Bioinformatics: An Introduction

David Cain

2012-08-30

Background

About me

- Education
  - Colby College, 2013
  - Majors:
    - computer science
    - economics
- Work history
  - Sasisekharan Lab: 2010 - present

About the Sasisekharan Lab

SASISEKHARAN LABORATORY
HARVARD-MIT DIVISION OF HEALTH SCIENCES AND TECHNOLOGY
DEPARTMENT OF BIOLOGICAL ENGINEERING

Discoveries

- Important mutation point discovered in H1N1
- Heparin contamination

Research

- Specializes in glycobiology
- Focus extends into multiple fields
- Multidisciplinary approach to research

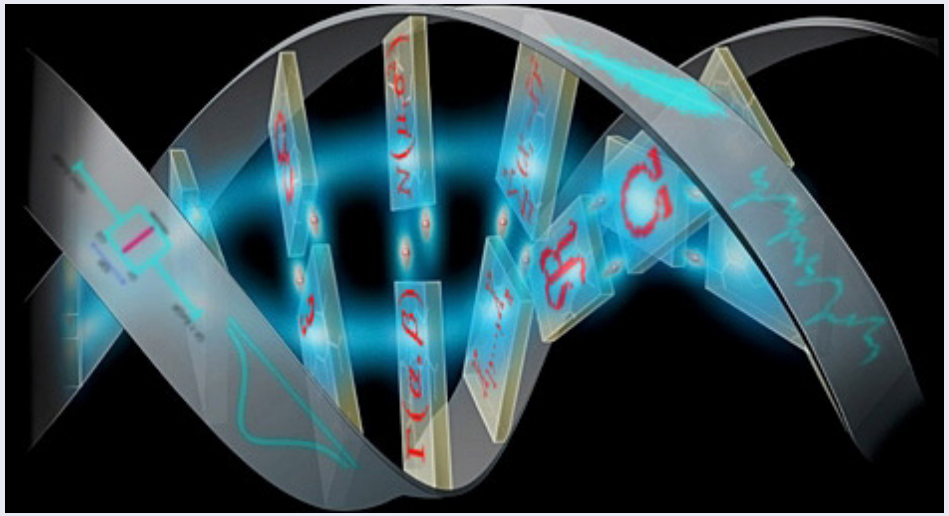
Affiliations

- Harvard-MIT Division of HST
- SMART

What *is* bioinformatics?

Definition

"A branch of biological science which deals with the study of methods for storing, retrieving and analyzing biological data."

