Artificial Worlds and Economics

David A. Lane

SFI WORKING PAPER: 1992-09-048

SANTA FE INSTITUTE

Artificial Worlds and Economics

David A. Lane

92-09-048

A working paper from the Santa Fe Institute Economics Research Program.

This research was supported in part by grants to the Santa Fe Institute, including core funding from the John D. and Catherine T. MacArthur Foundation, the National Science Foundation (PHY-8714918), and the U.S. Department o Energy (ER-FG05-88ER25054).

ARTIFICIAL WORLDS AND ECONOMICS

David A. Lane[*]
School of Statistics
University of Minnesota

0. Introduction

In this paper, I describe a class of models, called Artificial Worlds (AWs), that are designed to give insight into a process called emergent hierarchical organization (EHO). I argue that many economic phenomena seem to manifest EHO, and so economists might be interested in studying this process -- and in making use of AWs to do so. There are, however, some formidable inferential difficulties that will have to be overcome before AWs can become socially acceptable research tools.

The paper is organized as follows. Section 1 briefly describes EHO. Section 2 introduces AWs and some of their attendant inferential problems. Section 3 introduces two abstract AWs that address important general problems in EHO and then briefly describes an economic phenomenon, the coming into being of new industries, in which these problems appear to play a key role. Section 4 describes a particular kind of AW, classifier systems, that can be used to represent agents that are capable of generating complex behaviors in response to intermittent rewards from an "environment" of which they are a part. A collection of such agents, engaging in "economic" interactions with one another, produces another kind of AW, in which such interesting aggregate behaviors as the formation of bubbles and crashes and technical trading in an artificial "stock market", may arise. Section 5 considers the idea of an Artificial Economy -- an AW that can provide a dynamic, nonequilibrium, microfounded account of such aggregate-level or macroeconomic phenomena as stable growth paths, business cycles, and Pareto firm-size distributions.

[*] Work on this paper is supported by grants from the National Science Foundation and Citicorp. For many helpful conversations around the ideas presented in this paper, I would like to thank my colleagues Jim Dickey of the University of Minnesota and Brian Arthur, John Holland, Walter Fontana and Jim Pelkey of the Santa Fe Institute; my collaborators Franco Malerba and Luigi Orsenigo of Bocconi University and Giovanni Dosi and Marco Lippi of the University of Rome; and my students in the Artificial Worlds seminar at the University of Minnesota.

1. Emergent Hierarchical Organization

Many systems, in chemistry and biology as well as in human society, appear to have the capability of achieving, over time, a more and more complex organization. The process through which this organization is achieved, emergent hierarchical organization, typically displays two characteristic features.

First, the organization is hierarchical. That is, the systems are composed of a number of different levels, each level consisting of entities that interact with one another. Lower-level entities may actually be components of higher-level ones. The higher in the hierarchy is the level, the longer is the time-scale and the more extended the space-scale in which it is natural to describe the interactions between the relevant entities. For example,

- biological systems include entities and processes at levels ranging from molecular to cellular to organismic to ecologic;

- economic activities involve interactions between individual "decision-makers", firms and households, industries, and national economies.

Second, the systems appear to produce their own order. The actions of lower-level entities are channelled -- in effect, coordinated -- by higher-level structures that themselves arise from the lower-level entities' interactions. For example,

- informal trading networks transform into formally organized impersonal markets;

- neurons firing in response to sensory stimuli or the firing or other neurons with which they are connected produce predictable organism-level behavioral responses to particular patterns of environmental activity -- or may even give rise to action-guiding "concepts".

The order induced by this kind of hierarchical coordination is never static, since the interactions between higher-level entities change the environment in which lower-level interactions take place, and hence in the higher-level structures that develop out of them. Thus, the system as a whole is characterized by perpetual novelty at all its levels.

2. What are Artificial Worlds -- and What Might We Learn from Them?

Artificial Worlds are computer-implementable stochastic models, which consist of a set of "microlevel entities" that interact with each other and an "environment" in prescribed ways. AWs are designed so that they themselves may, under some conditions, *manifest* EHO. As a result, AWs represent an engineering approach to the study of EHO.

The entities built into an AW and their modes of interaction may be quite abstract, or they may be closely linked to objects and relations occurring in some real-world system of interest. In the former case, the AW may be used to investigate general principles underlying EHO, while in the latter the AWs may help us to understand how particular aggregate properties of the modelled real-world system depend on the characteristics of the lower-level processes that underlie them.

Formally, an AW consists of a set of *microlevel entities* (MEs), an *environment* and a *dynamic*. Each ME has attributes and modes of interactions with other MEs. The environment has a state.

When two or more MEs interact, their attributes may change. The changes are determined by the MEs' interaction modes. In addition, they may depend on the MEs' current attributes and the current state of the environment. Interactions between MEs can also change the state of the environment.

The dynamic, which may be in part stochastic, specifies the order in which interactions occur. The dynamic also imposes rules that determine when MEs die and when new ones come into the World (and with what attributes).

The initial conditions of an AW determine a state of the World: the state of the environment, a population of MEs, and the attributes of each of the MEs. These initial conditions, together with the dynamic of the AW, generate a *history* -- that is, a time-ordered sequence of states of the World. (With a stochastic dynamic, of course, the same initial conditions generate a probability distribution over a space of possible histories.)

