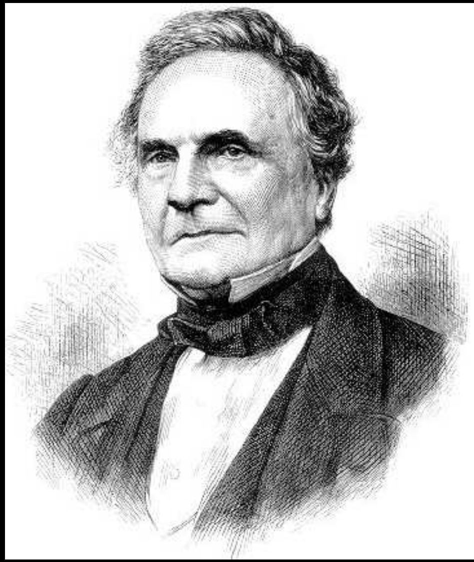
Computational Views of Evolution

Christos H. Papadimitriou
The Simons Institute

An early computational view of evolution

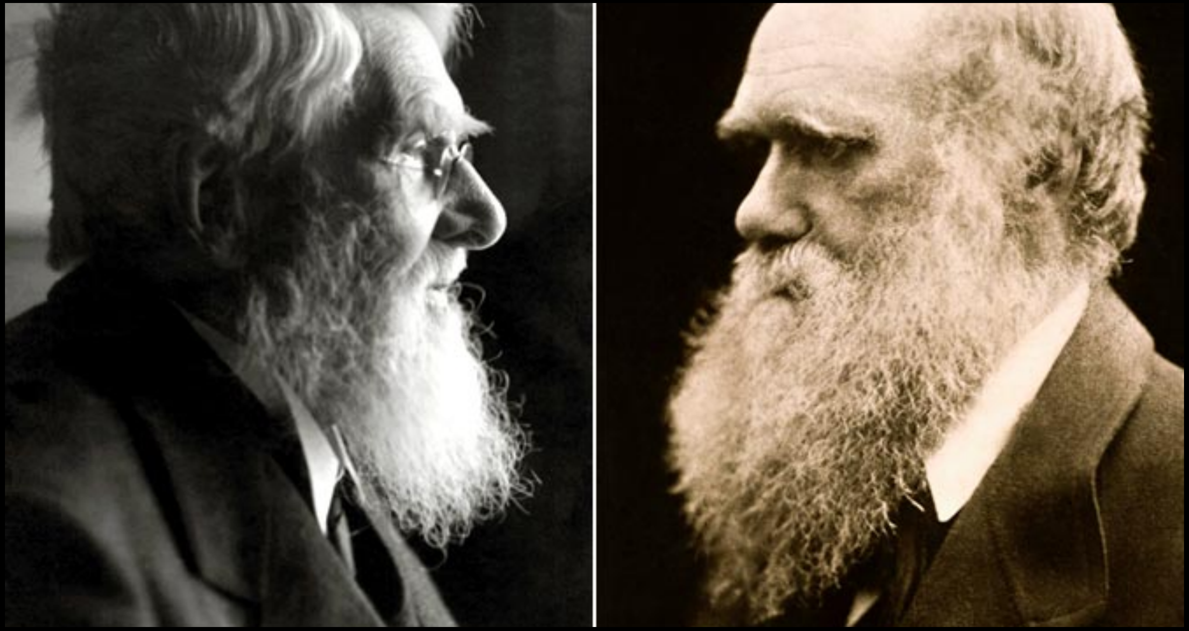
- Charles Babbage

Ninth Bridgewater treatise
ca. 1830 (paraphrased):



The Supreme Being created not species, but the algorithm for creating species

Wallace-Darwin 1858:
Exponential growth
is incompatible with Life



The Origin of Species

- Natural Selection
- Common Ancestry
- Possibly the world's most masterfully compelling scientific argument
- The six editions:1859, 1860, 1861, 1866, 1869, 1872

Cryptography against Lamarck

A. Weismann

[ca. 1880, paraphrased]

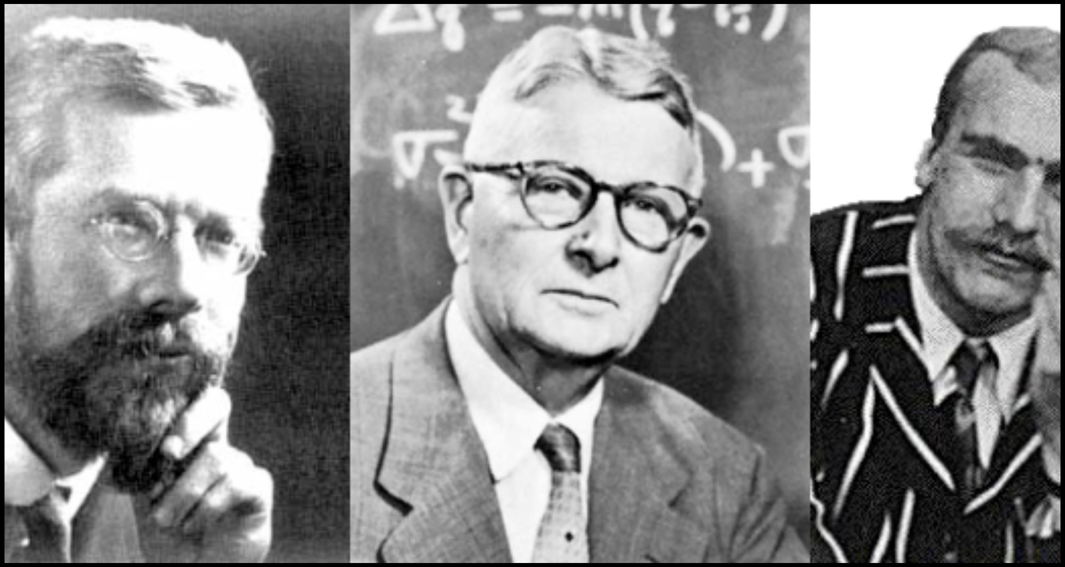
"The mapping from genotype to phenotype is one-way"

