

15-499: Algorithms and Applications

Computational Biology IV
- Sequencing the "Genome"

Thanks to: Dannie Durand for some of the slides.
Various figures borrowed from the web.

15-499

Page 1

Tools of the Trade

Cutting:

Arber, Nathans, and Smith, **Nobel Prize in Medicine** (1978) for "the discovery of restriction enzymes and their application to problems of molecular genetics".

Copying:

Mullis, **Nobel Prize in Chemistry** (1993) for "his invention of the polymerase chain reaction (PCR) method"

Reading: (sequencing)

Gilbert and Sanger, **Nobel Prize in Chemistry** (1980) for "contributions concerning the determination of base sequences in nucleic acids"

15-499

Page 2

Cutting

Cutting:

- Restriction Enzymes:
Cut at particular sites, e.g. ACTTCTAGAT
- Chemical, physical or radiation cuts
Cut at random locations

15-499

Page 3

Copying

Copying:

Cloning a strand of DNA

- Cosmids: clones sequences up to 40K bps
- BAC, PAC: up to about 200K bps
- YAC (yeast artificial chromosomes): up to 1 M

Copying between two specific sites

- PCR (polymerase chain reaction): 500 bps

15-499


Page 4

Cloning (copying fragments)

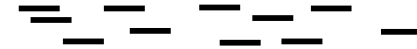
Isolate DNA 

15-499

Page 5

Isolate DNA 

fragmentation

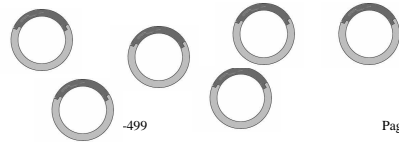
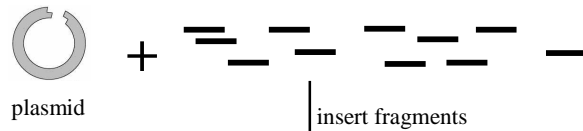



15-499

Page 6

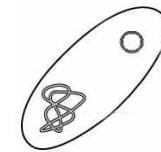
Isolate DNA 

fragmentation

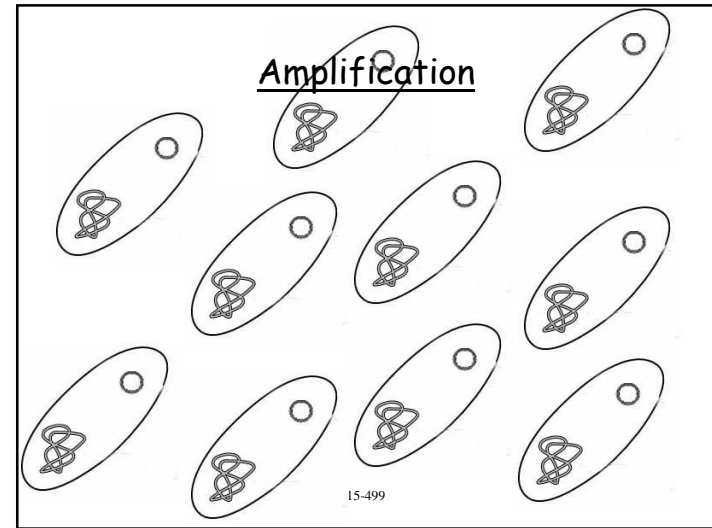
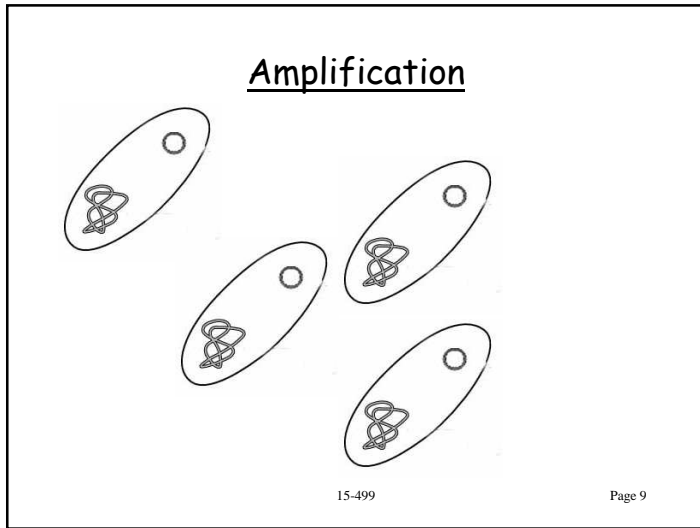
Page 7

Amplification



15-499

Page 8



PCR (Polymerase chain reaction)

Select two sequences that appear in the DNA sequence (e.g. ATACTTAATG and TCTAAGATAG)
 Design two synthetic "**primers**" identical to sequences

REPEAT:

1. **Denature:** Heat DNA to split into two strands
2. **Anneal:** cool and let primers attach
3. **Replicate:** let DNA attach in both directions

Note: cells copy DNA strands character by character

15-499 Page 11

PCR (Polymerase chain reaction)

15-499 Page 12

Reading: sequencing a fragment

Currently too expensive to actually read each bp.