The aim of AW modelling is to discover whether (and under what conditions) histories exhibit interesting *emergent properties*. An emergent property is a feature of a history that (i) can be described in terms of aggregate-level constructs, without reference to the attributes of specific MEs; (ii) persists for time periods much greater than the time scale appropriate for describing the underlying micro-interactions; and (iii) defies explanation by reduction to the superposition of "built in" micro-properties of the AW.[1]

For example, imagine an Artificial Economy in which MEs represent traders exchanging a set of commodities according to some prescribed rules that do not single out any particular commodity as a medium of exchange: the replacement of a barter system with the exclusive use of one of the commodities as a "money" would be an emergent property (see Section 4.5 below). Similarly, in an Artificial Economy in which some MEs produce machines for sale to other MEs who in turn produce consumer goods for sale to other MEs (who work for one or the other producer MEs), the evolution of a stable growth rate for "GDP", or of sector-specific Pareto-distributions for firm size, might be an emergent property (Section 5).

As these examples indicate, some emergent properties can be described in terms of variables that aggregate over the attributes of many MEs (like GDP), while others refer to "real" higher-level structures (like money). Both give evidence of self-organization in the AW -- coordination among the MEs induced by their interactions, leading to system meta-stability. More is possible: higher-level "entities" may arise. These entities are composed of sets of MEs that display coordinated patterns of behavior. They may even reproduce themselves (Section 3.2) and develop modes of interaction between one another (Sections 3.1 and 4.4), leading to even higher-level emergent properties. In such cases, the AW exemplifies EHO.

What can we hope to learn from AWs? We have to begin by considering "about what" we can learn. First, the AW itself might be the primary target of inference, and we might want to discover just which emergent properties it manifests, and how they depend on the system rules and initial conditions. Second, the AW might be regarded as a model of some real-world phenomenon in which we might be interested. In this case, we might want to determine whether (and if so, how) certain "lower"-level interactions in the real-world "cause" higher-level structures and processes to arise -- and how these higher-level structures and processes then change the nature of the lower-level interactions. Third, we might want to learn about EHO as an abstract phenomenon, investigating such questions as the following:

- What properties must a system have for EHO to occur?[2]

[1] Obviously, what "defies explanation" to one person may be explicable by another. What is required here is a negative assertion by the modeller, to the effect that the aggregate-level property in question is not deducible from the model's micro-properties by any argument substantially shorter than producing that property by running the model. I will discuss later some maneuvers that might lend "public" credibility to such an assertion. Notice that the modeller's assertion is not equivalent to the statement that he assigns low a priori probability to the property manifesting itself when he runs the model: after all, he may have other reasons than deductive argument for believing that systems with the micro-properties he built in to his model tend to exhibit aggregate-level regularities analogous to the property in question!

[2] See Kauffman (1990) and Rasmussen et al. (1990) for some interesting speculation on this question.

- Is there a taxonomy of possible forms of emergent organization? In particular, are all emergent organizational forms hierarchical?

- How do the properties of emergent higher-level entities and their interactions depend on the properties of the lower-level entities from which they arise?

- What kinds of interactions are possible between the levels of a hierarchically organized system? In particular, how autonomous are the processes of different levels? Under what circumstances can the evolution of a system process be predicted on the basis of observations only of the attributes of entities at the same level as the process (that is, without detailed information about processes at lower or higher levels)?

- What are the dynamical properties of emergent processes? For example, are "punctuated equilibria" (Section 3.2) generic?

While computer scientists might be interested in an AW for its own sake, economists presumably would study AWs in order to get insights into what might be going on in economies. Whatever the goal, to learn anything useful about any of the three inferential targets described above, we need strategies for designing appropriate AWs and for generating and processing useful data from them. There are some formidable difficulties standing in the way of this endeavor. I conclude this section by mentioning four of them:

<u>The need for computer-implementation</u>

AWs are well-defined mathematical models, but it is unlikely that interesting theorems about their emergent properties will be proved with tools currently available. I offer three reasons for this assessment:

First, AWs are designed to be innovatory or open-ended systems. Their emergent properties are only meta-stable, not equilibria or asymptotic states. By changing the environment of the lower-level entities that give rise to them, emergent structures induce processes leading to their own transformation (or demise). As a result, it will be difficult to apply the rich repertoire of mathematical methods that compute equilibria or asymptotic states, and there is no corresponding methodology for studying the properties of transient phenomena.

Second, emergent properties are necessarily complicated functions of the history of the attributes of the ME's from whose interactions they are formed (if this were not so, it would be easy to explain them by superposing the AW's micro-properties, and they would not qualify as emergent properties!). Since the dynamics of AWs are specified in terms of these micro-interactions, it is hard to imagine that the mathematical description of emergent properties will be analytically tractable.

Third, it seems to be a plausible (albeit ill-defined) hypothesis that the capability of a system to produce EHO is a function of its complexity, either in the attributes or arrangements of its component entities or in their patterns of interaction. As a result, the mathematician's ploy of constructing a highly simplified, tractable model that can be proved to display an interesting behavior observed in some more complicated system will not work in the context of EHO phenomena.

