Deepmod: An Over-the-Air Trainable Machine Modem for Resilient Communications

Dr. Adam L. Anderson
Oak Ridge National Lab
Tennessee Tech University

GRCon '18

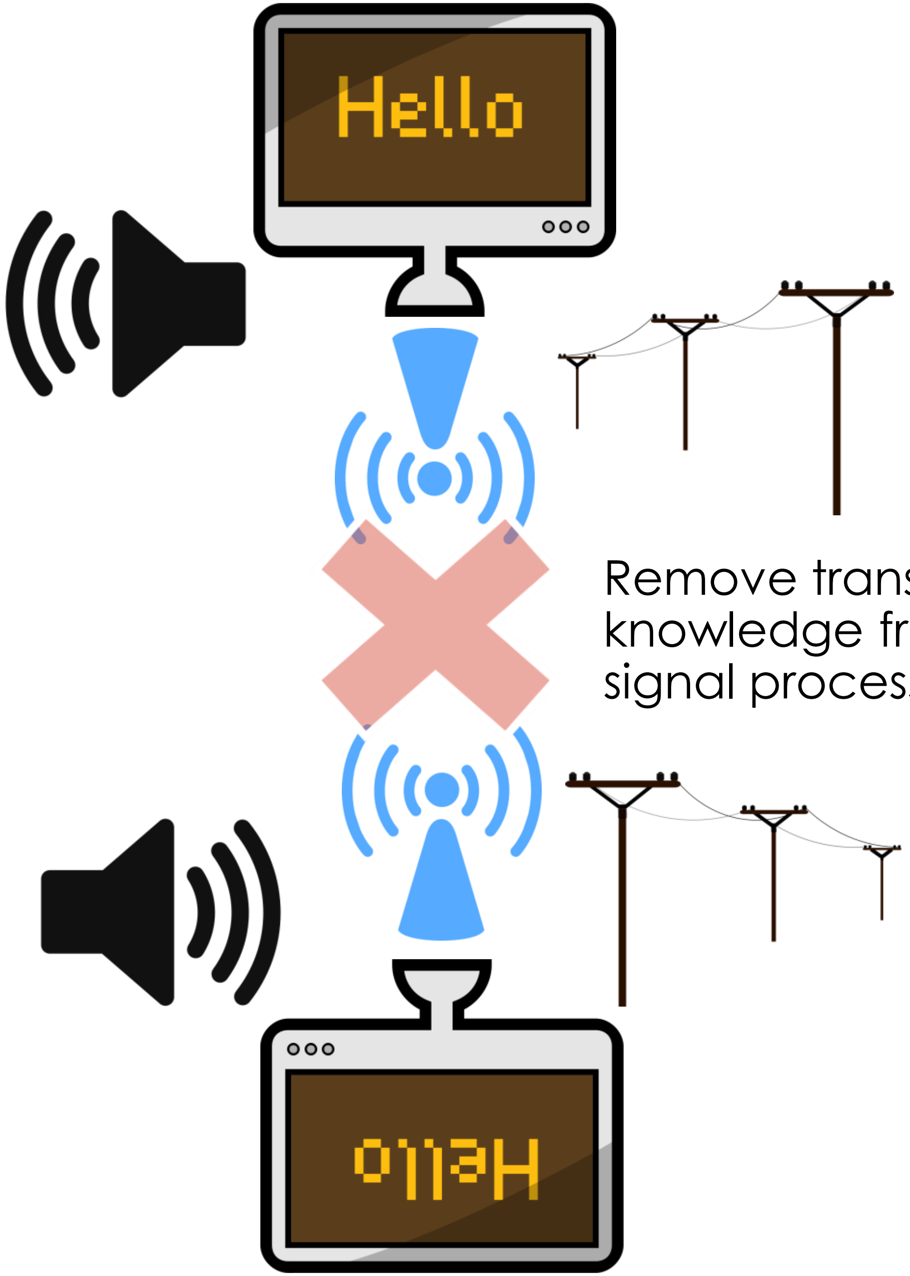
Specific Goal: Security

<u>Definition of Security [44 U.S.C., Sec. 3542]</u>

- **Confidentiality**: “Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information….”
- **Integrity**: “Guarding against improper information modification or destruction, and includes ensuring information non-repudiation and authenticity….”
- **Availability**: “Ensuring timely and reliable access to and use of information….”

<u>Deepmod</u>

“… by generating, on-the-fly, viable physical layer <u>modem</u> protocols, using <u>deep</u> learning, in a variety of media or channel conditions.”

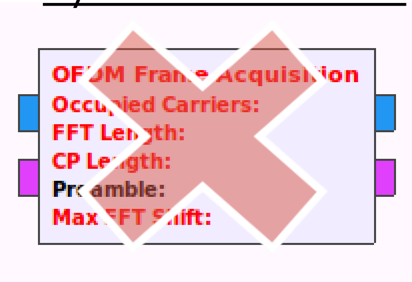


Remove transducer knowledge from signal processing

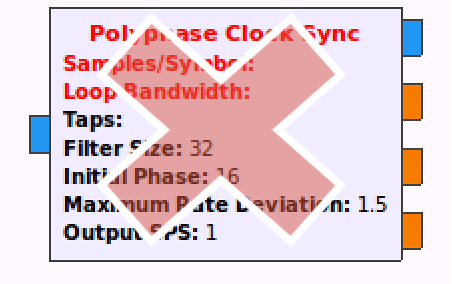
Pie-in-the-Sky Goal: Throw Out 75 Years of Digital Comms

Figure out: How much of traditional PHY can Deepmod actually learn?

