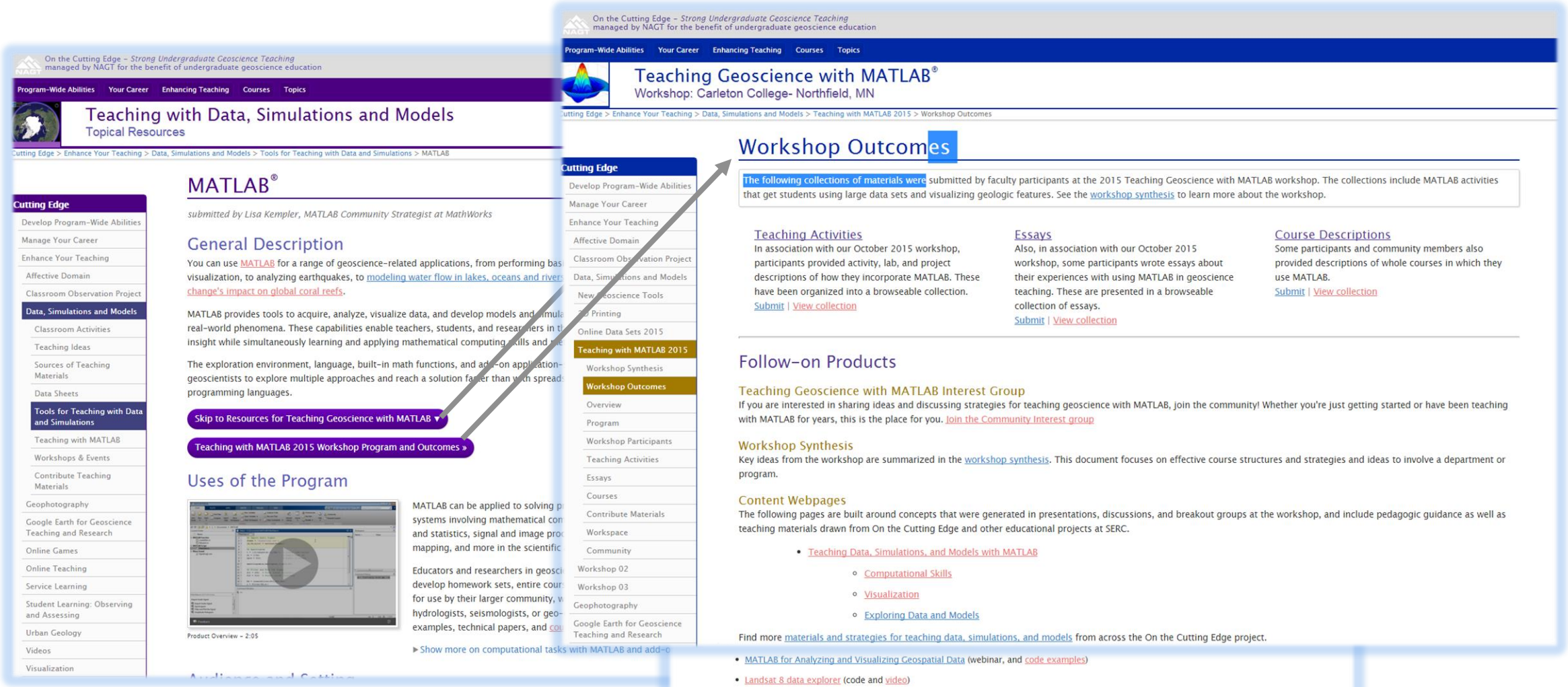
There's been lots of buzz around the topic of Reproducible Research. Proponents as well as the researchers who would be impacted by new reproducibility expectations are raising a lot of questions, questions about

- Definition and Scope:

```
- Simply put, what does Reproducible Research mean?
```

Teaching Geoscience with MATLAB: Resources, Community

(Also relevant for researchers, and educators in physics, applied math, statistics, etc.)



http://serc.carleton.edu/NAGTWorkshops/data_models/toolsheets/
https://serc.carleton.edu/NAGTWorkshops/data_models/matlab15/

SERC Site and Workshop Results:
Teaching Geoscience with MATLAB

MATLAB page for Geoscience Educat

- http://serc.carleton.edu/NAGTWorkshop
tml
- Useful content for multiple science and
disciplines, and researchers

Google
"MATLAB SERC"

Workshop Outcomes and Resources

- https://serc.carleton.edu/NAGTWorkshops/data_models/matlab15/outcomes.html
- Pay special attention to
  - Activities (lesson plans) and essays on teaching
  - Teaching approaches pages for philosophy re: computing, quantitative skills development

Should Computational Thinking be Ubiquitous?

Learning to Code? + **Coding to Learn?**

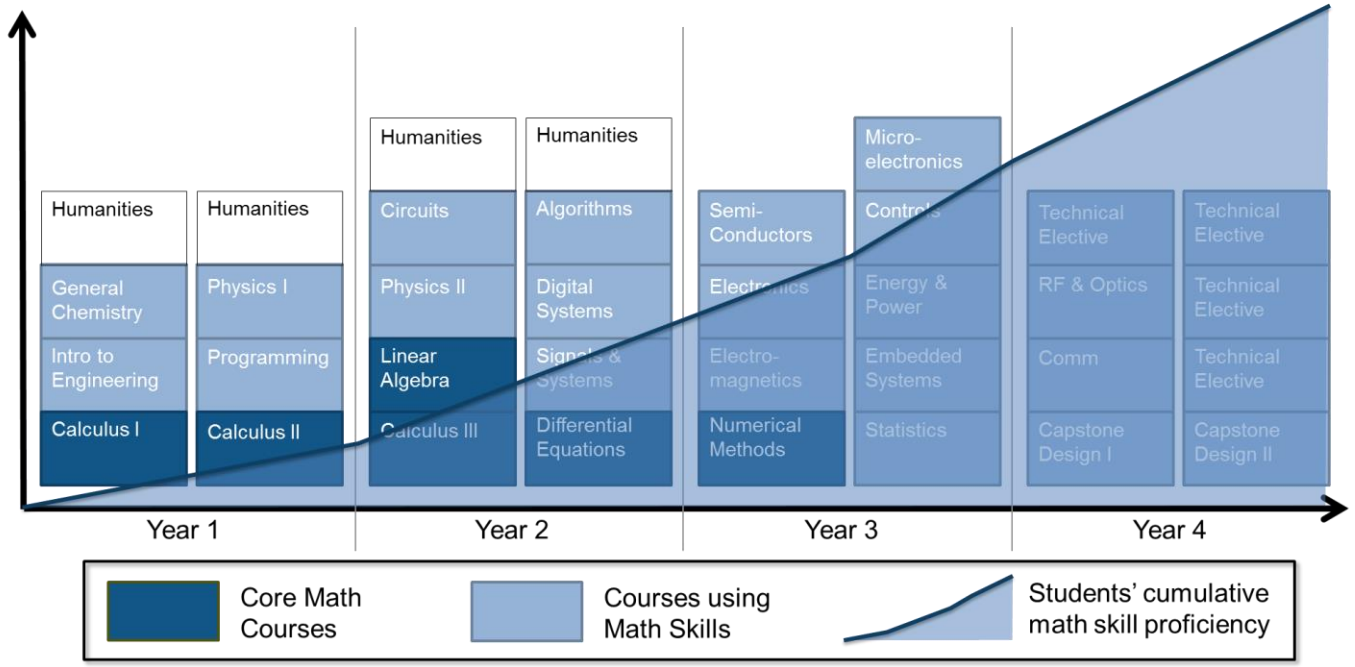
"Computational thinking is a fundamental skill for everyone, not just for computer scientists."

