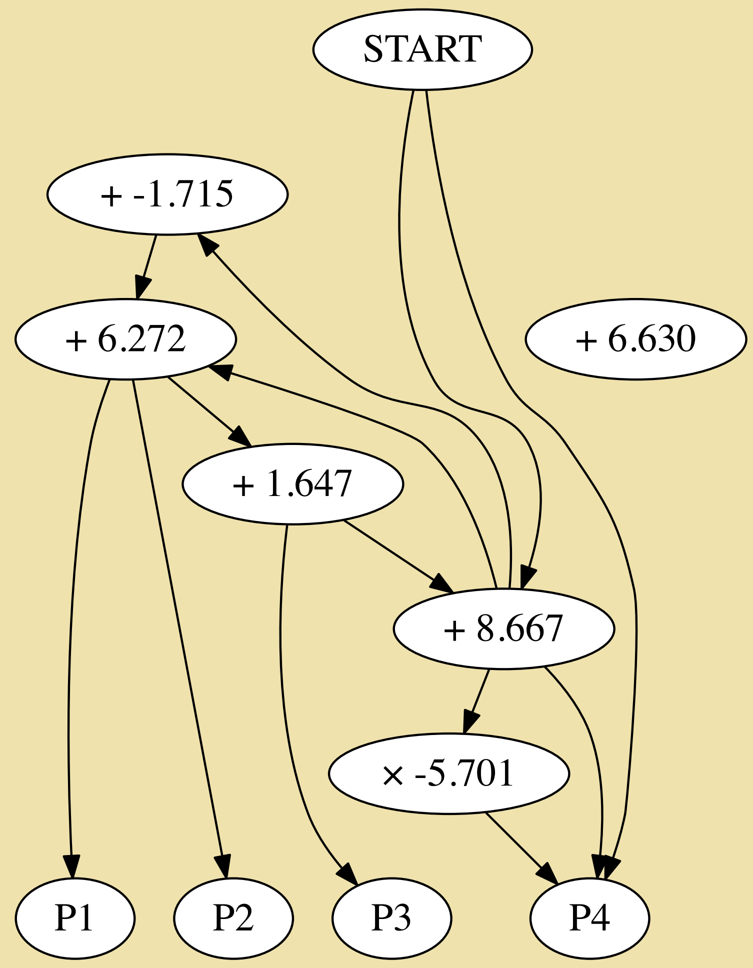
Experiments with Cascading Design



Ben Kovitz
Fluid Analogies Research Group
Indiana University

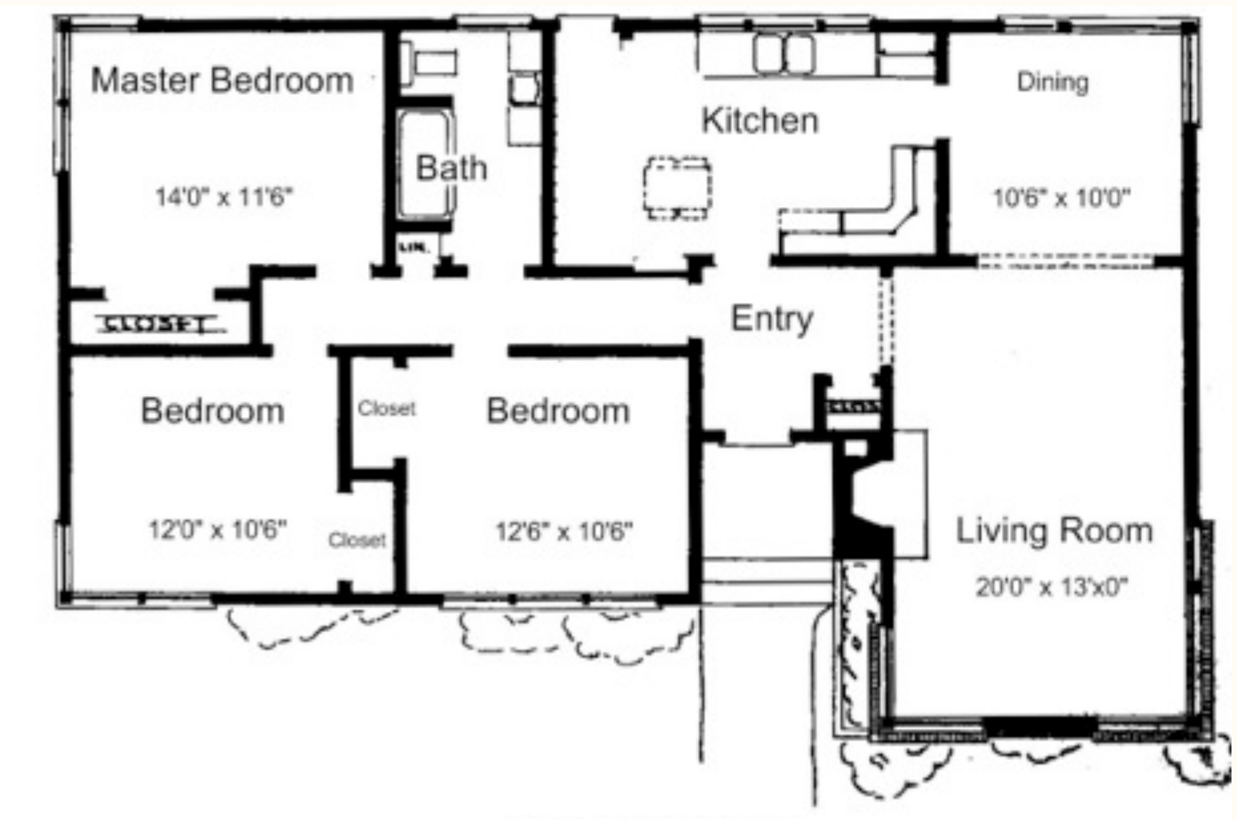
The coordination problem

An old creationist argument:

> The more refined the design, the harder to improve. Therefore evolution should slow down as complexity and refinement increase.

"Ripple" in engineering:

> Whenever you change one part of a design, you must change other parts to maintain integrity. The more complex the design, the more ripple.
>
> So, in an evolutionary algorithm, improvement usually requires multiple, simultaneous beneficial mutations.