Finding the length is cheap.

- The speed of a fragment in a gel when an electric charge is applied is proportional to its length (DNA has slight negative charge at one end).

Lengths are what are used in Forensic DNA analysis and for DNA "fingerprints"

Gilbert and Sanger got the Nobel Prize for figuring out how to use lengths to "read" a DNA strand from one end.

Currently only good for about 500 bp.

15-499

Page 13

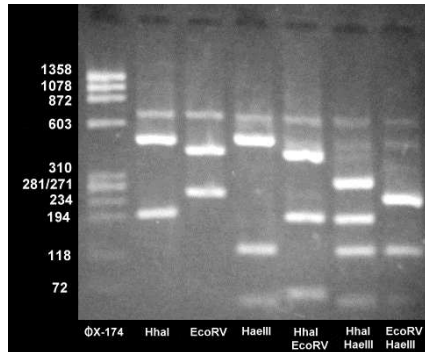
Forensic DNA Analysis

For the two samples, and some "control" DNA

1. Copy using PCR if sample is small
2. Use restriction enzymes to cut up DNA at particular sites (e.g. AATGATGGA)
3. Tag DNA with radioactive (or florescent) tracer
This is a strand that will attach to particular sites of the cut DNA.
4. Put each sample on its own track on a gel
5. Apply charge for fixed time
6. Expose film to see pattern of lengths

15-499

Page 14



7 different DNA sequences and their "fingerprints"

15-499

Page 15

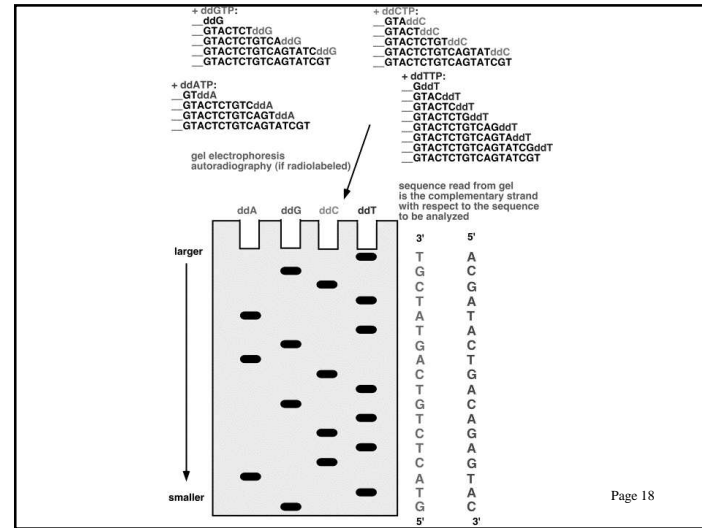
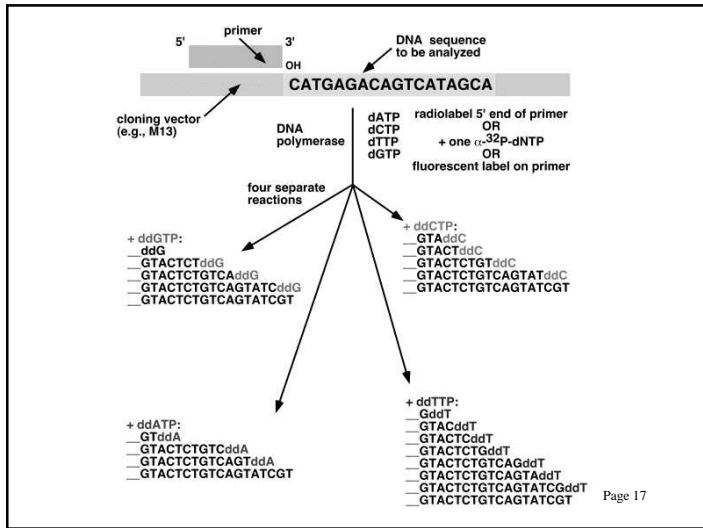
Reading using lengths

Can use special base-pairs that stop growth: DDC, DDA, DDT, DDG.

Will generate all prefixes that end in A, T, C or G.

15-499

Page 16



Improvements

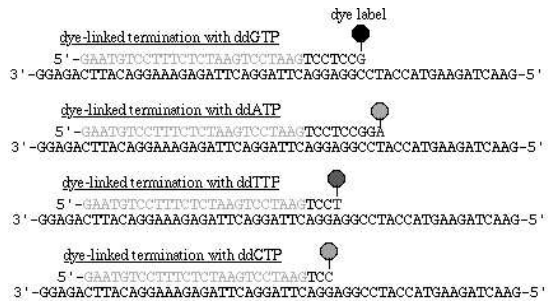
Use fluorescent dyes on the base pairs and laser to excite the dye as it passes a certain point on the gel.