Thus, it seems likely that we will learn about EHO from AWs only by implementing them computationally and observing what happens. As a result, we can learn about their emergent properties only inductively, and our success in that enterprise will depend on our ability to develop appropriate statistical tools, for the design as well as for the analysis of "evolutionary" experiments.

<u>Identifying Emergent Properties</u>

The very nature of emergent properties makes it problematic for us, as observers of the AW, even to formulate them, let alone discover whether or not they in fact obtain. Emergent properties represent innovations in the organization of the AW, and, to describe them, a new vocabulary is required, beyond the modelling language used to express the attributes and interactions of the AW's micro-entities. After all, emergent properties cannot be compactly expressed in the modelling language itself -- and, by definition, they "defy explanation" in terms of the constructs of that language. So how do we develop the right aggregate-level language to define -- and guide our search for -- potentially emergent properties?

AWs that model a real-world system have a natural vocabulary to express potentially emergent properties: the language that describes higher-level patterns and structures observed in the modelled system. Some[3] of these higher-level constructs may suggest AW analogs that can be expressed as functions of AW histories, and the words that describe the real-world constructs may be appropriated to define these functions. Thus, the modeller can build a glossary that semantically links higher-level real-world constructs with particular functions of AW histories. Any real-world

[3] But certainly not all. After all, the modeller abstracts only a small subset of entities, attributes and interactions to incorporate into the Artificial World, and only those higher-level constructs for which it is meaningful to aggregate only over this subset can be translated as a function on Artificial World histories. The determination of which higher-level constructs are meaningful in the Artificial World -- and how -- can be an important exercise for understanding the meaning and role of these constructs in the real world system itself.

phenomenon that can be described by these constructs translates, via the glossary, to a candidate for an emergent property of the AW -- provided, that is, that it satisfies the metastability and "explanation-defying" definitional requirements. Candidates generated in this way might be described as "expected emergent properties" of the AW.

"Unexpected emergence" -- an aggregate-level coordination phenomenon in the AW unmotivated by any real-world analogy -- is harder to find. This is particularly troublesome for abstract AWs, which lack a natural real-world reference vocabulary. In fact, most of the work that goes into studying such AW models as Coreworld (Rasmussen et al., 1990), Tierra (Ray, 1992) and Function-Object Gas (Fontana, 1992 -- see Section 3.1 below) consists in poring over output, attempting to identify features that display the "right" kind of coherence and temporal stability -- and then formulating a vocabulary, with both mathematical and "natural language" variants, in which to express them. Whether this search can be in some way "automated" is an important conceptual and practical problem.[4]

Finding Conditions of Emergence

When potentially emergent properties have been identified and translated into the behaviors of appropriate functions on histories, the next question to ask is: under what initial conditions (and, for stochastic dynamics, with what probability) will they obtain? Developing strategies to answer this question is difficult, since the space of initial conditions typically has a very high dimension, and interesting emergent properties may well depend on complicated interdependencies among the system parameters that define these dimensions.

Moreover, the relevant search space is even larger, because it has a time dimension. Well-defining the function on histories that determines whether a particular property emerges requires a specification of how long that property must persist -- and this specification must always be somewhat arbitrary. In addition, whether a particular property emerges or not depends not only on initial conditions, but on the length of time the history is observed -- so negative results may just mean that longer observation times are required, not that the initial conditions are insufficient to support the emergent property in question.

Causality and Emergence

Suppose a potentially emergent property of an AW has been identified and defined in terms of some function of histories -- and, with some set of initial conditions, a history has been generated and the property obtained. What kind of claim can be made about what "caused" this property -- in particular, is it meaningful to think of emergence itself as a cause?

To interpret emergence as a cause, we mean to say that the property formed because of the interactions amongst a dense network of entities -- and this formation depended on the denseness of this network, and perhaps the richness of the structure of the entities and their interactions. Thus, it is not enough merely to produce the property in the AW from some particular set of initial conditions: that set would have to be embedded in a hierarchy of sets, ordered by a "complexity" measure that increased with the network's "denseness" and the structural "richness" of the MEs and their interactions. Emergence as a cause would then require demonstration that the property fails to appear for low values of this measure -- but does, beyond some threshold value.[5]

Such a complexity measure imposes a structure on the high-dimensional AW parameter space. Without this structure, it is hard to see how one could begin to infer about what causes emergent properties -- and it is equally hard to see how any causal inference could be made that is independent of the particular measure used to induce the structure.

Now suppose we know how to infer about emergence-as-cause inside the AW. Suppose further that we believe that a particular aggregate-level feature in the AW is indeed an emergent property, and we have determined how "complex" the AW needs to be in order to support the feature's emergence. Suppose in addition that this emergent property is semantically linked to some real-world higher-level pattern or structure: what can we infer about the "cause" of this feature in the real world?