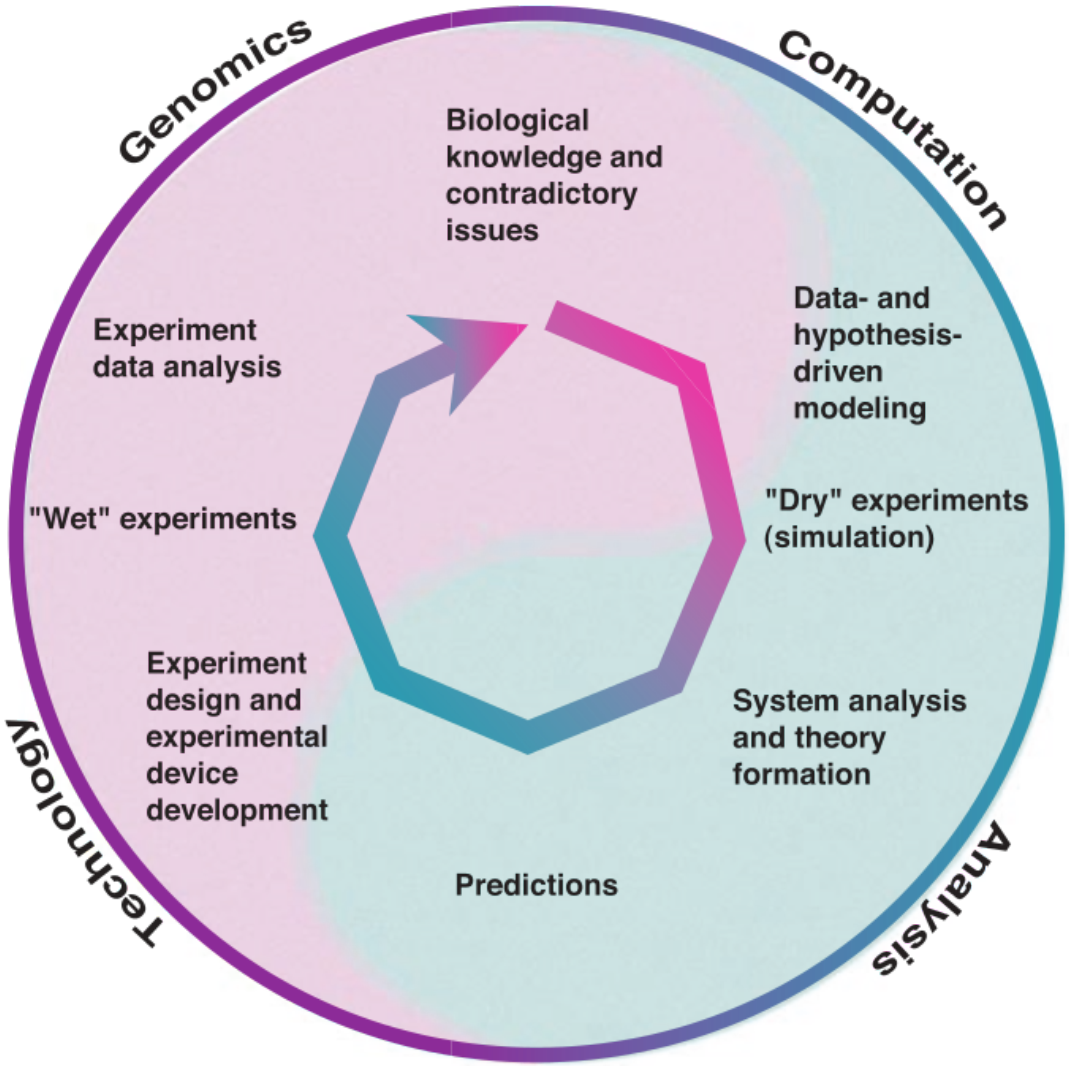


Systems biology

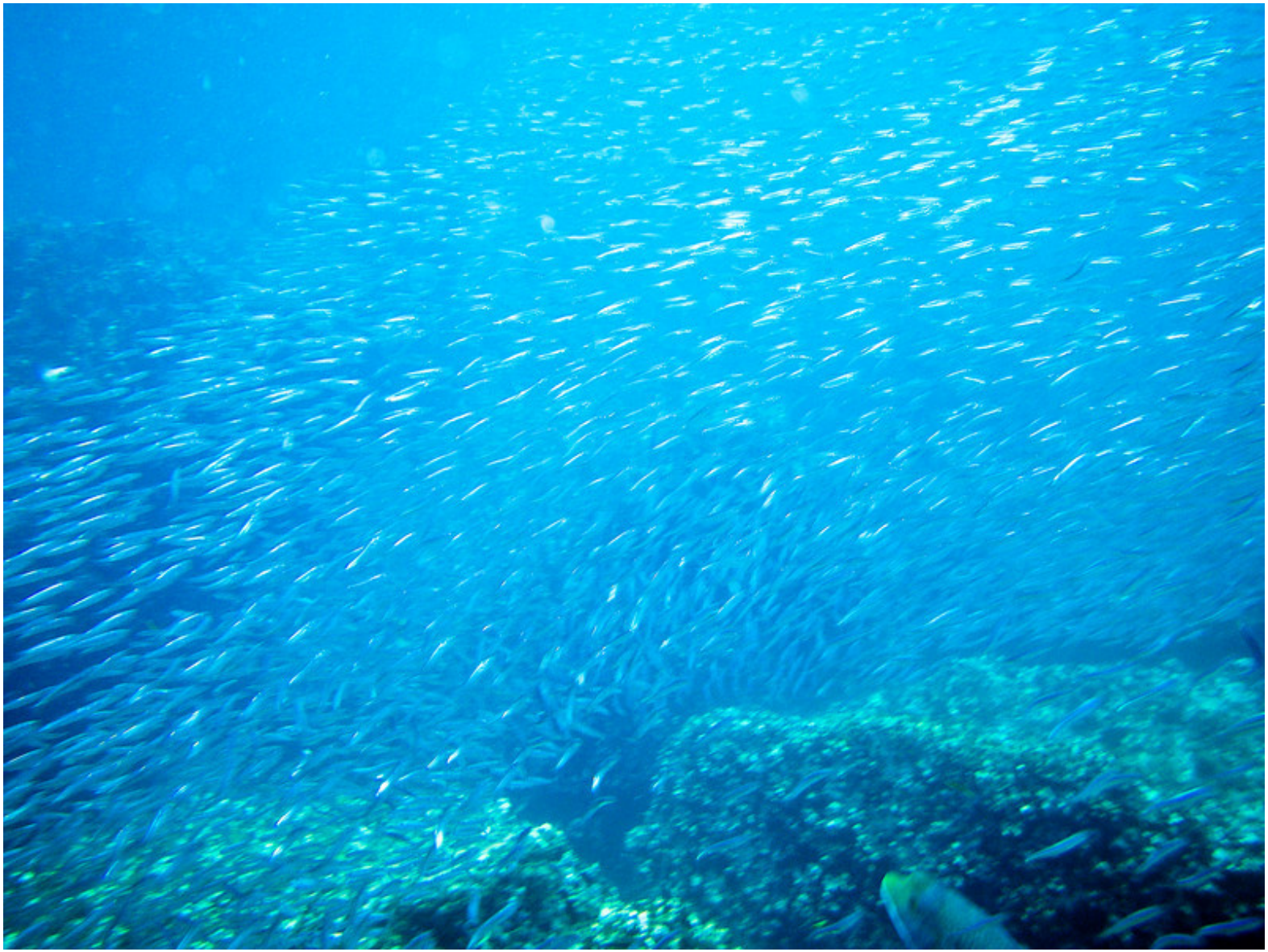
What is it?

- Seeks to understand how units in a system *interact*.
- Dynamic systems as a whole
- Build hypotheses and construct computational models
- Doesn't entirely fall under the classification of bioinformatics; it's quite multidisciplinary.