Improvements (1)



4 "test tubes", single track.

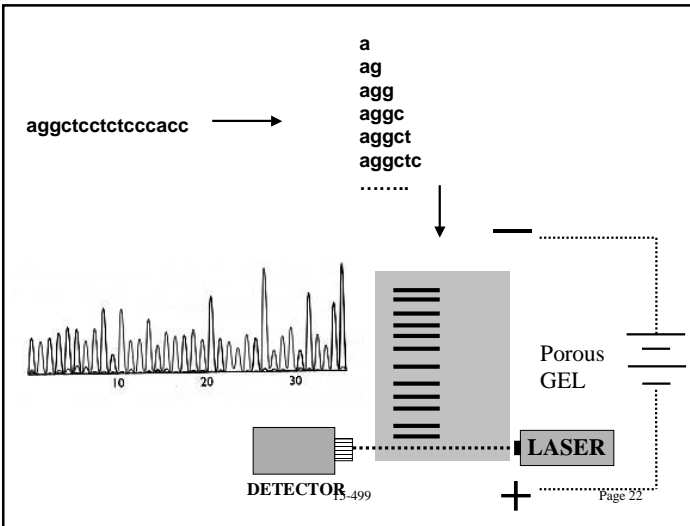
Improvements (2)



Single "test tube", single track

15-499

Page 21



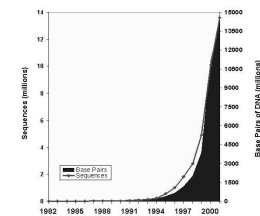
ABI 3700 sequencer

15-499

Page 23

History of Sequencing

- 1971 Nobel prize for restriction enzymes
- 1973 First recombinant DNA
- 1980 Nobel prize for DNA sequencing
- 1988 Congress establishes Genbank
- 1995 First genomic sequence
- 1998 First multicellular organism
- 2000 Fly genome
- 2000 First plant genome
- 2001 Human genome
- 2003 Mouse genome



22 million sequences

28 billion base pairs

15-499

Page 24

Sequencing the Whole Genome

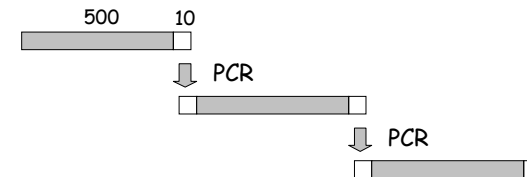
Problem: we only know how to sequence about 500 bps at a time in the lab.

1. Linear sequencing
2. The shotgun method
3. The double-barreled shotgun method.
4. Mapping

15-499

Page 25

Linear Sequencing



Each step takes too long. Requires "wet" runs.
e.g. if each step took 4 hours, sequencing the human genome would take
 $4 \times 3 \times 10^9 / 500$ hours = 3000 years
Also no interesting Computer Science ☺

15-499

Page 26

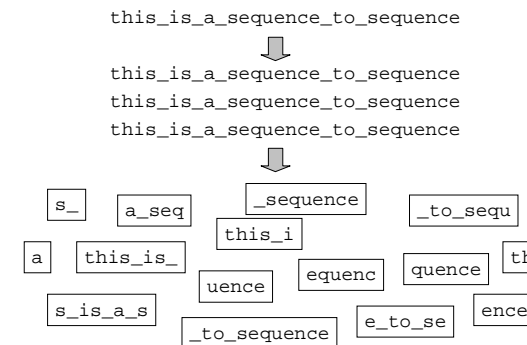
The Shotgun Method

1. Make multiple copies of the sequence.
2. Randomly break sequences into parts (e.g. using radiation or chemicals).
3. Throw away parts that are too small or too large.
4. Read about 500bp from the end of each part
5. Try to put the information together to reconstruct the original sequence

15-499

Page 27

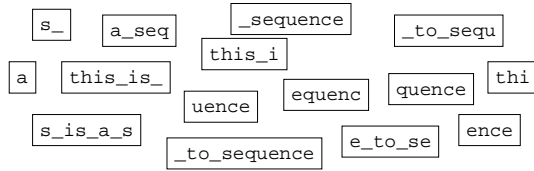
Example



15-499

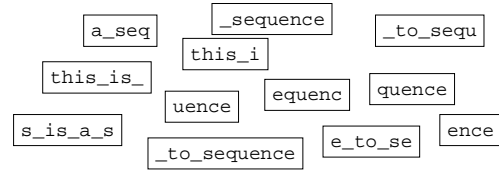
Page 28

Example



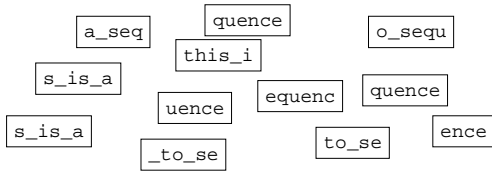
Remove strands that are too short (or too long)