Frame Synchronization

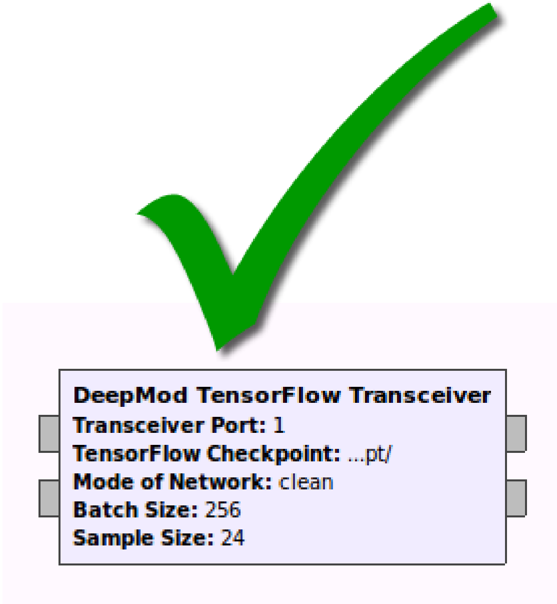


Timing Synchronization



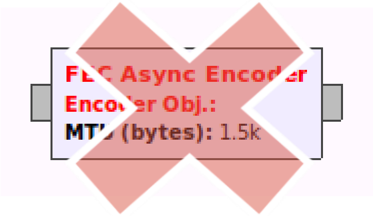
Digital Modulation

Phase Correction

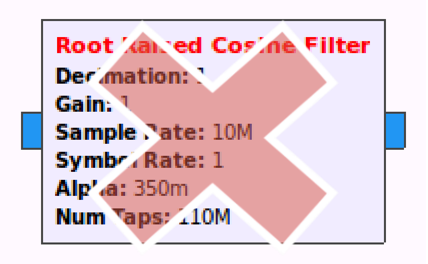


Automatic Gain Control

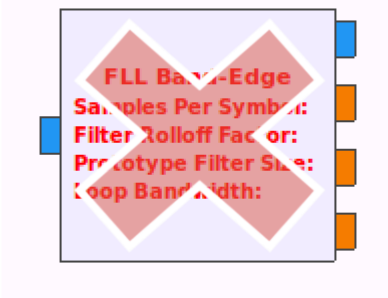
Error Correction



Matched Filtering



CFO Correction



We want to replace ***all*** signal processing blocks with a single learner

Recent Work on "Machine Modems"

- "Machine speak: Left to their own devices, computers can figure it out," https://www.ornl.gov/news/machine-speak-left-their-own-devices-computers-can-figure-it-out
- "Machine Learning Modems: How ML Will Change How We Specify And Design Next Generation Communication Systems, " (www.comsoc.org)
- T. J. O'Shea and J. Hoydis, "An introduction to machine learning communications systems,"ArXiv e-prints, Feb. 2017.
- T. OShea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, Dec 2017.
- S. Drner, S. Cammerer, J. Hoydis, and S. t. Brink, "Deep learning based communication over the air," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 132–143, Feb 2018.
- ECE 7970: Statistical Learning, Fall 2015, Class Project

A Note on Vocabulary

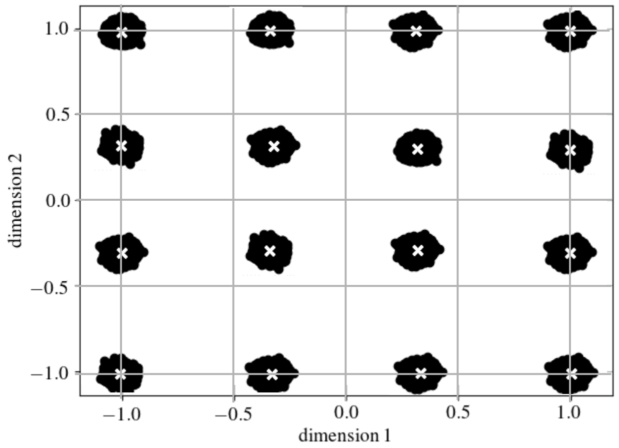
Some ambiguity between comms and ML communities

- **Network**: The communication network in question
- **Graph**: The “brain” of the neural network
- **Sample**: This is what comes out of, or into, your USRP
- **Example**: This is what goes into your machine learner
- **Spectral Efficiency**: We’ll call it bits-per-sample. In traditional comms you have samples-per-symbol, bits-per-symbol, r=1/2 FEC, etc. With Deepmod you have number of classes (bits-per-class) and samples-per-example. Both lead to the same definition of spectral efficiency.

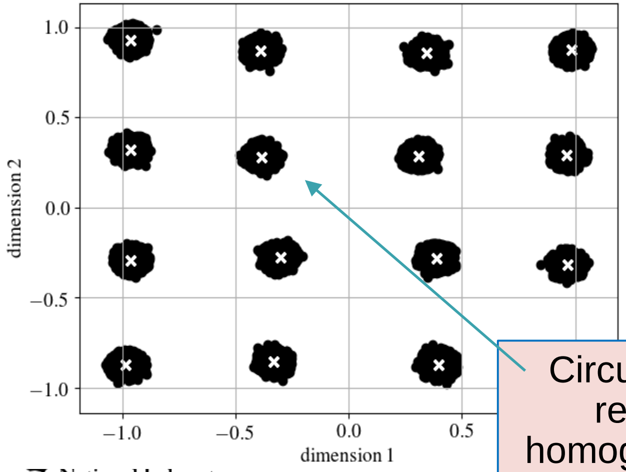
How Resiliency? A Toy Example

1) Homogeneous noise channel

Human Invented 16-QAM



Machine Invented 16-QAM

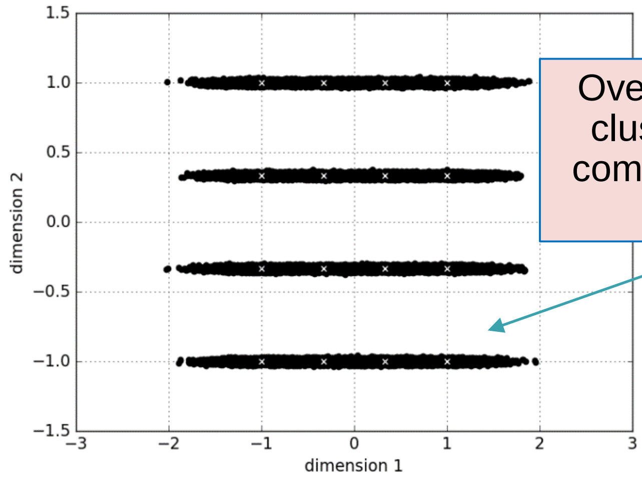


Circular clusters result from homogenous noise

- Human coded information flow goes to zero when the channel changes drastically.
- The machine is able to “learn” around the attack and maintain identical flow. The machine discovers latent communications.
- Sure, a human could invent 8x2-QAM but we don’t want to wait two-years for hardware upgrades.

