What is “Neural Signal Processing”? 

- The processing (performing operations on) of signals (current / voltage waveforms) from neurons (brain or peripheral nerves) 
- Traditional digital signal processing 
  - convolution 
  - frequency-domain analysis 
  - filtering 
  - compression 
  - encoding/decoding 
- Neural signal processing 
  - filtering 
  - spike detection 
  - spike sorting
Why Record Neural Signals?

- To learn about the brain
  - Study correlations between stimulus and activity of individual neurons
  - Study correlations between behavior and activity of individual neurons

- Medical applications
  - Epilepsy research: predict seizures before they occur based on neural activity (spikes, LFP, and EEG); localize origin of seizures
  - Neural prosthetics for amputees, quadriplegics
    - “Decode” motor intentions to control computer cursor, wheelchair, prosthetic limb, own limb

- Military applications
  - “super-soldier”

- Entertainment applications?

---

Example of Monkey Controlling Neural Prosthesis

http://motorlab.neurobio.pitt.edu/
Brain Areas

Some Motor, Sensory, and Association Areas of the Cerebral Cortex

- Functions are localized in the brain

The Motor Cortex

EE219A – Spring 2008 Lecture 16
The Brain

- Gray matter: neuronal cell bodies and dendrites (action potential generation; “computers”)
- White matter: axons (action potential transfer; “cables”)

The Neuron

- Dendrites: receive input (current) from other neurons
- Axon: transmit action potentials (APs) to other neurons
- Axon hillock: site of AP generation
- Myelin sheath: increases resistance b/t cell membrane and extracellular material → increases speed of AP propagation
**The Action Potential**

- AP begins at axon hillock
- For myelinated neurons, ion channels are only exposed at nodes of Ranvier, so AP is re-generated at each node
- In this way, APs propagate down an axon
Recording Action Potentials

- Two electrodes are placed inside brain in extracellular medium (one for recording, one for reference).
- During an action potential, influx of sodium into cell causes large change in voltage potential in the surrounding area relative to distant areas.
- This causes a voltage difference between the recording electrode and the (distant) reference electrode.
- Electrode tip should be less than 140 µm from neuronal cell body to record an AP
- Electrode tip diameters ~ 10 µm
- Single electrode may hear signals from multiple neurons

Electrodes for Single-Unit Recordings

- Traditional microwires made from tungsten, gold, platinum, or platinum-iridium, and coated with insulator
- New multi-channel electrode probes use microfabrication techniques, often with a silicon substrate
Properties of Neural Signals

- Peak membrane voltage ~40 – 50 mV
- Recorded voltages ~0.5 – 1 mV
- Firing rates of neurons can be 1 – 100 Hz, depending on brain area

Frequency Components of Neural Signals

- < 300 Hz: Local Field Potentials (LFP)
  - < 3 Hz: Delta (slow wave sleep)
  - 4 – 7 Hz: Theta (wake and REM)
  - 8 – 12 Hz: Alpha (drowsiness)
  - 12 – 30 Hz: Beta (decision-making, motor planning)
  - 36 – 100 Hz: Gamma (sensory attentive)

- 300 – 6000 Hz: Single-Unit Activity (“spikes”)
Spike Sorting

- In a neural recording, one electrode typically receives electrical signals from multiple (1-7) neurons simultaneously.
- **Spike Sorting**: The process of assigning spikes to different neurons.
- **Why spike sort?**
  - Functional reasons
    - Adjacent cells can encode completely different information
    - Neuroscientists want to know which spike came from which neuron in order to learn how the brain processes information
    - BMIs often depend on single-unit activity as input
  - Practical reasons
    - Electrode arrays have increasing numbers of channels (100, 128, 256, 1000, etc.) → increasing amount of data
    - Need to compress large amounts of data in order to transmit wirelessly

---

Spike Sorting

- Spike sorting is possible because each neuron produces a different, distinct spike shape during a recording that remains constant throughout the recording

- **Three Main Steps**
  1. **Spike Detection**: Separating spikes from noise.
  2. **Feature Extraction**: Transforming spikes into a certain set of features (e.g., principal components).
  3. **Clustering**: Classifying spikes into different groups (neurons) based on extracted features.
Some Existing Algorithms

1. Spike Detection
   – Amplitude Thresholding
   – Energy Operators
   – Wavelet Transform

2. Feature Extraction (FE)
   – Principal Component Analysis (PCA)
   – Wavelet Transform
   – Integral Transform (IT)
   – Discrete Derivatives (DD)

3. Clustering
   – k-Means
   – Single-Linkage
   – Valley Seeking
   – Superparamagnetic Clustering (SPC)
   – Online clustering

Spike Detection Algorithms

Apply threshold to...

- Amplitude of signal

- Absolute value of amplitude

\[
Thr = 4\sigma_N \\
\sigma_N = \text{median}\left\{\frac{|x(n)|}{0.6745}\right\}
\]
**Spike Detection Algorithms**

Apply threshold to...