Dr. Jeannette Wing, Vice President of Microsoft Research
Former Department Head of CS at Carnegie Mellon University

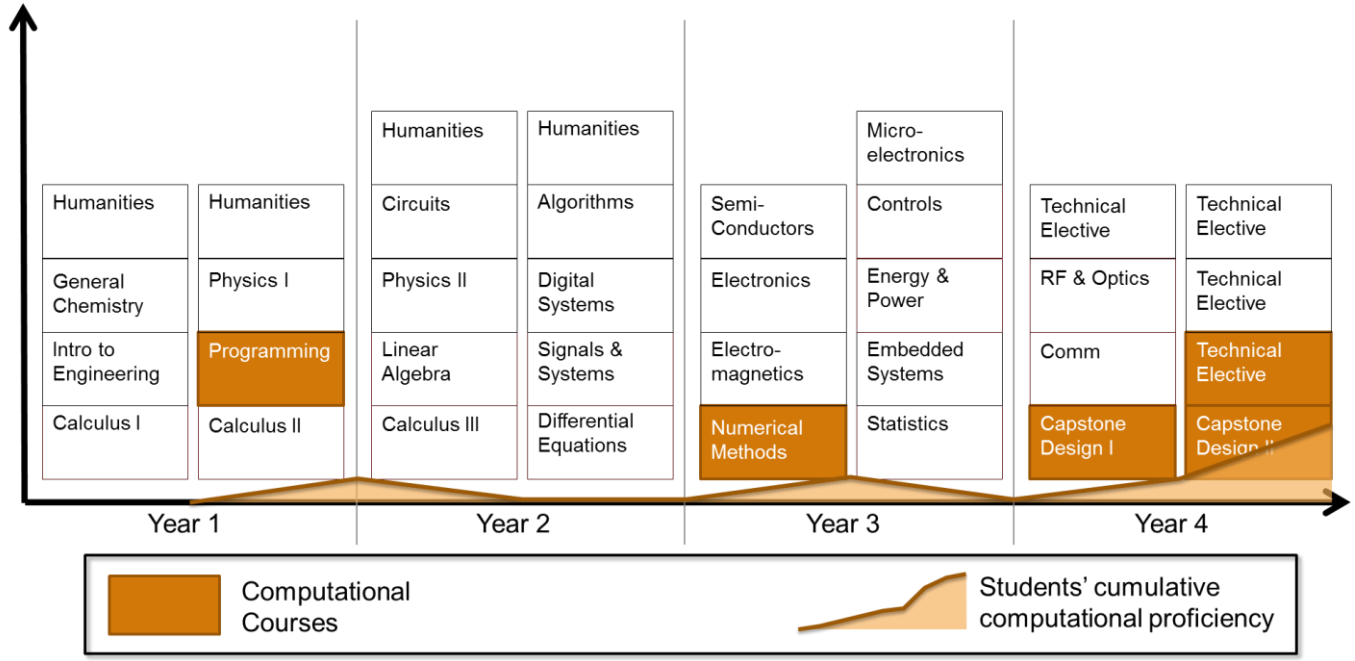
"Coding teaches me to think in a logical way"

Trinity School high school student
Accepted at MIT

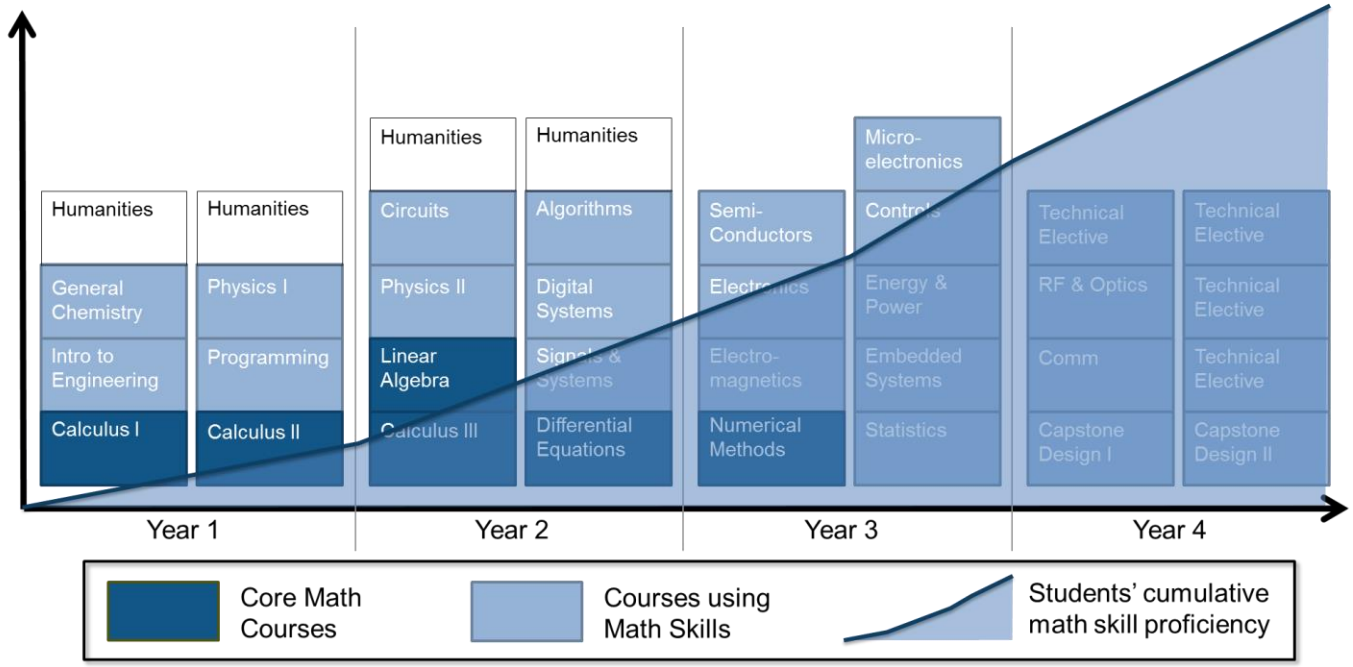
How Math is introduced in the curriculum