Example



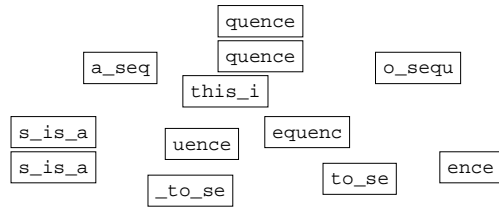
Sequence *k* characters from each (e.g. 6), from either end.

Example

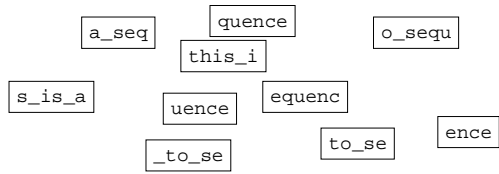


Find overlaps

Example



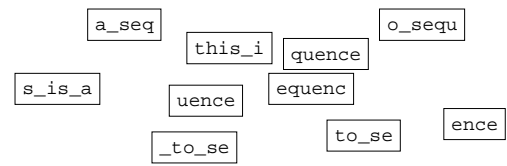
Example



15-499

Page 33

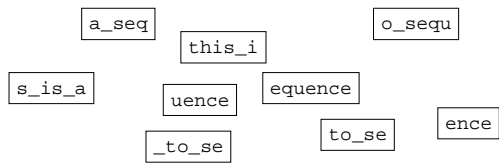
Example



15-499

Page 34

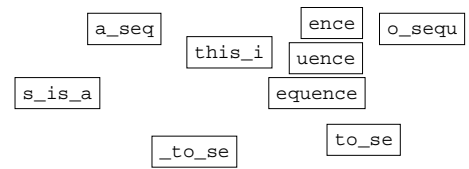
Example



15-499

Page 35

Example



15-499

Page 36

Example

a_seq o_sequ
this_i
s_is_a equence
_to_se to_se

15-499

Page 37

Example

a_seq o_sequ
this_i equence
s_is_a
_to_se to_se

15-499

Page 38

Example

a_seq o_sequence
this_i
s_is_a
_to_se to_se

15-499

Page 39

Example

a_seq o_sequence
this_i _to_se
s_is_a
to_se

15-499

Page 40

Example

a_seq
this_i _to_sequence
s_is_a
to_se

15-499

Page 41

Example

a_seq
this_i _to_sequence
s_is_a to_se

15-499

Page 42

Example

a_seq
this_i _to_sequence
s_is_a

15-499

Page 43

Example

a_seq
s_is_a
this_i _to_sequence

15-499

Page 44

Example

a_seq

_to_sequence

this_is_a

Having a single character overlap might not be enough to assume they overlap.

15-499

Page 45

Example

a_seq this_is_a _to_sequence

15-499

Page 46

Example

a_seq this_is_a _to_sequence

We are left with **gaps**, and unsure matches.
Each covered region (e.g. `this_is_a`) is called a **contig**

Is there a systematic way to find or even define a "best solution"?

15-499

Page 47

The SSP: an attempt

The shortest superstring problem: given a set of strings s_1, s_2, \dots, s_n find the shortest string S that contains all s_i .

NP-Hard, but can be reduced to TSP and solved approximately (nearly optimally in practice).
Even if easy to solve, are we done?

Our example gives:

`this_is_a_seq_to_sequence`

but this is the best we can do given the data.

This problem is caused by repeats.

Other problems?

15-499

Page 48

Problems

In practice the data is noisy.

- Reads have up to a 1% error rate
- Samples could have contaminants
- Fragments can sometimes join up

The reads could be in either direction (front-to-back or back-to-front). Cannot distinguish.

15-499

Page 49

Assembly in Practice

Score all suffix-prefix pairs

gatcgaat_ga

aattgactactatg

- This can use a variant of the global alignment prob. It is the most expensive step (n^2 scores).

Repeat:

- Select best score and check for consistency
- If score is too low, quit
- If there is a good overlap, merge the two.

Determine consensus:

- We know the ordering among strands, but since matches are approximate, we need to select bps. Can use, e.g., multiple alignment over windows.

15-499

Page 50

Some Programs for Assembly

Phrap
SEQAID
CAP
TIGR
Celera assembler
ARACHNE

After using one of these programs to generate a set of "contigs" with some gaps, one can use the linear method to fill in the gaps (assuming they are small).

atgattagccagtacgttt

tcagcatcccagtacgttatgcac

tttagccaga

15-499

Page 51