Friday, July 24, 15

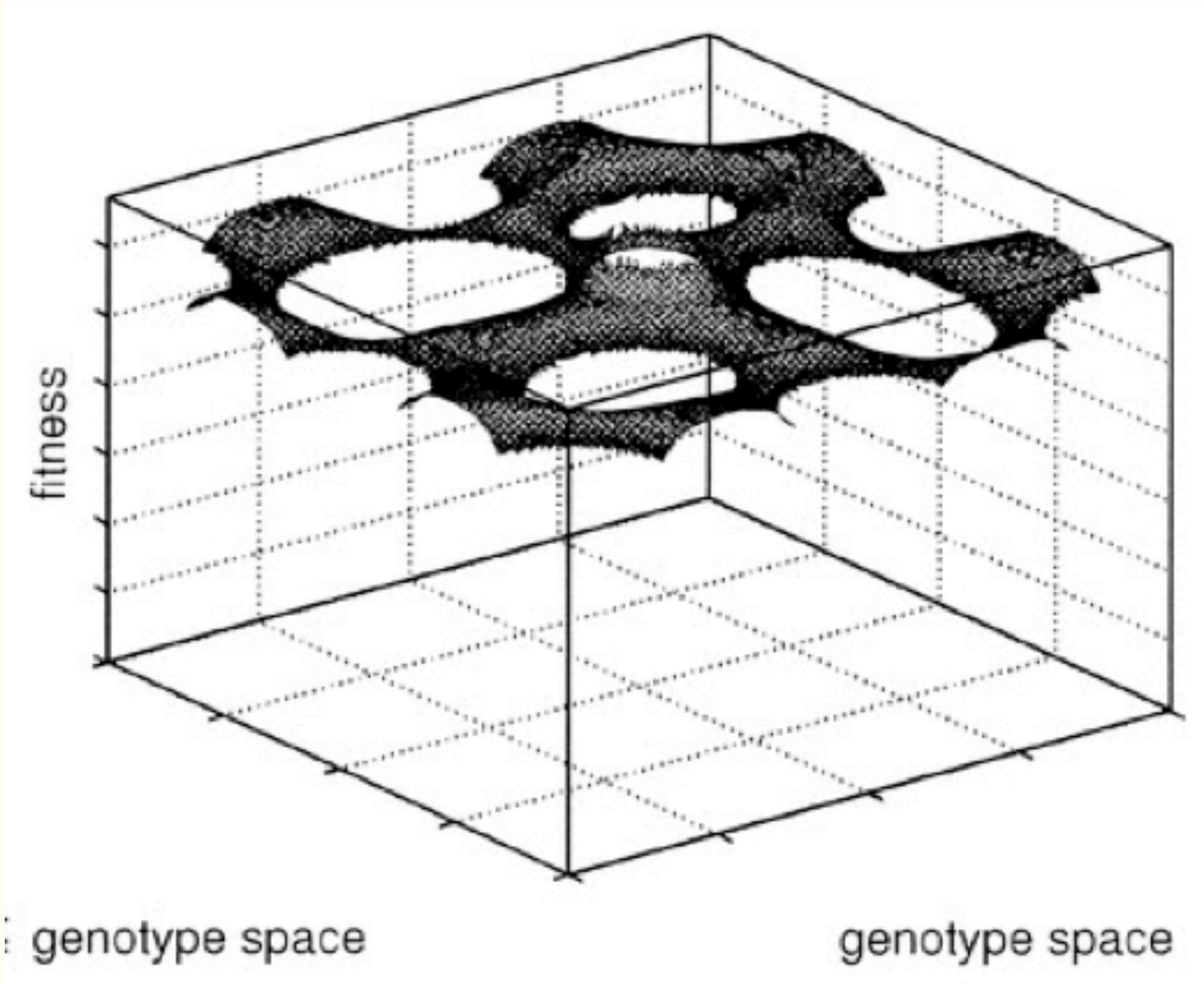
Everyone in software engineering "knows" that it becomes progressively more expensive to add features to a software system as the system grows and matures with real use.

The coordination problem

The requirement for coordination creates "holey landscapes".

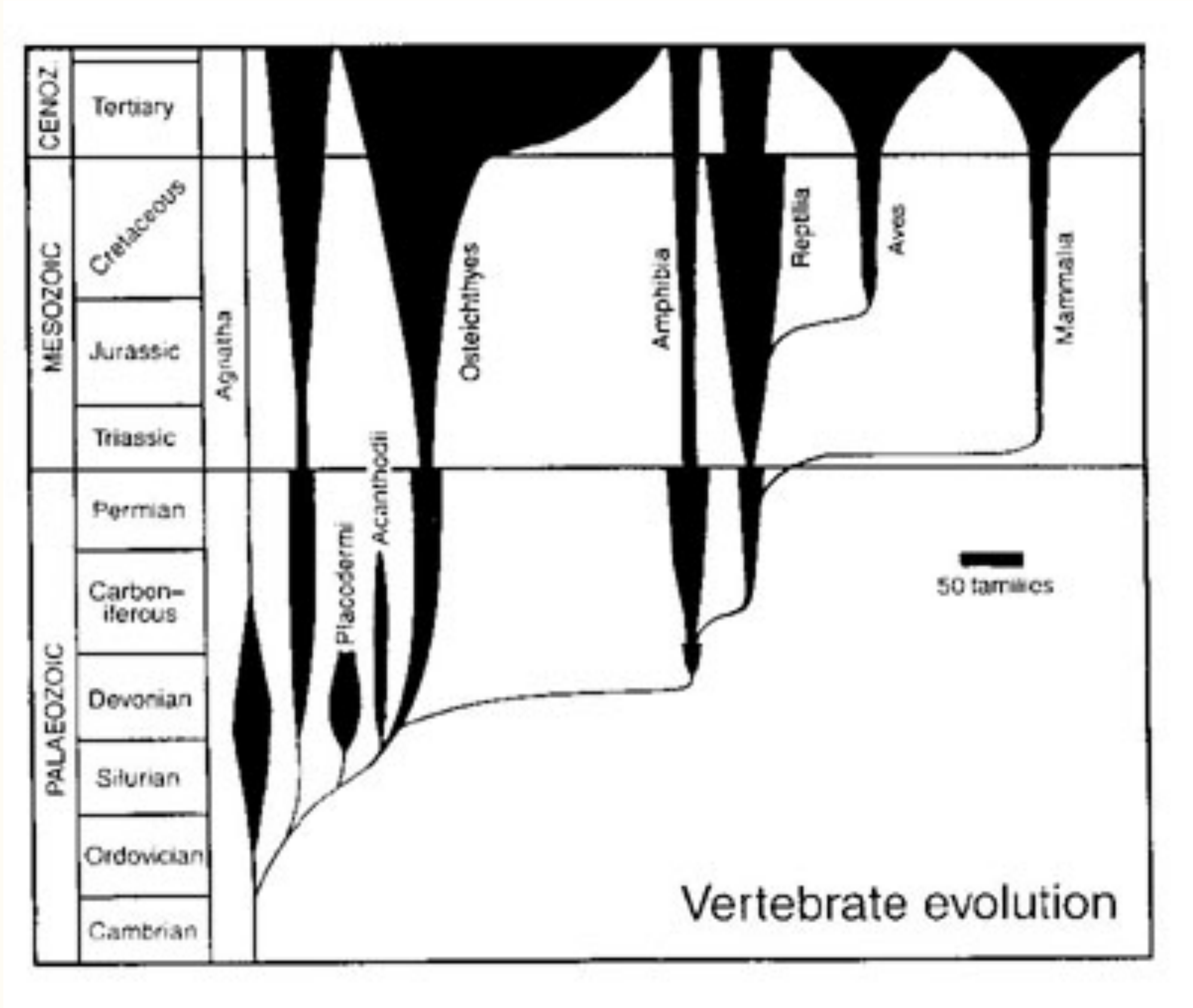
Most small changes are lethal.

Only big changes are good.



Gavrilets, Sergey. "Models of speciation: what have we learned in 40 years?" *Evolution* 57.10 (2003): 2197-2215.

