Echoing Emergence

John H. Holland

SFI WORKING PAPER: 1993-04-023

SFI Working Papers contain accounts of scientific work of the author(s) and do not necessarily represent the views of the Santa Fe Institute. We accept papers intended for publication in peer-reviewed journals or proceedings volumes, but not papers that have already appeared in print. Except for papers by our external faculty, papers must be based on work done at SFI, inspired by an invited visit to or collaboration at SFI, or funded by an SFI grant.

©NOTICE: This working paper is included by permission of the contributing author(s) as a means to ensure timely distribution of the scholarly and technical work on a non-commercial basis. Copyright and all rights therein are maintained by the author(s). It is understood that all persons copying this information will adhere to the terms and constraints invoked by each author’s copyright. These works may be reposted only with the explicit permission of the copyright holder.

www.santafe.edu

Many of our most troubling long-range problems -- trade balances, sustainability, AIDS, genetic defects, mental health, computer viruses -- center on certain systems of extraordinary complexity. The systems that host these problems -- economies, ecologies, immune systems, embryos, nervous systems, computer networks -- appear to be as diverse as the problems. Despite appearances, however, the systems do share significant characteristics, so much so that we group them under a single classification at the Santa Fe Institute, calling them complex adaptive systems (cas). This is more than terminology. It signals our intuition that there are general principles that govern all cas behavior, principles that point to ways of solving the attendant problems. Much of our work is aimed at turning this intuition into fact.

Even a brief inspection reveals some characteristics common to all cas: All cas consist of large numbers of components, agents, that incessantly interact with each other. In all cas it is the concerted behavior of these agents, the aggregate behavior, that we must understand, be it an economy's aggregate productivity, or the immune system's aggregate ability to distinguish antigen from self. In all cas the interactions that generate this aggregate behavior are nonlinear, so that the aggregate behavior cannot not be derived by simply summing up the behaviors of isolated agents. It is this latter characteristic that makes the study of cas so difficult.

Another feature comes close to being a trademark of cas: The agents in cas are not only numerous, they are also diverse. An ecosystem can contain millions of species melded into a complex web of interactions. The mammalian brain consists of a panoply of neuron morphologies organized into an elaborate hierarchy of modules and interconnections. This diversity is not just a kaleidoscope of accidental patterns. The persistence of any given part (agent) depends directly on the context provided by the rest. Remove one of the agent types and the system reorganizes itself with a cascade of changes, usually "filling in the hole" in the process. Moreover, the diversity evolves, with new niches for interaction emerging, and new kinds of agents filling them. As a result, the aggregate behavior, instead of settling down, exhibits a perpetual novelty, an aspect that bodes ill for standard mathematical approaches.

There is a less obvious feature of cas that is important. Cas agents employ internal models to direct their behavior. (Murray Gell-Mann uses the term schema to describe this aspect of agents, but I have used that term in the past as a technical term in the study of genetic algorithms, so I cling to 'internal model'). An internal model can be thought of, roughly, as a set of rules that enables an agent to anticipate the consequences of its actions. We are most conscious of our own internal models when we do lookahead in a complex game such as chess or Go. We try to move in such a way as to set the stage for future favorable configurations, as when we sacrifice a piece or make a "positional move" in order to capture an important piece later. However, even an agent as simple as a bacterium employs an "unconscious" internal model when it swims up a glucose gradient in the search for food. And humans make continual prosaic use of internal models. Consider our unconscious expectation that room walls are unmoving -- should this expectation be violated our attention immediately shifts to that part of the environment.
Internal models add still further to the complexities of aggregate behavior. Anticipations based on internal models, even when they are incorrect, may substantially alter the aggregate behavior. The anticipation of an oil shortage can cause great changes in aggregate behavior, such as stockpiling and price runups, even if the shortage never occurs. The dynamics of cas will certainly remain mysterious until we can take such effects into account.

2. Two Important Questions.

Two questions stemming from these observations are, for me, "right questions", in that they lead to useful thoughts about general principles:

Why do cas evolve toward diverse arrays of agents rather than toward optimal agents?

How do agents create and exploit internal models?

These questions might seem a bit to one side of central issues, but the following arguments make me think otherwise.

2.1 Diversity.

Consider first the question of diversity, and the attendant perpetual novelty. Diversity and perpetual novelty do blunt some of our most powerful tools for understanding complexity, but they also provide an entering wedge to deeper insight. If we look at real cas closely, it is clear that the diversity, in every case, is the product of progressive adaptation. It is a dynamic feature, much like the standing wave produced by a rock in a fast-moving stream: If you poke a "hole" in the flow, it quickly "repairs" itself. It is the pattern that persists, not its perpetually changing components.

The standing wave simile can be carried further if we think of cas as involving flows of resources through agents. Each agent is a kind of middleman, accepting resources from other agents, modifying them in some way, and passing them on to still other agents. The 'births' and 'deaths' of agents in a cas produce a continual turnover in the component parts of the flow. If you remove one kind of agent, the flow of resources is temporarily redirected, but the evolutionary mechanisms soon produce a new kind of agent to exploit the abandoned 'niche' (the 'convergence' of evolutionary biologists). For cas, however, the standing wave simile does not go far enough because the complexity of the interactions increases over time. Each time a new kind of agent arises it opens possibilities for interaction -- new niches -- for still newer agents. The pattern evolves.

The complexities of these long-term adaptive progressions are best exemplified by the evolutionary patterns studied by paleontologists. These progressions exhibit a complex hierarchical organization that is characteristic of all cas; we do not find real cas consisting of a few highly adapted individuals that exploit all opportunities. Paleontologists have an accepted principle that should be applied, mutatis mutandis, to the study of this universal characteristic of cas: To understand species, understand their phylogeny. If we can find common mechanisms that give rise to hierarchical organization, with the diversity and perpetual novelty it entails, we will have taken a substantial step toward uncovering general principles that govern cas dynamics.