- Nonlinear Energy Operator (NEO)

\[
\psi[x(n)] = x^2(n) - x(n+1) \cdot x(n-1)
\]

\[
Thr = C \frac{1}{N} \sum_{n=1}^{N} \psi[x(n)]
\]

---

**Spike Detection Algorithms**

Apply threshold to...

- Stationary Wavelet Transform Product (SWTP)

1. Calculate SWT at 5 consecutive dyadic scales:
   \[W(2^j, n), j = 1, \ldots, 5\]
2. Find the scale \(2^{j_{\text{max}}} \) with the largest sum of absolute values:
   \[
j_{\text{max}} = \arg \max_j \left( \sum_{n=1}^{N} |W(2^j, n)| \right)
   \]
3. Calculate point-wise product between SWTs at this scale and at the two previous scales:
   \[
P(n) = \prod_{j=0}^{2} |W(2^j, n)|
   \]
4. Smooth with Bartlett window
5. Threshold:
   \[
   Thr = C \frac{1}{N} \sum_{n=1}^{N} w(n) \ast P(n)
   \]
Feature Extraction Algorithms

- Principal Component Analysis (PCA)
- Integral Transform (IT)
- Discrete Wavelet Transform (DWT)
- Discrete Derivatives (DD)

- Special case: tetrode recordings

Principal Component Analysis (PCA)

- Also known as Karhunen-Loève transform
- Projects data onto an orthogonal set of basis vectors such that the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component).
- The theoretically optimum transform for a given data in least square terms.

- Algorithm
  1. Calculate covariance matrix of data (spikes) ($N$-by-$N$)
  2. Calculate eigenvectors ("principal components") of covariance matrix ($N$ 1-by-$N$ vectors)
  3. For each principal component ($i = 1,\ldots,N$), calculate the $i^{th}$ "score" as the scalar product of the data point (spike) and the $i^{th}$ principal component
  4. Each data point can be reconstructed by multiplying each score for that data point by each PC and summing over $i$
PCA for Spike Sorting

PCA: Pros and Cons

- **Pro**
  - Efficient (coding): can represent spike in 2 or 3 PCs

- **Cons**
  - Inefficient (implementation): hard to perform eigendecomposition in hardware
  - Only finds independent axes of the data under the Gaussian assumption. For non-Gaussian or multi-modal Gaussian data, PCA simply de-correlates the axes.
  - There is no guarantee that the directions of maximum variance will contain good features for discrimination.
Integral Transform (IT)

- Most spikes have a negative and positive phase.

**Algorithm**
1. Calculate integral of negative phase
2. Calculate integral of positive phase

\[
I_A = \frac{1}{N_A} \sum_{n=1}^{N_A} x(n)
\]
\[
I_B = \frac{1}{N_B} \sum_{n=1}^{N_B} x(n)
\]

- **Pros and Cons**
  - **Pros**
    - Efficient implementation (accumulators, no multiplies)
    - High dimensionality reduction
  - **Con**
    - Does not discriminate between neurons well

---

Discrete Wavelet Transform (DWT)

- Wavelets computed at dyadic scales form an orthogonal basis for representing data.
- Convolution of wavelet with data yields wavelet “coefficients”
- Can be implemented by a series of quadrature mirror filter banks.

**Pros and Cons**
- **Pro:** Accurate representation of signal at different frequencies
- **Con:** Requires convolutions \(\rightarrow\) multiple multiplies/adds per sample
DWT: Example

Discrete Derivatives (DD)

- Like a simplified version of DWT
- Differences in spike waveform are computed at different scales:
  \[ dd_\delta(n) = x(n) - x(n - \delta) \]
- Dimensionality is reduce by performing K-S test for normality
- Pros and Cons
  - Pros
    - accurate
    - very simple implementation
  - Cons:
    - Increases dimensionality
    - K-S test increases complexity

Samar, et al.

Nadasdy, et al.
Special Case: Tetrode Recordings

- **Tetrodes**: Probe with 4 electrodes
- Electrodes are close enough together to record signals from same neurons
- Amplitude of signal from a given neuron is highest on closest electrode and lowest on farthest electrode
- This information can be used for spike sorting
- Simplifies calculations!

Takahashi, et al.
Clustering

- Partitioning a data set into subsets ("clusters"), so that the data in each subset share some common trait - often proximity according to some defined distance measure
- Previous slides show scatter plots of one extracted feature vs. another (e.g., PC2 vs. PC1, amplitude on electrode4 vs. electrode1)
- In order to determine which neuron generated each spike, we must partition these scatter plots
- Types of clustering:
  - Hierarchical: Successive clusters are found using previously established clusters
    - Agglomerative – "bottom-up"
    - Divisive – "top-down"
  - Partitional: All clusters are determined at once

Clustering Algorithms

- k-Means
- Single-Linkage
- Valley-Seeking
- Superparamagnetic Clustering (SPC)
- Online
K-Means Clustering

- Most commonly-used clustering method
- Partitional
- Based on distance measure

Algorithm
1. Define $k$ (number of clusters/neurons)
2. Randomly define the $k$ cluster centroids
3. Assign each data point to the cluster with the closest (usually by Euclidean distance measure) centroid
4. Recompute each cluster centroid as mean of cluster
5. Repeat Steps 3-4 until convergence criterion is met (e.g., assignments stop changing)