The coordination problem



Benton, M. J. (1998). The quality of the fossil record of the vertebrates. *The adequacy of the fossil record.* Wiley, Chichester, UK, 269–303.

The fossil record suggests that natural evolution has found a solution to the coordination problem.

Time axis goes up. K-T boundary: 66 Mya. Permian-Triassic extinction: 252 Mya. Cambrian explosion: 542 Mya. Width of line is number of distinct families. Kingdom-Phylum-Class-Order-Family-Genus-Species. This is not conclusive, but it suggests that at least in some cases, greater complexity opens up more opportunities for radical changes. The mutation rate is basically constant—"molecular clock", Vince Sarich.

The coordination problem

“Cuban man dubbed ‘Twenty-Four’ proud of his four extra digits.” *The Blaze,* August 21, 2011

This guy might have a clue about why.

There’s a single allele on the human genome where a tiny mutation causes this. The allele does not contain all the information needed to build arteries, veins, capillaries, nerves, skin cells, bone, etc.

Cascading design

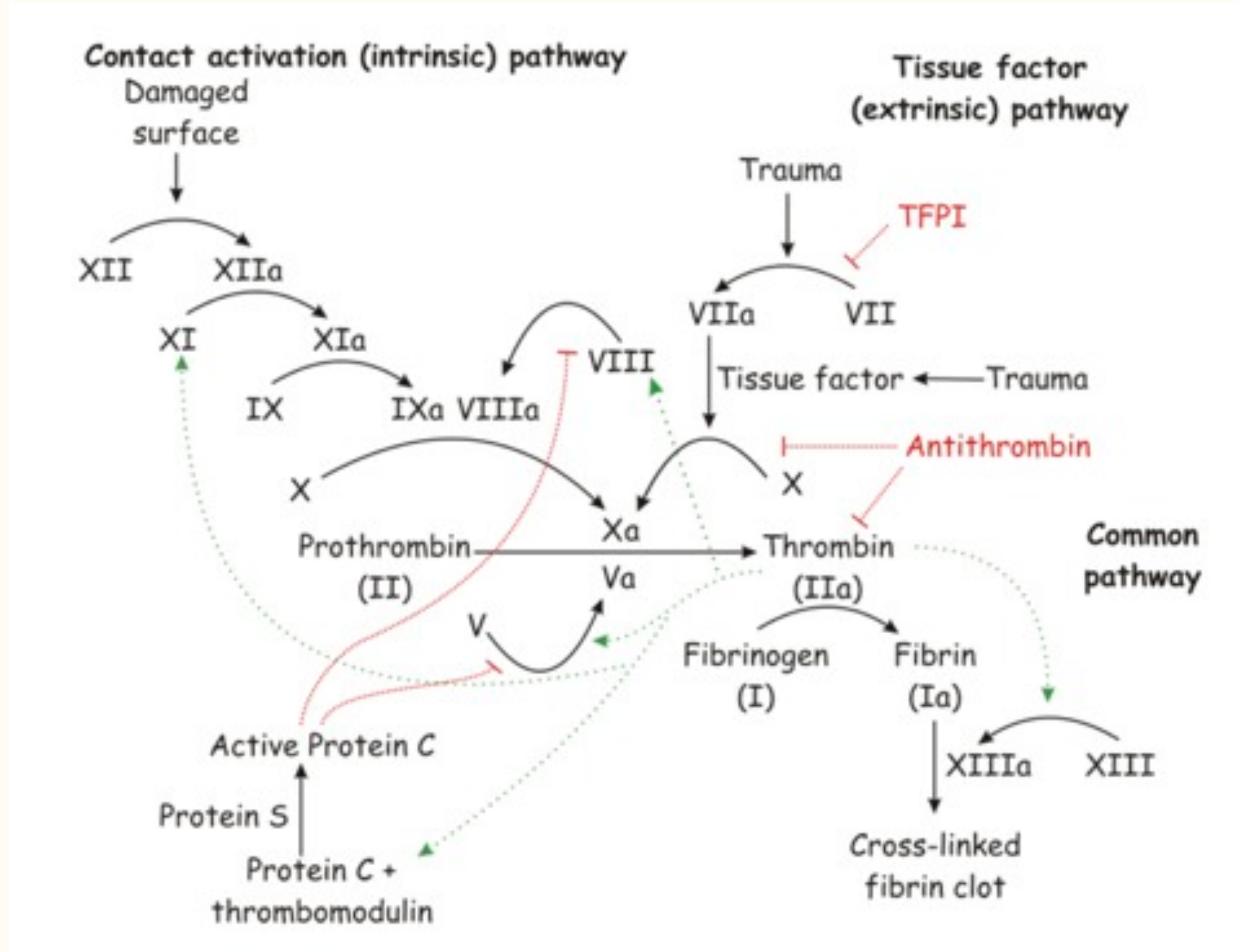
A **cascading design** is one in which each element does very little other than activate other elements.

Metabolic networks

Genetic regulatory networks

Neural networks

Object-oriented programs



Fibrin clotting pathway (simplified)

http://chemwiki.ucdavis.edu/User:Delmar/Blood_Clotting_Cascades

Market economies? Minds? Internet?

Friday, July 24, 15

Cascading design roughly means a directed graph where the edges are some sort of simple causal influence.

Make use of things that are not you. In market economies, you try to exploit comparative advantage as much as you can, and keep your own job simple and efficient. In minds, we constantly reuse and adapt. The Internet is a more-questionable example: link rather than try to provide all information yourself. The story of Hotmail's quick launch illustrates at least that the Internet exploits cascading design's ability to evolve fast.

Questions

1. Is cascading design enough to solve the coordination problem?

    Will a small mutation tend to produce a large, coordinated change in the phenotype?