A useful analogy for exploring the origin of hierarchies comes from the study of developing default hierarchies in classifier systems (for details see Holland et al. on Induction 5). In a classifier system, the first rules that establish themselves are 'generalists', rules that are
satisfied by many situations and have some slight competitive advantage. They may be "wrong" much of the time, but on average they produce interactions that are better than random. Because their conditions are simple, such rules are relatively easy to discover, and they are tested often because they are satisfied often. The frequent tests assure that survivors exploit real recurring features that offer advantages over random action. That is, the repeated tests provide a "statistical" confirmation of the generalist's "hypothesis" about its world. Once the 'generalists' are established, they open possibilities — niches — for other rules. A more complicated rule that corrects for mistakes of an over-general rule can benefit both itself and the over-general rule. It benefits from its own useful actions, while it prevents the 'generalist' from making a mistake. A kind of symbiosis results. Repetitions of this process produce an increasingly diverse set of rules that, in aggregate, handle the environment with fewer and fewer mistakes. As we will see, the mechanisms involved here have counterparts in other cas.

Along similar lines, there are good reasons why single "super-agents" that fill all niches do not appear. The exploitation of resources generated by the aggregate behavior of a diverse array of agents is much more than the sum of the individual actions — the nonlinearity of the interactions again. For this reason, it is a complex task to design or discover a single agent with the same capabilities for exploiting the resources. It is simpler to approach this capability step-by-step using a distributed system, as in the case of a developing default hierarchy. There will be more to say about this point in the discussion of internal models.

To return to the basic point, in cas it is the evolving patterns of interaction that are important. The patterns of interaction most familiar from ecology -- symbiosis, parasitism, mimicry, biological arms races, and so on -- all are best described in terms of flows of resources through agents, and all have counterparts in other cas. Agents direct these flows, sometimes rearranging the basic 'elements' defining the resources, but generally conserving the elements themselves. When groups of agents cause cyclic flows, resources are held within the system. Resources so retained become more readily available to other agents. Cas thrive in proportion to their ability to keep resources around, and their ability to do so increases as the number of interactions, particularly cyclic interactions, increases (another point that will be elaborated later). This is simply natural selection writ large. Parts of cas that exploit these options persist and expand; parts that fail to do so lose their resources to those that do. Thus, there are strong selective pressures on cas to discover and retain agents that provide such interactions. Each new interaction offers possibilities for still further interactions and redirection of flows, pushing diversity still further. Technically, cas search for a diversity of agents that provide progressively refined covers of the range of possibilities, rather than trying to design an optimal agent that handles all possibilities.

2.2 Internal models.

Let me turn now to the processes whereby agents create and exploit internal models. Here, there is a difficult definitional question: How can we know from the specifications of an agent that it has, or will build, an internal model? What characterizes agents that have internal models?

We might start with the idea that an agent has an internal model if we can infer something about the agent's environment by just inspecting the agent's structure. It is certainly true that we infer a great deal about the environment of any organism by studying its morphology and biochemistry. We can infer a nocturnal environment from big eyes, and we can infer a mosquito-rich, malaria-infested environment from sickle cells. But this is an inadequate diagnostic for internal models. Consider a meteorite. We can infer much of its environmental history from its composition and surface condition, but it is fruitless, even metaphorically, to attribute an internal model to meteorites.
Somehow it doesn't make sense to attribute an internal model to an agent unless the model takes an active role in determining the agent's behavior. The model should suggest current actions that make future environmental states accessible, and the agent should act on these suggestions. Here we can make a useful distinction between *tacit* and *overt* models. A tacit internal model simply prescribes a current action under an implicit prediction of some desired future state. For example, a bacterium swimming up a chemical gradient employs a tacit internal model that implicitly predicts valuable resources at the apex of concentration. (We might, as earlier, call tacit models "unconscious", if that weren't such an ill-defined word). On the other hand, an agent uses an overt model for active internal, or virtual, explorations of alternative lines of action. This virtual exploration is often called *lookahead*, the quintessential example being exploration of alternative move sequences in games like chess. Both tacit and overt internal models actively affect the agent's behavior and *cas* employ internal models of both kinds. The internal models of agents in an immune system are at the tacit end of the scale, while the internal models of agents in an economy are both tacit and overt.

In realistic situations, internal models must be based on limited samples of an agent's environment. In a perpetually novel environment, generalization from limited samples is a sine qua non for exploiting experience. Only with generalization can the model be useful in situations not yet encountered. Stated another way: When situations never recur, the model must treat situations not previously encountered as equivalent to situations already seen. Technically, the model must be built in terms of equivalence classes over the set of environmental states. The problem then is to find useful equivalence classes, classes that capture regularities in the agent's environment.

How easy is it to discover and exploit regularities? Artificial worlds in which useful regularities are rare or non-existent are easy enough to construct. However, that does not seem to be the case in the real world. We can describe almost anything in the real world in terms of simpler component parts, so-called 'building blocks'. Moreover, limited numbers of building blocks can be recombinated in many ways to yield descriptions and models of diverse arrays of real objects. This is true of everything from the morphology or biochemistry of a living organism to "a red Saab by the side of the road with a flat tire". The building blocks let us construct reasonable descriptions, and hypotheses, for situations we've never before encountered.

Here the discussion of internal models merges with the earlier discussion of diversity, extending our discussion of classifier systems and default hierarchies. Earlier, the search was for a diversity of rules that, working together, responded usefully to the environment. Rules were constructed by representing frequently encountered regularities in the environment. Technically, each regularity corresponds to an equivalence class over the possible states of the environment, the elements of the equivalence class being all environmental states involving the given regularity. When the regularity is captured in the condition for a rule, the corresponding equivalence class has been turned into a building block. The question becomes: Do internal models foster rules that correspond to building blocks that can be combined in a variety of ways?