Pros and Cons
- **Pro**: very simple
- **Cons**
  - Need to know $k$
  - Assumes spherical distribution of data

Single-Linkage Hierarchical Clustering

- Hierarchical, agglomerative
- Based on a distance metric

Algorithm
1. Initialization: Create one cluster for each data point.
2. Find the two closest clusters $c_1$ and $c_2$
3. Merge these clusters
4. Compute distance $D(c_1, c_2)$ between the new cluster and all of the old clusters
5. Repeat steps 2 – 4 until either:
   - $D(c_1, c_2)$ exceeds a certain threshold (distance criterion) or
   - A certain number of clusters is reached (number criterion) or
   - All objects are in one cluster. Requires building the entire hierarchical tree, then cutting the $k-1$ longest links to get $k$ clusters.

Pros and Cons
- **Pro**: conceptually simple
- **Cons**
  - computationally complex (requires many searches)
  - not automatic (requires setting of either $k$, distance criterion, or number criterion)
Valley-Seeking Clustering

- Hierarchical, agglomerative (extension of single-linkage)
- Algorithm
  1. Calculate distance matrix $D = (d_{ij})$
  2. Calculate neighbor number matrix $L = (l_{ij})$
  3. Calculate $S = (s_{ij})$, where $s_{ij} = (l_{ij} + l_{ji})/2$
  4. Calculate NDD matrix $J = (J_{ij})$, where $J_{ij} = \frac{|l_{ij} - l_{ji}|}{s_{ij}^{1/\alpha}}$
  5. Estimate convexity matrix $D^2 = (d^2_{ij})$, where
     $d^2_{ij} = \left( \frac{\sum_{i} l_{ij} + l_{ji} \sum_{i} s_{ij}}{\sum_{i} s_{ij}^2 / \sum_{i} l_{ij}^2} \right)$
  6. Construct connectivity matrix $C = (c_{ij})$, where $c_{ij} = I(s_{ij} \leq l_{ij}, J_{ij} \leq l_{ij}, d^2_{ij} \leq l_{ij})$
  7. Assign data to clusters using connectivity matrix
- Pros and Cons
  - **Pros**: automatic, makes no assumption about shape of data
  - **Con**: complex

Zhang, et al.

---

Valley-Seeking: Examples

Zhang, et al.
Superparamagnetic Clustering (SPC)

- Hierarchical, divisive
- Data is modeled as a granular magnet, where each point is assigned a spin.
- The model is heated from low temperatures to high temperatures
  - Low temperatures = “ferromagnetic region”: all spins are aligned
  - High temperatures = “paramagnetic region”: system is disordered and all spins are random
  - “superparamagnetic region”: spins within the same high-density region are aligned while the spins of different high-density regions are not aligned. At these temperatures, the clusters are revealed.

SPC: Pros and Cons

- **Pros**
  - Automatic (don’t need to know the number of clusters/neurons a priori)
  - Makes no assumptions about shapes of clusters

- **Cons**
  - extremely complex
SPC: Algorithm

1. Calculate strength of interaction between each point and every other point
   \[ J_{ij} = \begin{cases} \frac{1}{\beta} \exp\left(-\frac{\psi_i - \psi_j}{\beta}\right) & \text{if } \psi_i \text{ and } \psi_j \text{ are neighbors} \\ 0 & \text{otherwise.} \end{cases} \]

2. Randomly assign each point a spin
3. Run \( N \) Monte Carlo simulations to determine whether each particle’s spin changes (points that are close together will change their states together)
   \[ p_{ij} = \exp\left(-\frac{E_{ij}}{k_B T}\right) \]
4. Calculate the correlation \( G \) between each point and every other point
   \[ c_{ij} = \begin{cases} 1 & \text{if } \psi_i \text{ and } \psi_j \text{ belong to the same SW cluster} \\ 0 & \text{otherwise.} \end{cases}, \quad C_y = \langle c_y \rangle, \quad G_y = \frac{(q - 1) C_3 + 1}{q} \]
5. Assign points to the same cluster if their correlations are above a certain threshold
6. Repeat Steps 3-5 for each temperature
7. Find temperature corresponding to superparamagnetic region
   \[ Blatt, et al. \]

Online Clustering

- Developed for human closed-loop experiments
- Algorithm
  1. Initialization: Assign first data point to its own cluster.
  2. Calculate distance between next data point and each cluster centroid
  3. If smallest distance is less than a threshold, assign point to nearest cluster and recompute that cluster’s centroid. Otherwise, start a new cluster.
  4. Check the distances between each cluster and every other cluster. If any distance is below a threshold, merge those two clusters.
  5. Repeat 2-4 indefinitely.
- Pros and Cons
  - Pros
    - Computationally simple
    - Realtime
    - Automatic
  - Con: Assumes spherical distribution of data
Commercial Spike-Sorting Software

Plexon
Dallas, TX

References: Spike Sorting Algorithms

References: Neural Recording