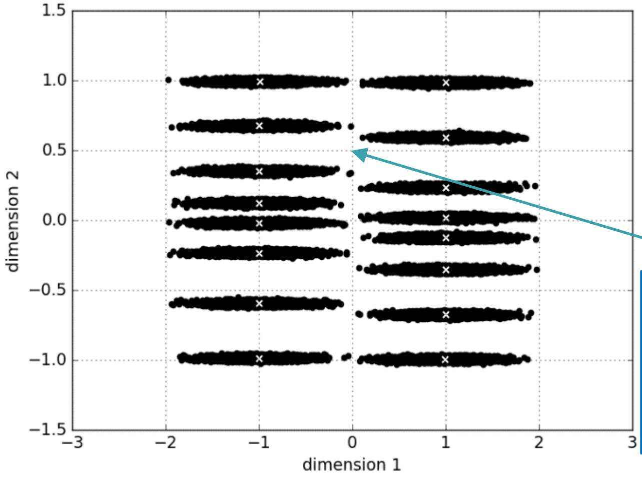
2) Heterogeneous noise channel

Human Invented 16-QAM



Overlapping symbol clusters means no communications can take place

Machine Invented 8x2-QAM

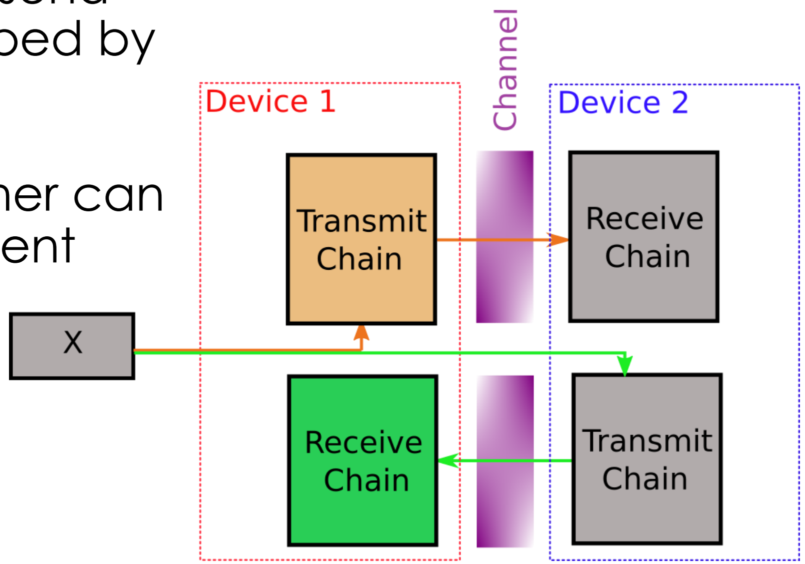


No overlap. Symbol centers rearranged around heterogenous noise

A Real Example: Over-the-Air Training with Autoencoders

Autoencoders are a classic first approach to unsupervised learning.
1. Encoder (transmit chain) converts classes to samples.
2. Decoder (receive chain) converts samples to class estimates.

Transmit chain is trained to send message that is least disturbed by channel
- Trivial loss function
- Doesn't ensure partner can understand what is sent



Receive chain is trained to decode from known sequence

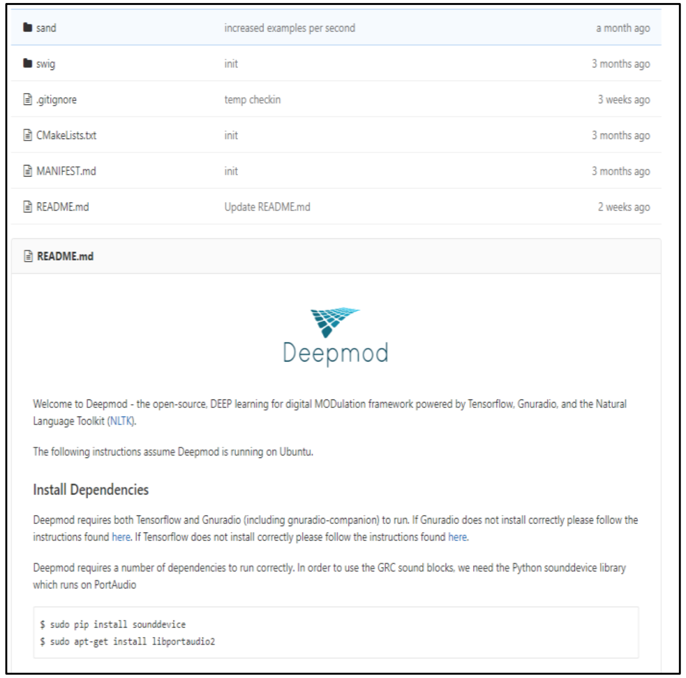
No backpropagation is allowed through the wireless channel!

Result: Better than just inventing constellations but not much learning on transmit side

OAK RIDGE National Laboratory

Deepmod: Acoustic Channel with Autoencoder

Step 1) Make Fancy GIT Repo



sand	increased examples per second	a month ago
swig	init	3 months ago
.gitignore	temp checkin	3 weeks ago
CMakeLists.txt	init	3 months ago
MANIFEST.md	init	3 months ago
README.md	Update README.md	2 weeks ago

README.md

Deepmod

Welcome to Deepmod - the open-source, DEEP learning for digital MODulation framework powered by Tensorflow, Gnuradio, and the Natural Language Toolkit (NLTK).