At the least, we can certainly argue against the necessity of any alternative explanation that assigns a causal role either to other real-world aggregate-level features that do

[4] Bedau and Packard (1992) propose a statistic whose purpose is to diagnose the arrival of an "innovation" into an Artificial World. Their statistic seems to depend on a genotype-phenotype distinction: the microentities in the World are replicators, whose behaviors are coded by a genome; selection operates on the coded behaviors; innovations in behavior depend on the introduction of a new genotype; and successful innovations are marked by the initiating genotype's ability to persist in the population over time. The Bedau and Packard statistic tracks such persistence at the genomic level. But the generality of this approach seems questionable: not all higher-level innovations depend upon the persistence of single micro-innovations, even in biological evolution. To paraphrase the evolutionary perspective persuasively set forth in Buss (1989): on an evolutionary time scale, genotypes are transient, while phenotypic organization is here to stay.

[5] One might suspect that typically, as the complexity measure increases above this value, a second threshold might be obtained, beyond which the system again fails to manifest the property in question -- just as turning up the heat applied to the bottom of a beaker of fluid results first in the formation of convection cells and, at even higher temperature, their degradation into a regime of turbulence. See Kauffman (1992) and Langton (1992) for stimulating discussions on this theme.

not have analogs in the AW or to attributes of lower-level "agents" that are not possessed by the MEs of the AW. For example, an Artificial Economy in which, say, a stable growth path for GDP emerged from sufficiently rich patterns of micro-interactions would thus argue against the necessity of invoking the existence of Walrasian equilibrium to explain macro-coordination -- or against the proposition that such macro-coordination depended upon the assumption of optimizing agents capable of forming rational expectations.

But we would like to infer more than this. Can we argue that the real-world aggregate regularity is indeed "caused by" the entities and interactions we abstracted out of it and built into the AW, in which the analog of that regularity was identified as an emergent property? That is, can we infer emergence as a "causal mechanism" in the real world, once we have so identified it in the AW?

Certainly, the AW demonstration ought to raise our probability that such a mechanism operates in the real world, just as it diminishes the probability of alternative causal stories that credit features and attributes not detected or built into the AW. But the real world necessarily contains many more entities and interactions than the AW, operating at levels below, at and above that of the focal regularity. Surely, it is possible that the causal mechanism hinted at in the AW is swamped by the additional "turbulence" in the real world, and some entirely different sets of interactions or direct effects drive the formation of the feature of interest. It is not clear how to determine how plausible is this possibility -- but of course, the more specific one can be about just which additional interactions or effects might provide the alternative causal story, the more plausible it would appear to be.

3. Abstract AWs and the Lawfulness of EHO

In this section, I describe two abstract Artificial Worlds, Walter Fontana's Function-Object Gas (Fontana, 1992) and Kristian Lindgren's Evolutionary Prisoner's Dilemma (Lindgren, 1992). Function-Object Gas is directed primarily to an exploration of the relation between structure and function, Evolutionary Prisoner's Dilemma to the dynamics of evolutionary processes.

While much work remains to be done before AWs yield deep insight into these two themes, the themes themselves are fundamental to an understanding of many real-world processes. The section concludes with a discussion of an economic example of such a process, the coming into being of a new industry.

3.1 Function-Object Gas: Function and Organization

Function-Object Gas (FOG) is designed to explore how higher-level structure emerges from micro-level function. The notion of function on which FOG is based is abstracted

from chemistry. A chemical entity functions by acting on other chemical entities to produce new chemical entities. Similarly, in FOG, all interactions between MEs are of a single type: a ME A acts on a ME B to produce a new ME A(B).[6]

FOG also abstracts from chemistry the relation between structure and function at the micro-level. Which new entities are produced when chemical entities interact are completely determined by the structure of the interacting entities: the components from which they are built up and the way in which these components are arranged . Thus, a chemical entity is both a syntactic and a semantic object. Syntactically, it is built up from component objects, according to well-defined rules. Semantically, its "meaning" (that is, its function), coded by its structure, is revealed in the chemical reactions in which it partakes. The dual character -- syntactic and semantic -- of chemical entities is most striking in catalysis: the syntactic form of the catalyst is unchanged, even as it accomplishes its function of transforming the structure of other chemical entities.

In FOG, each ME has a syntactic representation in terms of more elementary components. This representation never changes during the lifetime of the ME. An ME's representation codes for its semantics, in that the representations of the interacting MEs determine the outcome of the interaction. That is, the representations of the MEs A(B) and B(A) can be "computed" from the representations of A and B, for every pair of allowable syntactic representations A and B.[7,8] In FOG, all interactions are doubly catalytic: neither A nor B is "destroyed" by their interaction. So, A+B -> A+B+A(B).

Thus, in chemistry and FOG alike, micro-level function is determined by micro-level structure. However, this is by no means the end of the function-structure story: micro-level

[6] The interacting entities are ordered: A(B) need not be the same as B(A). In addition, A(B) is not defined for all MEs A and B.

[7] Technically, this is achieved by using Alonzo Church's λ-calculus to represent MEs as λ-objects -- mathematical functions in intensional form, that act on other functions to yield new functions according to nine axioms of construction and syntactic transformation. Computationally, then, a λ-object is both function and data. The components of a λ-object are variable names, the abstraction symbol λ, and three structural symbols (period and left and right parentheses). The set of λ-objects are defined recursively by the three construction axioms: variables are λ-objects; if x is a variable and M an λ-object, then $\lambda x.M$ is a λ-object; and if M and N are λ-objects, so is M(N). The semantics governing function evaluation are incorporated in the other five axioms. The λ-calculus is computationally complete; every recursive function can be represented as a λ-object. See Barendregt (1984) for details.

