Bioinformatics Methods

Kevin Karplus

Biomolecular Engineering Department
University of California, Santa Cruz
karplus@soe.ucsc.edu

1 August 2011

Outline of Talk

- What is Bioengineering? Biomolecular Engineering? Bioinformatics?
- Protein Structure Prediction and Protein Design
- Genome Assembly

What is Bioengineering?

Three concentrations:

- Biomolecular
- Rehabilitation
- Bioelectronics

What is Bioengineering?

Three concentrations:

- Biomolecular
  - Drug design
  - Biomolecular sensors
  - Nanotechnology
  - Bioinformatics
- Rehabilitation
- Bioelectronics

What is Bioengineering?

Three concentrations:

- Biomolecular
- Rehabilitation
  - Systems to held individuals with special needs
  - Cell-phone-based systems to reach large numbers of people.
  - Novel hardware to assist the blind
  - Robotics for rehabilitation and surgery applications.
- Bioelectronics

What is Bioengineering?

Three concentrations:

- Biomolecular
- Rehabilitation
- Bioelectronics
  - Implantable devices
  - Interfacing between organisms and electronics
  - Artificial retina project

What to take early

- Mathematics
- Chemistry and then biology
- Introductory bioengineering courses
- Declare your major immediately!

What to take early

- Mathematics
- Chemistry and then biology
- Introductory bioengineering courses:
  - BME80G, Bioethics (F)
  - BME5, Intro to Biotechnology (W, S)
  - CMPE80A: Universal Access: Disability, Technology, and Society (W, S)
- Declare your major immediately!

What to take early

- Mathematics
- Chemistry and then biology
- Introductory bioengineering courses
- Declare your major immediately!
  - You can always change to another one later.
  - Bioengineering is one of the most course-intensive majors on campus.
  - Many courses have prerequisites.
  - It is important to get staff and faculty advice early.

What is Biomolecular Engineering?

Engineering **with**, **of**, or **for** biomolecules. For example,

with: using proteins (or DNA, RNA, . . .) as sensors or for self-assembly.

of: protein engineering—designing or artificially evolving proteins to have particular functions

for: designing high-throughput experimental methods to find out what molecules are present, how they are structured, and how they interact.

What is Biomolecular Engineering?

Engineering **with**, **of**, or **for** biomolecules. For example,

with: using proteins (or DNA, RNA, . . .) as sensors or for self-assembly.

of: protein engineering—designing or artificially evolving proteins to have particular functions

for: designing high-throughput experimental methods to find out what molecules are present, how they are structured, and how they interact.

What is Biomolecular Engineering?

Engineering **with**, **of**, or **for** biomolecules. For example,

with: using proteins (or DNA, RNA, . . .) as sensors or for self-assembly.

of: protein engineering—designing or artificially evolving proteins to have particular functions

for: designing high-throughput experimental methods to find out what molecules are present, how they are structured, and how they interact.

What is Biomolecular Engineering?

Engineering **with**, **of**, or **for** biomolecules. For example,

with: using proteins (or DNA, RNA, . . .) as sensors or for self-assembly.

of: protein engineering—designing or artificially evolving proteins to have particular functions

for: designing high-throughput experimental methods to find out what molecules are present, how they are structured, and how they interact.

What is Bioinformatics?

Bioinformatics: using computers and statistics to make sense out of the mountains of data produced by high-throughput experiments.

- Genomics: finding important sequences in the genome and annotating them.
- Phylogenetics: “tree of life”.
- Systems biology: piecing together various control networks.
- DNA microarrays: what genes are turned on under what conditions.
- Proteomics: what proteins are present in a mixture.
- Protein structure prediction.
- …

Outline for proteins

- What is a protein?
- The folding problem and variants on it:
  - Local structure prediction
  - Fold recognition
  - Comparative modeling
  - “Ab initio” methods
  - Contact prediction
- Protein Design

What is a protein?

- There are many abstractions of a protein: a band on a gel, a string of letters, a mass spectrum, a set of 3D coordinates of atoms, a point in an interaction graph, ...
- For us, a protein is a long skinny molecule (like a string of letter beads) that folds up consistently into a particular intricate shape.
- The individual "beads" are amino acids, which have 6 atoms the same in each "bead" (the *backbone* atoms: N, H, CA, HA, C, O).
- The final shape is different for different proteins and is essential to the function.
- The protein shapes are important, but are expensive to determine experimentally.