Surprise! Inheritance is discrete

- Gregor Mendel [1866]
- Number of citations between 1866 and 1901:

3

The “Modern Synthesis”
1918 - 1940



Fisher – Wright - Haldane

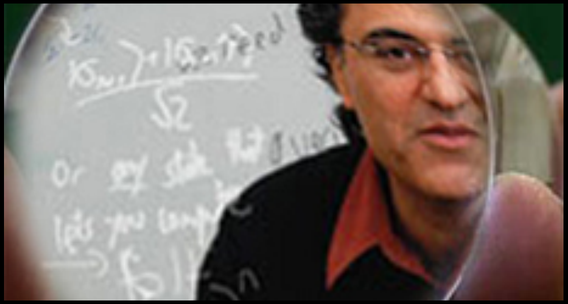
Meanwhile at the farm…
(1929 – 1946)

Gödel - Turing - von Neumann

Theory of Computing
(last six decades)

A mathematical framework, stance, and methodology for understanding the capabilities and limitations of the computer

The Lens of Computation



- When the point of view of the Theory of Computing is applied to a field of science, progress often happens

- E.g.: Statistical physics, quantum mechanics, game theory and economics, social science, molecular biology and evolution

Btw: the special affinity between computation and biology

There is “innate explicit code” in Life

The Theory of Computing, in a nutshell

- Life is hard
- computers can occasionally help

  → algorithms
- other times, they can't

  → complexity

Algorithms

- Computational problem:
- An infinity of inputs, each seeking an output
- The output must be in a particular relation to the input
- Inputs and outputs are strings of bits
- Graphs, matrices, etc. can be so represented

Algorithms (cont.)

- Algorithm A for computational problem C
- Must be correct (= eventually stop with the right output for each input of C)
- $T_{A,C}$ (n) = the number of elementary steps A takes until completion, when supplied with an input of length n, maximized over all inputs of length n ("worst-case analysis")

Examples of computational problems

- Linear programming
- Shortest path from s to t in a graph
- Traveling salesman problem
- Integer programming
- Sequence alignment
- Sequence centroid

Sequence alignment (or edit distance)

- Input: two sequences ACGGTGT… and CTAGTAA… and parameter d
- Output sought: An alignment with at most d skips/overwrites
- There is an algorithm A with $T_{A,C}(n) = O(n^2)$
- (When d is small, can be solved in linear time *cf.* BLAST)

Sequence centroid

- Given s sequences ACC…, GCC…, ACT… etc. and a parameter d
- Output sought: A new sequence AGC… which has edit distance $\leq$ d from each
- Can be found in time $T_{A,C}(n) = O(2^n)$
- Fact: all algorithms known for this problem require exponential time

Is exhaustive search ever necessary?

- NP: all search problems
- P: all search problems solvable in polynomial time (e.g., sequence alignment)
- Conjecture 1: $P \neq NP$
- Conjecture 2: Sequence centroid is not in P
- Fact: These two conjectures are equivalent
- Sequence centroid is NP-complete

Sooooo, the Theory of Computing

- A comprehensive methodology for dealing with computational problems
- Develop efficient algorithms for them
- Or establish complexity lower bounds, such as NP-completeness
- Plus more complex strategies, such as approximations and heuristics

Life algorithms (and complexity)

- Protein folding and the Levinthal paradox
- The H-P model [Ken Dill, ca 1990]
- PHPPHPHPPHPHP: fold it!

```
H – P
|   |
P – P – H
|       |
P – H – P — P
|       |   |
P – H – P – H
```

score = 4

Trouble in Life…

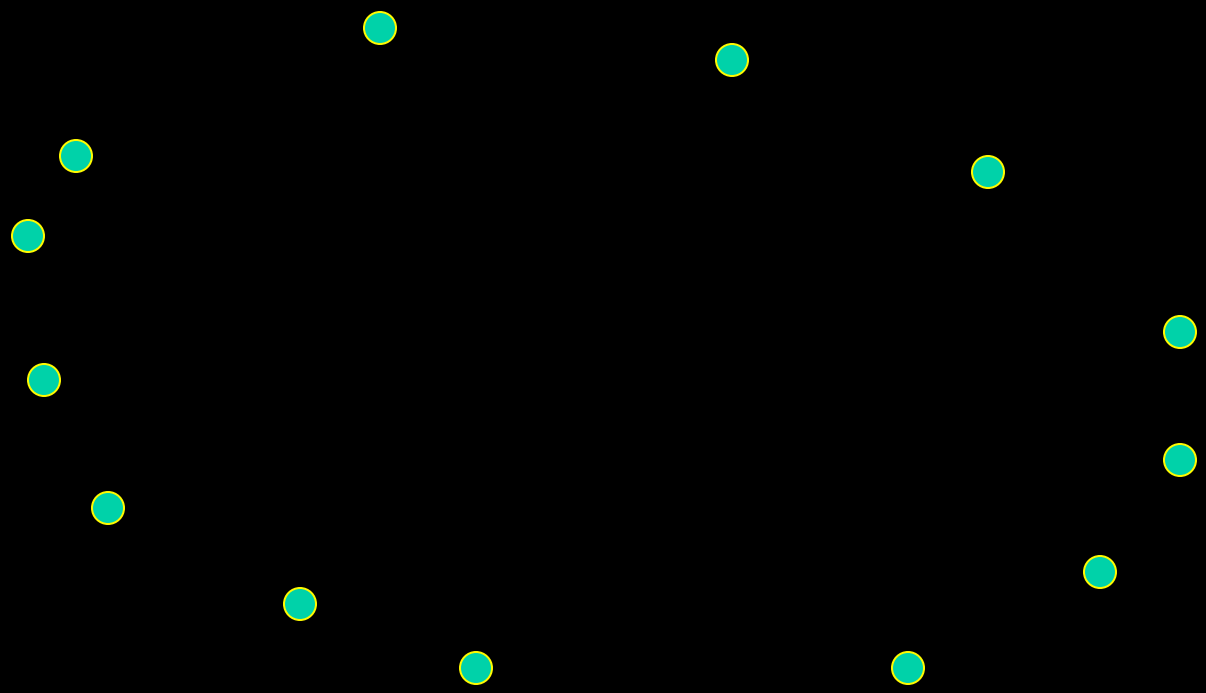
Theorem [CGPPY98, BL98]: The HP folding problem is NP-complete