The following instructions assume Deepmod is running on Ubuntu.

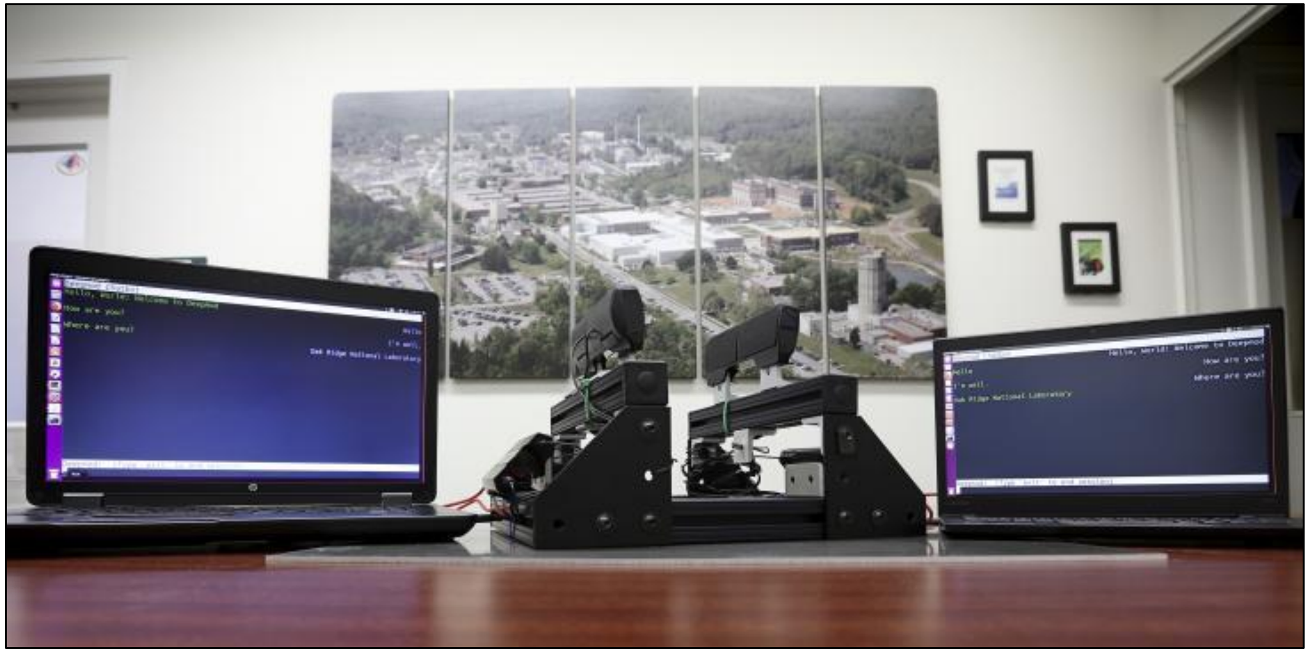
Install Dependencies

Deepmod requires both Tensorflow and Gnuradio (including gnuradio-companion) to run. If Gnuradio does not install correctly please follow the instructions found here. If Tensorflow does not install correctly please follow the instructions found here.

Deepmod requires a number of dependencies to run correctly. In order to use the GRC sound blocks, we need the Python sounddevice library which runs on PortAudio

```
$ sudo pip install sounddevice
$ sudo apt-get install libportaudio2
```

Step 2) Make Fancy Hardware



Step 3) Demo "Machine Speak"

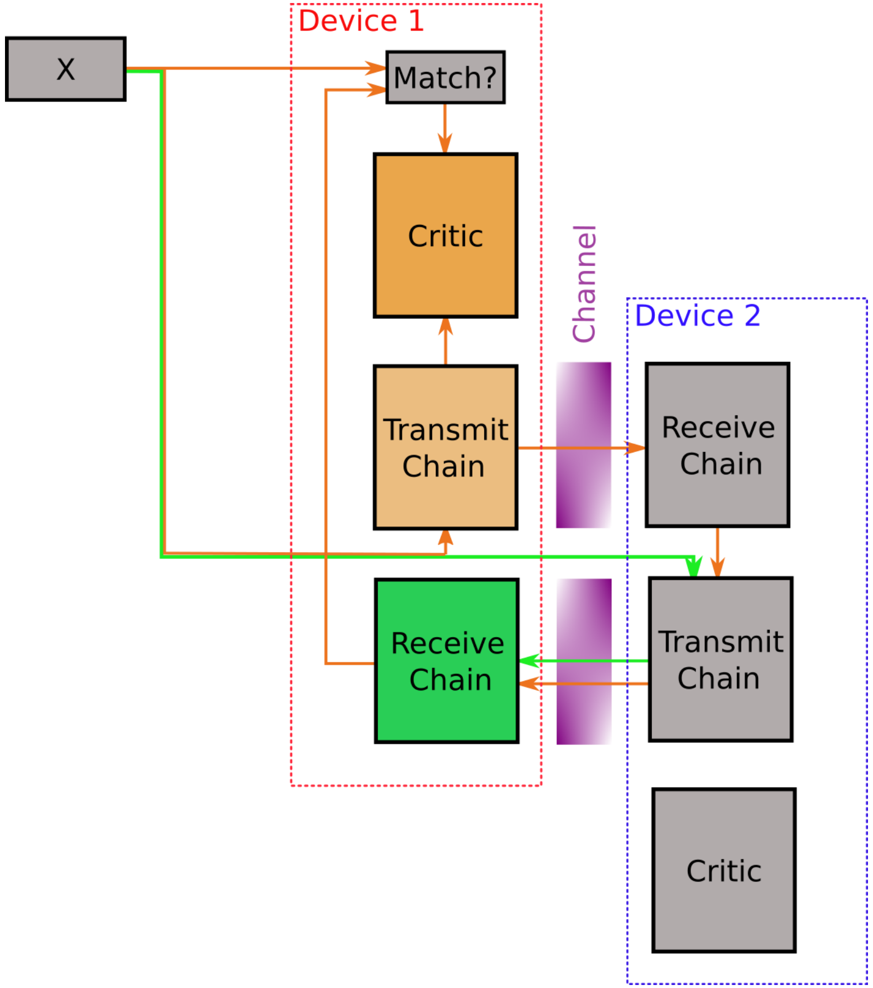
- Deep convolutional neural network autoencoder
- Single transducer rig (soundcard, microphone, speaker)
- GNU Radio, TensorFlow, Natural Language Toolkit
- Classes are English phonemes
- Listen as computers communicate via chat terminal