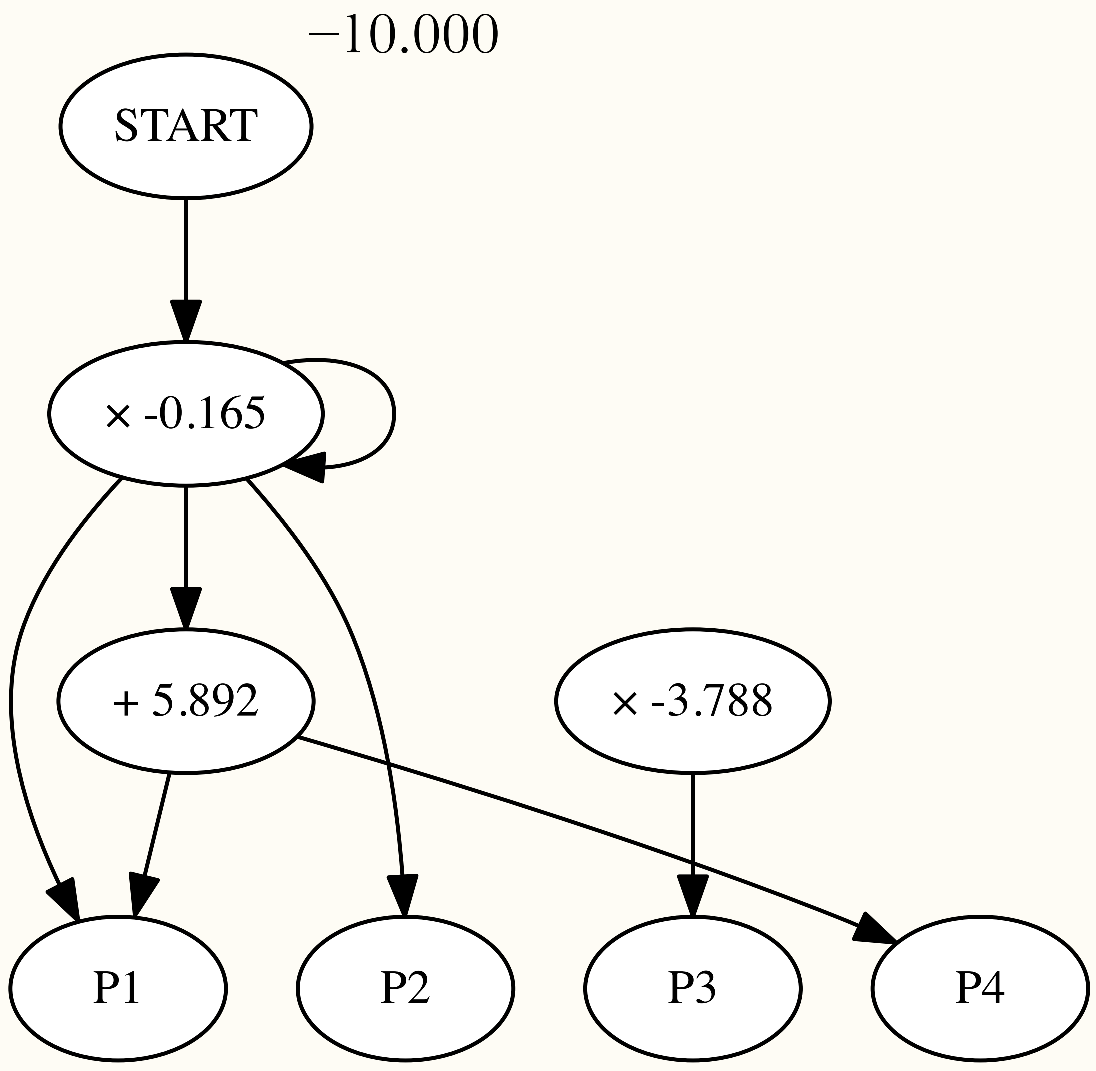
2. Do cascading designs become progressively more evolvable?

    Without "bolting on" extras like genes to alter mutation rate.

If selective pressure tends to make small mutations produce coordinated changes, then the expected value of single mutations should tend to increase.

Experimental approach

Genotype



Phenotype

10.570	1.650	*none*	7.542

I've run many experiments to test this hypothesis. In the rest of this talk, I'll describe two of the more interesting ones.

Compare the simplest meaningful cascading design against direct evolution of the phenotype.

Genotype: Directed graph of arithmetic operations that feed into the phenotype vector.

Phenotype: Vector of four real numbers

Mutations that alter only a number are much more common. Number mutations are limited to a small range.

Similar to Cartesian Genetic Programming, except nodes can be introduced and deleted by mutations. Unlike CGP, this is not intended to generate practical results. This is just a way to see the effect of cascading design, in a simple model of the common denominator of many kinds of cascading design: metabolic networks, OO code, etc.

Experimental approach

Fitness

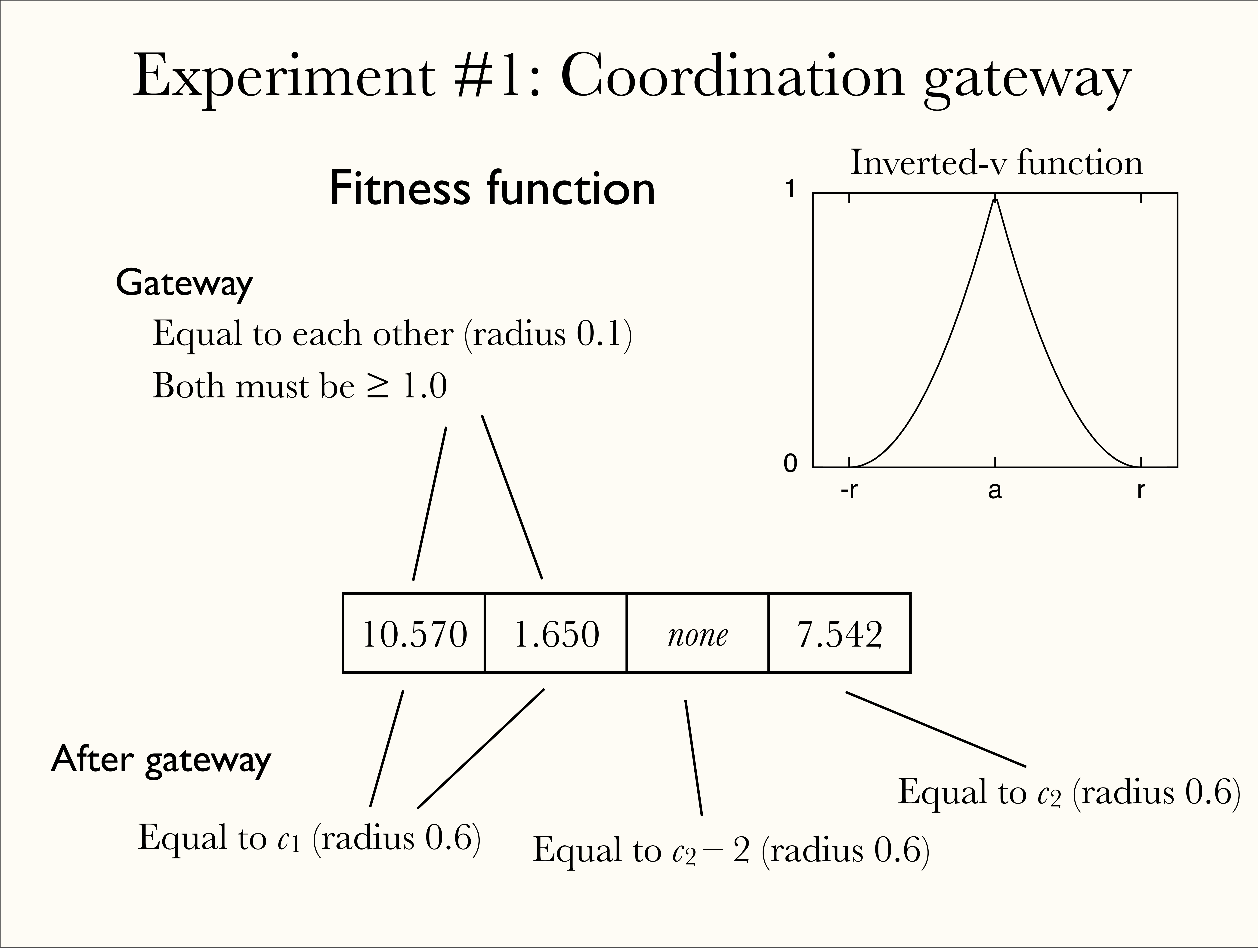
Reward a mathematical relationship between phenotype elements ("coordination") as well as absolute numbers.