- Levinthal's paradox sharpened
- Remember: exponentials incompatible with Life
- Is the real problem simpler than the HP cartoon? (hard to believe…)

Or could it be that…

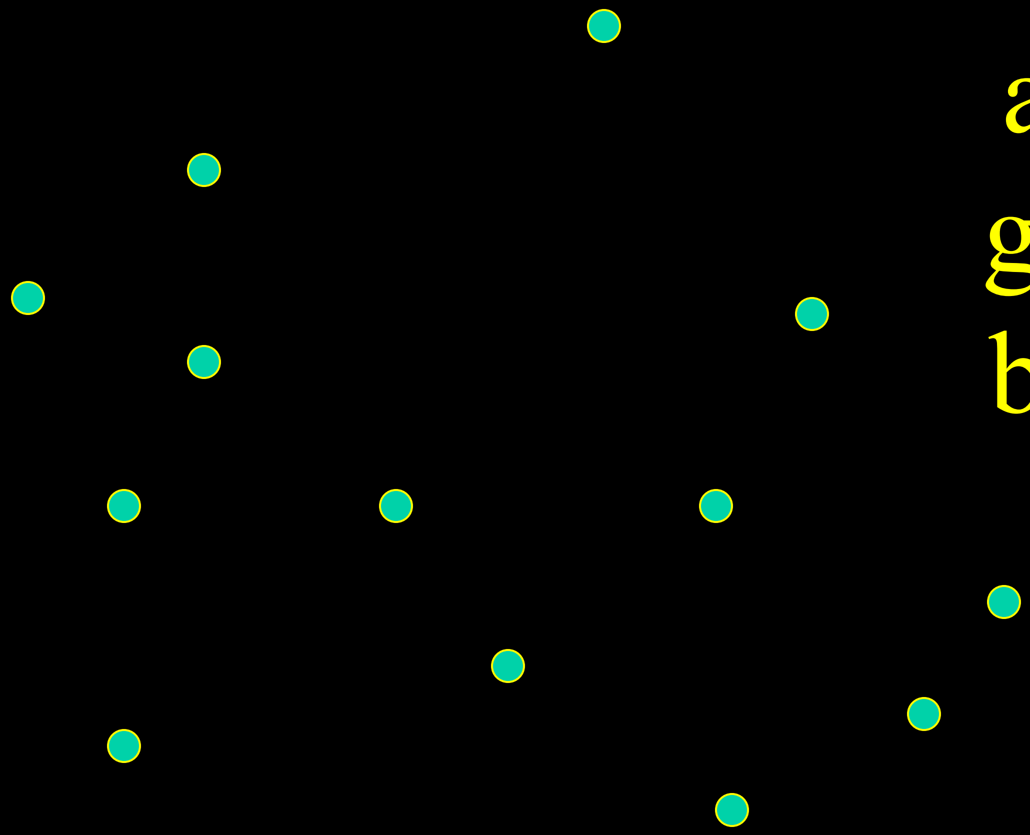
- …proteins have been selected so that they fold easily?
- Remember worst case: even the hardest problems have easy inputs