OAK RIDGE National Laboratory

Over-the-Air Adversarial Training

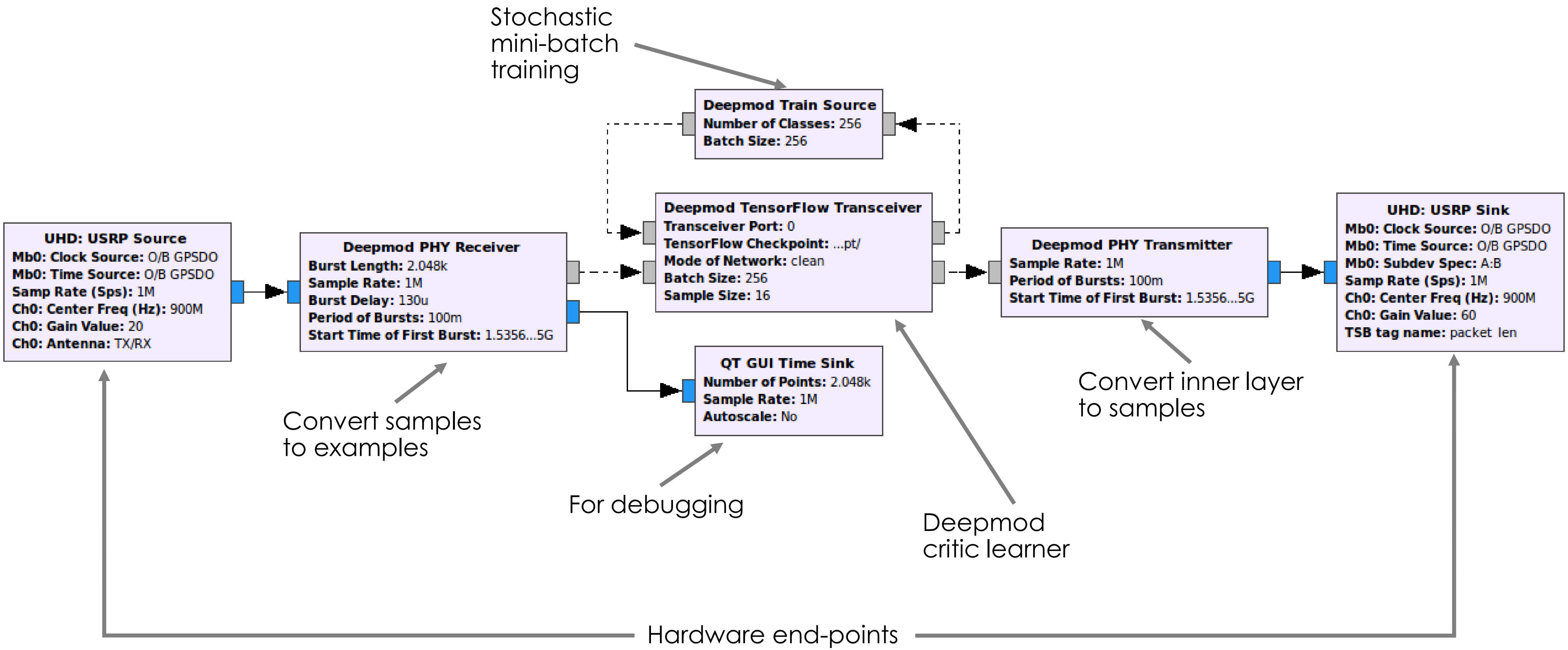
We need a method for the transmit chain to learn as well

1. Stochastic mini-batches. 256 “classes” or 8 bits per class
2. Device 1 converts classes to samples and transmits
3. Device 2 replies with what is heard not what is known
4. Device 1 separates correctly decoded classes from incorrect
5. Transmit chain (encoder) is updated to optimize understandable classes



Receive chain is still trained in same manner

Obligatory GNU Radio Companion Screenshot



Deepmod: RF Channel

Step 1) Add a whole lot of gnuradio and tensorflow code to the repo

Name	Last commit	Last update
apps	init	7 months ago
cmake	init	7 months ago
docs	almost ready for powerline	a week ago
examples	jammer	a week ago
grc	jammer	a week ago
include/deepmod	init	7 months ago
lib	init	7 months ago
python	jammer	a week ago
sand	rf dm	3 months ago
swig	init	7 months ago
.gitignore	prepping for branch	4 months ago
CMakeLists.txt	init	7 months ago
MANIFEST.md	init	7 months ago
README.md	Update README.md	5 months ago
ber.npy	about to branch gps	3 weeks ago

README.md

Deepmod

Welcome to Deepmod - the open-source, DEEP learning for digital MODulation framework powered by Tensorflow, Gnuradio, and the Natural Language Toolkit (NLTK).