Systems biology



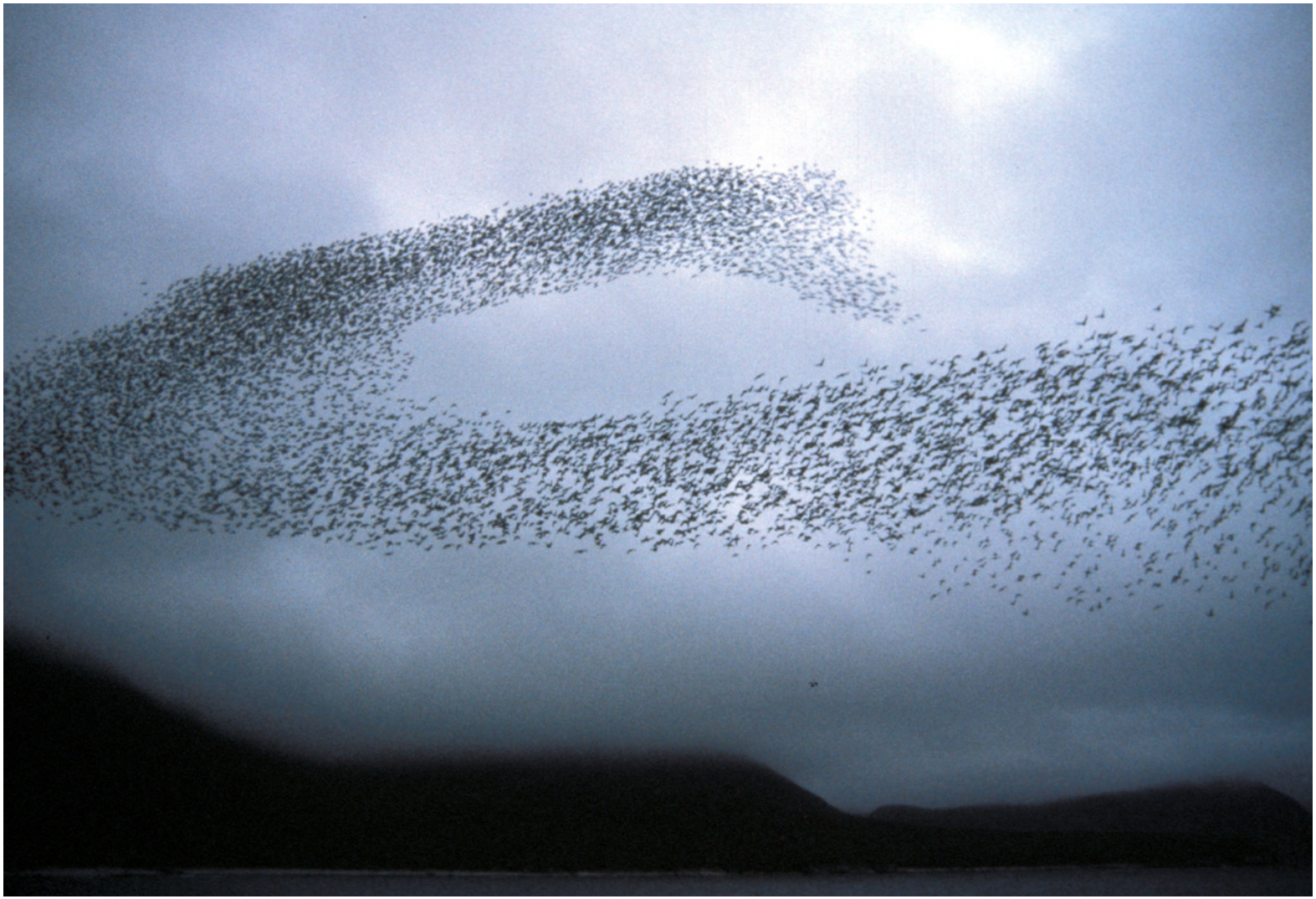
Systems biology



Systems biology

Systems biology

Systems biology



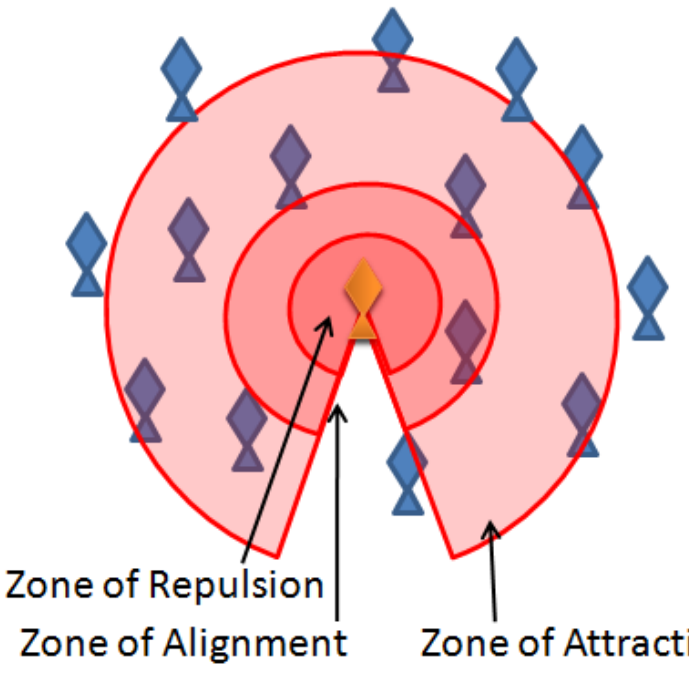
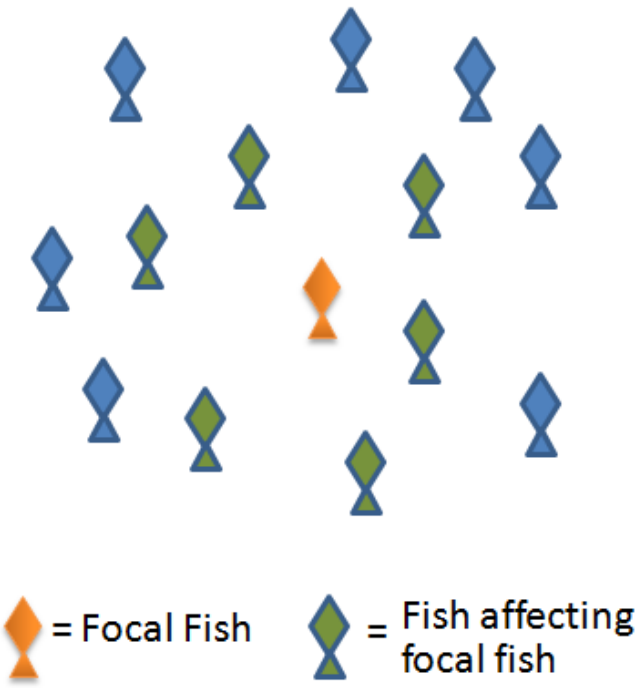
Systems biology

Systems biology

Systems biology

Common theme?

- Swarm behavior
- Swarm intelligence

Metric Distance Model	Topological Distance Model
	
Focal fish pays attention to all of the fish within a certain distance	Focal fish only pays attention to the 6 or 7 fish closest to itself, regardless of distance

Systems biology

Simulators

- Boids – Craig Reynolds, 1986

Example #2

Other applications:

Systems biology

Simulators

- Boids – Craig Reynolds, 1986

Example #2

- Circadian rhythm

Other applications:

Systems biology

Simulators

- Boids – Craig Reynolds, 1986

Example #2

- Circadian rhythm

Other applications:

- genetic algorithms

Systems biology

Simulators

- Boids – Craig Reynolds, 1986

Example #2

- Circadian rhythm

Other applications:

- genetic algorithms
- simulating evolution

Systems biology

Simulators

- Boids – Craig Reynolds, 1986

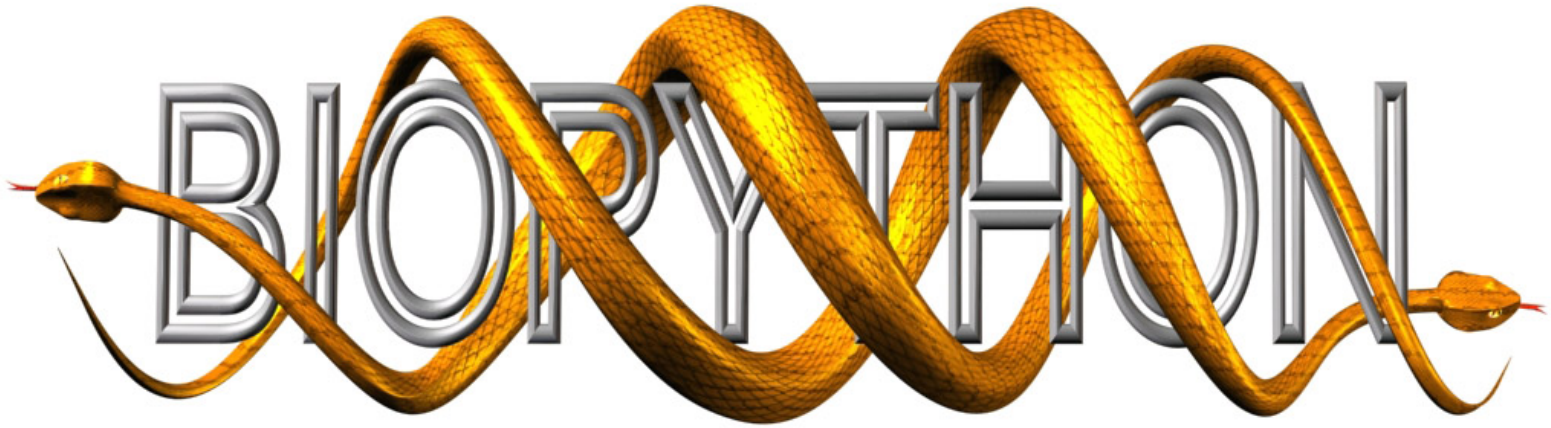
Example #2

- Circadian rhythm

Other applications:

- genetic algorithms
- simulating evolution
- cellular subsystems

Bioinformatics tools



About Biopython

- "A set of libraries to provide the ability to deal with 'things' of interest to biologists working on the computer."
- Package modules to address many areas within bioinformatics
- Open source, completely free/libre

Working with Sequences

Sequences

- A biological sequence is the result of reading a biological object (e.g. a protein or a string of DNA) end to end.
- Represented as a string of characters
- Characters belong to a pre alphabet
  - Protein alphabet
  - DNA/RNA alphabet

Working with Sequences: Simple Example #1

- DNA is double-stranded
- Each side of a double helix is the complement of the other

Using Biopython:

```
>>> from Bio.Seq import Seq
>>> from Bio.Alphabet import IUPAC
>>> dna_seq = Seq("CGATATAGGATCG", IUPAC.unambiguous_dna)
>>> dna_seq
Seq('CGATATAGGATAA', IUPACUnambiguousDNA())
>>> print dna_seq.complement()
Seq('GCTATATCCTATT', IUPACUnambiguousDNA())
```

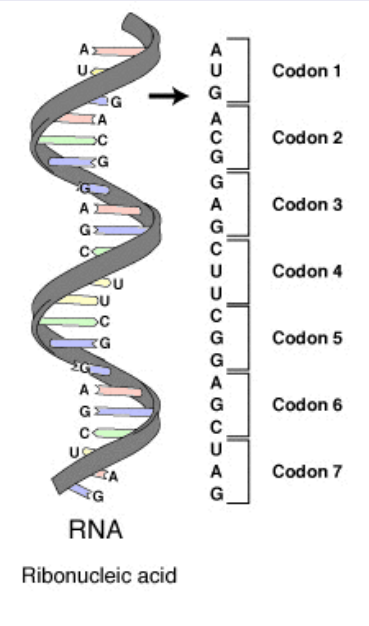
Working with Sequences: Simple Example #2

If we assume this DNA sequence to be a template strand, we can easily find the mRNA sequence that would create the fragment:

```
>>> dna_seq
Seq('CGATATAGGATAA', IUPACUnambiguousDNA())
>>> dna_seq.reverse_complement().transcribe()
Seq('UUAUAUAGGAUCG', IUPACUnambiguousRNA())
```

Sequence mutation

- DNA mutation -> protein mutation
- redundancy in codon to amino acid translation
- radical changes are selected against



Probability of mutation

- Given amino acids are more likely to transform to certain amino acids than they are to others.
  - Properties of codon -> amino acid code
  - Evolutionary pressures
- Substitution matrices
  - Store probability that one amino acid will mutate to another
  - Metric of evolutionary similarity

Sequence alignment

BLOSUM substitution matrices

- **BLO**ck **SU**bstitution **M**atrix
- Constructed with local alignments of evolutionarily divergent proteins
- Observed conserved sequence blocks to build probabilities
- Accounts for the bias given by very similar sequences
  - Threshold percent identity
  - BLOSUM62 is very good at aligning distant sequences

Alignment

- Indels
- Gap penalties

Sequence alignment

	Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
Ala	4																			
Arg	-1	5																		
Asn	-2	0	6																	
Asp	-2	-2	1	6																
Cys	0	-3	-3	-3	9															
Gln	-1	1	0	0	-3	5														
Glu	-1	0	0	2	-4	2	5													
Gly	0	-2	0	-1	-3	-2	-2	6												
His	-2	0	1	-1	-3	0	0	-2	8											
Ile	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

Alignment example

- Input: various H1N1 strains
- Tool: Clustalx

Finding similar sequences

BLAST

- **B**asic **L**ocal **A**lignment **T**ool
- A similarity search against all known sequences
- Web services to handle BLAST queries

- Can be run locally
- Explanation of BLAST algorithm

Finding similar sequences: example

Example BLAST query

- BLAST tool: protein blast
- Input: a subsequence of the H1N1 virus, found in 1935
- Output: evolutionarily similar sequences?

Finding similar sequences: example

[Influenza A virus (A/Puerto Rico/8/**1934**(H1N1))]
[Influenza A virus (A/Puerto Rico/8-JV1/**1934**(H1N1))]
[Influenza A virus (A/Puerto Rico/8-KV7/**1934**(H1N1))]
[Influenza A virus (A/Puerto Rico/8-KV3/**1934**(H1N1))]
[Influenza A virus (A/Puerto Rico/8-CV6/**1934**(H1N1))]
[Influenza A virus (A/Puerto Rico/8-JV3/**1934**(H1N1))]
[Influenza A virus (A/Puerto Rico/8-CV10/**1934**(H1N1))]
[Influenza A virus (A/lvPR8/**34**(H1N1))]
[Influenza A virus (A/Alaska/**1935**(H1N1))]
[Influenza A virus (A/Mongolia/153/88(H1N1))]
[Influenza A virus (A/Puerto Rico/8-KV1/**1934**(H1N1))]
[Influenza A virus (A/Puerto Rico/8-KV8/**1934**(H1N1))]
[Influenza A virus (A/Puerto Rico/8-KV6/**1934**(H1N1))]
[Influenza A virus (A/Puerto Rico/8-KV5/**1934**(H1N1))]

Secondary structure

Structures

- Alpha helixes
  - Most common, easily predictable
  - N-H group to C=O group 4 residues earlier
- Beta sheets
  - Parallel/antiparallel
  - N-H to C=O bond

Obtaining secondary structure

- `dssp`: secondary structure from a PDB file

Mutation analysis

Mutations

- A few are beneficial

Mutation analysis

Mutations

- A few are beneficial
- Most are benign

Mutation analysis

Mutations

- A few are beneficial
- Most are benign
- Some are *deadly*.