To see why internal models favor diversity of this kind, consider the commonplace building blocks for a face (hair, forehead, eyebrows, eyes, etc.). Consider a set of rules, wherein each rule responds to a particular alternative for one of these components (a particular hair color, a hairline, a shape of forehead, an eyebrow shape, or the like). With a few dozen such rules it is possible to describe millions of faces. Each rule, then, is active in a variety of contexts -- the different alternatives for other parts of the face -- gaining utility accordingly. A novel situation is handled by simultaneously activating various rules dealing with the building blocks of the situation. A face never before seen can be handled by an appropriate selection of rules describing its component parts. "A red Saab by the side of the road with a flat tire" evokes rules dealing with "red", "roadsides", "tires", "cars", etc. Thus, the discovery of building blocks plays a key role in the formation of useful internal models, assuring both diversity of agents and diversity of interactions.
In constructing internal models, parsimony is a problem, even when the model condenses environmental states into equivalence classes. To see why this is so we must look more carefully at the way a model is specified. A model, a fortiori an internal model, is specified by its transition function: A transition function f specifies what the next state of the model S(t+1) will be when the input I(t) to the model and its current state S(t) are given; i.e. f = IxS → S. For an internal model, there is one element of S for each equivalence class of environmental states. Each action of the agent constitutes an input to the environment, so I contains one element for each possible action.

Simple models, those dealing with a few hundred or a few thousand alternatives, can be presented explicitly by a table. There is one line in the table for each pair in IxS, and for that pair the table specifies the element of S that is expected to follow. For realistic cases, though, the number of lines required in a tabular presentation is just too great. Chess, which presents a problem much simpler and more one-sided than most realistic environments, would require a table with substantially more than $10^{50}$ lines (an average of 10 alternative actions, leading to a distinct configuration, on each of 50 successive moves). Even if the board configurations are condensed into equivalence classes, so that several alternatives at each move were treated as already given in earlier lines of the table, the number of lines would still easily exceed $10^{40}$. Somehow, the model must be compressed.

Compression usually amounts to finding a set of generators for the model (much as one uses a set of generators and relations to generate the multiplication table for a finitely-generated mathematical group). An appropriate set of n generators can generate a table of $2^n$ lines. We can think of rules as generators that are activated in clusters to cover a great diversity of conditions, as in the example of the faces or the "red Saab". The dynamic of discovering these generators is interesting and important. At first the model consists of easily discovered generalizations that are often wrong but better than random. This coarse model is steadily refined by 'exception rules' as experience builds up, as described earlier in the classifier system example. New information is gracefully incorporated, without destroying the parts of the model already constructed. Classifier systems were in fact designed to illustrate this process for discovering and elaborating parsimonious internal models. Such models are called quasi-homomorphisms (see appendices 2A and 2B in Holland et al. for a more formal description).

There is a final point concerning overt internal models. If the model is to be useful for lookahead, the agent must be able to execute the component rules 'fast time'. That is, there must be a fast dynamic that lets the agent run through successive rules faster than the corresponding states occur in the environment. Then the agent can extrapolate from the current situation, using the model to anticipate the effects of different action sequences. The fast dynamic lets the agent "run ahead" of the environment. This use of the overt model entails an additional requirement: The agent must be able to exercise the kinds of control on the model that it would use on the environment. For this to be so, the agent must have two sets of rules: (1) The rules that define the overt model, and (2) the rules that determine action. The second set of rules must include rules that allocate action between the model and the real world. With this arrangement the agent bases current actions in the environment on their predicted future effects as derived from the model. Predictions that are subsequently falsified can trigger the agent into correcting the parts of the model responsible for those predictions. That is, the agent can improve its ability to predict even when rewards (payoffs, reinforcements) are absent. This is a valuable asset in a world where rewards are usually intermittent or sparse.

The Echo models are not meant as a vehicle for studying the acquisition of overt internal models; classifier systems are better suited to that study. However, tacit models play a key role in Echo. Tacit models, while not providing the advantage of lookahead, do offer the advantage of
tacit anticipation of the future consequences of current actions, as in the case of the bacterium with a tacit model that causes it to swim up a chemical gradient. The parsimony of the model enters quite directly, because it costs resources to implement the model. An agent with a tacit model will proliferate only if the resources required to implement the tacit model are less than the additional resources acquired because of the model (suitably discounted, if acquired over an extended period of time). As with other characteristics that enhance an agent's ability to reproduce in its niche, tacit models are acquired over successive generations as the space of possibilities is searched. It is reasonable to believe that there are building blocks for different kinds of tacit models, and that these building blocks will proliferate through a variety of agents.

3. The Echo Models.

The Echo models all have a common framework, though there are several variants, so I will refer to the class as if it were a single model hereafter. Echo may be able to simulate actual ecosystems, but that is not its primary purpose. It is designed to facilitate computer-based gedanken experiments. More specifically, Echo is designed to facilitate explorations for mechanisms that generate major cas phenomena such as diversity and internal models.

Because Echo is a computer-based simulation, it allows no unarticulated or ambiguous assumptions. The generated behavior is a precise consequence of the assumptions implemented. This rigor, combined with Echo's ability to handle very complex systems, provides possibilities not available through traditional mathematical analysis. However, these possibilities are bought at the cost of the generality that usually accompanies mathematical models. Though parts of Echo can be analyzed mathematically, providing important guidelines for the model-building process, the results of individual simulation runs are simply points in the space of possibilities.