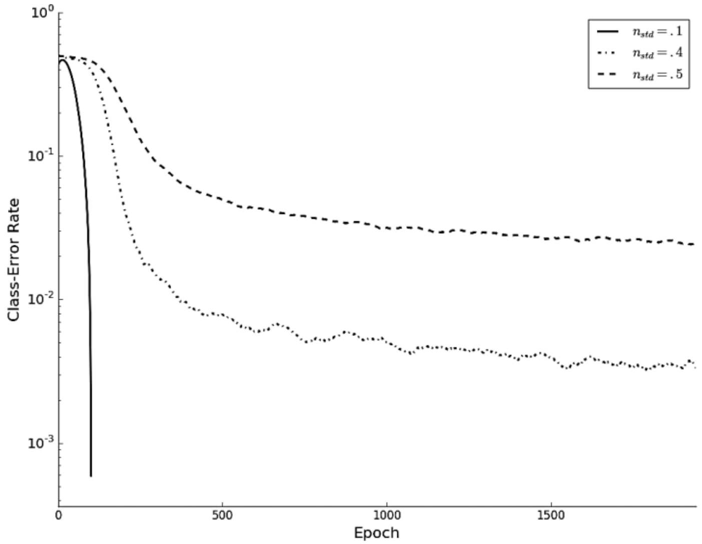
Step 2) Plug into a different set of transducers (USRPs in this case)

Step 3) Demo Deepmod

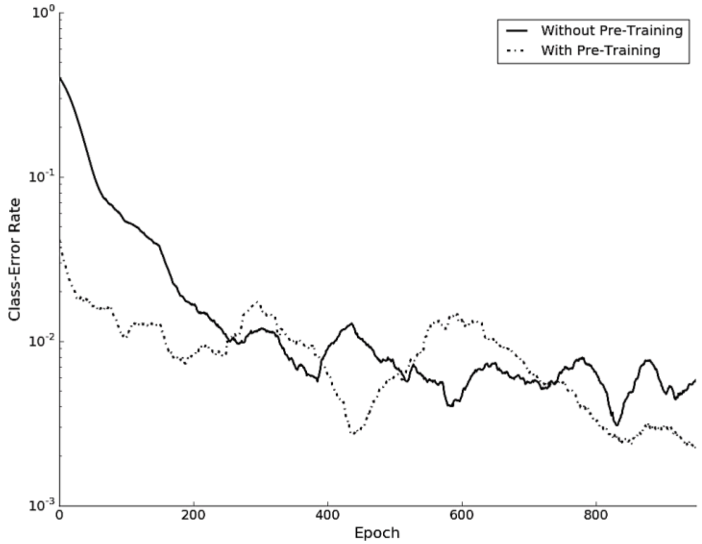
- Sample rate = 1 Msps
- Center frequency = 900 MHz
- SNR adjusted to match simulation
- 256 classes, 16 (real) samples, **1 bit per sample**
- **0 signal processing blocks**
- High rate communications

Simulated vs. Over-the-Air

Simulated class error rate vs. training epoch



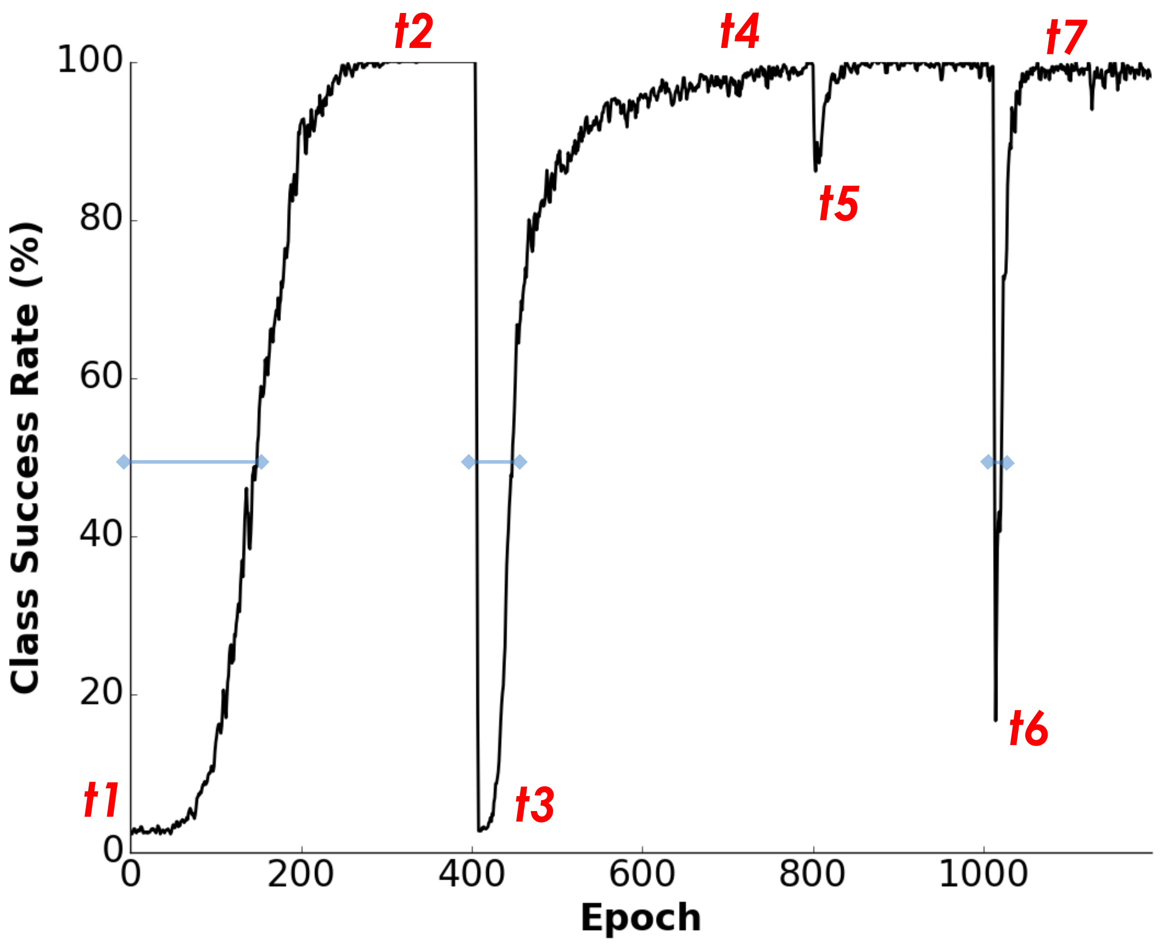
Over-the-air class error rate vs. training epoch and graph initialization