Mutation analysis

Mutations

- A few are beneficial
- Most are benign
- Some are *deadly*.
- **What makes a mutation dangerous?**

Mutations

H1N1: a case study

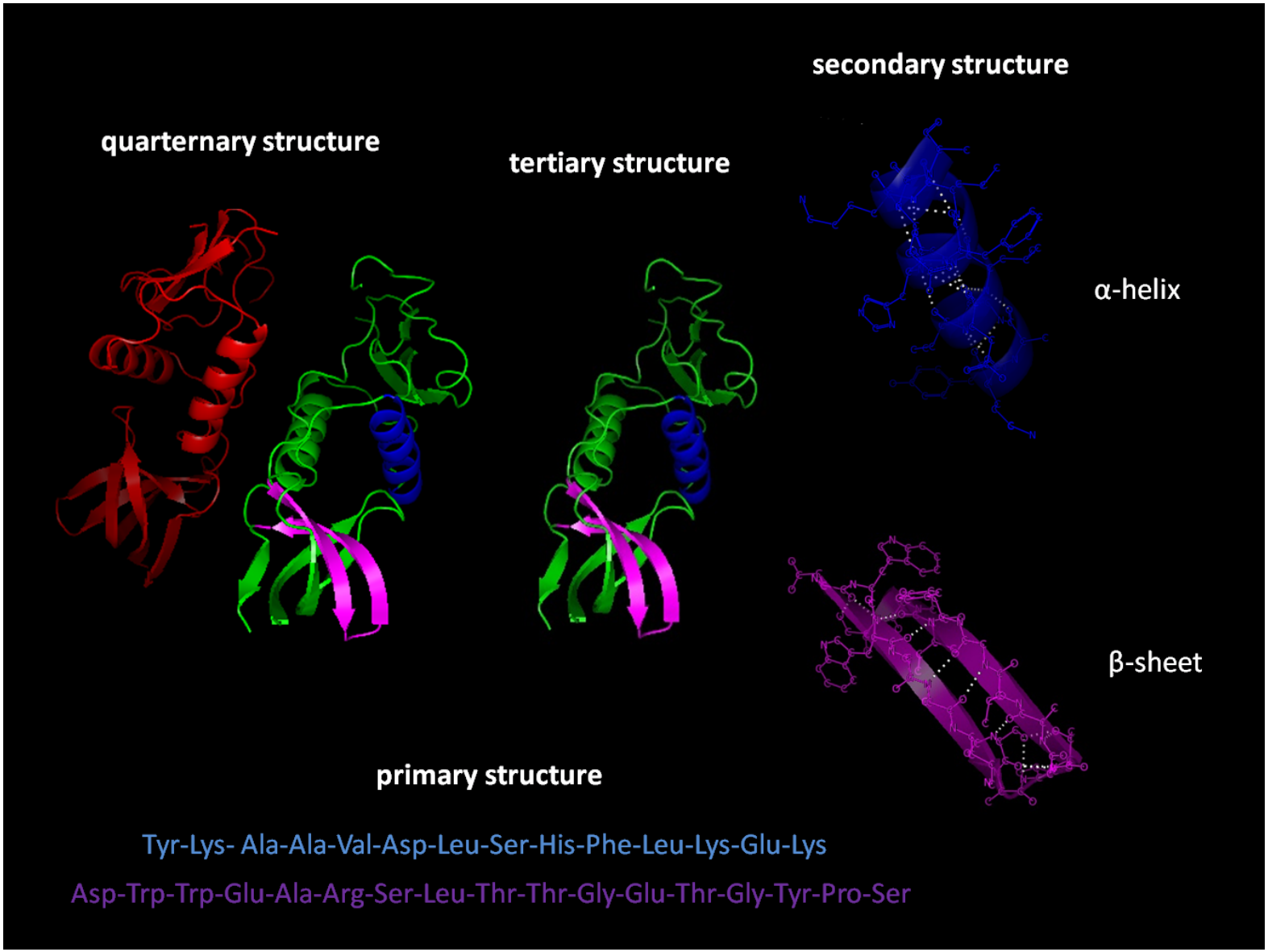
- WHO feared a large-scale pandemic with a high death toll.
- Far fewer people died than expected.
- That might not be the case next time.
- Important mutation point discovered in H1N1