Folding Problem

The *Folding Problem*:
If we are given a sequence of amino acids (the letters on a string of beads), can we predict how it folds up in 3-space?

```
MTMSRRNTDA ITIHSILDWI EDNLESPLSL EKVSERSGYS KWHLQRMFKK
ETGHSLGQYI RSRKMTEIAQ KLKESNEPIL YLAERYGFES QQTLTRTFKN
YFDVPPHKYR MTNMQGESRF LHPLNHYNS
```

↓

Too hard!

Fold-recognition problem

The *Fold-recognition Problem*:
Given a sequence of amino acids A (the *target* sequence) and a library of proteins with known 3-D structures (the *template* library),
figure out which templates A match best, and align the target to the templates.

- The backbone for the target sequence is predicted to be very similar to the backbone of the chosen template.

New-fold prediction

- What if there is *no* template we can use?
- We can try to generate many conformations of the protein backbone and try to recognize the most protein-like of them.
- Search space is huge, so we need a good conformation generator and a cheap cost function to evaluate conformations.

Secondary structure Prediction

- Instead of predicting the entire structure, we can predict local properties of the structure.
- One popular choice is a 3-valued helix/strand/other alphabet. Typically, predictors get about 80% accuracy on 3-state prediction.
- Many machine-learning methods have been applied to this problem, but the most successful is neural networks. (Random forests also doing well.)
- Using Conditional Random Fields can improve sampling of sequences, without improving accuracy on individual residues.