e.g., the traveling salesman problem

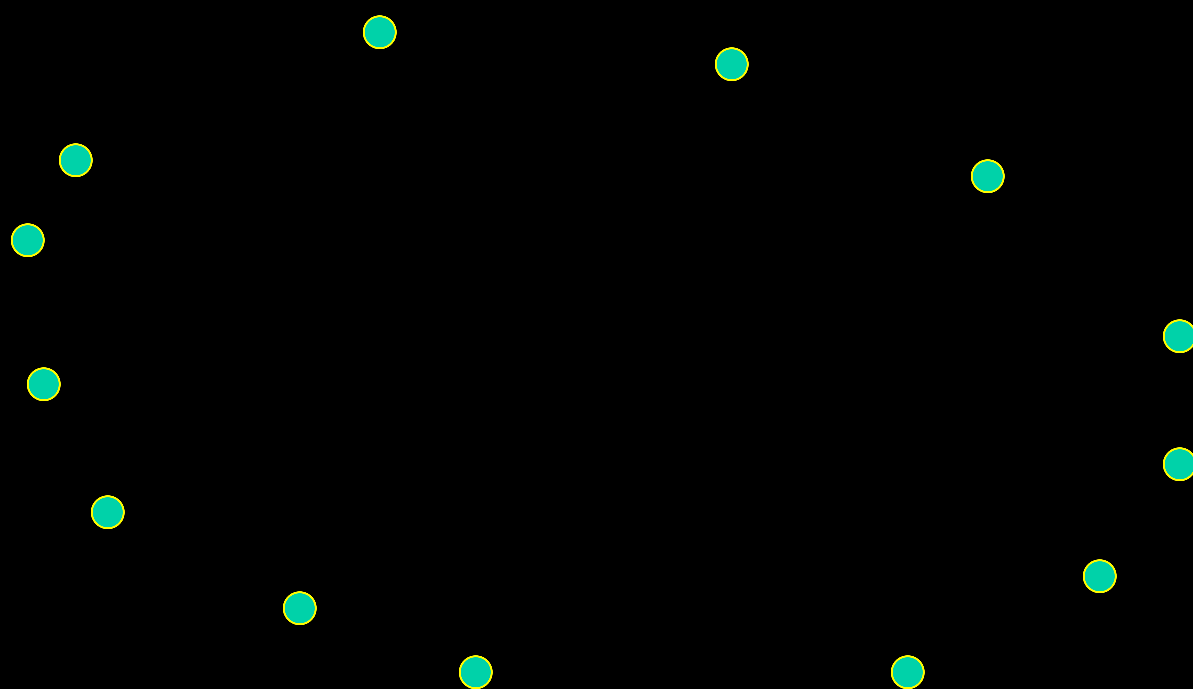


Or could it be that…

- …proteins in real organisms have been selected so that they fold easily?
- Remember worst case: even the hardest problems have easy inputs
- Life is hard, but natural selection can favor easy inputs…
- [CHP, Sideri 1999] experiments with the traveling salesman problem: evolve a population of TSP inputs, fitness = "ease"



after a few generations becomes…



Online algorithms and the experts problem

- Every day you must choose one of n experts
- The advice of expert i on day t results in a gain G[i, t] in [-1, 1]
- Challenge: Do as well as the best expert *in retrospect*
- Surprise: It can be done!
- Hannan 1958, T. Cover 1991, Winnow, Boosting, no-regret learning, MWUA, …

Multiplicative weights update

- Initially, assign all experts same weight/ probability
- (Or think of the distribution on the experts as a stock portfolio)
- At each step, increase the weight of each by $(1 + \varepsilon\, G[i, t])$ (and then normalize)
- **Theorem**: Does as well as the best expert

Intuition

- Each day t, p_i becomes
  $p_i(1+\varepsilon G[i, t]) \approx p_i \exp(\varepsilon G[i, t])$
- After many days, $p_i \approx \exp(\varepsilon \Sigma_t G[i, t])$
- The protfolio will consist almost exclusively of the best performing stock – in hindsight.
- (Unless there are near ties, in which case we do not care much…)

There is more…

The same algorithm solves zero-sum games, linear and convex programming, network congestion,…

Computer scientists find it hard to believe that such a crude technique solves all these sophisticated problems

Heuristics inspired by Evolution

- Local search [Croes 58, Bock 58]
- [Dunham, Fridshal, Fridshal, North 61]
  "Design by natural selection"
- Simulated annealing [Kirkpatrick et al. 83]
- "Go with the winners" [Aldous-Vazirani 93]
- Tabu search [Glover 84]
- ….

Genetic algorithms

- Maintain a population of solutions
- Encoded as some kind of genotype
- Fitness = goodness as a solution
- Next generation created by mutations and (uaually) recombination
- Influentially proposed by [Holland 80]

More…

- Evolutionary strategies
- Evolutionary programming
- Genetic programming
- Differential evolution
- …and not to mention ant colony algorithms, bee hive algorithms, cuckoo algorithms,…
- Artificial life (e.g. Avida)

Rough classification of evolution-inspired heuristics

- **Simulated annealing:** variants of the local search algorithm, one solution or very few solutions maintained, mutation but no recombination → asexual evolution
- **Genetic algorithms:** population of solutions maintained, genetic encoding, new generation produced through mutation plus recombination → sexual evolution

Comparison

- Genetic algorithms encoding is very hard to do right – must reflect latent modularity in the solution space
- Not many practical successes known
- In contrast, simulated annealing heuristics are often the best known algorithms for certain applications

Back to Evolution: it is full of fascinating problems

- The role of sex
- The maintenance of variation
- The emergence of novelty
- …among many others

(Remembering G. H. Hardy, 1908:

"I am reluctant to intrude in a discussion concerning matters on which I have no expert knowledge")

The role of sex

- Sex is ubiquitous in Life
- Despite its multifaceted costs

[Barton and Charlesworth "Why sex and recombination?", 1998]

- Which makes the apparent advantage of simulated annealing (asexual evolution) over genetic algoorithms (sexual evolution) hard to explain…

A Radical Thought

- ***What if sex is a mediocre optimizer of fitness?***

[A. Livnat, J. Doushoff, P., M. Feldman, *PNAS* 2008]