Mutation analysis

Other major fields

- Understanding cancerous mutations

Tertiary structure



Homology modelling

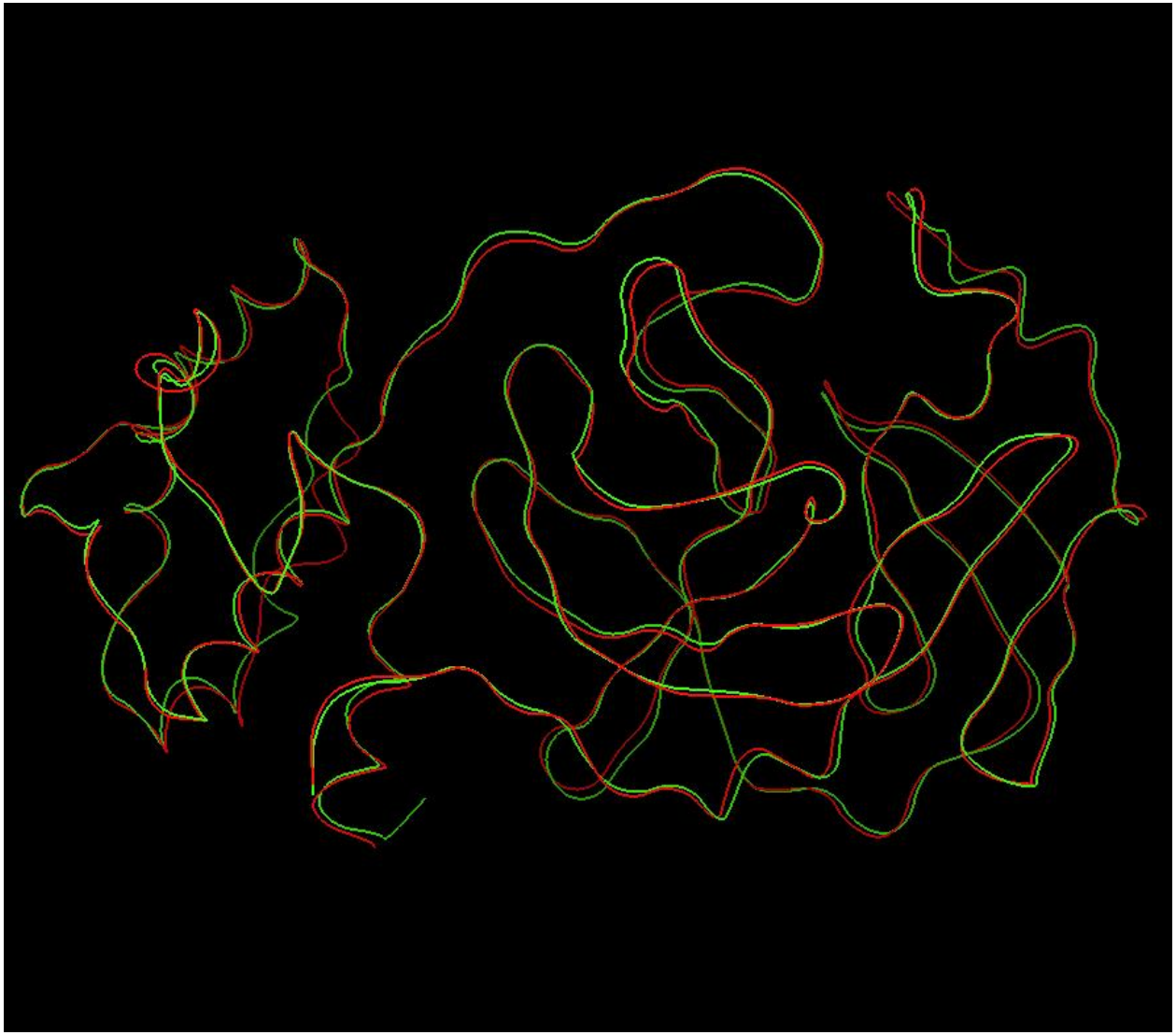
Why it's done

- Absence of structural data
- Difficulties of crystallography
- Structures conserve better among homologues

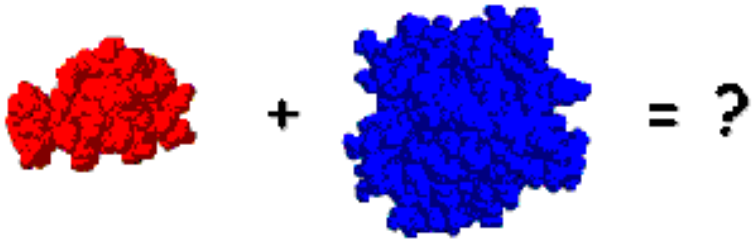
How it's done

- Target to template
- Primary -> tertiary structure

Homology modelling



Protein Docking



What is docking?

- Seeks to predict strongest interactions
- Structures are fixed, conformations variable
- Parameters to customize the process:
  - active site
  - distance constraints

Fast-evolving field

- Computationally expensive operation
- Hardware advances are changing the field
- Increasing algorithm complexity, rising success rates

Protein Docking

Popular tools:

- AutoDock
- HADDOCK
- PatchDock
- RosettaDock
- ZDOCK

Protein Docking

Scoring conformations

- Geometric fit
- Strength of interactions

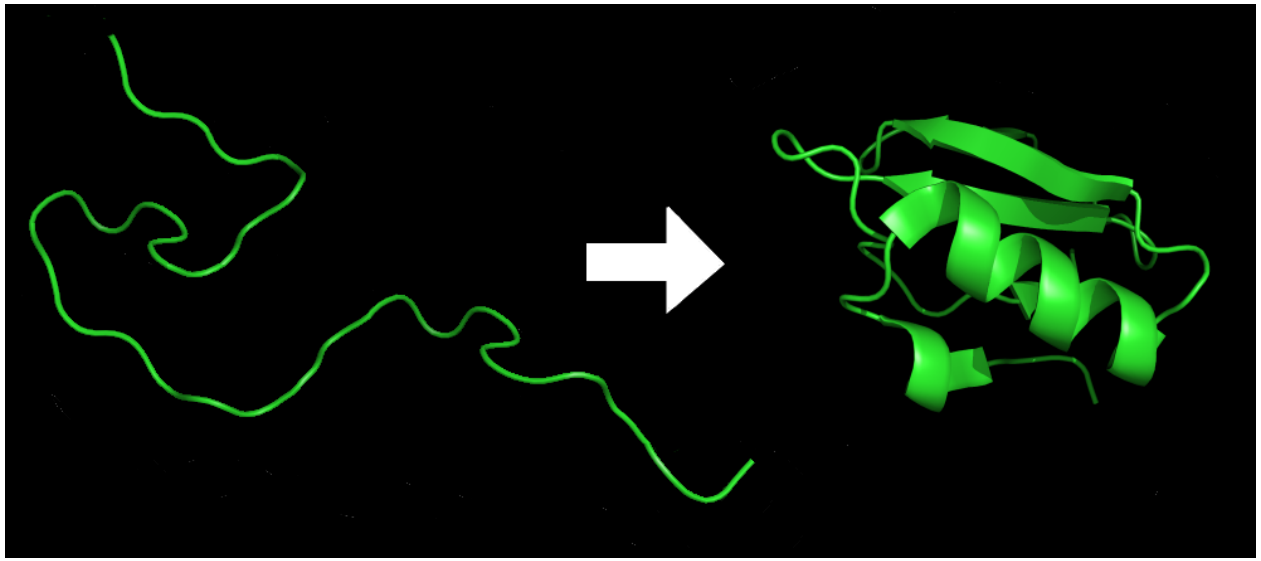
Measuring success

- Take real complex, split up receptor & ligand
- Feed receptor and ligand to docking software
- See how closely results match actual complex