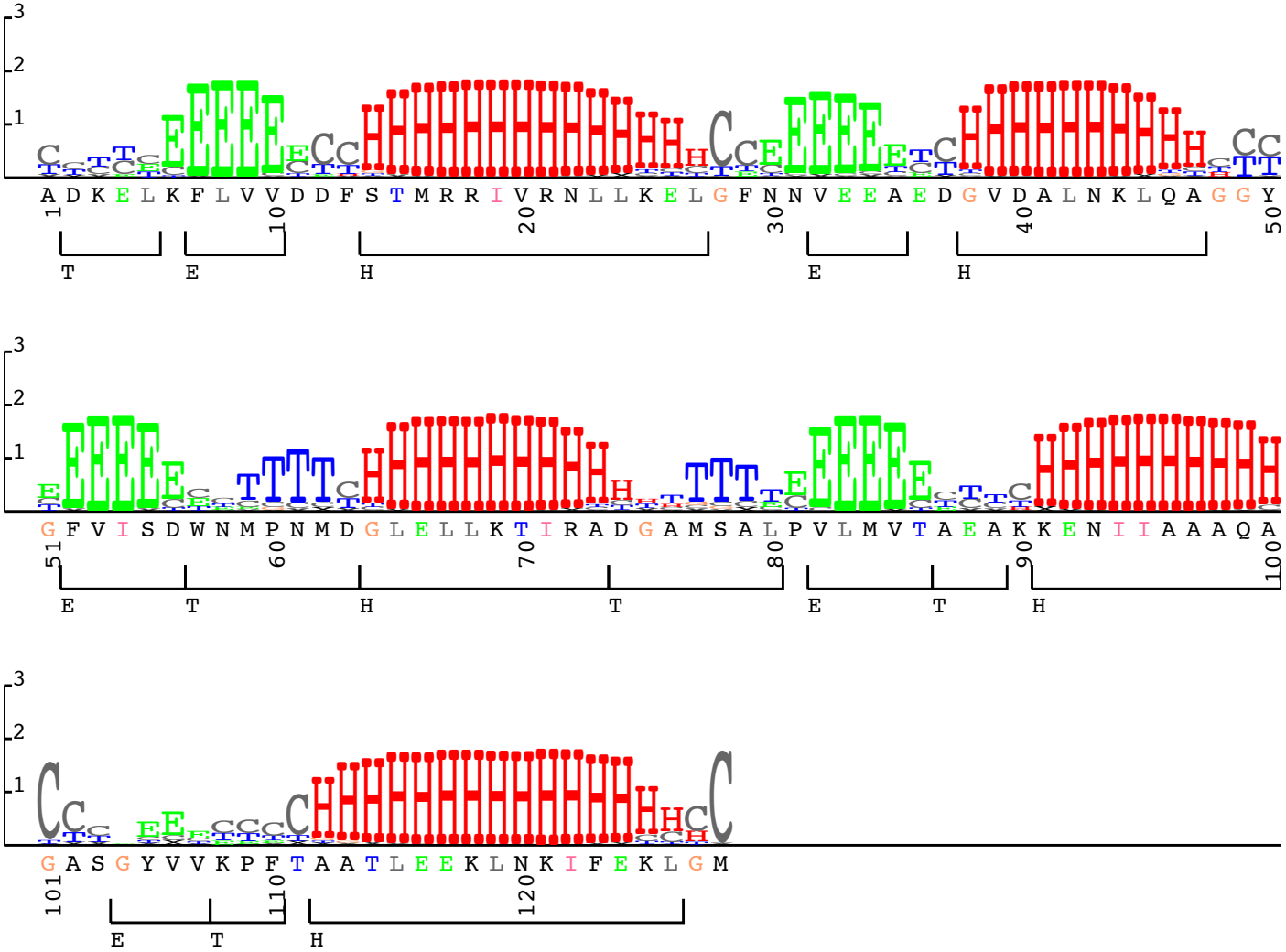
Local Structure Alphabets

- What local properties do we choose?
- We want properties that are well-conserved through evolution, easily predicted, and useful for finding and aligning templates.
- We have investigated many alphabets.
- Current favorites are str2, a 13-state secondary-structure alphabet that distingusishes between different β strands, and near-backbone-11, an 11-state burial alphabet.

Sequence logos (NN)

Summarize local structure prediction:

nostruct-align/3chy.t2k EBGHTL



Fold recognition

- Do iterative search to find similar sequences in databases of other proteins
- Use multiple sequence alignment to do local structure prediction.
- Build HMM that has multiple tracks (amino-acid and local structure alphabets).
- Search PDB using final HMM.