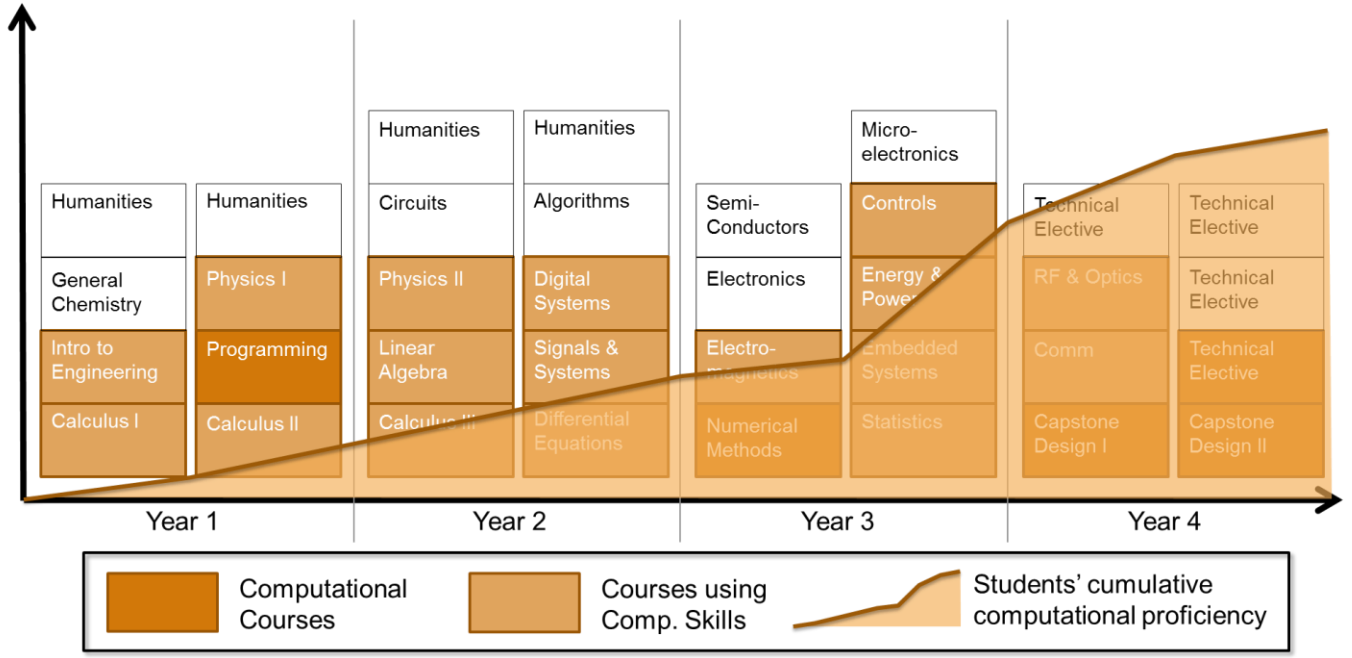
How is Computational Thinking introduced?



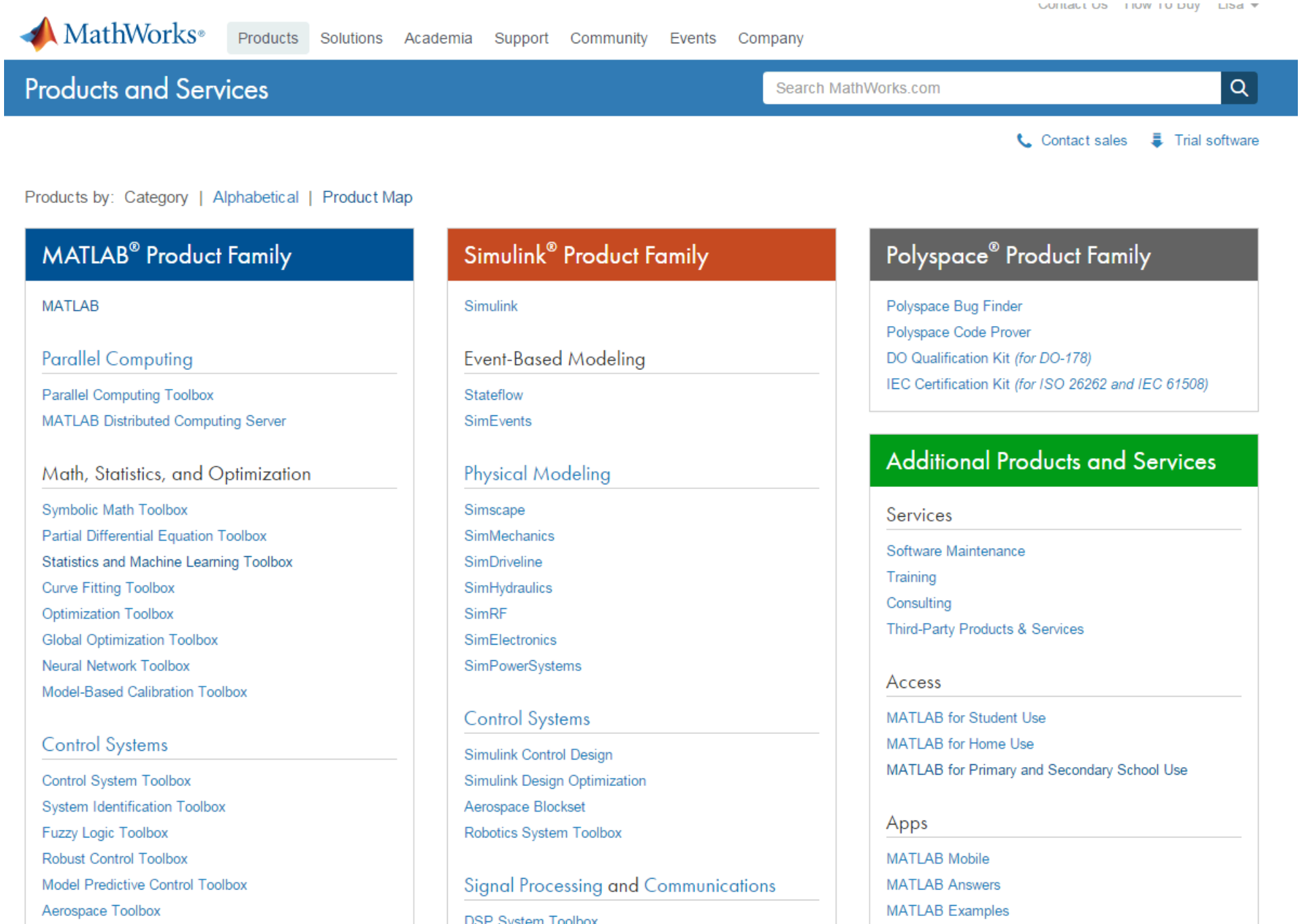
How Math is introduced in the curriculum



