

Class 35: Decoding DNA



DNA Helix Photomosaic from cover of *Nature*, 15 Feb 2001 (made by Eric Lander)

Sign up for your PS8 team design review!

CS150: Computer Science
University of Virginia
Computer Science

David Evans
<http://www.cs.virginia.edu/evans>

Speculations

- Must study math for 15+ years before understanding an (important) open problem
 - Was ~10 until Andrew Wiles proved Fermat's Last Theorem
- Must study physics for ~6 years before understanding an open problem
- Must study computer science for 1 semester before understanding the most important open problem
 - Unless you're a 6-year old at Cracker Barrel
- But, every 5 year-old understands the most important open problems in biology!

CS150 Fall 2005: Lecture 35: Decoding DNA

2 Computer Science
University of Virginia

Biology's Open Problem

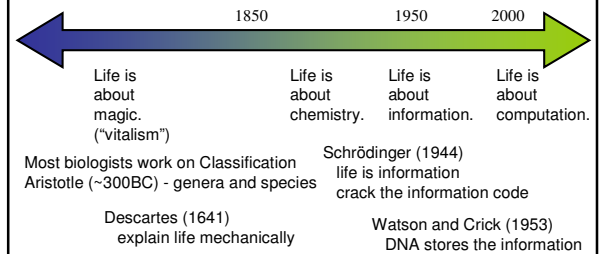


How can a (relatively) simple, single cell turn into a chicken?

CS150 Fall 2005: Lecture 35: Decoding DNA

3 Computer Science
University of Virginia

Brief History of Biology

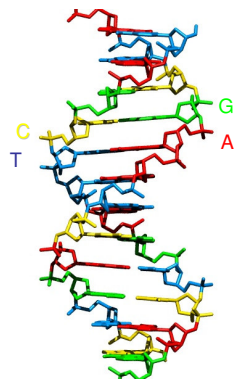


CS150 Fall 2005: Lecture 35: Decoding DNA

4 Computer Science
University of Virginia

DNA

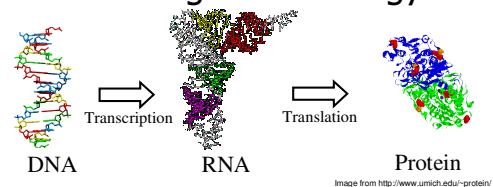
- Sequence of nucleotides: adenine (A), guanine (G), cytosine (C), and thymine (T)
- Two strands, A must attach to T and G must attach to C



CS150 Fall 2005: Lecture 35: Decoding DNA

5 Computer Science
University of Virginia

Central Dogma of Biology



- RNA makes copies of DNA segments
- RNA describes sequences of amino acids
- Chains of amino acids make proteins

CS150 Fall 2005: Lecture 35: Decoding DNA

6 Computer Science
University of Virginia

Encoding Proteins

- There are 4 nucleotides: adenine (A), guanine (G), cytosine (C), and thymine (T) (replaced with uracil (U) in RNA)
- There are 20 different amino acids, and a stop marker (to separate proteins)
- How many nucleotides are needed to encode one amino acid?

with 2, could encode 16 things: $4 * 4$
with 3, could encode 64 things: $4 * 4 * 4$

Codons

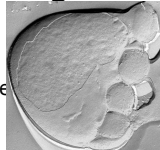
- Three nucleotides encode an amino acid
- But, there are only 20 amino acids, so there may be several different ways to encode the same one

		SECOND POSITION					
		U	C	A	G		
FIRST POSITION	U	phenylalanine Phe	leucine Leu	tyrosine Tyr	tryptophan Trp	U	THIRD POSITION
	C	leucine Leu	proline Pro	histidine His	arginine Arg	C	
A	isoleucine Ile	methionine Met	asparagine Asn	glutamine Gln	A		
G	valine Val	alanine Ala	serine Ser	glycine Gly	G		

From <http://web.mit.edu/esgbio/www/dogma/dogma.html>

Shortest (Known) Life Program

- *Nanoarchaeum equitans*
 - 490,885 bases (522 genes)
 - $490,885 * \frac{1}{4} * \frac{21}{64} = 40,268$ bytes
 - Parasite: no metabolic capacity, must steal from host
 - Complete components for information processing: transcription, replication, enzymes for DNA repair
- Size of compiling C++ "Hello World":
 - Windows (bcc32): 112,640 bytes
 - Linux (g++): 11,358 bytes



<http://www.mediacover.net/Extremophiles.cfm>
KG Sletten and Dr. Rachel Reichberg

How Big is the Make-a-Human Program?