[8] For λ-objects A and B, B is not in the domain of A if the computation implied by the transformation axioms applied to A(B) does not halt. In FOG, there is a limit placed on transformation steps, and any interaction whose associated computation exceeds this limit produces no product.

function can in turn give rise to *higher*-level structures. Consider an autocatalytic network: a set of chemical entities that (perhaps in the presence of some "food set") catalyze reactions among its members (and the food set), such that each member of the network is a product of at least one of these reactions. Thus, an autocatalytic network reproduces itself -- collectively, not necessarily individually. Take away some of its members, and an autocatalytic network may "disappear" as one after another of its members fail to be produced by reactions involving remaining members; while the removal of others of its members may not matter, as they are soon replaced from transformations among the "survivors". Thus, even though the functionality of a particular chemical entity may be latent in its structure, the organizations of chemical entities to which this functionality may give rise are really aggregate-level or population concepts.

To see how FOG can be used to address the problem of the emergence of higher-level structure from micro-level function, I first describe how to generate a FOG history. Start with a population of MEs[9] (λ-objects: see footnote 7) -- these are typically generated at random. Next, select a pair of these MEs at random, say A and B, and let them interact as described above. If the computation for A(B) terminates, add this ME to the population and select another ME at random and remove it from the population. This dynamic keeps the population size constant. Now iterate the interaction-deletion steps many times.[10]

The population of MEs in the FOG after many interactions may display structure at the syntactic or the semantic level. Syntactic structure refers to common features of the representations of the members of a set of MEs. For example, the set of λ-objects of the form $A_{ij} = \lambda x_1.\lambda x_2. \ldots \lambda x_i.\ x_j$, $j<$ i, exhibits syntactic structure.

Semantic structure depends on the production pathways involving reaction products from interactions between members of the set. For example, suppose A, B, C, and D are MEs, with A(B) = C, B(C) = D, C(D) = A and D(A) = B. Then, regardless of the other interactions of these MEs, the set {A, B, C, D} is *self-maintaining*, in that each can be formed from interactions between members of the set. (This property is analogous to the concept of an autocatalytic network). Note also that {A,B}, {B,C}, {C,D} and {A,D} are all *seeding sets*, in that the entire set can be reconstructed by interactions involving the elements in each of these subsets and their "descendant" products. A set that contains all of the products from interactions between set members is *closed*. Closed self-maintaining sets are *self-reproducing*.

Self-maintaining sets are not guaranteed to survive under FOG dynamics, since MEs are removed randomly from the population. Clearly, MEs that belong to a self-reproducing subset with several small seeding sets have a better chance of persisting in a population that contains that seeding set than does an ME that belongs to no such subset. One way in which a FOG population can display semantic structure is if it can be decomposed into a number of such self-reproducing subsets. These subsets in turn can have a variety of semantic structures, which may be represented by means of interaction graphs, as in Fontana (1992).

So far, there have been no constraints imposed on interactions in FOG, except for the upper bound on allowable computation time (see footnote 8). It turns out, however, that what higher-level structures form depends crucially on which interactions are allowed to take place. For example, some MEs may reproduce themselves (that is, A(A) = A) or other MEs (A(B) = B). Clearly, if the set of MEs reproduced by an ME A contains A, it is self-maintaining, in a trivial way. Fontana (1992) reports that, without constraints in interactions, FOG tends to organize around production pathways that end in an ME that reproduces every ME in the pathway. Starting with 1000 random MEs, after tens of thousands of collisions, the FOG population is typically closed and consists of one or more self-reproducing subsets, each with its own identity function.

Thus, to explore a greater range of interesting emergent structures in FOG, Fontana has begun to investigate what happens when he constrains the permissible set of interactions. He does this in two ways, which correspond to syntactic and semantic constraints. For example, barring copy reactions is a semantic constraint, since whether a reaction copies one of the reactants is a function of the interaction, not just the product of the reaction. In general, though, it is difficult to formulate semantic constraints. Syntactic constraints bar interactions that produce reaction products with specified structure. Thus, they amount to restricting the FOG population to particular subsets of λ-objects.

To determine which products to prohibit, Fontana has taken advantage of a peculiar finding: FOG tends to produce organization on *both* the syntactic and semantic level. That is, when the FOG achieves a metastable, closed population, this population exhibits patterns both in the structure of their MEs and in their production pathways. Thus, it is possible to prevent a particular semantic organization from occurring by prohibiting reaction products that have its corresponding syntactic features.

For example, when copy reactions are prohibited, families consisting of MEs of the form $A_{ij} = \lambda x_1.\lambda x_2. \ldots \lambda x_i.\ x_j$, $j< i$,

[9] There is no (external) environment in FOG.