It is true that simulation runs can suggest generalizations, but this depends upon assigning real world meanings to numerical results. Such interpretations can be misleading or false. There are cases in the literature where a relatively simple algorithm, such as linear regression, is identified with some sophisticated real world process, such as 'perception'. Or some generated numerical sequence that bears a superficial similarity to collected data is labelled with the same name, even when there has been no attempt to compare the underlying mechanisms. With care, simulations can suggest genuine generalizations, generalizations that can be rigorously established, but indifferent attention to problems of interpretation can destroy the usefulness of computer-based models. Facile labelling of what are, after all, streams of numbers in a computer, leaves too much to the eye of the beholder.

I have taken two steps to meet this 'eye of the beholder' difficulty. First, I have tried to select primitive mechanisms that have direct counterparts in the various cas. The interpretation of the behavior generated by these mechanisms is thereby constrained to be consistent with the interpretation of the mechanisms. Second, Echo incorporates, as special cases, a wide range of well-established paradigmatic models -- Wicksell's Triangle, Overlapping Generation models, Prisoner's Dilemma games, 2-armed and n-armed Bandits, abstractions of the antigen-antibody matching process, biological arms races, cyclic food webs, and so on. These paradigmatic models have undergone intense scrutiny in the disciplines in which they originated -- economics, political science, immunology, and so on -- and have been adjudged to be relevant abstractions of critical problems in those disciplines. Their appearance in Echo forms a bridge from the abstractions in Echo to familiar, accepted abstractions in the relevant disciplines, again constraining the interpretation.
3.1 The organization of Echo.

Echo provides for the study of populations of evolving, reproducing agents distributed over a geography, with different inputs of renewable resources at various sites. In the simulation the resources are designated by letters drawn from a small finite alphabet. Each agent has simple capabilities defined by a set of 'chromosomes': Resource acquisition (from the environment or by interaction with other agents), resource transformation, and chromosomal exchange. In the simulation these capabilities are defined by a combination of behavior-defining strings (rules) and tags (playing the role of "identifiers" such as banners, trademarks, active surface elements, or the like). The strings are defined over the same alphabet as the resources; that is, the strings are defined in terms of the resources required to construct them.

Though these capabilities are simple, and simply defined, they provide for a rich set of variations. Collections of agents can exhibit analogues of a diverse range of phenomena, including ecological phenomena (e.g., mimicry and biological arms races), immune system responses (e.g., interactions conditioned on identification), economic phenomena (e.g., trading complexes and the evolution of "money"), and even evolution of multi-cellular organization (e.g., emergent hierarchical structure).

3.2 Primitive elements.

Agents play a central role in Echo, but the basic primitive is a component I'll call a compartment. Agents are collections of compartments, much as a cell consists of a collection of organelles. The compartments in an agent interact with each other and with compartments in other agents. In any given interaction resources are exchanged between compartments; individual compartments may also transform a resource from one kind (letter) to another.

The full capabilities of a compartment are specified by four strings put together from the resource alphabet:

1. A tag that serves as a (phenotype) marker that is visible to other compartments.
2. An interaction condition that specifies the kinds of compartments -- as determined by their tags -- the given compartment will interact with.
3. An offense string that is used to determine a compartment's ability to "force" the outcome of an interaction (cf. the notion of induction in developmental biology).
4. A defense string that is used to determine a compartment's ability to "resist" attempts to force the outcome of an interaction (cf. the notion of competence in developmental biology).

In addition, the compartment has a reservoir in which it stores the resources it acquires.

3.3 Reproduction.

Reproduction of an agent.

Agents reproduce by reproducing their component compartments. To reproduce a compartment the agent must provide copies of each of the four strings that define that compartment. Because the strings consist of sequences of letters from the resource alphabet the agent must collect enough of the proper resources to make copies of these strings. If the agent has more than one compartment then, before it can reproduce, it must collect enough resources to make copies of the strings defining all of its compartments.

In more detail: By absorption of resources from the site, and through interactions with other compartments, each compartment in an agent collects resources in its reservoir. For purposes of reproduction, an agent is assumed to have access to the resources in all the reservoirs of its component compartments, so that resources in the reservoir of one compartment can be used to
Figure 1. General Overview. *Echo* simulates a connected array of sites, each of which has its own distribution of renewable resources. Agents within a site can take up resources from the site and may acquire resources by interacting with other agents at the site. Agents may migrate from site to site.
Figure 2. A Compartment. Agents are composed of compartments and interactions between agents are mediated by their compartments. The resources exchanged in an interaction are collected in the reservoirs of the interacting compartments. The form of the interaction is controlled by four basic components -- a condition, a tag, an offense string, and a defense string -- each of which is specified by a string defined over the alphabet of primitive resources. When two compartments come into contact, each attempts to match its condition against the other's tag. Matches may be bilateral ("trade"), unilateral ("combat"), or null (no interaction). If an interaction takes place, the amount of resource exchanged is determined by cross-matching each compartment's offense string against the other's defense string. The outcome may range in severity from a simple exchange of surplus resources, from the reservoirs, to complete destruction of either or both of the compartments. From the time of its first completed interaction onward a compartment is said to be in active status.
Figure 3. Details of an Interaction. When two agents come into contact, one compartment of each agent is selected for testing. The interaction condition of each selected compartment is checked against the tag of the other compartment. If one or both conditions are satisfied, the interaction proceeds. The intensity of the interaction is determined by matching the offense string of each compartment against the defense string of the other compartment. The score, for each compartment, determines the amount of the compartment's resources -- resources in its reservoir and, possibly, resources defining its structure -- transferred to the reservoir of the other compartment.
replicate another compartment. When an agent's compartmental reservoirs, taken as a whole, have enough resources to make copies of all the strings defining all the agent's compartments, the agent is ready to replicate itself (subject to conditions we'll come to shortly).