- 3 Billion Base Pairs
 - Each nucleotide is 2 bits (4 possibilities)
 - $3 \text{ B pairs} * 1 \text{ byte}/4 \text{ pairs} = 750 \text{ MB}$
- Every sequence of 3 base pairs one of 20 amino acids (or stop codon)
 - 21 possible codons, but $4^3 = 64$ possible
 - So, really only $750 \text{ MB} * (21/64) \sim 250 \text{ MB}$
- Most of it (> 95%) is probably junk

1 CD ~ 650 MB

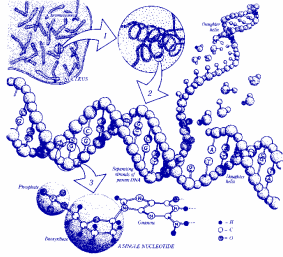


People are almost all the Same

- Genetic code for 2 humans differs in only 2.1 million bases
 - 4 million bits = 0.5 MB
- How big is 0.5MB?
 - 1/3 of a floppy disk
 - ~22 times the size of the PS6 adventure game code



Is DNA Really a Programming Language?



CS150 Fall 2005: Lecture 35: Decoding DNA

13 Computer Science

Stuff Programming Languages are Made Of

- **Primitives**
codons (sequence of 3 nucleotides that encodes a protein)
- **Means of Combination**
?? Morphogenesis? Not well understood (by anyone).
This is where most of the expressiveness comes from!
- **Means of Abstraction**
DNA itself – separate proteins from their encoding
Genes – group DNA by function (sort of)
Chromosomes – package Genes together
Organisms – packages for reproducing Genes

CS150 Fall 2005: Lecture 35: Decoding DNA

14 Computer Science

Jacob and Monod, 1959

- Not so simple: cells in an organism have the same DNA, but do different things
 - Structural genes: make proteins that make us
 - Regulator genes: control rate of transcription of other genes

The genome contains not only a series of blue-prints, but a coordinated program of protein synthesis and the means for controlling its execution.
François Jacob and Jacques Monod, 1961

CS150 Fall 2005: Lecture 35: Decoding DNA

15 Computer Science

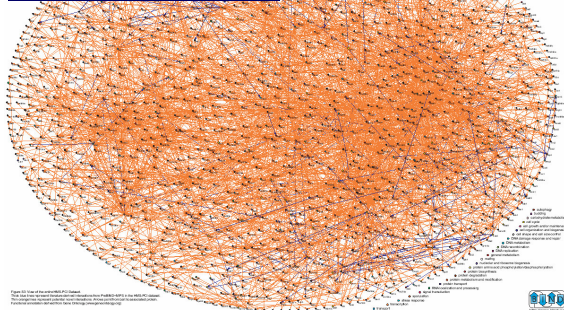
Split Genes

- Richard Roberts and Phillip Sharp, 1977
- Not so simple – genome is spaghetti code (exons) with lots of noops/comments (introns)
- Exons can be spliced together in different ways before transcription
- Possible to produce 100s of different proteins from one gene

CS150 Fall 2005: Lecture 35: Decoding DNA

16 Computer Science

Saccharomyces Cerevisiae
(Yeast) Protein Interactions,
4825 proteins, ~15,000 interactions
Bader and Hogue, Nature 2002



CS150 Fall 2005: Lecture 35: Decoding DNA

17 Computer Science


Most Important Science/Technology Races

- 1930-40s: Decryption Nazis vs. British
Winner: British
Reason: Bletchley Park had computers (and Alan Turing), Nazi's didn't
- 1940s: Atomic Bomb Nazis vs. US
Winner: US
Reason: Heisenberg miscalculated, US had better physicists, computers, resources
- 1960s: Moon Landing Soviet Union vs. US
Winner: US
Reason: Many, better computing was a big one
- 1990s-2001: Sequencing Human Genome

CS150 Fall 2005: Lecture 35: Decoding DNA


18 Computer Science

Human Genome Race



Francis Collins
(Director of
public National
Center for
Human Genome
Research)
(Picture from
UVa Graduation
2001)

VS.


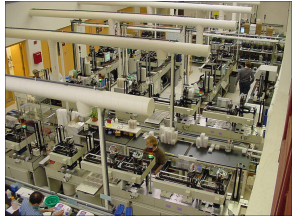



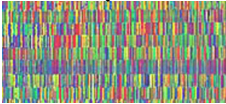
Craig Venter
(President of
Celera
Genomics)

- UVa CLAS 1970
- Yale PhD
- Tenured Professor at U. Michigan
- San Mateo College
- Court-martialed
- Denied tenure at SUNY Buffalo

CS150 Fall 2005: Lecture 35: Decoding DNA 19 Computer Science

Reading the Genome

Whitehead Institute, MIT

CS150 Fall 2005: Lecture 35: Decoding DNA 20 Computer Science

Gene Reading Machines

- One read: about 700 base pairs
- But...don't know where they are on the chromosome