Selection at two loci

- Fitness landscape of a 2-gene organism

3	2	4	5	4
1	0	0	7	2
2	1	0	4	3
1	8	1	3	2

Rows: alleles of gene A

Entries: fitness of the combination

Columns: alleles of gene B

Asexual evolution

- Asex will select the largest numbers

3	2	4	5	4
1	0	0	7	2
2	1	0	4	3
1	8	1	3	2

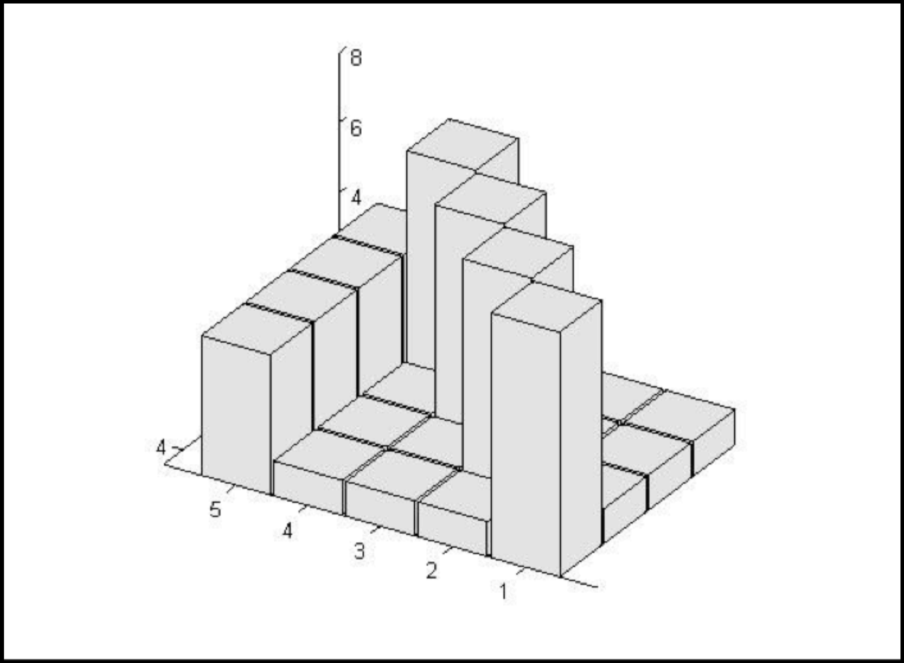
Mixability

- But sex favors the alleles that perform well with many genetic partners

3	2	4	5	4
1	0	0	7	4
2	1	0	4	3
1	8	1	3	2

In pictures

[Livnat, P., Feldman
J. Th. Bio 2011]



Unless
peaks > 2×plateau
the plateau
will prevail under sex

Weak selection

1.03	1.02	1.04	.97	1.01
1.01	.96	1.03	1.03	1.02
1.02	1.02	1.01	1.04	1.03
.99	.98	1.04	1.03	1.02

$$w_{ij} = 1 + s\,\Delta_{ij}$$

with $s << 1$

Linkage equilibrium [Nagylaki 1993]

Under weak selection, $p_{ij} = x_i\, y_j + o(s^2)$

(after log n generations)

where $x_i = \Sigma_j\, p_{ij}$ and $y_j = \Sigma_i\, p_{ij}$

The Fisher-Wright equations become

$$x_i^{t+1} = x_i^t\,(1 + s\,\Sigma_j\, y_j^t\, \Delta_{ij})$$

Remember multiplicative updates?

Under weak selection, evolution becomes a *game*

- The players = the loci
- The strategies = the alleles
- The common utility = the organism's fitness

    (*coordination game*)
- ***The players play by MWUA***

[E. Chastain, A. Livnat, P., U. Vazirani, 2013]

Reinterpret as an online optimization problem

At each generation, each locus maximizes

the cumulative expected fitness of the organism over all previous generations

+

(1/s) times the *entropy* of the alleles' distribution

Changing the subject: Pointer Dogs

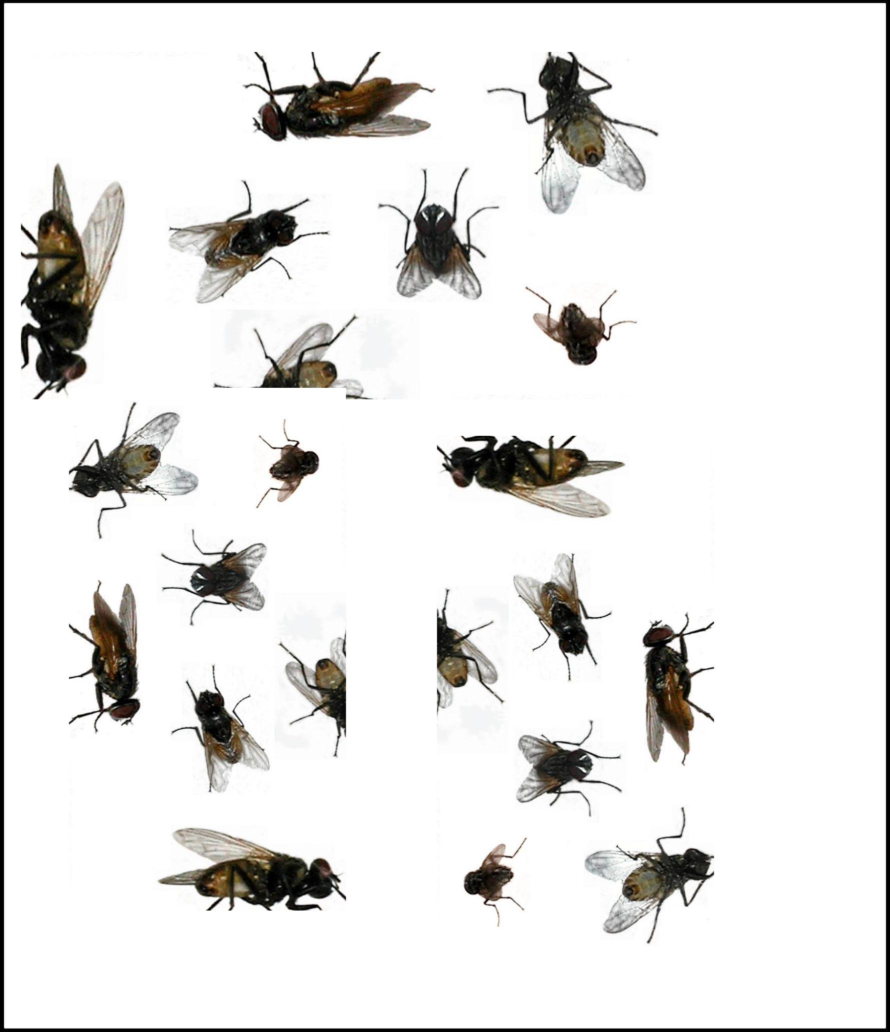
Pointer Dogs

C. H. Waddington

Waddington's Experiment (1952)