With this provision, an agent's rate of reproduction depends entirely on its ability to collect the necessary resources to copy its compartments. A complex agent, with many compartments and complex capabilities (long strings defining the compartments) will have to collect many resources in order to reproduce. There is no explicit fitness parameter; indeed the fitness of any given agent is dependent upon the context provided by its site and the other agents at that site.

Reproduction of multi-agent organizations.

One of the most interesting features of real cases is their hierarchical organization. If we look to a multi-cellular organism, we see that it consists of a great many different kinds of cells, ranging from nerve cells to muscle cells. Nevertheless, all of these cells share the same basic blueprint. The chromosomes of all the different cells in the multi-cellular organism are identical. The differences come about because only some of the genes in each cell are expressed; the genes expressed determine the detailed structure of the cell.

To translate this style of organization to Echo, we must supply each agent with a 'chromosome' that specifies the set of compartments it can have. Each 'gene' in the 'chromosome' specifies a compartment. Then we must make some provision for turning genes on and off, so that the actual compartments the agent contains depend upon what genes are on at the time the agent is formed. That is, when the agent replicates, a compartment gets copied only if the corresponding gene is on (cf. repressors for an operon in a bacterial chromosome). Thus, the offspring of an agent only has those compartments specified by the genes that are on in the parent, though the offspring carries the whole chromosome of the parent.

The question now becomes: What determines whether a given gene is on or off? The object is to make the condition endogenous and subject to selection. A simple version associates a condition with each gene; the gene is turned on only if the associated condition is satisfied. To make the condition endogenous, I will make it dependent upon activity within the agent. Since all activity centers on compartments, this means defining the condition in terms of the activity of some particular (kind of) compartment.

In what follows, I will define a compartment as active if it has undergone an interaction (processed some resources) for the agent prior to the time replication starts. A gene's condition then becomes a requirement that some particular (kind of) compartment is active. Note that this may be a compartment quite different from the one the gene is associated with. (Just as the metabolic product of one organelle in a cell can repress or de-repress the activity of a gene associated with a different organelle).

With this provision it is possible to arrive at a multi-agent organization consisting of differentiated agents with a common chromosome. Assume, at the time of reproduction, that the offspring agent "sticks" to its parent, instead of floating away to become a "free agent". That is, the offspring agent becomes part of the multi-agent organization. It may, however, have a different set of compartments than its parent, because this depends upon the genes that are on in the parent. Nevertheless it has the same chromosome as its parent. As this process is iterated, all agents in the multi-agent organization share a common chromosome, but they may be greatly different in their compartmental composition.

A multi-agent can be generated from a single agent, a 'seed', via the iterative production of "sticky" offspring. If, at some point, one of the offspring does not "stick" to the cluster, then it
Figure 4. An Agent. Agents are the basic self-replicating entities. Replication is controlled by a replication condition defined over the set of possible compartmental tags; replication proceeds only if there is an active compartment in the agent that has a tag satisfying the replication condition. In addition each compartment has an associated on/off condition, also defined over compartmental tags and satisfied in the same way as the replication condition. A compartment is copied at the time of replication only if its on/off condition is satisfied. The replication will be executed only if the agent has collected enough resources in its compartmental reservoirs to make copies of the four defining strings of each of the compartments with satisfied on/off conditions; the resources in all of the compartmental reservoirs are shared for this purpose. If there are not enough resources in the reservoirs the replication is aborted.

The replication condition, together with the set of on/off conditions and one other set of conditions described in Figure 5, constitutes the agent's 'chromosome'. Any or all conditions may be null; a null condition is treated as satisfied at all times. The full chromosome is always copied during replication, but it may undergo mutation and recombination during the process.
Figure 5. Development of an Hierarchical Meta-agent. To allow agents to organize themselves into more complex structures, each agent has one or more additional chromosomal conditions, called level conditions. As with other chromosomal conditions, level conditions are defined over the set of compartmental tags and are satisfied if there is an active compartment with a tag belonging to the set specified by the condition. If the agent has only null level conditions, or if none of its level conditions is satisfied at the time of replication, then its offspring is "free-living" agent. If one or more level conditions is satisfied then the offspring "sticks" to its parent and the pair move and interact as a unit, a meta-agent. Further offspring with satisfied level conditions will be added to the unit, yielding a more complex meta-agent. A level condition may also specify that the offspring be placed "interior" to its parent, so that the parent and any other agents at the parent's level form a "shell" around the offspring. Iterations of this procedure can yield complex hierarchical meta-agents. If, after successive replications, a given parent has no level conditions satisfied, its offspring will be "ejected" as a free-living agent. If this offspring has an appropriate set of chromosomal conditions, it may become the "seed" for a new meta-agent of similar organization. As always in Echo, a meta-agent will be selected for its ability to collect the resources necessary for its replication.
becomes the potential 'seed' of a new multi-agent. Selection for the ability to gather in the appropriate resources for replication, will act on the multi-agent through selection of the component agents. Only those organizations that provide advantages in collecting resources will survive, and only those organizations that provide for timely release of 'seed' agents will survive. Agents are to multi-agents much as compartments are to agents, the difference being that the agents in the multi-agent organization have the same blueprint (the chromosome), whereas the compartments in an agent each have a different blueprint (the genes).

3.4 Variation.

With these preliminaries in place, we can now address the main question: How can Echo generate, and sustain, an increasingly diverse array of agents and multi-agent organizations? The preliminaries account for selection. Different kinds of agents, and multi-agents, will persist if they regularly collect enough resources to produce lines of progeny. In Darwinian terms, this leaves us with the question of variation. How are new variants introduced?