- Current Deepmod achieves realistic spectral efficiency (1 bit per sample)
- At acoustic rates also achieves good throughput
- At RF rates Deepmod is **throttled due to TensorFlow bottleneck**
- Need to get Deepmod closer to traditional PHY layer throughputs*

* Inference is MUCH faster than training fortunately

Resiliency to Extreme Channel Changes



t1: Two Deepmod enabled radios are powered on

t2: Radios converge to a viable communications protocol

t3: Narrowband jammer is engaged; performance bottoms out

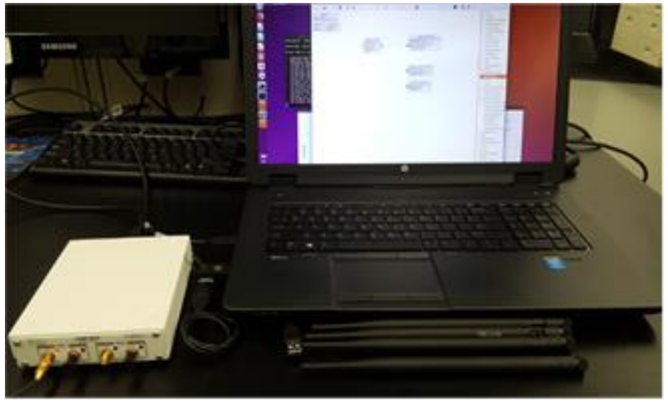
t4: Deepmod recovers by "learning" new jammed channel

t5: Jammer is turned off confusing Deepmod temporarily

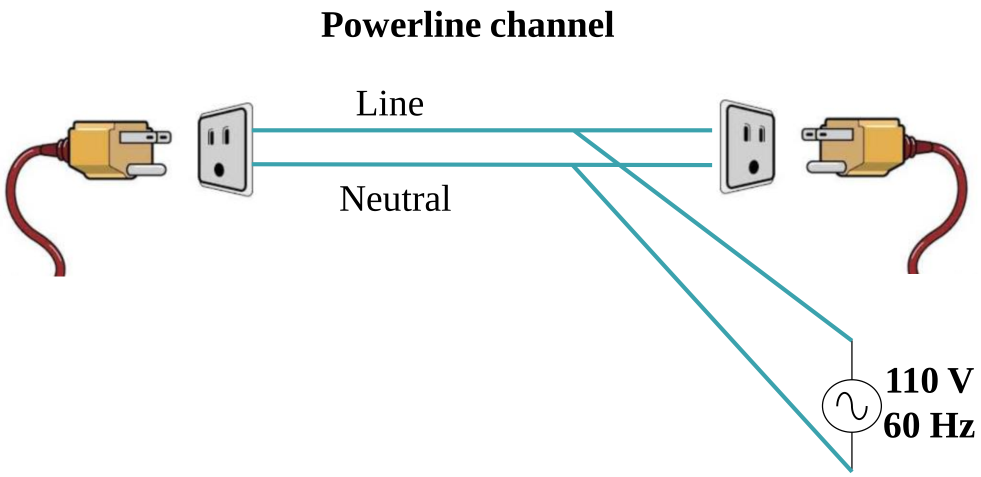
t6: Jammer is turned back on

t7: Deepmod recovers quickly; retained knowledge of both "on" and "off" jammer states

Deepmod: Powerline Communications (in progress)



High-Pass, 60 Hz Isolation Filter w/Plug-in Shielded Enclosure

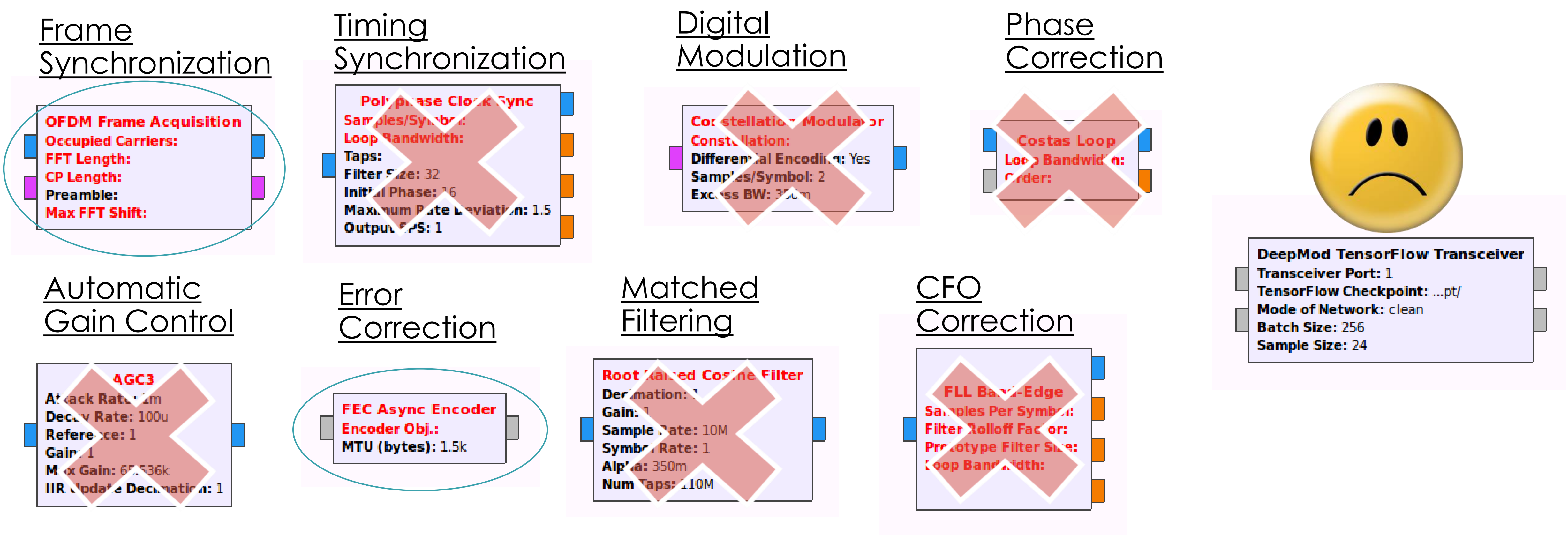


Signal down-converted from 75 MHz carrier

- Powerline channel is extremely noisy and difficult to model all possibilities. Every channel is different.
- Perfect environment for a learnable PHY layer!
- However, Deepmod isn't magic; it won't increase your SNR or range. Still need some signal to make it across channel.

