ENSURING REACHABILITY IN A GRAPH-BASED MODEL OF EVOLUTION

PAUL CHIUSANO

1. Model Description

Consider a graph \( G = (V, E) \), where each vertex \( v \in V \) holds a set of tokens \( D_v \) taken from some alphabet, \( A \).\(^1\) Tokens can be either marked or unmarked and we denote the subsets of marked and unmarked tokens in \( D_v \) as \( M_v \) and \( U_v \), respectively. We will often refer to \( D_v \) as “the (set of) tokens at \( v \),” \( M_v \) as “the marked tokens at \( v \),” and \( U_v \) as “the unmarked tokens at \( v \).” We call \( G \) with its distribution of tokens a \textit{share-network}.

We define the following procedure on a share-network:

1. Choose an unbuilt vertex \( v \in V \) with \( D_v \) having no more than \( \ell \) unmarked tokens \( (|U_v| \leq \ell) \) and mark these tokens. If there are no such vertices, terminate.
2. Visit all immediate neighbors of \( v \), marking any tokens that are common with those in \( v \). (For each neighbor \( i \) of \( v \), modify each \( U_i \) to be \( U_i - D_v \).) Mark \( v \) as \textit{built}.
3. Repeat.

Using this procedure, is it possible to eventually build all vertices in the graph? Or, supposing the graph is infinite, will the system get “stuck” or will it continue to build vertices indefinitely? What makes some vertices “reachable” (capable of being built eventually) and others not? Will there be some general pattern to the record (sequence of built vertices) of the entire run? The answers to these questions will depend on the topology of the graph, the distribution of tokens among vertices, and the parameter \( \ell \).

The Correspondence with Evolution. The simplest evolutionary interpretation of this model is that each vertex in the graph corresponds to a problem that the evolutionary system might be “interested” in solving, and building a vertex in the model is finding a solution to the problem posed by that vertex.

To the extent that the (token sets at) two vertices \( a \) and \( b \) overlap, it indicates that knowing the solution to \( a \) gives some information about how to find the solution to \( b \) and vice versa; i.e., the two problems are related. There are degrees of overlap. At one extreme, problems \( a \) and \( b \) may have nothing to do with one another, so knowing the solution to \( a \) will not help at all in finding the solution to \( b \). At the other extreme, problems \( a \) and \( b \) might be so closely related that knowing the solution to \( a \) makes finding the solution to \( b \) trivial. The tokens themselves do not necessarily

\(^1\)A generalization is to allow duplicates in \( D_v \). Then if we have two connected vertices \( a \) and \( b \) with \( U_a = \{1,1,1,2,2,3\} \) and \( D_b = \{1,2,3\} \), after building vertex \( b \) we would have \( U_a = \{1,1,1,2,2,3\} - \{1,2,3\} = \{1,1\} \).
ensuring reachability in a graph-based model of evolution

equate to things like “sections of the genome” or “steps in a proof” or “physical parts in a machine”; they exist mainly implement this concept of relatedness in a precise way.

The number of tokens at vertices can be thought of as an indicator of how much knowledge a solution to that problem would need to draw upon (knowledge is being used in a very broad sense here), or how many dependencies would need to be satisfied in order for a solution to that problem to become possible—roughly speaking, it is an indicator of “difficulty.” Put another way, the number of tokens at a vertex indicates how likely it would be for a solution to that problem to spontaneously burst into existence. For instance, it might be conceivable for a society with zero scientific knowledge to accidentally discover general relativity, but the chances of that actually happening are essentially zero. On the other hand, such a society might be quite capable of originating some myth to explain the existence of stars. The share-network for this system would therefore have significantly more tokens at the vertex corresponding to general relativity than at the vertex corresponding to the star-existence myth.

Note that overlap between $D_a$ and $D_b$ only means that it is possible to apply knowledge of how to solve $a$ to problem $b$ in principle; edges between vertices signify that the evolutionary system can actually take advantage of this relatedness (if it exists) in practice. An edge may connect vertices whose problems are not related; likewise, not all vertices which are related are necessarily connected.

Since this description is quite abstract, here are several examples of how different evolutionary systems map onto the model:

- **Biology.** Vertices correspond to viable strategies for existence in the biosphere. Building a vertex is evolving an organism which implements that strategy. Vertex overlap indicates that the strategies share some underlying commonality which could be exploited by a designer with perfect knowledge. Edges connecting vertices indicate that this commonality, if it exists, can be used to enable actual discovery. Most edges will be between the class of strategies that constitute a species (members of a species can transfer adaptations to one another), though there might be inter-species links (since events such as symbiogenesis can occur).

- **Technology.** Vertices correspond to useful technological functions. Building a vertex is creating technology which fulfills that function. Vertex overlap between two pieces of technology indicates that ideas, knowledge, or some aspect of the work from one technology could be used in developing the other. Edges connecting vertices $a$ and $b$ indicate cases where the society’s communication network allows people working on technology $a$ to become aware of and draw ideas from technology $b$.

- **Mathematics.** Vertices correspond to theorems the mathematics community is interested in proving true (or false). Building a vertex is proving that theorem. Overlap between two vertices indicates that proving one theorem could in principle aid in proving the other theorem. Edges connect the vertices if there are mathematicians aware of both problems and who are capable of transferring the knowledge about one problem to knowledge of the other problem. It may be the case that two obscure problems are related, but that mathematicians fail to actually see the connection between them (the vertices overlap but are unconnected by an edge).
• *Science*. Vertices correspond to unsolved questions or problems about phenomena in the universe. Building a vertex is finding a model or theory which answers the question or explains the phenomena. Vertex overlap means that two scientific theories are related somehow; edges connecting vertices means that scientists are aware of this relatedness and can take advantage of it. For instance, certain questions about the evolution of cooperation in biology are related to models and theories in the domain of game-theory and this connection has been exploited by evolutionary theory (concept of ESS).

The parameter $\ell$ is the *lookahead* of the system. Lookahead can be thought of as the system’s ability to discover solutions in the absence of viable intermediate configurations. Another way of looking at it is that it captures how much previously nonexistent knowledge the system can synthesize in order to solve some problem which is of interest. If a group of cavemen were asked to build a cart for carrying supplies, they could conceivably manage even if they had not yet discovered the wheel. But they would not likely be capable of building a modern computer unless viable subgoals existed. Likewise, in biology, a primordial cell might evolve a form of locomotion “on its own” but would be unlikely to spontaneously develop into a dolphin unless there were other viable configurations “along the way.” In general, although there is often some sort of gradient to follow toward viable configurations, this gradient may not extend over large “distances” in the adaptive space; this is captured by the possibility that $\ell$ may be finite and quite small in comparison to the number of unmarked tokens at vertices in the graph. Clearly, if $\ell$ is infinite, any vertex is reachable, while if $\ell$ is finite, some share-networks will have unreachable vertices.

If this model is an appropriate cartoon of evolution, it is not obvious why technology, mathematics, biology, or any seemingly open-ended evolutionary system should have managed to avoid getting stuck. Presumably, the share-networks that correspond to these systems have an “adequate mix” of network topology, token distribution, and lookahead. But what is this adequate mix? Are certain classes of share-networks more likely than others to be fully reachable? Understanding what enables ongoing reachability in this simple domain may shed light on how to build artificial systems with increased evolutionary potential.