The compartments can be varied quite simply by adding and deleting letters in the defining strings, a simple kind of mutation. A more sophisticated variation results when strings belonging to different compartments are recombined; this brings into play the full powers of a genetic algorithm. Both of these methods produce new kinds of compartments, but they do not yield variation in the number of compartments belonging to an agent. To accomplish this the number of genes in the chromosome defining the agent must be changed. Genetics suggests an interesting mechanism for doing this, intrachromosomal duplication: Genes sometimes are duplicated within the chromosome, the result being two adjacent copies of the gene where there was one before. In Echo this would yield an agent with two copies of the same compartment. A subsequent mutation in one of the duplicate genes would yield two different compartments in place of two identical compartments. If the resulting agent survives, then further intrachromosomal duplications and mutations can yield agents with still more, varied compartments.

3.5 Capabilities.

The foregoing is only a general description of the compartment-based version of Echo, and this version has not been tested, but results from earlier versions motivate the mechanisms described. The next few paragraphs describe my expectations based on tests of the earlier versions.

In the earlier models, the agents at a site were treated as thoroughly mixed and randomly paired for interaction, rather like the billiard ball model of gases. The present model proceeds in much the same way but now it is compartments, rather than agents, that are paired. That is, in the simplest version, pairs of agents from the site are selected at random, and then a compartment from each agent is selected at random for possible interaction. Compartments within the same agent may also be paired for interaction. In either case, an interaction actually takes place only if the tag of one of the compartments chosen satisfies the interaction condition of the other compartment. If the condition of one compartment is satisfied but not the condition of the other, then the second may abort the interaction, say with some predetermined probability; if both conditions are satisfied, the interaction takes place unconditionally.

In earlier models, agent A could enforce an unilateral flow of resources from agent B, if A 'offense' string closely matched B 'defense' string, while B's offense poorly matched A's defense (for details, see Chapter 10 of Adaptation in Natural and Artificial Systems). This unilateral interaction was called 'combat'. The use of a pair of strings in each agent allows intransitivity in the evolving 'food webs': Agent A can "eat" agent B, agent B can "eat" agent C,
Figure 6. From Single-compartment Agent to Multi-compartment Agent. The basic evolutionary operator for increasing the complexity of agents is *intrachromosomal duplication*. When the intrachromosomal duplication (icd) operator is applied, a segment of the chromosome is doubled. If one or more compartment-associated on/off conditions have been doubled, the offspring produced at the next replication will contain a second copy of the compartments corresponding to the doubled on/off conditions. Subsequent replications may involve a mutation (or other modification) that provide a modification of the added compartment, yielding an agent with differentiated compartments. The resulting multi-compartment agent will be subject to resource-gathering selective pressures and will survive only if the new compartments somehow give it an advantage that compensates for the additional resources it must collect to copy the additional compartments.
but agent A cannot "eat" agent C. There was also provision for a bilateral exchange, called "trade", that involved a cooperative matching of another set of strings. It is clearly of greater interest if these different kinds of interaction can be unified, so that evolution can adaptively select the conditions and kinds of interactions an agent will undertake. Let's see how compartmentation makes this possible.

The flow of resources between compartments can be mediated by the same string matching technique used for "combat", but only the resources of the respective compartments are at risk now. (Remember that compartments will only interact if their respective interaction conditions are satisfied, so the flows are conditional on the interaction conditions). Under this restriction, the trick to unifying "combat" and "trade" is to make the flow dependent upon the match scores. The better the match between the offense string of y and the defense string of x, the higher the resource flow from x to y. Specifically, the flow is given by an S-function, with low match scores resulting in acquisition of some fraction of the surplus in the reservoir, higher scores drawing more from the reservoir than just surplus, and still higher scores acquiring a progressively higher fraction of the "structural" resources (resources tied up in the defining strings) of the other compartment, thereby "killing" it (see Figure 3). A bilateral exchange of surpluses between the reservoirs of x and y, "trade", results when their respective match scores are both low. A unilateral exchange of resources, "combat", occurs if there is a lopsided difference in the respective match scores.

In order to approach questions of emergence with a sparse set of mechanisms, several "shortcuts" have been used in the early versions of Echo:
(1) Details of metabolism and assembly of resources into compartmental structures are omitted. Once the resources are acquired, they are automatically assembled into the required structures with no attempt to simulate the chemistry involved. (By progressively adding resource transformation capabilities to compartments, under control of the chromosome, the evolution of metabolism can be modeled with increasing verisimilitude.)
(2) There is no resource cost in duplicating the chromosomes that provide a compartment's blueprints. The only "costs" associated with a longer chromosome are the costs incurred in duplicating the more complex compartmental structures they describe. (This can be easily modified by charging a resource cost for reproducing a chromosome, akin to the cost of recharging ADP in a biological cell. The ratio of this cost to the compartment reproduction cost then becomes a parameter of the system.)
(3) The activity of a compartment directly determines the activation of the corresponding gene. In real chromosomes the activation of genes depends upon complex metabolic feedbacks; here it is assumed that the activity of a compartment "stands for" the string of metabolic intermediates that it induces.

As in earlier models, the condition/tag combination provides for selective interaction. This, followed by the match score computation, makes possible a diverse range of interactions. We can design systems with counterparts of ecological phenomena (e.g., parasitism, symbiosis, biological arms races, and mimicry), economic phenomena (e.g., trading complexes, and the evolution of "money"), immune system responses (e.g., interactions conditioned on identification), and so on. It also easy to show that Echo subsumes a wide range of standard models such as the Prisoner's Dilemma, the Two-armed (n-armed) Bandit, Wicksell's Triangle, Overlapping Generation models, and so on. Compartments make possible still more complex organizations, starting with simpler primitives. And, with the addition of agents that have on/off genes in their chromosomes, it is possible to investigate counterparts of metazoan formation and evolution (e.g., emergent hierarchical organization).

While it is useful that Echo can be set up to imitate such systems, that is not its primary purpose. Rather, Echo is intended as a vehicle for studying the emergence and evolution of such complexities. Some of the most interesting simulations start with simple initial configurations, so
that complexities that appear later must perforce be a consequence of the evolutionary mechanisms supplied by Echo. When complexity increases in interesting ways under these conditions, we at least have an existence proof for the sufficiency of the mechanisms supplied.