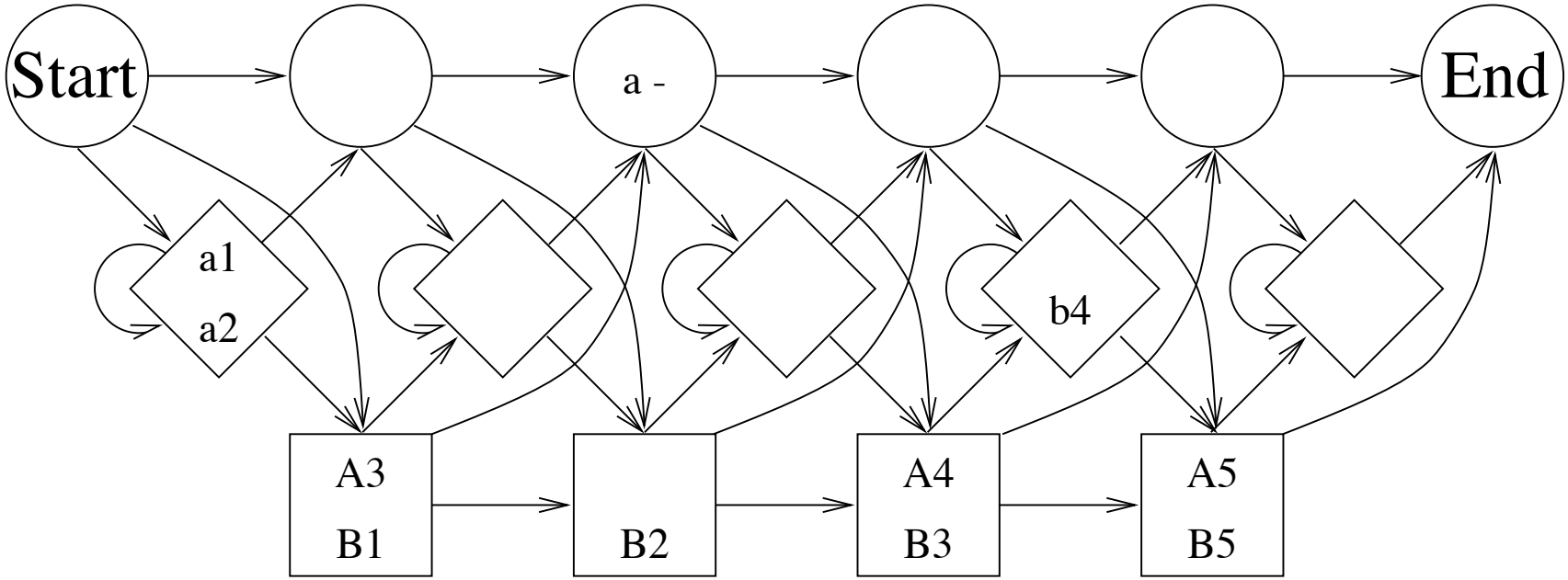
Fold recognition

- Do iterative search to find similar sequences in databases of other proteins:
  - Make a Hidden Markov Model from sequence or alignment.
  - Use HMM to search for similar sequences.
  - Retrain HMM on new set (or representative subset).
  - Align sequences using HMM.
  - Repeat.
- Use multiple sequence alignment to do local structure prediction.
- Build HMM that has multiple tracks (amino-acid and local structure alphabets).
- Search PDB using final HMM.

Fold recognition

- Do iterative search to find similar sequences in databases of other proteins
- Use multiple sequence alignment to do local structure prediction.
- Build HMM that has multiple tracks (amino-acid and local structure alphabets).
- Search PDB using final HMM.
  - Look for similar sequences in database of solved protein structures.
  - Use multi-track HMM to align target to solved stuctures.

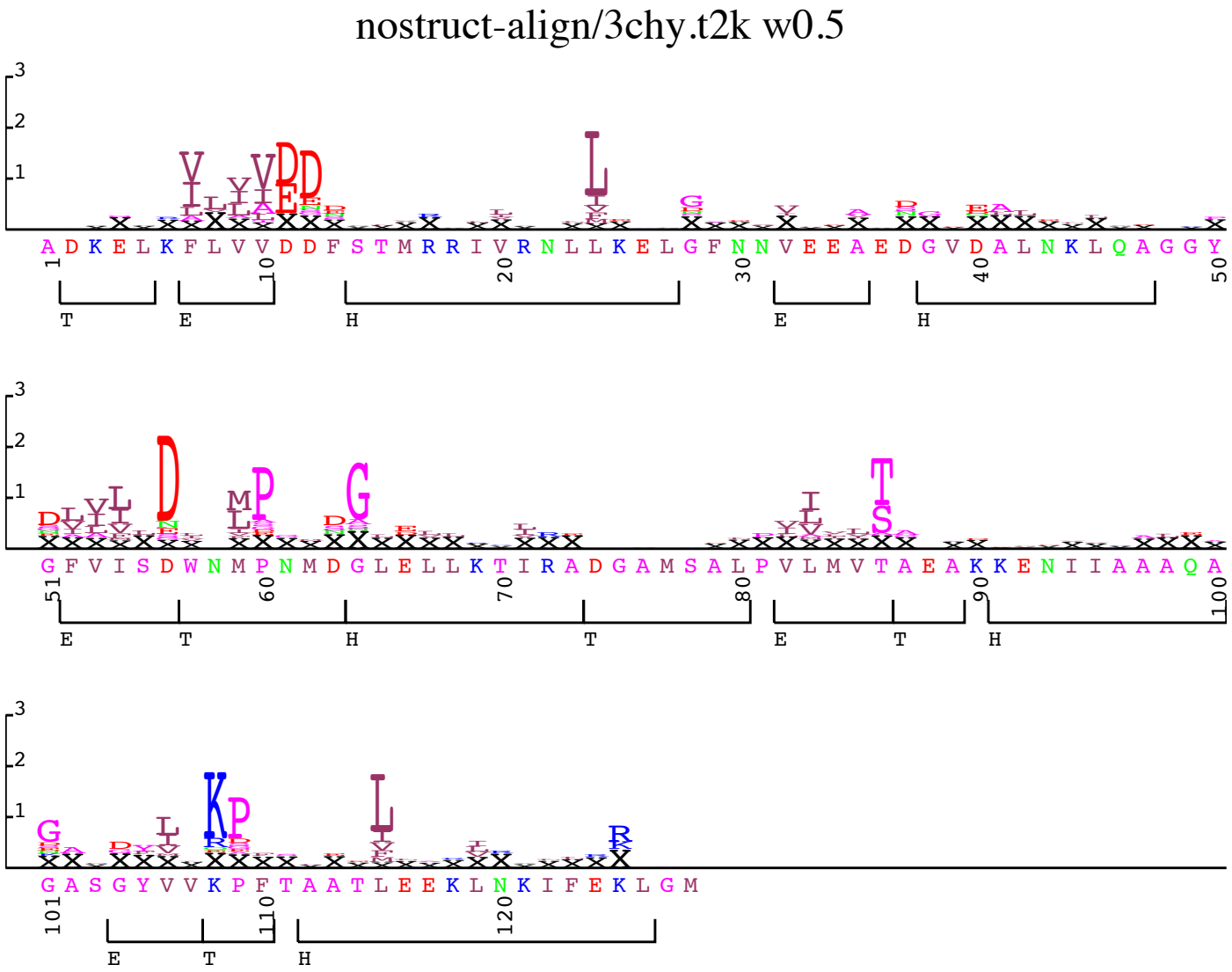
Profile HMM




```
a1 a2 A3     -     A4  .  A5
 .  .  B1     B2    B3 b4  B5
```