Generation 1

Temp: 20^o C



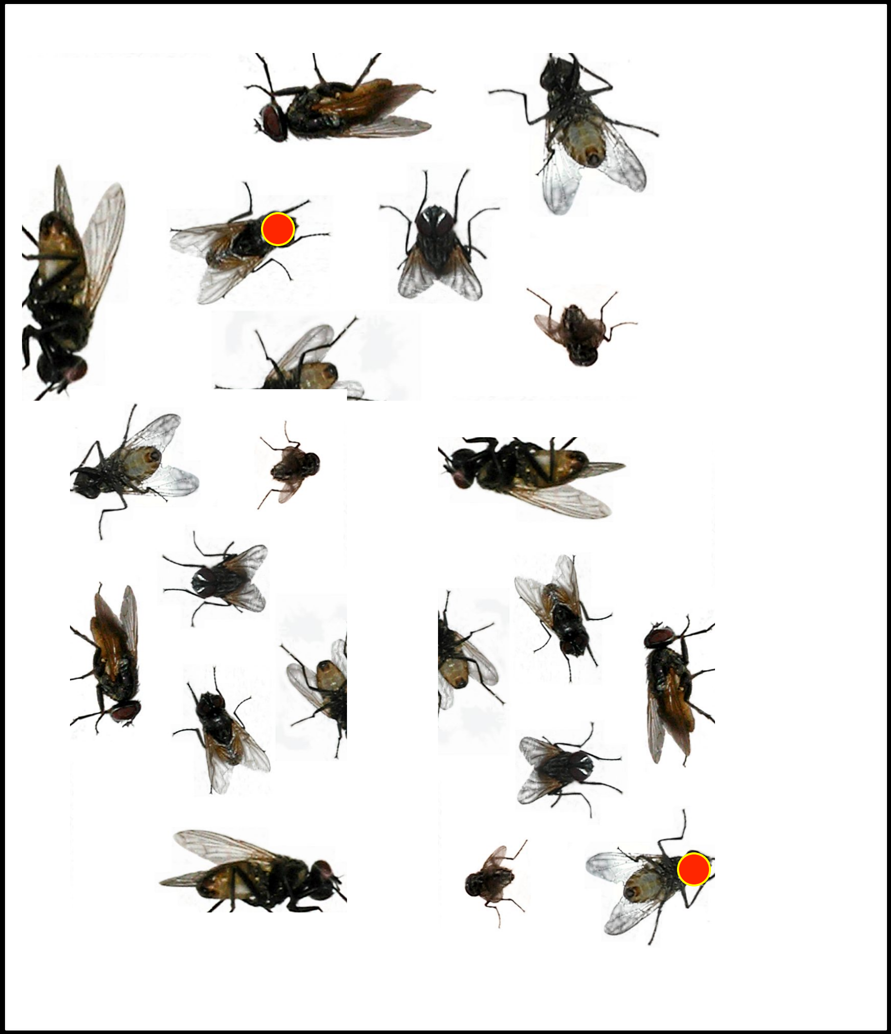
Waddington's Experiment (1952)

Generation 2-4

Temp: 40° C

~15% changed

Select and breed those



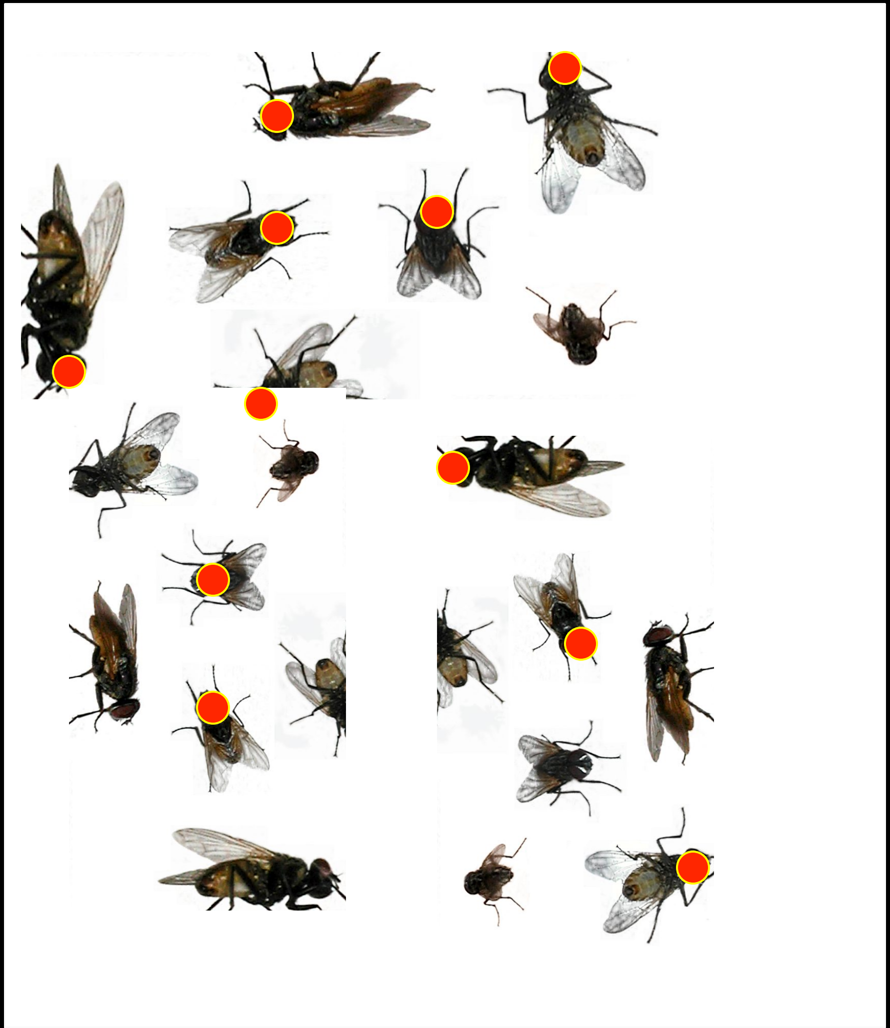
Waddington's Experiment (1952)