3.6 An example.

A population-based version of the Prisoner's Dilemma provides a simple example of the kind of investigation that Echo should facilitate. In this version, there is a population of agents that come into contact via random pairings. At each contact, the condition/tag mechanism is used to determine whether or not an interaction takes place. If an interaction does take place, then the pair executes one play of the Prisoner's Dilemma according to the strategies implied by their respective offense/defense strings. The four possible outcomes (cooperate-cooperate, cooperate-defect, defect-cooperate, defect-defect) determine an exchange of resources. The rate at which a given agent collects resources determines its rate of reproduction and, ultimately, the relative frequency of its kind in the overall population. As the frequency of a given kind of agent increases, the frequency of interactions involving that kind of agent increases.

Earlier experiments with selective mating based on tags are relevant here. In those experiments, an early, accidental association of a tag with a trait conferring a reproductive advantage is rapidly amplified because of the higher reproduction rate of the tag's carriers. For example, such an advantage is conferred by a tag associated with "compatible" mates that produce fewer lethal offspring under crossover. The tag, originally meaningless, takes on a meaning. It comes to stand for a particular kind of compatibility. By developing selective mating conditions based on the tags, the agents can react to this compatibility, thereby increasing their fitness.

In the population-based Prisoner's Dilemma, tag amplification provides a way for agents to make useful distinctions. For example, an agent developing a condition that identifies tags associated with 'cooperators' will prosper from the increased payoff that results. As in the selective mating experiments, there is strong selection for combinations of tags and conditions that favor profitable interactions. In effect, the agents develop tacit models, anticipating the effects of interacting with agents having certain kinds of tags.

This selective process, in turn, opens new niches. For example, mimicry becomes possible; an agent can present a tag with an established "meaning", while pursuing a different course of action. The mimics, as in biological studies of mimicry, can only occupy a small proportion of the population relative to the agents being mimicked, because the other agents begin to adjust to the deception when the proportion of the mimics becomes large. Even in an "ecosystem" with the limited possibilities offered by the Prisoner's Dilemma, an interesting diversity of strategies and internal models arise.


There is a more sophisticated version of Echo that emphasizes the flow of resources resulting from the interactions between agents. By assuming a kind of "rapid mixing" of resources among like agents this version makes better contact with mathematical models and allows for much more extensive computer explorations. It uses an integrated two-tiered format: (1) The Upper tier specifies the evolution of the genotype/phenotype relations for the adaptive agents, where the agents have the compartment-based structure described earlier. The Upper tier models the 'slow dynamics' of the system. Some techniques associated with classifier systems can be extended to the study of this tier. From a mathematical perspective, there seems to be a generalization of the 'schema theorem' for genetic algorithms that applies to the Upper tier. This version of the schema theorem does not depend upon an exogenous definition of fitness.
(2) The Lower tier uses a matrix of flow coefficients to describe the transfer of resources between different kinds of pairs of interacting adaptive agents. The phenotypic properties of the Upper tier precisely determine flow coefficients of the Lower tier. The Lower tier models the 'fast dynamics' of the system. The recursions based on the flow-matrix are natural discrete generalizations of the Lotka-Volterra equations; as such they are susceptible to mathematical study.

In more detail:
(1) The mapping from the real cas to the Upper tier of Echo is from repeated structural features (building blocks) of the cas to generators and relations (resources, tags, rules, etc.) in the Upper tier. As a result the behavior of interest in Echo is generated and emergent, rather than being given by a table of all possibilities. This provides an insurance policy against 'eye of the beholder' interpretations, so that correspondences, even qualitative correspondences, are non-trivial, enhancing Echo's use as a platform for gedanken experiments.
(2) The Lower tier simulates the Upper tier, rather than directly simulating the fast dynamics of the cas being studied, because this offers a considerable advantage in revising the model as it is tested. Whenever the flow-matrix in the Lower tier fails to capture some phenomenon of interest, the nature of the failure can be examined systematically by altering parameters in the Upper tier as well as the Lower tier. Such detailed control of the hypothesize-test-control cycle is rarely possible within the real cas; it is typically impossible to repeatedly 'restart' a real cas from the same initial conditions. There is an additional advantage. Regions of the parameter space that prove interesting can be defined precisely, allowing the construction of flow-matrices specialized for analysis of those regions.

The two-tiered Echo model provides several advantages in formalizing the idea of an adaptive landscape:
(1) A fixed flow-matrix corresponds to a 'steady-state' ecosystem (an ecosystem without innovations). The flow-matrix makes it easy to represent the effects, on flows, of lags and carrying capacity. Mathematically there are strong connections to the study of Queued N-armed Bandits, and to r- and K-selection. The flow-matrix also provides measures of the integrity or robustness of subsystems. In the particular case of economics, the Echo models provide interesting relations between new markets, technological innovation, and r-selection, on the one hand, and saturated markets, market share, and K-selection, on the other hand.
(2) The agent-determined flows are specified rigorously so that the instantaneous form of the landscape allows for a 'fast-dynamics', e.g. oscillations like the Linx-Hare oscillation.
(3) Processes of evolution, co-evolution, etc., define a trajectory through the space of allowed flow-matrices, so that features of those processes are well-defined properties of the trajectory. It is possible to define both a 'schema theorem' for these trajectories and a notion of neighborhood based on an 'operator metric', making it possible to study some of the main determiners of the rate of evolution.
(4) Fitness is endogenous, varying rapidly in some contexts. With fitness so-defined, it is possible to study context-dependent changes in fitness, inclusive fitness, etc., for single genotypes and sets of genotypes (e.g., arms races). For subsystems, the net resource income per unit of resource sequestered, an interest rate, provides more information than fitness.
(5) Epistasis is determined by evolving phenotypic interactions and is subject to selection, making it possible to study selection for or against particular forms of epistasis.