Example

- PDB structure 1ktr
- Antibody-antigen complex

Folds



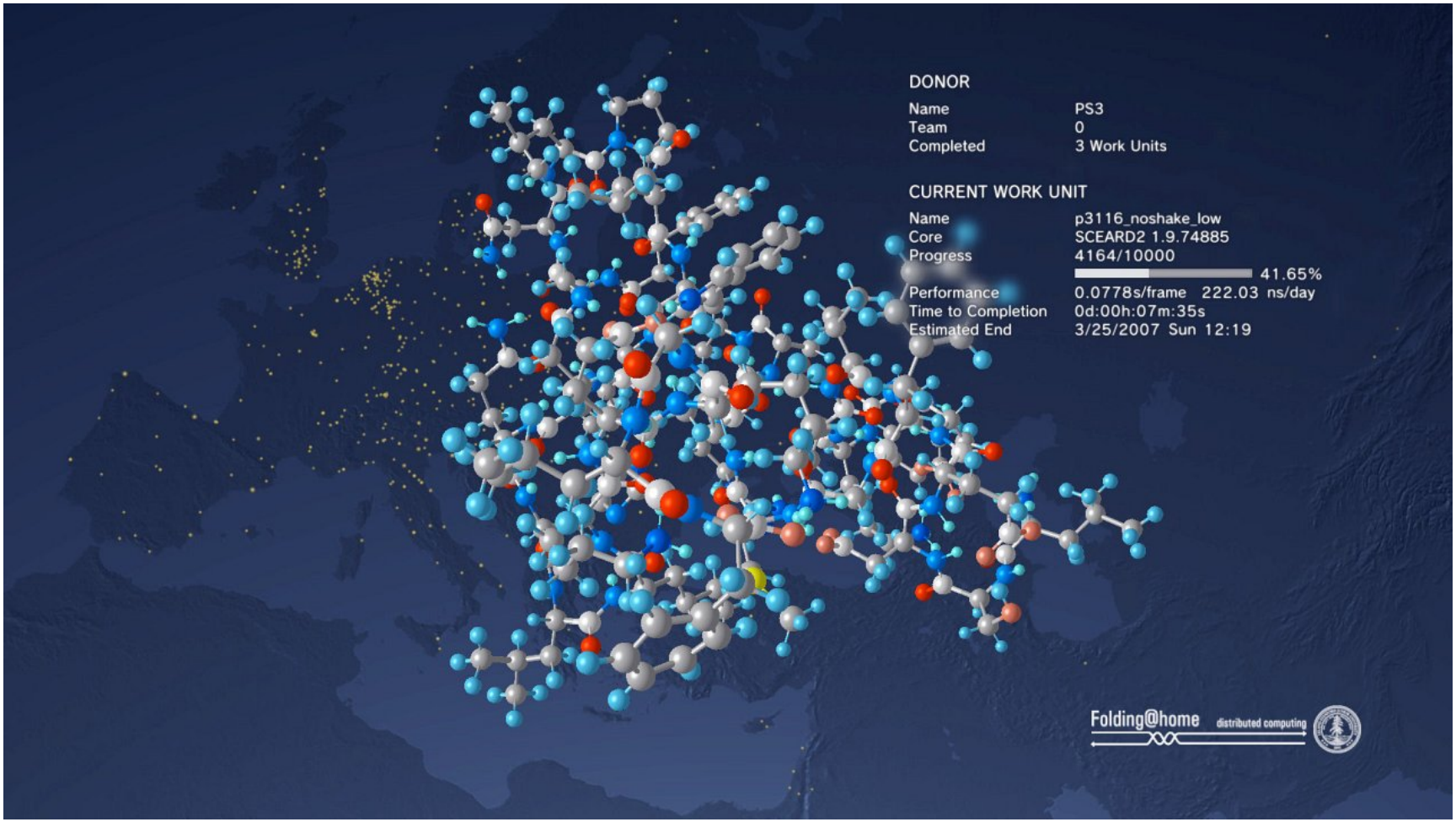
Fold prediction

- Modifying a sequence can introduce radical fold changes
- Field seeks to learn about how folds form, how to predict them

Crowd-sourcing, distributed computing

- Distributed computing: SETI@home
  - 3 million computers run SETI@home
  - Together, they analyze radio telescope data

Folds

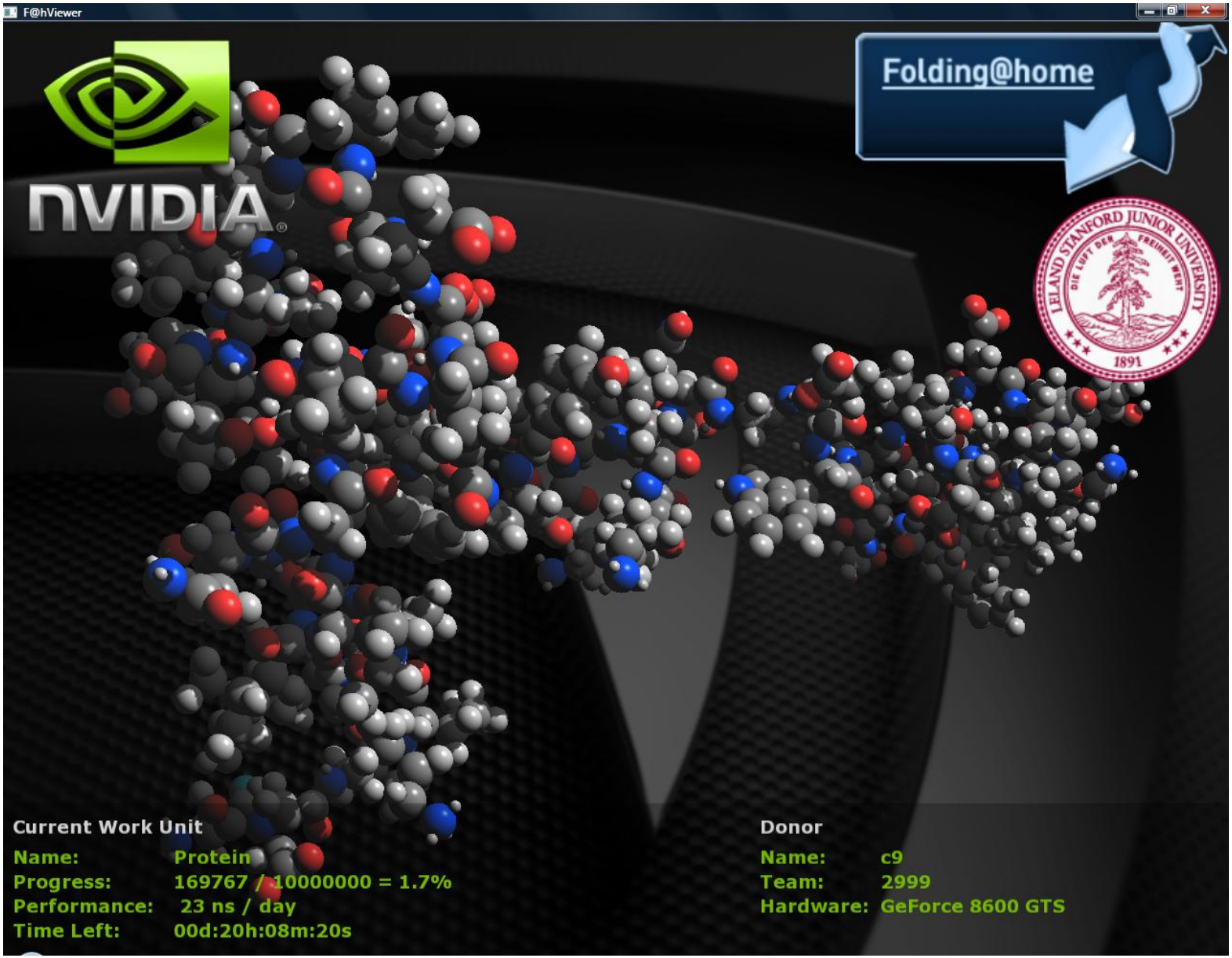

DONOR
Name PS3
Team 0
Completed 3 Work Units
CURRENT WORK UNIT
Name p3116_noshake_low
Core SCEARD2 1.9.74885
Progress 4164/10000
41.65%
Performance 0.0778s/frame 222.03 ns/day
Time to Completion 0d:00h:07m:35s
Estimated End 3/25/2007 Sun 12:19
Folding@home distributed computing