Sequence logos (MSA)

Summarize multiple alignment:



Contact prediction

- Predict that residues separated along the chain are close in 3-space.
- Use mutual information between columns.
- Thin alignments aggressively (30%, 35%, 40%, 50%, 62%).
- Compute e-value for mutual info (correcting for small-sample effects).
- Compute rank of log(e-value) within protein.
- Feed log(e-values), log rank, contact potential, joint entropy, and separation along chain for pair, and amino-acid profile, predicted burial, and predicted secondary structure for each residue of pair into a neural net.

Full 3D modeling

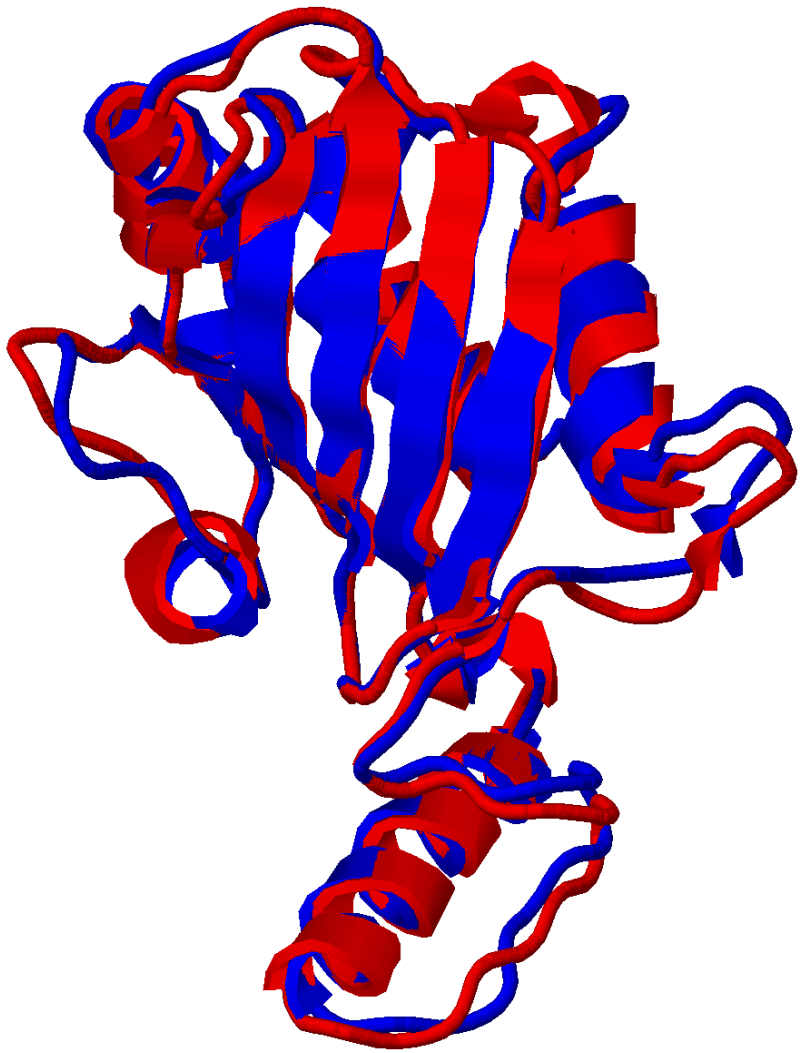
- Copy backbone atoms from aligned PDB file
- Copy fragments from shorter alignments to other PDB files.
- Combine randomly.
- Stochastic search to optimize “energy” function, which may include constraints from alignments, predicted contacts, local structure prediction, ...