Computer-based models are perforce caricatures of real systems. It is in the nature of modeling that certain features of the real system are emphasized, while details must be thrown away. This is no less true of mathematical models, though they often have the compensating advantage of generality. In both computer-based modeling and mathematical modeling, as in
cartooning, much skill and taste go into choosing what is to be emphasized and what is to be thrown away. And, again as in cartooning, the resulting caricature will only be informative if the choices point up essential characteristics. In modeling, unlike cartoons, there is the additional criterion that the results be protected from 'eye of the beholder' ambiguities.

Echo uses two techniques to avoid 'eye of the beholder' ambiguities. First, Echo is overtly designed to subsume specific models from other disciplines, such as Wicksell's Triangle and the Prisoner's Dilemma, in its larger framework. Because these specific models have already been closely examined for relevance in their respective disciplines, the larger framework inherits this relevance. Second, Echo uses a 'generators and relations' approach to modeling. Instead of using a long list of rules in an attempt to describe each action of the system being modeled, Echo models elemental processes and mechanisms. This approach avoids the "unwrapping" syndrome (where the "solution" is explicitly "wrapped" into the initial instructions of the program) that pervades much computer-based modeling. Echo, so-contrived, offers the possibility of studying interactions and generalizations of well-known special models in a larger, still rigorous, context.

The two-tiered version of Echo also provides some further protection from 'eye of the beholder' interpretation. It makes contact with extant mathematical studies, ranging from the Lotka-Volterra equations to the schema theorem of genetic algorithm studies. The relevant theorems, suitably adapted, offer guidelines as to parameter settings, and regions in the space of possibilities, that will prove interesting.

The design of Echo turns on the aspects of cas that are to be emphasized, with a focus on the evolution of diversity and internal models. The basic mechanisms center on the discovery of building blocks and the exploitation and elaboration of tags. The mechanisms for discovering and exploiting building blocks are modelled on the genetic algorithm's manipulation of schemata. The mechanisms for exploiting and elaborating tags are modelled on the manipulation of tags in classifier systems. Because tags control interactions between compartments, the only tags that persist over long periods are those that control useful interactions. That is, the tags in Echo provide a tacit model of the agent's world, as in the Prisoner's Dilemma example, implementing useful anticipations.

It is relatively easy to find counterparts of these mechanisms in other cas. Indeed the mechanisms were originally designed with such counterparts in mind, and the successful use of these mechanisms in earlier genetic algorithm/classifier system experiments bodes well for their use in Echo.

The central questions I intend to pose to Echo are the two questions concerning diversity and internal models. With the help of Echo's simulations I hope to (1) demonstrate that an increasingly diverse array of agents ('genotypes') incorporate certain building blocks discovered early on, a kind of 'founder effect' for building blocks, akin to the pervasive use of the Krebs cycle in organisms, and, using this information, I hope to (2) formulate a generalization of the schema theorem based on endogenous fitness. This version of the schema theorem would help to determine how 'innovations' in cas shift under the discovery of building blocks; it may show that combinations of building blocks providing innovation are close together in terms of the operator metric(s) defined by crossover, mutation, etc. The theorem should apply to building blocks for structural features, such as the compartment's offense and defense strings, as well as to building blocks for the tags that implement tacit models.

It is worth noting that the compartment-oriented version of Echo is suitable for studying the evolution of organizations, with the embryogenesis of metazoans providing a prime example (see, for example, the work of Buss).
under the constraint that it must not seriously damage the overall organization that assures the supply of critical resources, innovation takes some surprising turns. The sequestration of the germ-line in metazoans, or the genetic constraints that prevent the runaway reproduction we call cancer, are prime examples.

Among those who have carefully compared different cases, there is little doubt that they form a coherent subject matter. At the right level of abstraction, their mechanisms and processes can be given a unified description. Within this framework we begin to see common causes for common characteristics. Common characteristics such as diversity in components, perpetual novelty in behavior, exploitation of internal models, and persistent operation far from equilibrium, all seem to arise from similar mechanisms and processes. The challenge now is to provide a rigorous treatment of these observations.

The challenge is formidable because our traditional mathematical tools rely on linearity and equilibria -- fixed-points, basins of attraction, and the like -- features mostly missing from cases. Oscillations and recursive interactions are not features of linear systems, and the anticipations provided by internal models frequently destroy equilibria. To meet this challenge we need an unusual amalgam of techniques:

1. Interdisciplinarity. Different cases show different characteristics of the class to advantage, so that clues come from different cases in different disciplines.
2. A 'correspondence principle'. Bohr's famous principle translated to cases, means that our models should encompass standard models from prior studies of particular cases, not only to forestall 'eye of the beholder' errors, but also to assure relevance.
3. Computer-based gedanken experiments. Such models provide complex explorations not possible with the real system -- it is no more feasible to isolate and repeatedly "re-start" parts of a real case than it is to test "flameouts" on a real jet airplane carrying passengers -- suggesting critical patterns and interesting hypotheses to the prepared observer. Such experiments can also provide 'existence proofs', showing that given mechanisms are sufficient to generate a given phenomenon.
4. A mathematics of competitive processes based on recombination. Ultimately we need rigorous generalizations, something computer-based experiments cannot provide on their own. Mathematics is our most powerful method for attaining such ends. The mathematics needed must depart from traditional approaches, emphasizing persistent features of the far-from-equilibrium, evolutionary trajectories generated by recombination.

I believe this amalgam, appropriately compounded, offers hope for a unified approach to the difficult cases problems that stretch our resources and place our world in jeopardy.
References.