Generation 5

Temp: 40^{o} C

~60% changed

Select and breed those



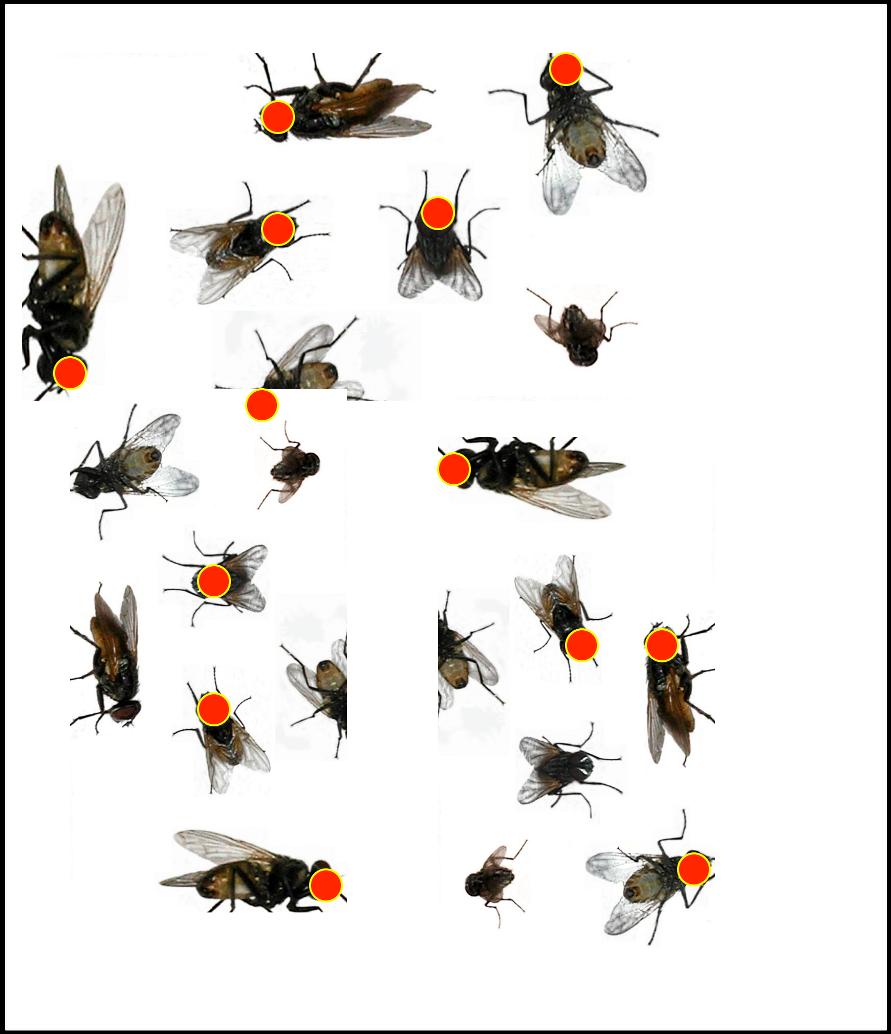
Waddington's Experiment (1952)

Generation 6

Temp: 40^{o} C

~63% changed

Select and breed those



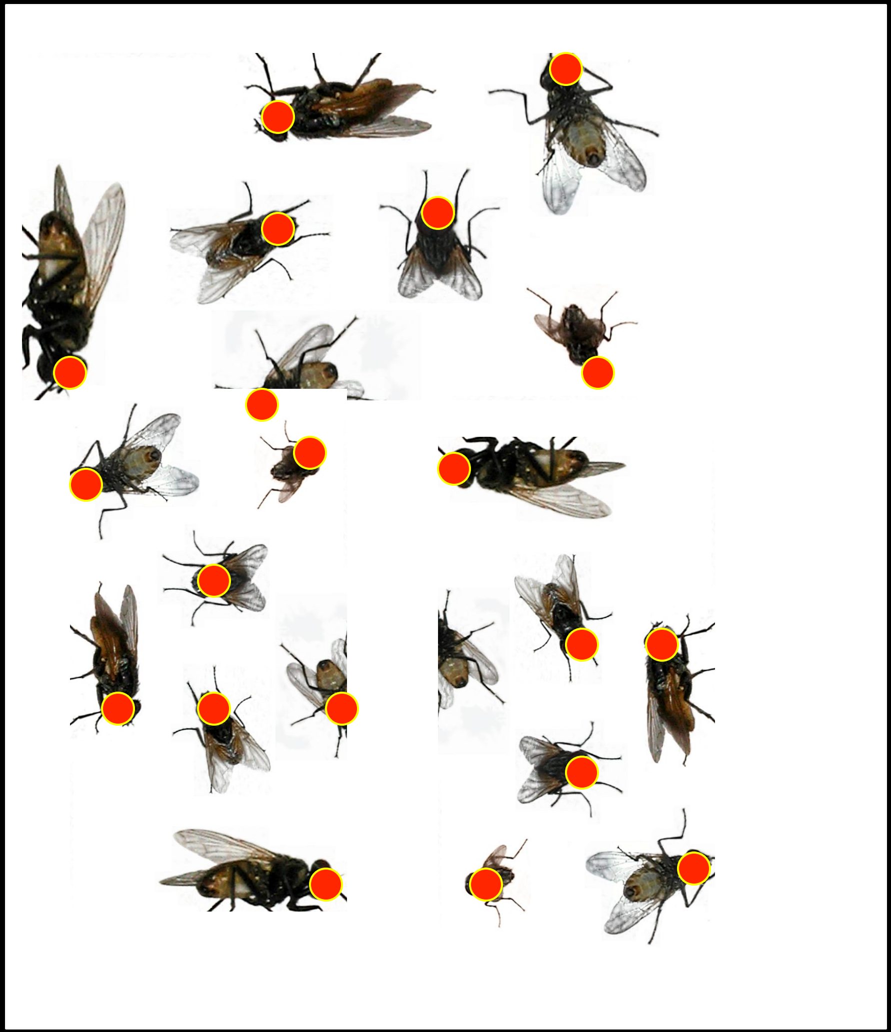
Waddington's Experiment (1952)