Randomly change constants every 20 generations ("epoch") while retaining the coordination.

Population: 40.

Phenotype

10.570	1.650	*none*	7.542

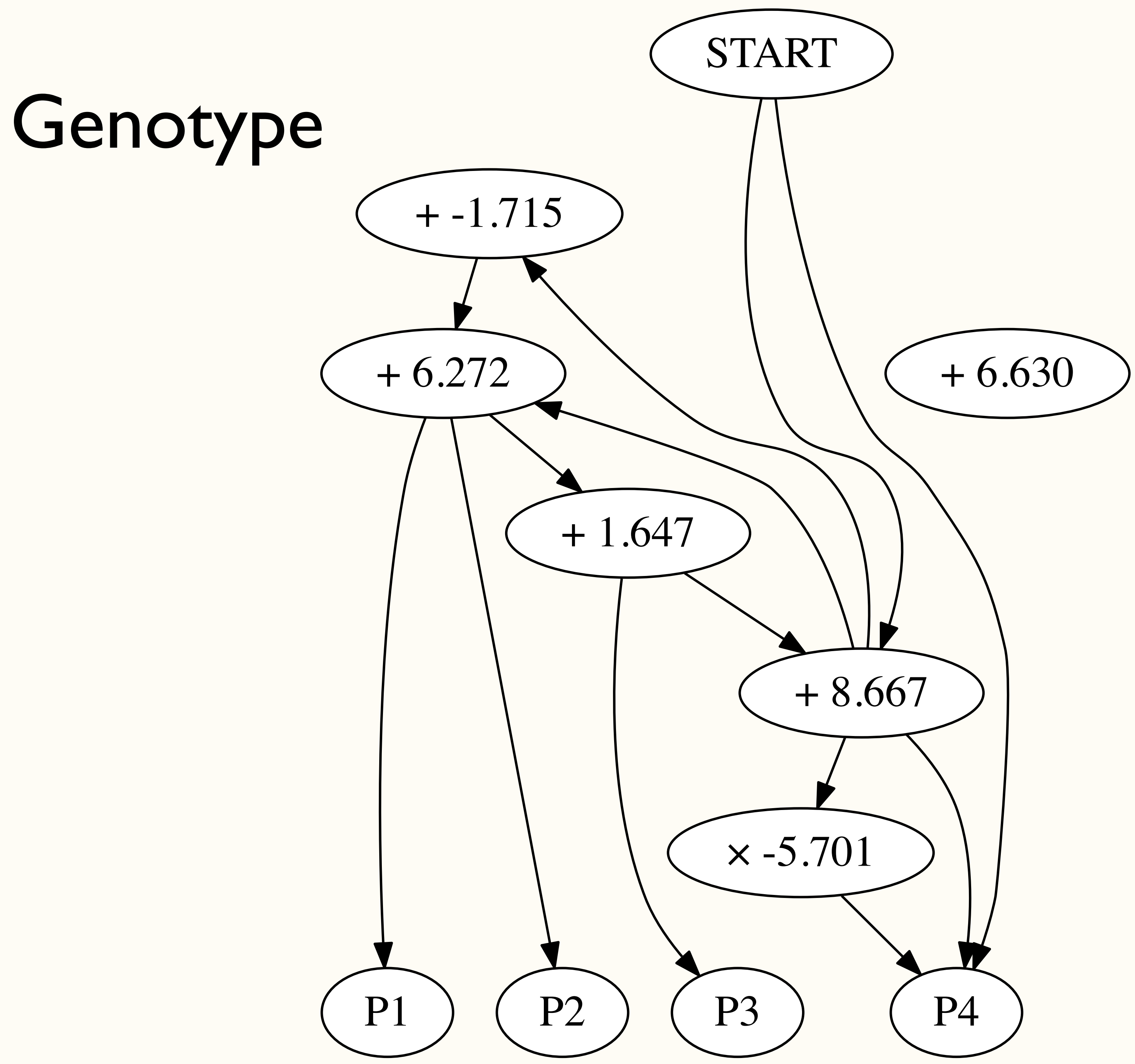


Friday, July 24, 15

Cartoon example: It helps if both your legs are the same length. If the environment rewards longer legs than you have now, but legs of unequal length are catastrophically bad, it helps a lot if a single mutation can lengthen both legs by a little bit. Then you have a fitness gradient that you can hill-climb. If each leg is lengthened by a separate mutation, you need two simultaneous mutations to advance in fitness.

The inverted-v function provides a steep hill to climb within the radius, and no gradient at all outside the radius.