Should Computational Thinking be built up like we introduce Math?

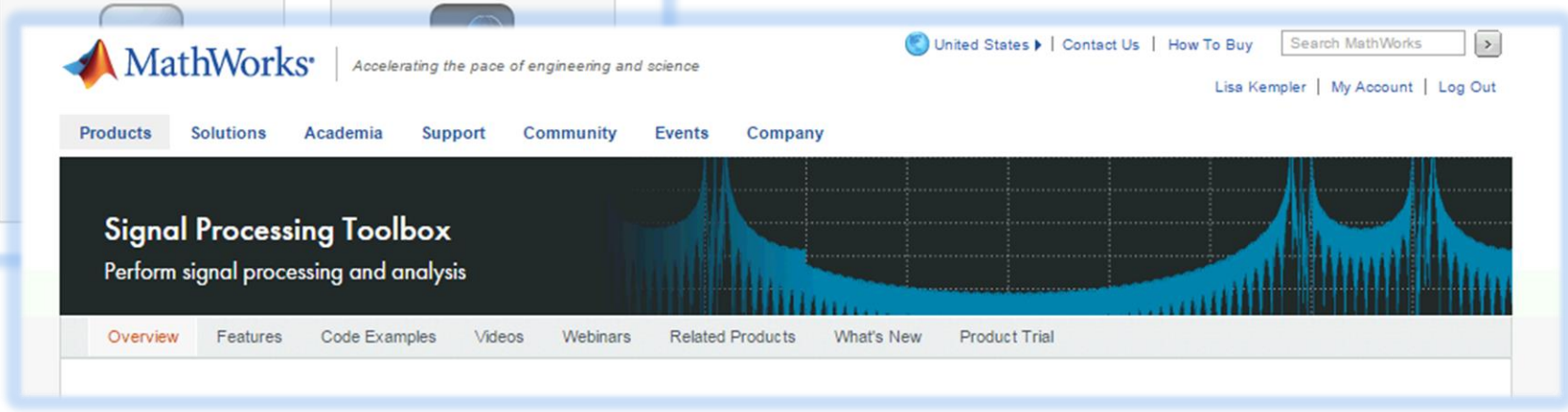


MATLAB Tools from MathWorks for Research and Teaching



[MATLAB and Simulink Products]