CASP ~~Competition~~ Experiment

- Everything published in literature "works"
- CASP set up as true blind test of prediction methods.
- Sequences of proteins about to be solved released to prediction community.
- Predictions registered with organizers.
- Experimental structures compared with solution by assessors.
- "Winners" get papers in *Proteins: Structure, Function, and Bioinformatics*.

T0298 domain 2 (130–315)

RMSD= 2.468Å all-atom, 1.7567Å C_α, GDT=82.5%
best model 1 submitted to CASP7 (red=real)



Computational Protein Design

- Train neural nets to take local-structure inputs and provide amino-acid outputs.
- Use RosettaDesign to design sequences, constrained by neural net outputs.
- Target applications: specific binding of carbon nanotubes, mimics for AGRP (agouti-related protein) binding to different melanocortin receptor.

Outline of genome assembly

- What is a genome?
- What sequencing technologies are currently used?
- The assembly problem
- Algorithms for assembly

What is a genome?

- Complete sequence of all DNA in a cell (exceptions for plasmids, viruses, organelles).
- Varies from cell to cell, so we usually approximate to get a “typical” genome.
- Usually want an *annotated genome* which has genes and other features labeled and indexed.

Current sequencing technologies

- Sequencing by size sorting
- Sequencing by ligation
- Sequencing by replication
- Single-molecule sequencing

Sequencing by size sorting

- Need pure sample: many copies of one DNA molecule.
- Generate "prefixes" of DNA, with known last base.
  - Maxam-Gilbert sequencing (obsolete): cuts DNA at specifc base.
  - Sanger sequencing: copies DNA stopping at specific base.
  - Hood variant: copies DNA using flourescent label for last base.
- Measure lengths of prefixes by electophoresis.
- About \$1.50/read, 800–1200 bases/read
- Error rate about 0.05% (1 in 2000)

Sequencing by ligation

- Only 1 platform (SOLiD)
- Shreds DNA, then does emulsion PCR to get beads with pure DNA fragments.
- Ligates small stretch of DNA to template.
- Unusual "color-space" reads. Color encodes 2 bases, but only 4 colors:

  0 (blue): AA, GG, CC, TT
  1 (green): AC, GT, CA, TG
  2 (yellow): AG, GA, CT, TC
  3 (red): AT, GC, CG, TA

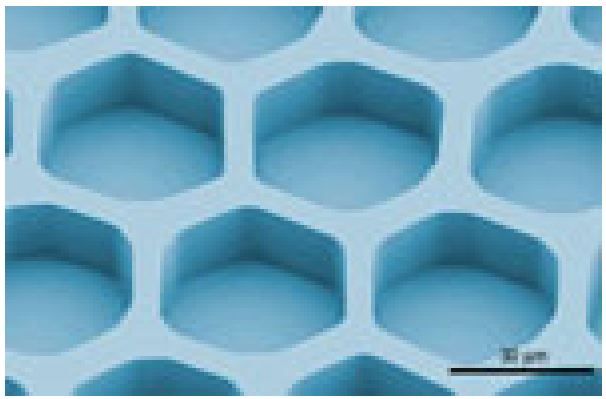
- Takes a week to process a sample
- Get about 200–300 million 50-base reads.
- Error rate about 1.6%

Sequencing by replication

- Bases added one at a time, with detector to tell whether a base is added (or which base is added).
- Pyrosequencing (454)
- Illumina/Solexa (Genome Analyzer)
- Ion Torrent

Pyrosequencing (454 machine)

- After shearing and size-selecting DNA, attach to beads.
- Do emulsion-PCR to get a polony on each bead.
- Put beads into one-bead wells in picotiter plate.