Deepmod Didn’t Learn Everything We Had Hoped

Remember, we wanted to replace ***all*** signal processing blocks with a single learner



Don’t give up yet!

A Quick Note on Frame (Example) Synchronization

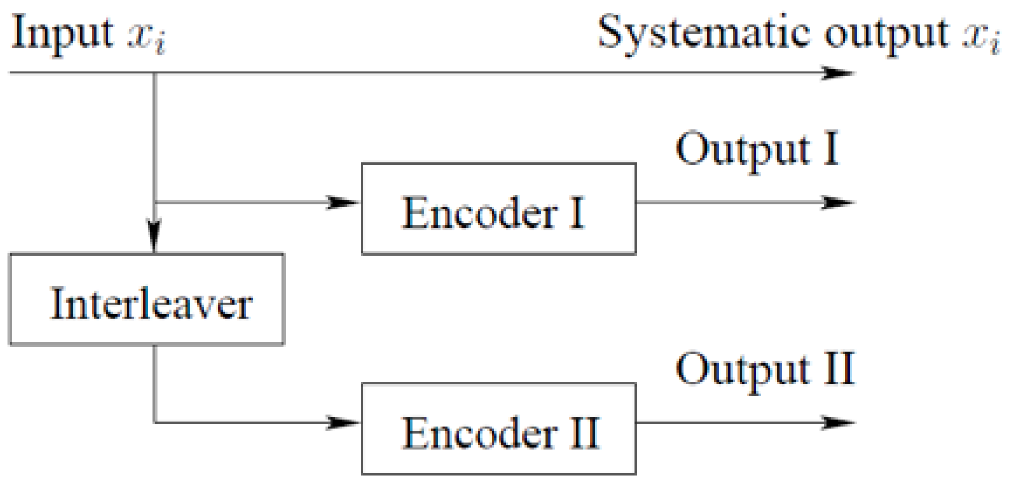
We tried a little bit of everything…

- **Continuous Tone-Coded Squelch System** (CTCSS) – only listen when a prearranged tone was present. Used in the original acoustic demo.
- **Energy Detection** – classic double-sliding window for packet edge detections.
- **Preamble** – this felt like "cheating" since it assumes a rudimentary "language" before training.
- **GPS** – only listen to channel in certain time slots. Okay to be off by a sample or two; Deepmod can work around that.
- **Machine Learner** – is where we are headed. Will probably need to be its own machine at the front end of Deepmod.

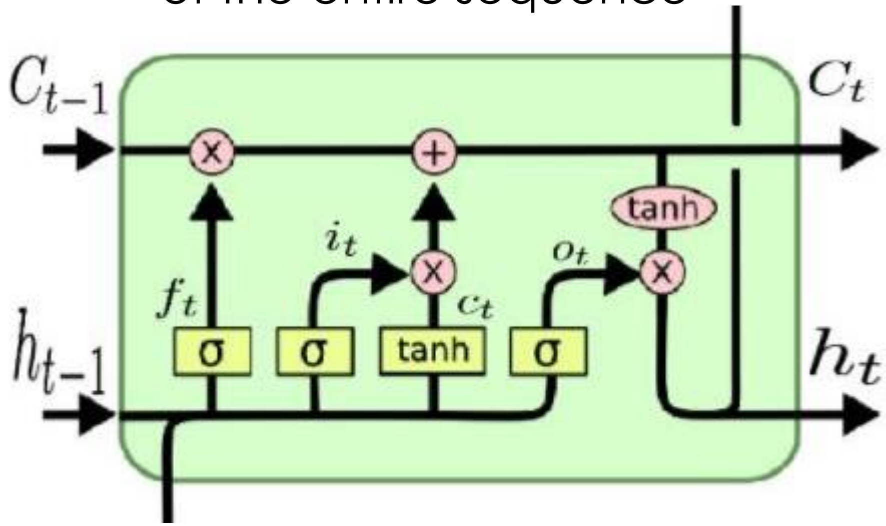
Moving from CNN to Recurrent Neural Network (RNN)

"Ordinary neural networks including CNNs don't perform well in cases where sequence of data is important"

Turbo Code
Output samples are functions of all input samples

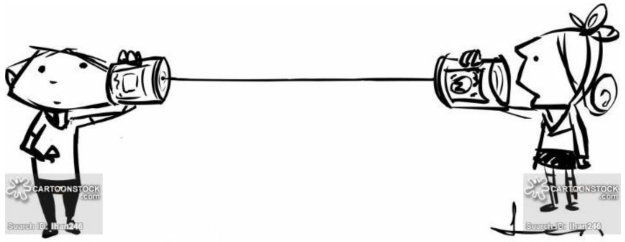


LSTM Cell
All classification is a function of the entire sequence



In comms, sequence of data is important. Deepmod is currently a CNN. Maybe we need it to be an RNN?

Conclusions and Future Work



"I said, how do you send a text with this thing?"

What We Did

- Deepmod can generate a new PHY protocol on-the-fly when:
  - The channel is severely distorted
  - The channel medium is completely altered
- Deepmod can learn what you make it capable of learning

What We Want to Do

- Move away from Deepmod CNN critic toward RNN critic
- Need high spectral efficiency AND high throughput
- Powerline comms, underwater comms, deep-space comms: whatever medium you can stick two transducers into