Read 3 TACCCGTGATCCA

Read 2 TCCAGAATAA

Read 1 ACCAGAATACC

Actual Genome AGGCATACCAGAATACCCGTGATCCAGAATAAGC

CS150 Fall 2005: Lecture 35: Decoding DNA 21 Computer Science

Genome Assembly

Read 1 ACCAGAATACC

Read 2 TCCAGAATAA

Read 3 TACCCGTGATCCA

Input: Genome fragments (but without knowing where they are from)

Output: The full genome

CS150 Fall 2005: Lecture 35: Decoding DNA 22 Computer Science

Genome Assembly

Read 1 ACCAGAATACC

Read 2 TCCAGAATAA

Read 3 TACCCGTGATCCA

Input: Genome fragments (but without knowing where they are from)

Output: The smallest genome sequence such that all the fragments are substrings.

CS150 Fall 2005: Lecture 35: Decoding DNA 23 Computer Science

Common Superstring

Input: A set of n substrings and a maximum length k .

Output: A string that contains all the substrings with total length $\leq k$, or no if no such string exists.

ACCAGAATACC

TCCAGAATAA

TACCCGTGATCCA

$n = 26$

→

ACCAGAATACC

TCCAGAATAA

TACCCGTGATCCA

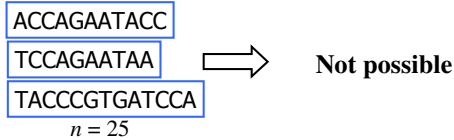
ACCAGAATACCCGTGATCCAGAATAA

CS150 Fall 2005: Lecture 35: Decoding DNA 24 Computer Science

Common Superstring

Input: A set of n substrings and a maximum length k .

Output: A string that contains all the substrings with total length $\leq k$, or no if no such string exists.



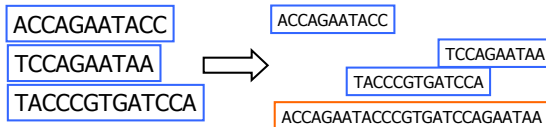
Common Superstring

- In NP:
 - Easy to verify a “yes” solution: just check the letters match up, and count the superstring length
- NP-Complete
 - Similar to Smiley Puzzle!
 - Could transform 3SAT into Common Superstring problem

Shortest Common Superstring

Input: A set of n substrings

Output: The shortest string that contains all the substrings.



Shortest Common Superstring

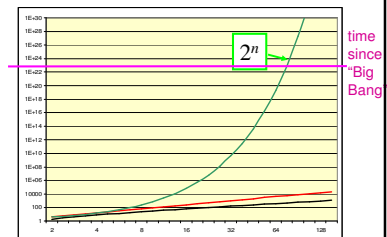
- Also is NP-Complete:
 - function** scsuperstring (pieces)
 - maxlen = sum of lengths of all pieces
 - for** $k = 1$ **to** $k = \text{maxlen}$ **step 1 do**
 - if** (commonSuperstring (pieces, k))
 - return** commonSuperstring (pieces, k)
 - end for**

Human Genome

- 3 Billion base pairs
- 600-700 bases per read
- $\sim 8X$ coverage required
 - $> (/ (* 8 (* 3 1000 1000 1000)) 650)$
 - 36923076 12/13
- So, $n \approx 37$ Million sequence fragments
- Celera used 27.2 Million reads (but could get more than 700 bases per read)

Give up?

No way to solve an NP-Complete problem (best known solutions being $O(2^n)$ for $n \approx 20$ Million)



Approaches

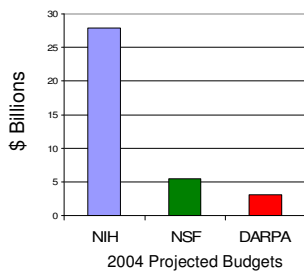
- Human Genome Project (Collins)
 - Start by producing a genome map (using biology, chemistry, etc) to have a framework for knowing where the fragments should go
- Celera Solution (Venter)
 - Approximate: we can't guarantee finding the shortest possible, but we can develop clever algorithms that get close most of the time in $O(n \log n)$

Result: Draw



President Clinton announces Human Genome Sequence essentially complete (with Venter and Collins), June 26, 2000

But, Human Genome Project mostly adopted Venter's approach.



So Why Haven't We Cured Cancer Yet?

Why Biologists Haven't Done Much Useful with the Human Genome Yet

They are trying to debug highly concurrent, asynchronous, type-unsafe, multiple entry/exit, self-modifying programs that create programs that create programs running on an undocumented, unstable, environmentally-sensitive OS by looking at the bits (and just figuring out the shape of a protein is an NP-hard problem)

Charge

- Meet with your project teams before the design review meeting
 - You don't need a formal presentation, but should have notes prepared