Experiment #1: Coordination gateway



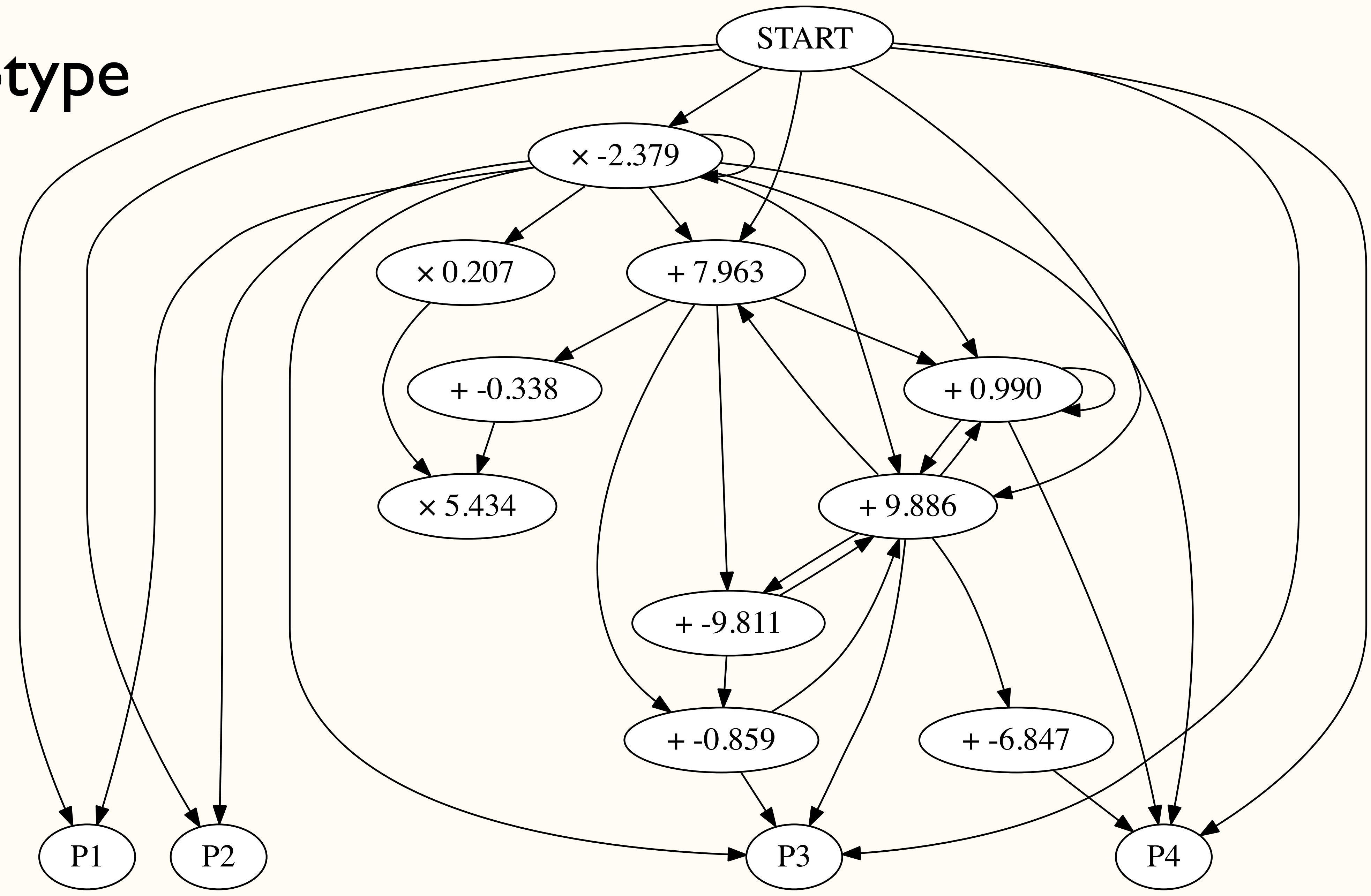
Phenotype

5.496	5.496	6.586	8.804

This is pretty typical. Most mutations don’t alter P1 or P2. Most mutations that do alter P1 and P2 leave them equal.

Experiment #1: Coordination gateway

Genotype



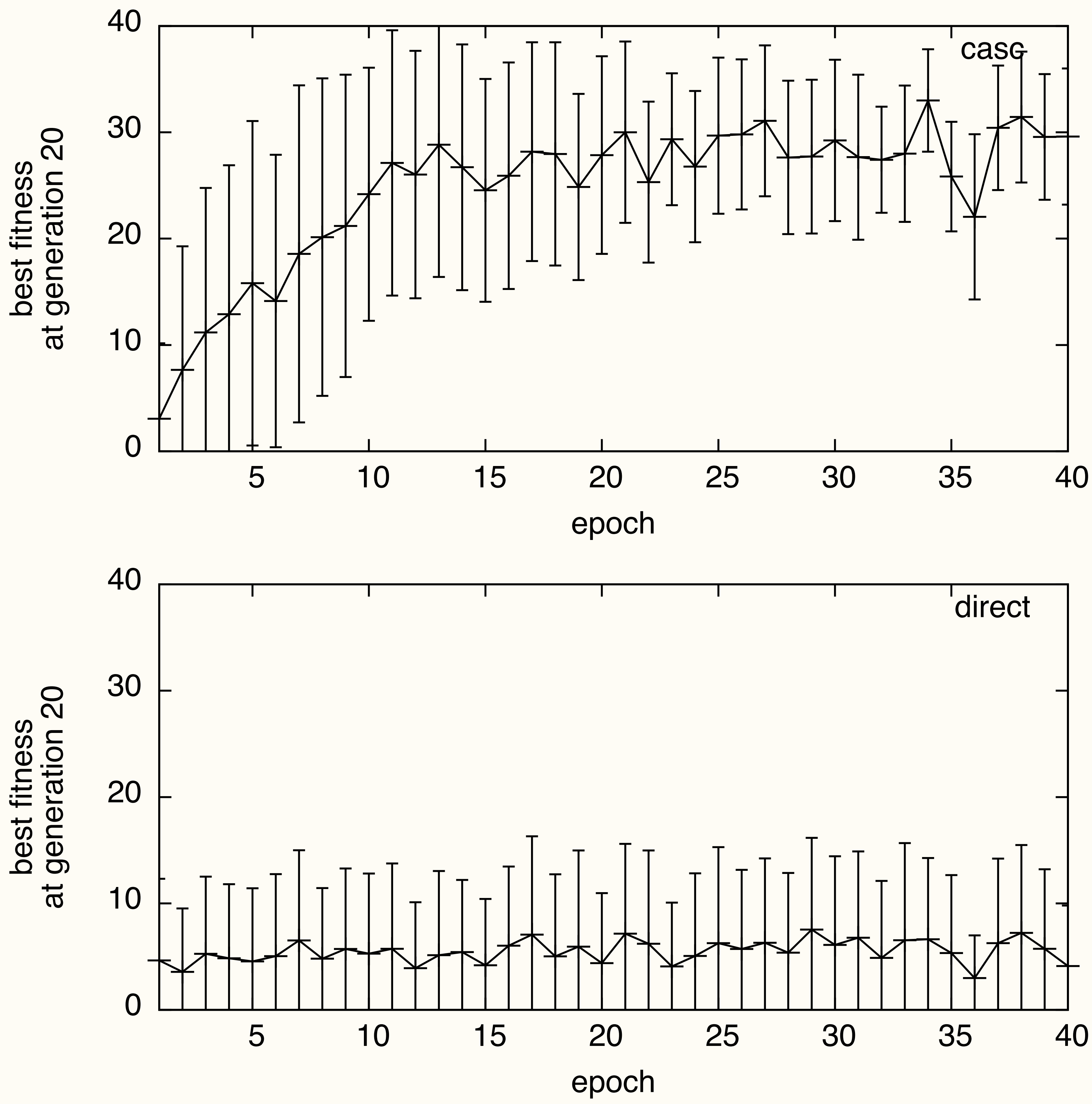
Phenotype

13.790	13.790	1.649	−3.349

Friday, July 24, 15

a common feature of many graphs that evolved in both experiments: genotype nodes occur in chains or well-connected communities that provide many mutation targets, each of which varies the same "global" parameter of the phenotype nodes of direct relevance to the fitness function. Thus, the probability is very high that any mutation will preserve coordinations that have proven crucial in previous evolution