Folds

Folding@Home

- "Help unlock the mysteries of disease"
- Advance Alzheimer's, Huntington's, and cancer research
- Gives visual feedback to current job
- Launched in 2000, has produced 100 papers
- Makes a competition out of "who can donate the most computing power"

Folds


F@hViewer
Folding@home
NVIDIA
LELAND STANFORD JUNIOR UNIVERSITY
1891
Current Work Unit
Name: Protein
Progress: 169767 / 10000000 = 1.7%
Performance: 23 ns / day
Time Left: 00d:20h:08m:20s
Donor
Name: c9
Team: 2999
Hardware: GeForce 8600 GTS

Engineering, drug design

Engineering proteins

- artifical antibodies to combat viruses
- modified viruses to actually help the body

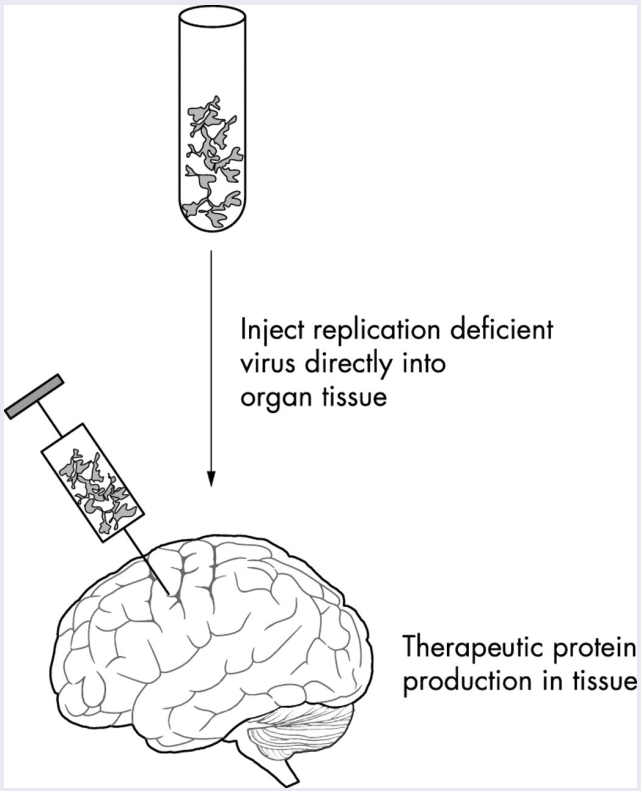
Relation to study of protein folding

- Fold contains the function
- Produce a drug to bind with the functional site, while maintaining a fold that holds its shape.

Engineering, drug design

Gene therapy

- Modified viruses deliver genes to human cells



Hardware

- Powerful, RAM-laden machines
- Impact of distributed computing & super computers

GPU computing

- Folding
- Systems biology

GPU Performance

- Cost per calculation
- Multithreading
- Overclocking
- low-level: C & assembly

Open source

Free vs. libre

- FOSS
- Free "as in beer" and free "as in freedom"
- copyright vs. copyleft
- Free Software Foundation

Well-known libre licenses

- GPL
- MIT
- BSD

Open source

Benefits

- Facilitates collaboration in the community
- Faster rate of scientific progress
- Many eyeballs make for better code
  - Secure, transparent code
  - The Cathedral and the Bazaar

Personal experience

- Biopython
  - Responsive developer community
  - Code affecting your work? Existing code flawed? Patch it.
  - Open source benefits everyone
- Why open source is important to me

My program

Basic properties

- All Python, interacting with third party utilities
- Medium in size and scope (approx. 6,000 lines)
- Only runs on GNU/Linux (limitation of third party tools)
- All components are free, most are libre

Project management

- Version controlled on Git
- Automated testing with `unittest` framework
- Written in Vim

Bioinformatics tools

Common programming languages

- Perl
- Python
- Java
- C, C++, Fortran, & Pascal (to a lesser extent)

Open source tools:

- OBF: Biopython, BioPerl, BioJava, BioSQL
- PyMOL

Commercial tools:

- MATLAB bioinformatics toolkit
- Discovery Studio

Online resources

- Large effort to create standalone servers, databases, etc.
- Massive, large-scale sharing of data, in the hopes of collaborative progress
- Databases of mutations, crystal structures, sequences, etc.

Well-known databases, servers

- BLAST
- SWISS-MODEL
- COSMIC

Closing notes

Importance of data structures & good design

- Time complexity is very, very important
- Big O notation- minutes vs. days
- The right data structure often makes all the difference

Getting involved

- Low barriers to entry
- Many aspects of CS have direct applications
  - machine learning
  - database theory
  - etc.
- Many open source projects need contributors
  - Biopython

Contact information

- davidjosephcain@gmail.com
- github.com/davidcain

Links

- *Links can be found throughout presentation*
- A summary of all used links (plus a few extra) follows.

SMART, Sasisekharan Lab

- SMART- Singapore-MIT Alliance for Technology and Research
- Sasisekharan Lab
  - News/discoveries
  - Important mutation point discovered in H1N1
  - Heparin crisis

Systems biology

- Boids, by Craig Reynolds, 1986

Links

Sequences

- BLAST
- Overview of BLAST
- Protein blast

Docking

- AutoDock
- HADDOCK
- PatchDock
- RosettaDock
- ZDOCK

Secondary structure

- dssp

Links

Sequence alignment tools

- MUSCLE
- Mustang
- Clustal

Biopython

- Biopython
- How to Contribute

Folding

- Folding@Home

Links

Open Source

- Free Software Foundation
- GNU

Online resources

- BLAST
- SWISS-MODEL
- COSMIC

Attributions

Most images used in this presentation are in the public domain. A good faith effort was made to properly attribute images according to their licensing terms.

- H1N1 photo- Jesus Perez
- Ants swarming
- Docking graphic
- Systems biology diagram
- School of mullets
- Protein structure graphic
- Homology model backbone
- Gene therapy in clinical medicine
- RNA codons