MATLAB Toolboxes

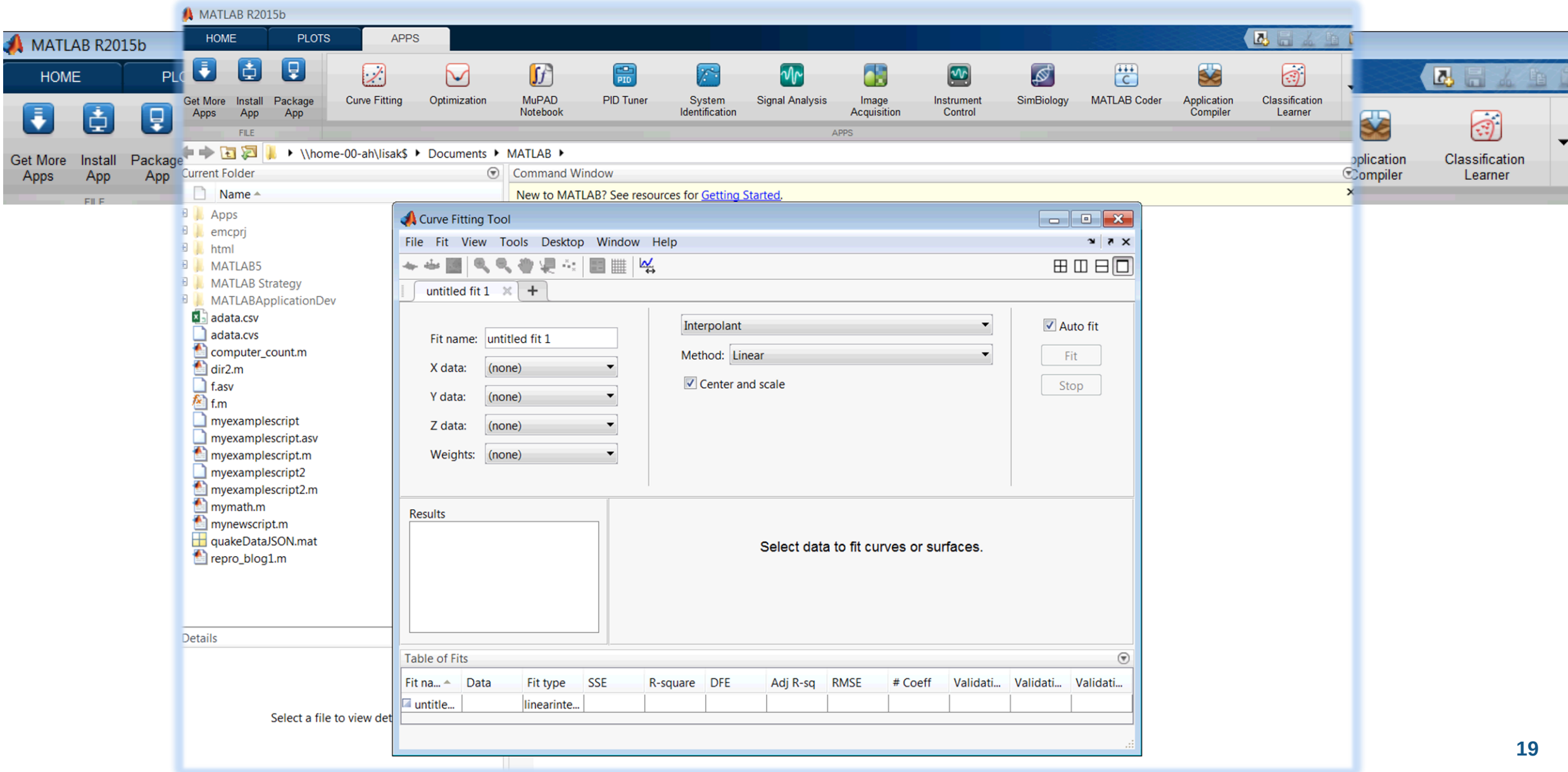


See code examples

Watch videos

Add your own documentation

MATLAB Apps Tab and Curve Fitting App



MATLAB in the Cloud and on Your Phone: MATLAB Online and MATLAB Mobile

Contact Us How To Buy Lisa

MathWorks Products Solutions Academia Support Community Events Company

MATLAB Online
Tech Preview

Search MathWorks.com

Overview | Features | Limitations | System Requirements

Access MATLAB from your web browser

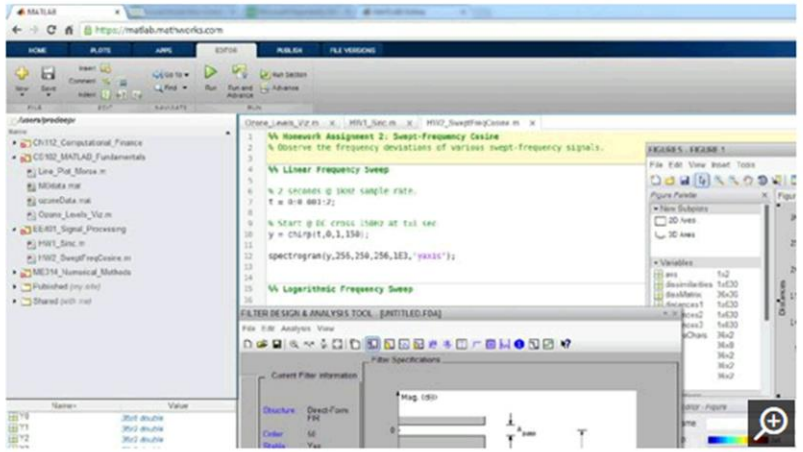
MATLAB Online™ provides access to MATLAB® from your web browser. Just log in to use MATLAB.

With MATLAB Online, your files are stored on MATLAB Drive™ and are available wherever you go. MATLAB Drive Connector synchronizes your files between your computers and MATLAB Online, providing offline access and eliminating the need to manually upload or download files. You can also run your files from the convenience of your smartphone or tablet by connecting to MathWorks Cloud through the MATLAB Mobile™ app.

MATLAB Online facilitates collaboration by enabling you to share files with friends or publish them as formatted HTML or PDF documents.

MATLAB Online is available for use with MATLAB Student and MATLAB and Simulink® Student Suite R2014a or later. The following toolboxes are available at this time:

- Computer Vision System Toolbox
- Control System Toolbox
- Curve Fitting Toolbox
- Econometrics Toolbox
- Financial Toolbox
- Financial Instruments Toolbox
- Fuzzy Logic Toolbox
- Global Optimization Toolbox
- Model Predictive Control Toolbox
- Neural Network Toolbox
- Optimization Toolbox
- Partial Differential Equation Toolbox
- Phased Array System Toolbox
- Robust Control Toolbox
- Signal Processing Toolbox
- Statistics and Machine Learning