Experiment #1: Coordination gateway

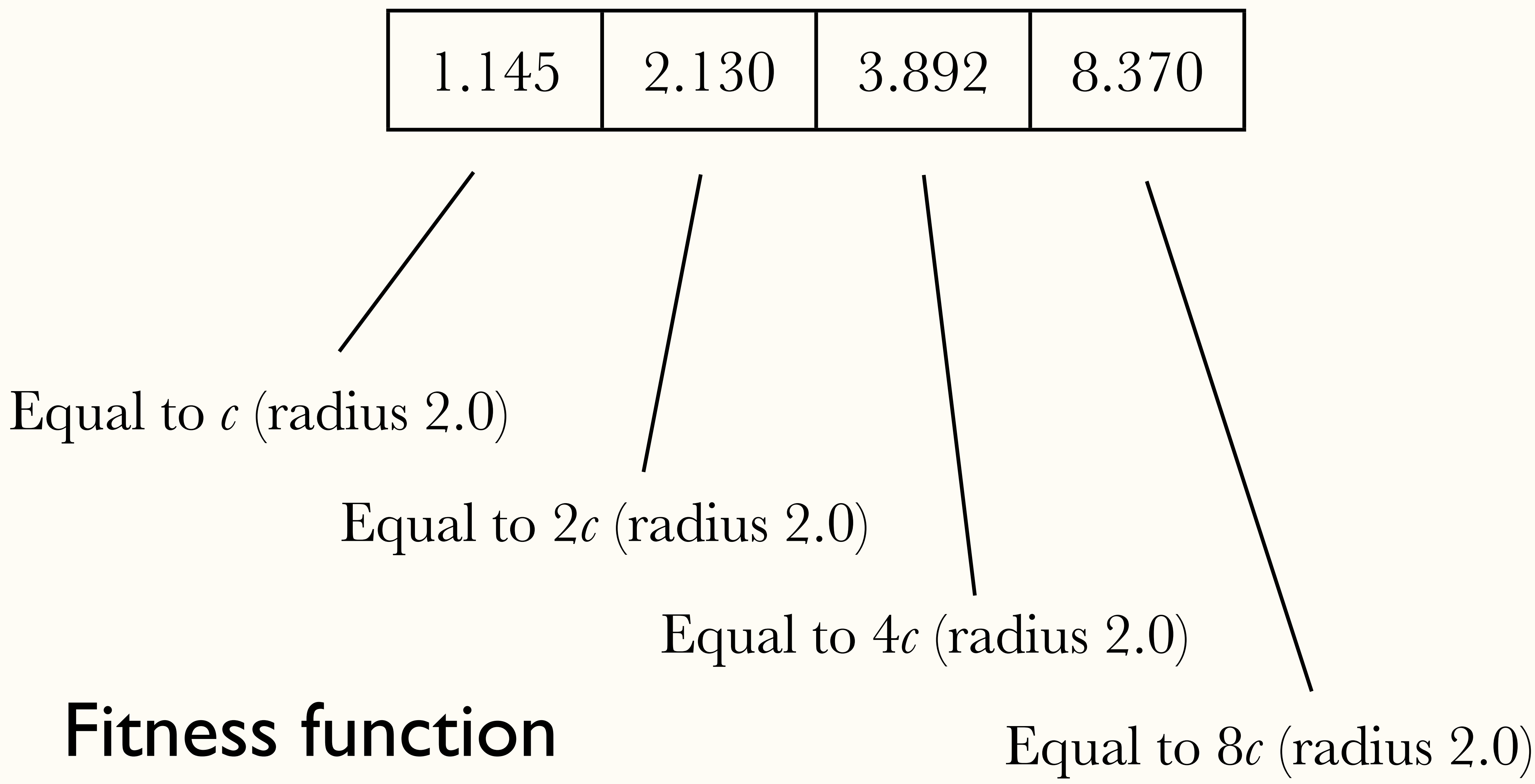


Friday, July 24, 15

Direct evolution did improve over the first few epochs. The reason is, usually the best phenotypes at the end of an epoch at least have approximately equal P1 and P2. But they find it hard to evolve much beyond that, since P1 and P2 must stay equal in order to gain benefits from the rest of the fitness function.

Cascading design: expected value of a mutation much lower, but max much higher. Most mutations are catastrophic, but a few are excellent, and that's all that matters.

Experiment #2: Invariant Ratio



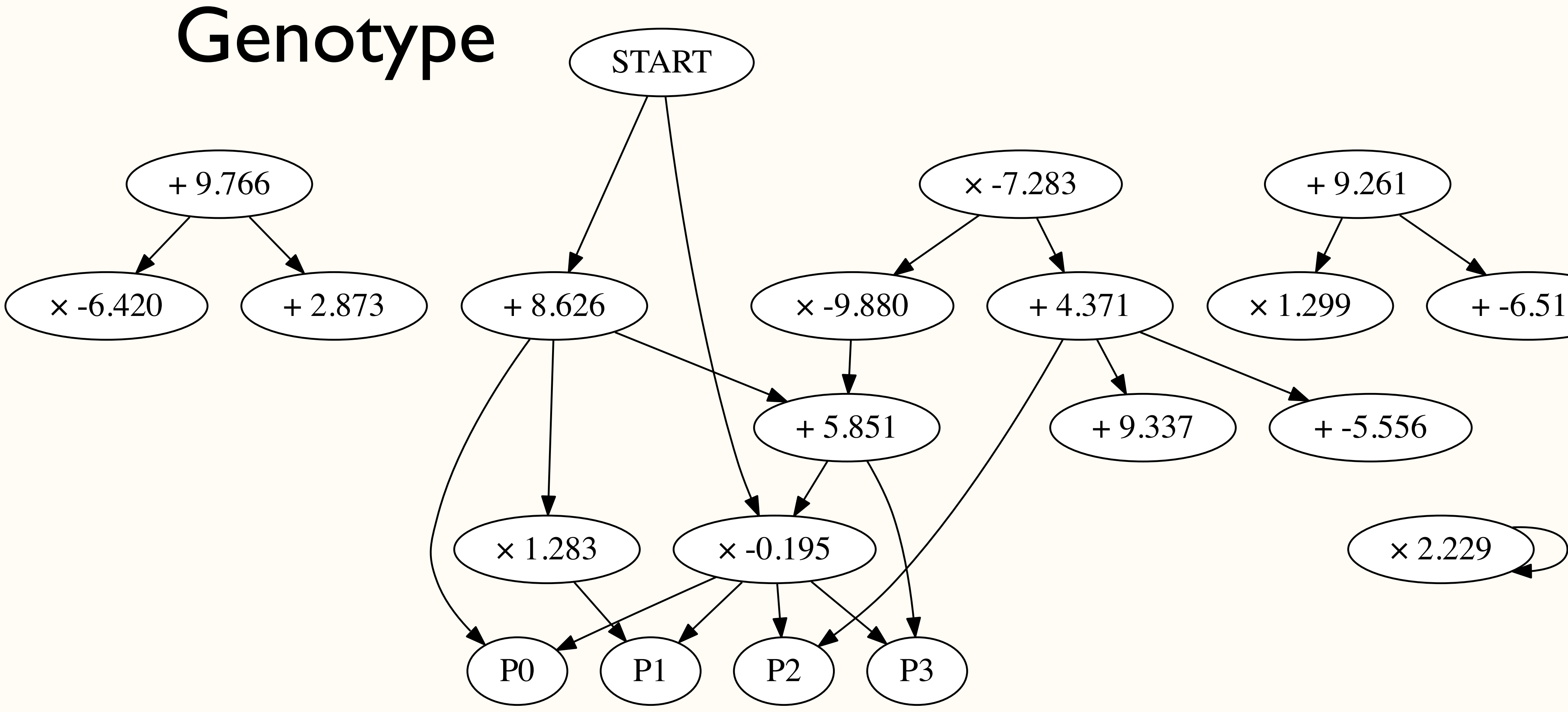
Pick a new random number -10.0 < c < 10.0 every epoch.

Ratio of 2 between consecutive phenotype elements stays constant. One random constant per epoch.

Note that the ratio is not itself rewarded. Only closeness to the absolute quantity is rewarded. 1 pt for each equality.