[10] Note that with this dynamic, FOG interactions are "on average" singly, not doubly, catalytic, since A is removed from the system with the same probability as it is selected to form a product A(C), for all C in the population.

as described above, proliferate. Their syntactic structure is clear. Semantically, according to the transformation rules of λ-calculus, these so-called projection functions satisfy

$$A_{ij}\ (A_{km}) = A_{i-1,j-1}, \qquad \text{if } j > 1$$
$$= A_{k+i-1,m+i-1}, \qquad \text{if } j = 1$$

Thus, start with, say, A_{i1} : this ME acts on itself to produce $A_{2i-1,i}$, which then acts on itself (or on any other member of the family) to produce (in turn) $A_{2i-s,i-s}$, for s = 2,…, i-1. These i MEs form a simple semantic structure, organized around the cycle $A_{i1} \rightarrow A_{2i-1,i} \rightarrow A_{2i-2,i-1} \dots \rightarrow A_{i1}$. Note that any member of this cycle is a seeding set for the cycle. According to Fontana (personal communication, 1992), FOG without copy reactions organizes into one or more of these families, with transient random selection between families (and victory tends to go to the largest).

So the next organizational question to investigate is: what structures emerge when all MEs of the form A_{ij} are prohibited? Once these are discovered and their syntactic regularities are found, a further constraint can be imposed, and additional organization forms obtained. By continuing in this way, Fontana is uncovering a hierarchy of increasingly complex organizational forms that can emerge in FOG, under increasingly complex constraints on allowable interactions. He is attempting to associate with each of these forms an underlying algebraic structure that describes its interaction graph. The hope is that these structures will provide the basis for a mathematical theory of organizational form.

Another direction of current research with FOG is to search for the emergence of structures at a higher level than the sets of MEs so far described. For example, can self-reproducing sets interact with one another to produce other sets with some metastable structure? An interaction between sets of MEs can be defined trivially to produce the union of all the pairwise interactions between elements of the two sets. It is not clear that this is a useful definition; nor is it yet clear what a reasonable alternative might be. It may also be necessary to introduce noise into the system, for example by occasionally perturbing the structure of individual MEs or the products of their interactions. This may "destabilize" emergent organizations, especially those that involve many MEs with complicated production pathways, with the result that the system will support more, smaller structures that may support or inhibit one another through their mutual interactions. At any rate, EHO is so far a one-level phenomenon in FOG.

To conclude this discussion of FOG, consider an alternative way of building a computational system in which entities interact with entities to produce new entities. An obvious strategy is to decide how many entities you want to have in the system, say *n*, and then randomly construct an n-by-n lookup table that gives the products of all possible pairwise interactions. Representing MEs as λ-objects has two principal advantages over this "random lookup" strategy:

- The λ-based system is computationally open-ended.[11] You are not limited to any pre-fixed number of MEs, and you can represent any imaginable relation between MEs, since any computable function can be expressed as a λ-object.

- In the λ-based system, the representation of MEs codes for their function. Thus, it is possible to explore relations between structure and function that have no counterparts in the "random lookup" scheme. In particular, any syntactically correct expression or family of expressions can be inserted (or deleted) from the system and the effects on organization monitored. Put another way, the λ-representation provides a true genotype-phenotype[12] distinction -- and a way of experimentally determining which "genes" are responsible for which "body plan" characteristics.

On the other hand, experiments with FOG alone cannot tell us whether the structure-function relations that they reveal depend upon the λ-representation of its MEs. That is,. we need other arguments to determine whether the algebraic structures of organization that Fontana is discovering are general principles of emergent organization or merely artifacts of his model (and perhaps reducible to theorems in λ-calculus itself). These arguments must be inductive in character. Can these structures (and not others!) be observed in other systems, from real or Artificial worlds, in which functional interaction can be interpreted as the creation of new entities?[13]

[11] At least in principle; in practice, one must introduce constraints on the number of steps in a computation, the length of the representation of objects and so forth.

[12] Here a self-maintaining set of λ-objects represents the "organism", with the syntactic structure of each λ-object representing a gene. The phenotype is the (semantic) structure of the interaction graph of the set and its reaction products.

[13] Fontana and biologist Leo Buss are currently translating some organizational experiments with FOG into the language of evolutionary biology, with promising results. A paper on "Algebraic Replicators and Units of Selection" is forthcoming. In particular, they provide new interpretations of the significance of "life cycles".

3.2 Evolutionary Prisoner's Dilemma: The Dynamics of Evolutionary Processes

Evolutionary Prisoner's Dilemma (EPD) is a simple example of an evolutionary process. The leading natural example of an evolutionary process is, of course, biological, and it is far from simple. It is hard to think about biological evolution now without taking account of its rich organizational structure, in particular the hierarchy of descent (replicating genes, interacting organisms, evolving species -- and beyond) and the economic or ecological network, with its complex of relations between organisms, revolving around energy production and exchange[14].

The concept of evolutionary process on which EPD is based abstracts away from all this structure. It starts with the notion of an entity as a set of attributes. Entities are capable of self-replication: that is, they can produce other entities that have the same set of attributes as themselves. Entities with the same set of attributes form an entity type. The entities in an evolutionary process form a population, and the population consists of more than one entity type. Different entities replicate at different rates, so that the distribution of entity types in the population changes over time.[15] The probability that an entity replicates at any given time depends not only on its own attributes but also on those of the other members of the population at that time. Finally, evolutionary processes include mechanisms whereby entities with new kinds of attributes enter the population. Frequently, these mechanisms depend upon innovation-generating errors that take place in the process of replication.