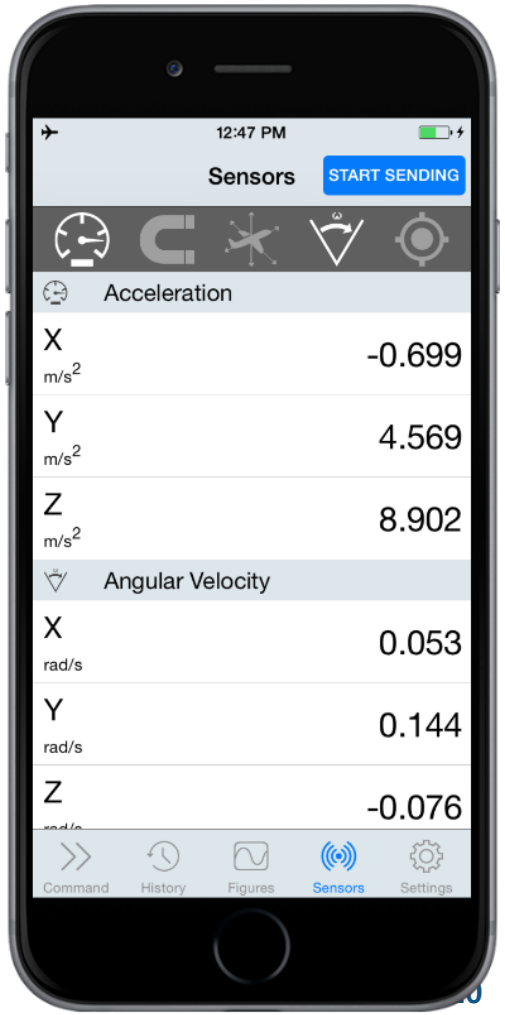


MATLAB Online

Start using MATLAB Online

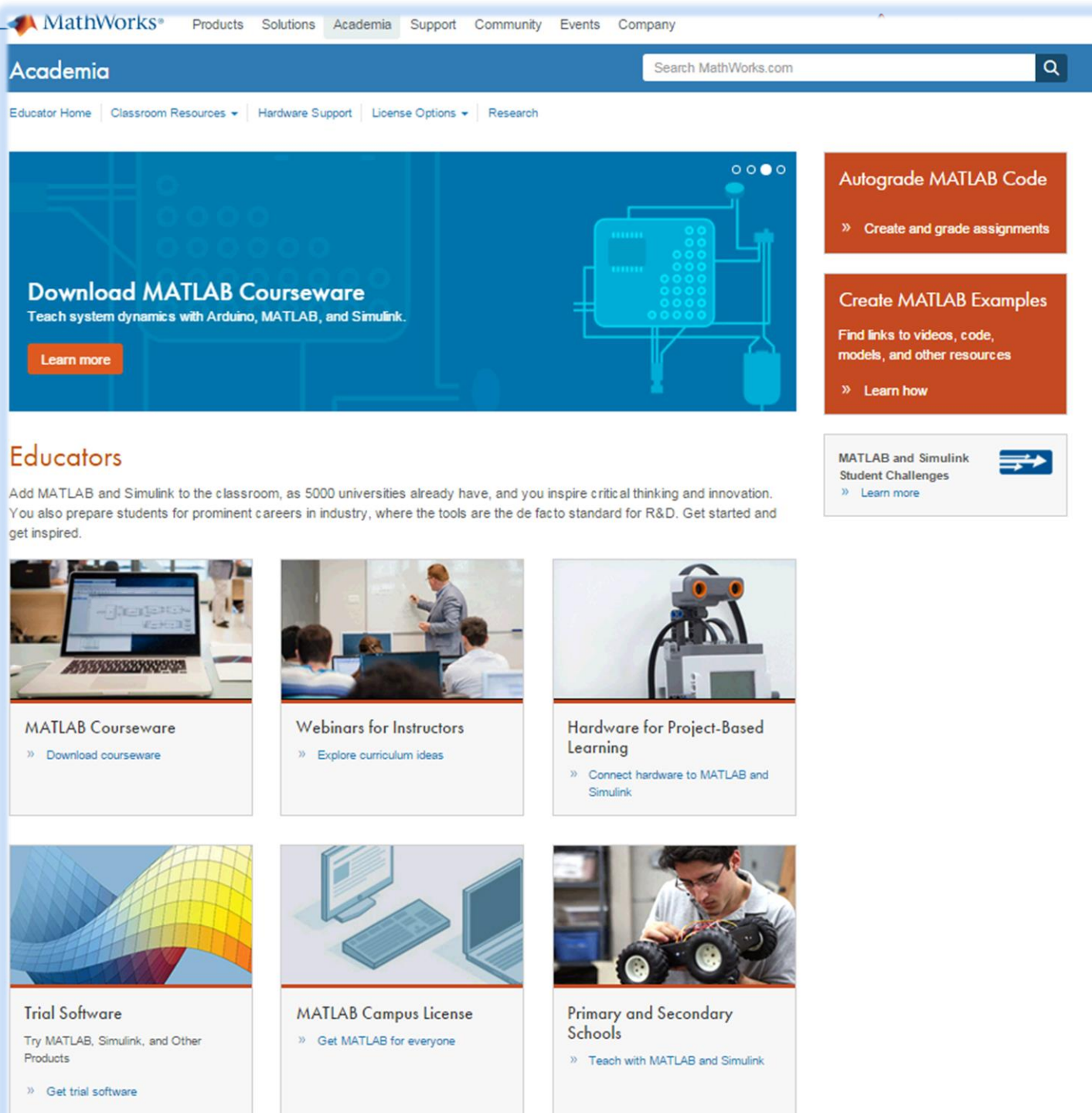
Synchronize files with MATLAB Drive

MATLAB Mobile for iOS or Android



Support for Educators

- Courseware
- Autograding
- On-Ramp Tutorials & Videos
- Code examples
- All-Campus License
- MATLAB Online



Next Steps

LOOP

- Tell each other what you're doing
- Make your tools and methods available for reuse

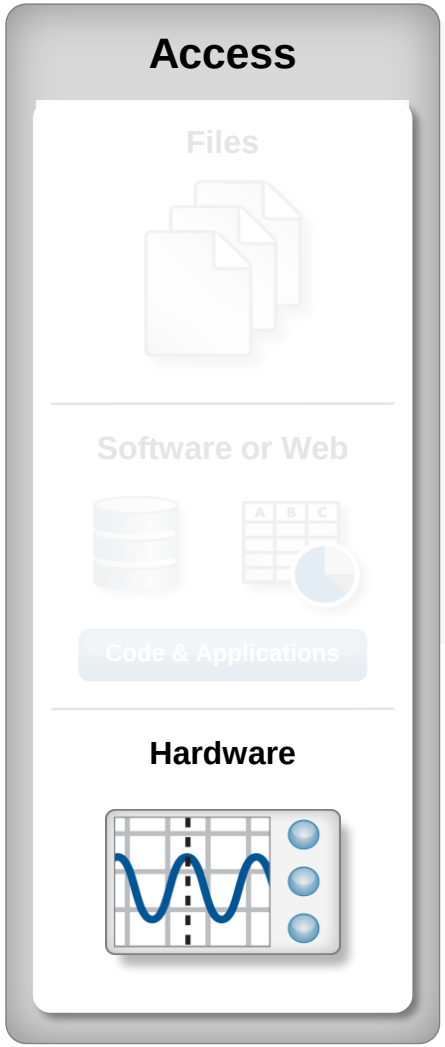
How will ARO share best practices and resources in 2016?

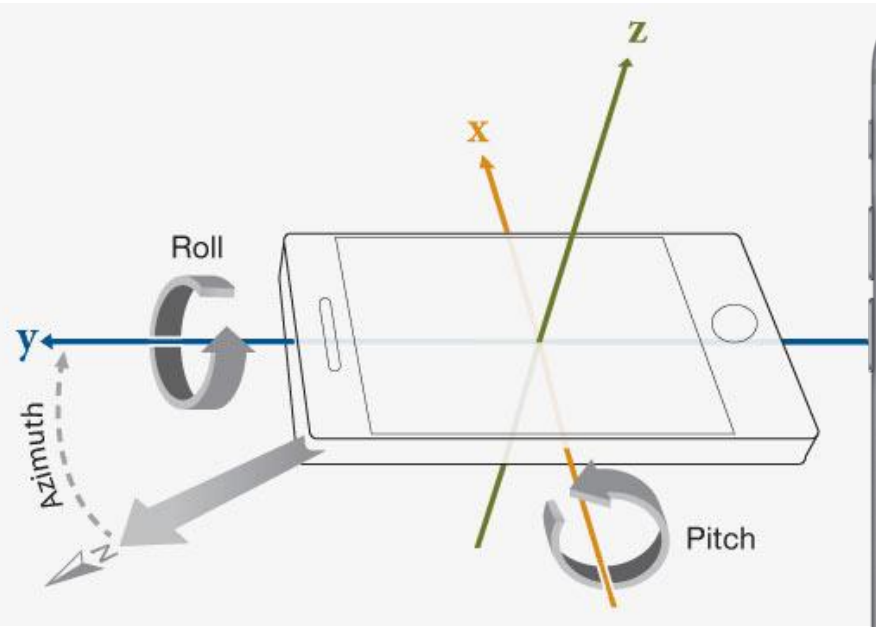
Continue the discussion post session and post ARO!