Nature Biotechnology 21, 1425–1427 (2003)

doi:10.1038/nbt1203-1425

- Do polymerization with one base type at a time.
- Use light emission to determine how many copies of base are added to end of chains.

Pyrosequencing (454 machine)

- After shearing and size-selecting DNA, attach to beads.
- Do emulsion-PCR to get a polony on each bead.
- Put beads into one-bead wells in picotiter plate.
- Do polymerization with one base type at a time.
- Use light emission to determine how many copies of base are added to end of chains.
- 1,000,000 reads, 500–1000 bases/read
- about $3k for a run
- Error rate about 0.9%
- When several bases in a row are identical, determining exactly how many bases of that type were present can be difficult. (homopolymer errors)

Illumina/Solexa

- Polonies grown as spots on a slide rather than separate beads.
- One base at a time reading, all 4 bases read at once (different color fluorophors).
- $\approx$ 5 billion 2×100-long paired-end reads.
- Error rate about 1.5%

Ion Torrent

- small, cheap machine (about $50,000)
- Electronic readout, no flourescent molecules, no optics
- medium throughput, fast, low cost per run
- same homopolymer problems as 454 technology
- reads under 100 long

Single-molecule sequencing

- Several new technologies that don't require amplifying DNA:
  - Pacific Bioscience (SMRT)
  - Helicos Bioscience (Helicos)
  - nanopores
- All have super high error rates (10–20%).
- Same molecule must be read repeatedly to get useful data.
- PacBio occasionally gets very long reads, but various tricks are needed, making data analysis difficult.

Characterisitics of data

platform	reads/run	read length	error rate	cost per base
Sanger	1–384	500–1200	very low	very high
454	1e6	500–1000	low	medium
Illumina	4e9	2×100	high	low
SOLiD	300e6	50	high	low

Different data representations

- base space
- flow space (454, Ion Torrent)
- color space
- Each sequencer and each program uses different data formats and different quality information.

The assembly problem

- Jigsaw puzzle with millions of pieces that overlap.
- Need much more DNA sequence than target genome (generally 15–100×)
- Want to end up with single sequence for each chromosome

Problems

- Sequence data is noisy.
- Repeats can have identical sequences in different parts of genome.
- DNA sample may have variations within sample.
- Data is huge (larger than computer memory).

Algorithms for assembly

- Overlap-consensus graph (needs long reads)
- de Bruijn graph (has trouble with high error rates and long reads)

Overlap consensus

- Each node is a single read. Edges represent overlaps between the end of one read and the beginning of another.
- Clusters of connected nodes can be used to build consensus contigs.
- Overlap must be large enough to be unique location in genome, or chimeric contigs can get built.
- Finding overlaps is expensive part.
- Clusters have to be broken where continuation of contig is ambiguous, so repeats tend to be represented by single consensus contig.
- Best method for 454 and Sanger data.

de Bruijn graph

- Each node is a k-mer. Edges connect window $[i, i+k)$ to window $[i+1, i+k+1)$ of read, and have counts of occurrence.
- Each read becomes a path in the graph.
- Contigs build from strongly supported paths.
- Errors create “bubbles” and “dead-ends” that need to be merged into main paths.
- No need to find overlaps, but graphs get huge.

Web sites

These slides:

`http://users.soe.ucsc.edu/~karplus/papers/tools-aug-2011.pdf`

UCSC bioinformatics info:

`http://www.bme.ucsc.edu/`

SAM-T08 prediction server: `http://compbio.soe.ucsc.edu/SAM_T08/T08-query.html`

CASP2 through CASP8—all our results and working notes:

`http://users.soe.ucsc.edu/~karplus/casp2/`

...

`http://users.soe.ucsc.edu/~karplus/casp8/`

Banana Slug Genomics wiki:

`http://banana-slug.soe.ucsc.edu/`