Thus, evolutionary processes are characterized by replication (the reproduction of existing entities), selection (the differential replication rates of different entity types), and variation (the generation of new entity types). To determine a particular evolutionary process, it is necessary to specify the following elements:

- a set of entity attributes;[16]

- a fitness function (which may be stochastic) that gives the replication rate for each entity type, given the current distribution of entity types in the population;[17]

- variation mechanisms whereby new entity types enter the population; and

- an initial population of entities.

For example, in population genetics models used in theoretical evolutionary biology, entity attributes are typically defined at the genotypic level. The variation mechanisms include such genetic operators as mutation and recombination. The most problematic element in these models is the fitness function, since relative replication rates depend on the interactions at the phenotypic level. Thus, a genotype's relative replication rate is a function not only of how phenotype is determined by genotype,[18] but also of the kinds of ecologic relations that different phenotypes have with one another (competition, predation, symbiosis and so forth). These underlying processes are not at all well understood, and so it is impossible to derive the form of the fitness function from first principles. In contrast, if entities were taken to be organisms (or even species), the relevant attributes might be structural or functional properties that could be directly related to relative replication rates -- but then the variation mechanisms could be modelled only phenomenologically.[19]

The designer of an AW evolutionary process faces two difficult challenges: how to determine the fitness function for an arbitrary population of MEs, and how to create variation mechanisms that can supply new types of MEs indefinitely. Lindgren solved these problems, and also

[14] For introductions to the literature on hierarchical views of evolution, see Hull (1988, 1989), Salthe (1985), and Eldredge (1985).

[15] Entities may also leave the population, for example by dying.

[16] Note that if the process is truly open-ended, S is an infinite set.

[17] Note that the domain of the fitness function is not the set of individual entity types, but the set of possible populations of entity types. The process described here is coevolutionary: the fitness of each entity type depends on what other entity types share its world. In this sense, the population is an "individual", with entities as its "parts", which itself undergoes evolution. Thus, no "landscape theory" that fixes a "fitness function" over the set of entity types can describe the dynamics of the kind of evolutionary process I am defining here, since such a "landscape" is continuously deforming as the distributions of the entity types in the population change.

[18] Which may of course depend in part on what other genotypes are in the relevant population, since this determination is "environmentally" mediated -- and the other entities in the population form part of a given entity's environment.

[19] An alternative approach to modelling evolutionary processes begins by positing two different types of entities: replicators and interactors. Replicators have a fixed structure that can be exactly replicated; variation mechanisms then introduce new types of replicators. On the other hand, replicators do not interact directly with one another; interactors do. So selection operates on interactors. The key modelling problem in this approach is to relate the replicators to the interactors: in particular, how do the functional properties of interactors depend upon the structure of replicators, and how do the interactions between interactors determine the differential rates at which the replicators replicate? The answers to these questions determine the analog of the fitness function described in the text. Hull (1988, 1989) argues exhaustively and convincingly for this approach to modelling biological evolution. In EPD, the MEs (or strategies, see text) are both replicators and interactors.

provided a natural language in which to describe his AW, by building EPD around a version of Iterated Prisoner's Dilemma.

Each EPD ME represents a strategy for playing a two-person game with two possible actions (say, 0 and 1).[20] This strategy is the only attribute of the ME. Each generation, MEs interact with one another in a round robin tournament: every ME in the population uses its strategy to play a particular version of Iterated Prisoner's Dilemma against every other ME.[21] The MEs then receive their average reward from these encounters, and they replicate in such a way that the expected number of replicates of each ME is proportional to its average reward.[22] Thus, the fitness function is determined by the representation of MEs, via the pairwise interaction rule of the round robin tournament and the interpretation of the representation as a Prisoner's Dilemma strategy.

Variation in EPD arises from three kinds of replication error, each of which occurs with a fixed probability, independently for each transcription event. First, any given bit may be transcribed incorrectly (here the probability is per bit transcription, so the greater is the length of the string representing the ME, the higher the probability of replication error). Second, the string may get adjoined to a copy of itself, doubling its length (for example, "01" is incorrectly copied as "0101"). This error is particularly important, since it makes the set of possible MEs infinite, so that EPD is potentially open-ended. Because of the way in which strategies are encoded (see footnote 20), the offspring ME resulting from this error has exactly the same strategic behavior as its parent.[23] However, its doubled length means that it takes account of one more previous move than its parent does -- and a subsequent transcription error in any of its bits will give rise to a different kind of strategic behavior than could arise from any transcription error in the parent type. Finally, the string may be cut in half, with either half chosen at random as the viable offspring (for example, "1101" might be incorrectly copied as either "11" or "01").

EPD dynamics exhibit interesting emergent properties. First, a succession of stable ecologies -- that is, distributions of entity types that persist for many generations -- form, dominate the EPD population, and then degrade. Both the individual ecologies and their succession may be regarded as emergent higher-level structures. Each ecology may possess one of a number of possible organizational forms: some are dominated by a single entity type; some have several symbiotic or competitive dominant types; in others, the dominant role is distributed among a number of "quasi-species" that share some key features and differ in others.