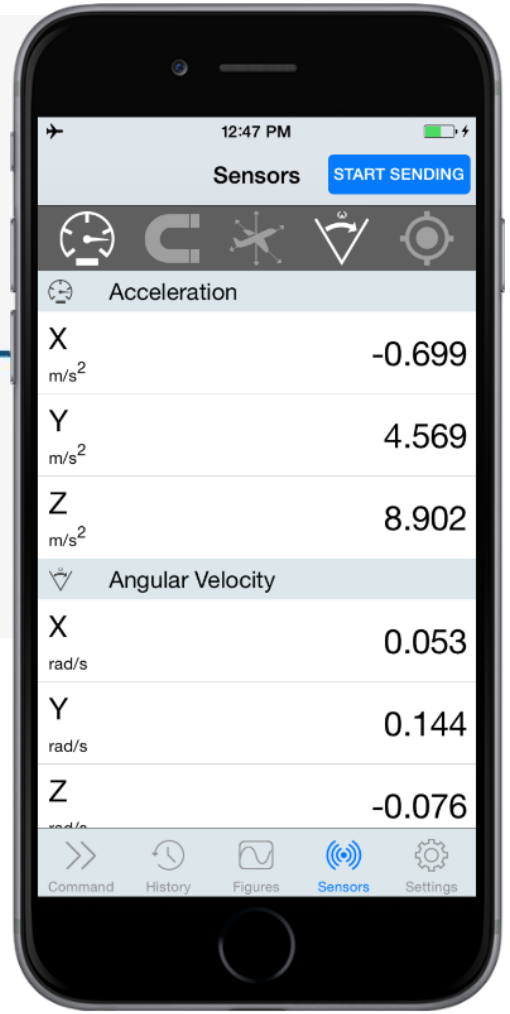
Panel Discussion

APPENDIX

Access Sensors on Mobile Devices

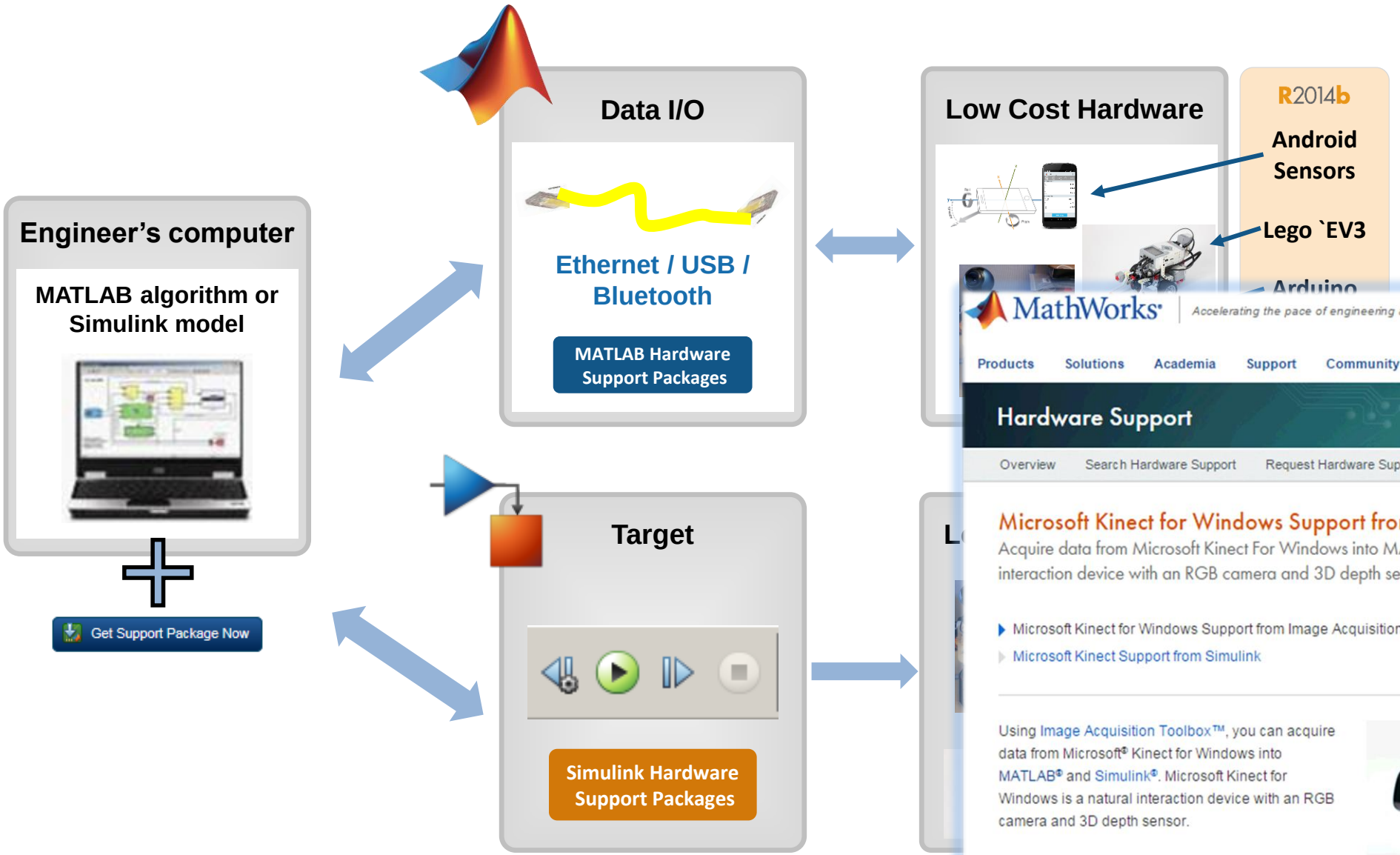


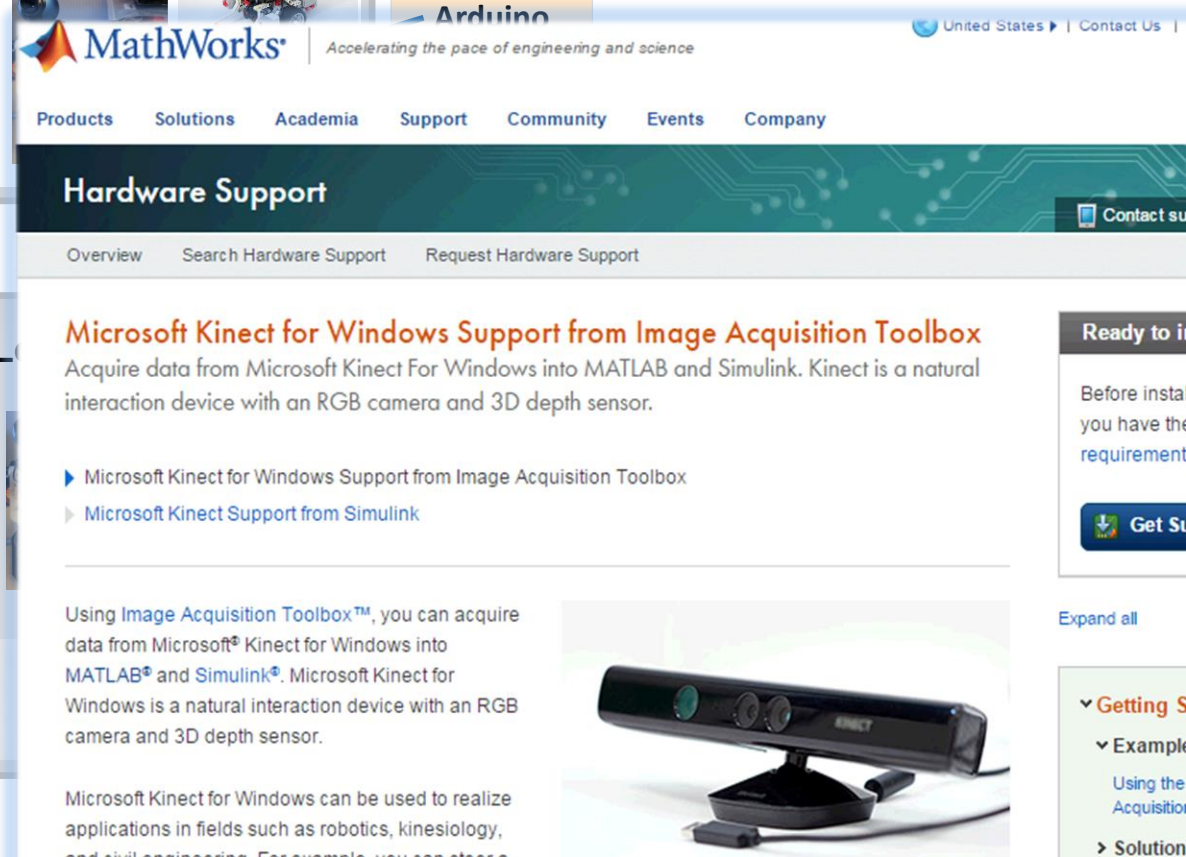




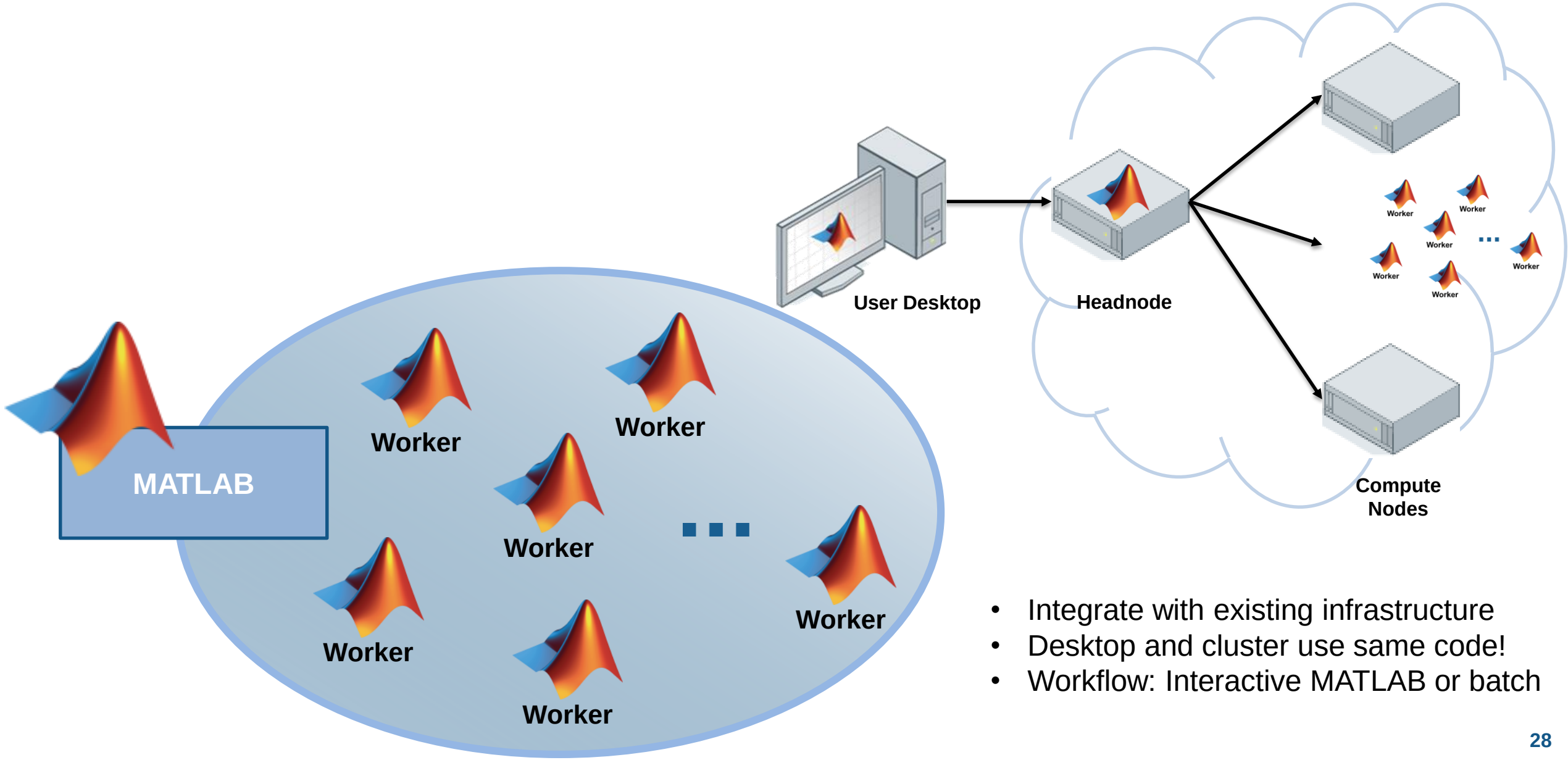
MATLAB Mobile for iOS or Android

Connecting to Low Cost Hardware





Beyond Serial MATLAB ➔ Integrating with Clusters and Cloud



- Integrate with existing infrastructure
- Desktop and cluster use same code!
- Workflow: Interactive MATLAB or batch

MATLAB for Parallel Computing: SNIC/XSEDE

30 XSEDE HPC Mentors at 17 Organizations –

Program initiated in 2015