(…)

Generation 20

Temp: 40^{o} C

~99% changed

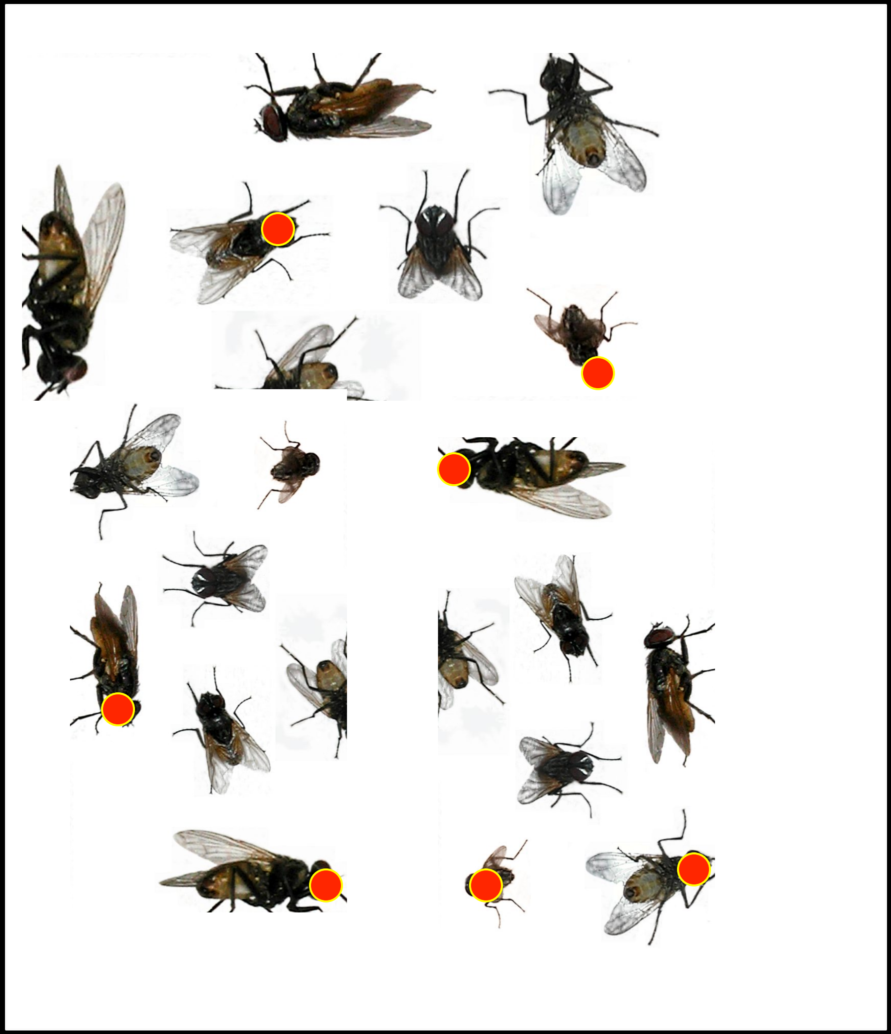


Surprise!

Generation 20

Temp: 20^{o} C

~25% stay changed!!



- Adapt ome
  geneti

Is There a Genetic Explanation?

Function f (x, h) with these properties:

- Initially, $\text{Prob}_{x \sim p[0]}\,[f(x, h = 0)] \approx 0\%$
- Then $\text{Prob}_{p[0]}[f(x, 1)] \approx 15\%$
- After breeding $\text{Prob}_{p[1]}[f(x, 1)] \approx 60\%$
- Successive breedings, $\text{Prob}_{p[20]}[f(x,1)] \approx 99\%$
- Finally, $\text{Prob}_{p[20]}[f(x, 0)] \approx 25\%$

A Genetic Explanation

- Suppose that "red head" is this Boolean function of 10 genes and "high temperature"

"red head" = "$x_1 + x_2 + \ldots + x_{10} + 3h \geq 10$"

- Suppose also that the genes are independent random variables, with p_i initially half, say
- All properties of the Waddington experiment satisfied
- [Stern *AN* 1958]

Arbitrary Boolean Functions

- What if we have an arbitrary Boolean function of genes (no environmental variable h)
- Suppose the satisfying genotypes have a fitness advantage ($1 + \varepsilon$ *vs.* 1, say)
- Will this trait be fixed eventually?

Arbitrary Functions: ***Yes!***

Theorem: Any Boolean function of genes which confers an $1 + \varepsilon$ selection advantage will be fixed (with high probability within poly generations and with poly population).

[2014; with Adi Livnat, Aviad Rubinstein, Greg Valiant, Andrew Won]

"Look, Ma, no mutations!"

Emergence of a trait in the whole population, without Fisherian propagation,

through the manipulation by selection of the allelic frequencies

Soooooo…

- Fascinating field, exquisite problems
- Computational insights appear to be reasonably productive
- Analytical proof of the mixability principle?
- Is implicit entropy maximization a more general phenomenon in evolution?


Thank You!