Experiment #2: Invariant Ratio

Genotype



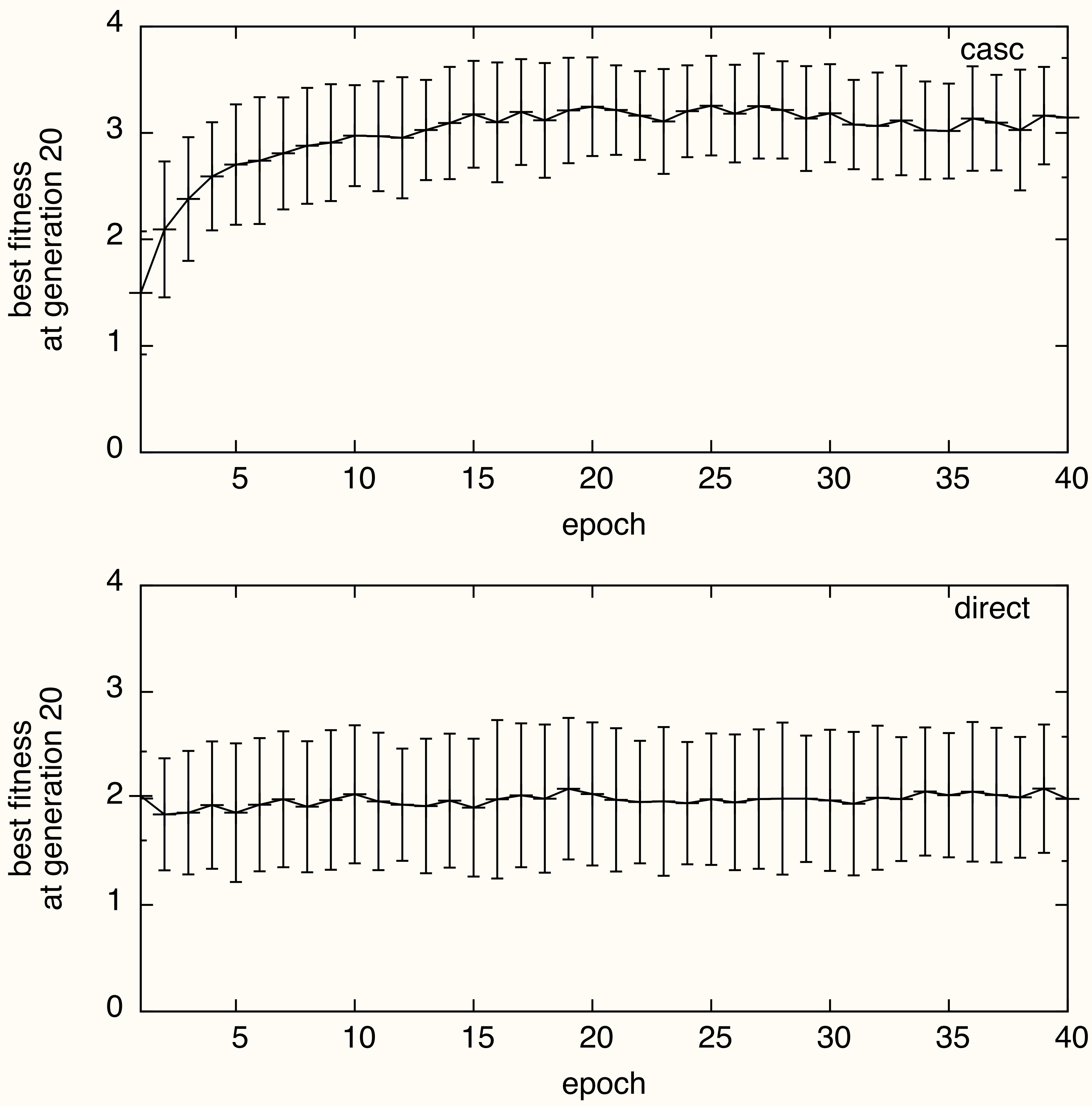
Phenotype

1.145	2.130	3.892	8.370

Friday, July 24, 15

The “rat’s nest” among the active nodes embodies the 2.0 constant ratio: change one number, and the ratio stays the same.

Experiment #2: Invariant ratio



Friday, July 24, 15

Direct evolution can usually get 2 of the nodes close to their target values, but no more.

Cascading design usually got 3 nodes about right, but seldom got all 4. The ratio as represented in the genome was usually a little off from 2.0, so single mutations would often gain at one node while losing at another. So, this fitness function still traps cascading design at local optima.

Conclusions

If…

phenotype is produced by "cascading" genome

and fitness functions reward coordination

and genome can model the coordination

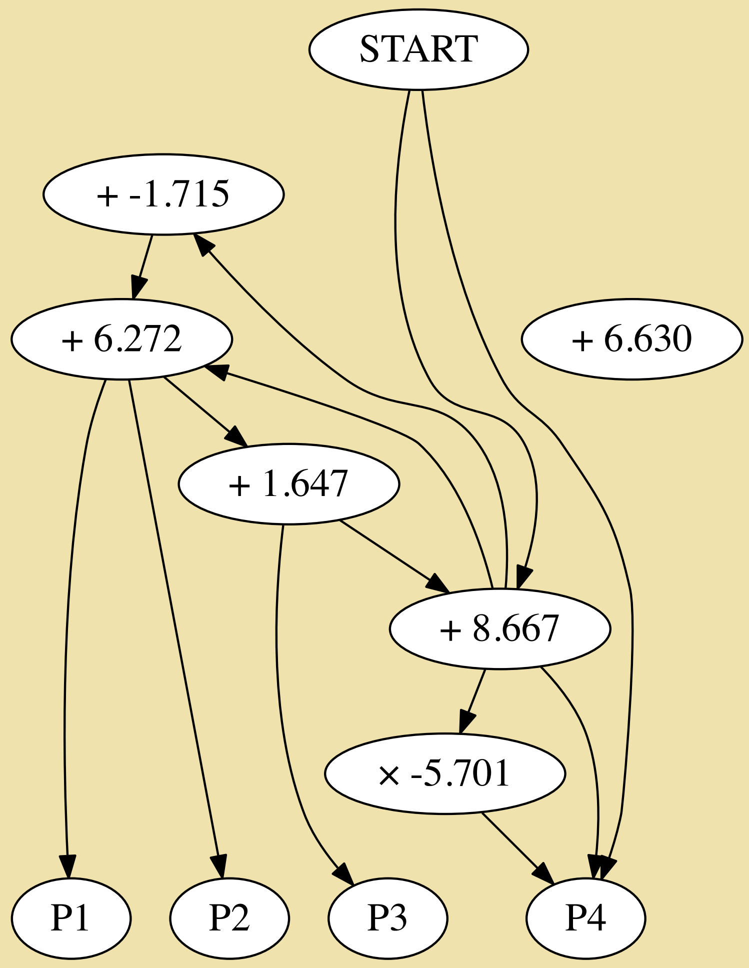
then…

selective pressure (selecting phenotypes by fitness) exerts indirect pressure to increase evolvability (of genotype).

Locality is bad! Epistasis is good!

Hypothesis: "Hillifying" the fitness landscape.

Selective pressure on structure of genotype "hillifies" the fitness landscape as seen by the mutations that only affect numbers.



The End