Second, the periods of stasis or "quasi-equilibrium" in which a stable ecology persists are interrupted by shorter periods of destabilization, which also display certain characteristic features. During a destabilization period, the number of entity types in the population fluctuates rapidly. Frequently, these periods begin with a large "extinction", in which the number of entity types drops rapidly. It is also typical that the average reward that MEs receives drops during the destabilization periods. In EPD, there is no exogenous "environment", so all destabilizations are endogenously generated: that is, such phenomena as mass extinction and structural disintegration do not necessarily require exogenous causes (like asteroid collisions or volcanic eruptions!). Destabilization periods end with the formation of a new stable ecology, in which the leading entity types were not present (or present only at low frequencies) in the previous "quasi-equilibrium".

Contingency plays an important role in EPD ecological succession. While it is easy to compute which strategies have relative advantages over which, it is not easy to predict which sets of strategies will dominate the emerging stable ecologies. Start with the same initial populations, and quite different successions can occur. For example, starting with particular values for the system parameters (growth and error rates) and an initial population consisting entirely of memory 1 strategies, with probability[24] about 0.9 EPD will end up (by 30,000 generations) in an ecology

[20] Each EPD ME is a string of 0's and 1's of length 2^m, where m is an integer. The strategy encoding for the ME works as follows: write the last m moves (in reverse order: the opponent's last move, your last move, the opponent's next-to-last move,…); read what you have just written as a binary number; go to that coordinate of the your strategy vector -- and play the number you find there.

[21] The version has the following features: a) the play is noisy: that is, if a player's strategy dictates that he play a "0", say, he plays a "1" with probability p (p is small, and does not depend on the player or the history of the game); b) the payoff per play is as follows: if both players choose 0 ("defect"), they each win 1; if they both choose 1 ("cooperate"), they win 3; otherwise, the one who chooses 0 wins 5 and the one who chooses 1 wins nothing; c) the iteration is infinite, and the reward to each player in the iterated game is average payoff per play given above.

[22] In Lindgren's version of EDP, population size is kept constant and the proportion of each entity type in the next generation is proportional to its average reward. If the proportion of any entity type falls below 1/N, where N is the nominal population size, the entity type is dropped from the population. In effect, rather than setting the probability of replication for each ME to be proportional to its average reward, Lindgren substitutes the expected number of replicates per type. While Lindgren's version gains computational efficiency at the cost of failing to be a true evolutionary process, it shares the qualitative dynamical features described below with the truly evolutionary probabilistic replication scheme.

[23] For example, 0101 is the same strategy as 01, since its play depends only on the opponent's last move, regardless of its own previous move.

[24] These probabilities, as reported in Lindgren (1992), are of course obtained as frequencies over many runs of EPD.

dominated by many different memory 4 entity types that share common features in their representation (1xx10xxx0xxxx001): Lindgren argues that this particular ecology cannot be destabilized by the low-frequency introduction of any possible entity type. On the other hand, with probability 0.1, this ecology will not form, and the system will follow some other succession, leading to ecologies whose dominant types have memory lengths of 5 or greater.

These features of EPD dynamics -- a contingent succession of "quasi-equilibria" interrupted by "catastrophic" destabilization periods -- resemble the "punctuated equilibrium" version of the history of biological evolution, as put forward by Eldredge and Gould (1972).[25] Their appearance in such a simple evolutionary process as EPD suggests that they may be generic, at least in some very general subclass of evolutionary processes. An important goal for future work with abstract AWs is to try to discover the defining properties of this subclass and to gain a better understanding of punctuated equilibrium dynamics. What characterizes the set of possible stable ecologies? How large is the set? To which perturbations is a stable ecology robust -- and which destabilize it? Why are the destabilization periods relatively short-lived, compared to the "quasi-equilibria"? Why are destabilization periods frequently initiated by rapid mass extinctions -- and what endogenous mechanisms drive these events? What determines the order of succession of stable ecologies -- and which successions are contingent and which (at least conditionally on some predecessors) necessary?

I conclude this discussion by pointing out two important phenomena in biological evolution that do not arise in EPD but could be the targets of future AW research. To explore these two phenomena would require evolutionary AWs with more structural possibilities for higher-level organization than are present in EPD:[26]

- A key ingredient of the "punctuated equilibrium" story is that fundamental structural innovation seems to arise only in brief destabilization periods, not in the intervening "quasi-equilibria", in which various "implications" of the fundamental innovations are worked out. Most dramatically, all existing animal phyla (and many more, since lost) appeared in the Cambrian explosion, a period lasting less than two million years, over 500 million years ago (Gould, 1989). That is, biological evolution seems to produce big differences first, in quick bursts, and slowly fills in the details.

[25] Somit and Peterson (1992) contains a very interesting series of essays on the meaning and scope of punctuated equilibrium.

[26] An interesting evolutionary AW that addresses at least the first of these issues is Thomas Ray's Tierra (see Ray, 1992).

- In biological evolution, selection operates at more than one level at the same time. Thus, within organisms, cellular selection continues to occur (for example, cancers represent successful selection at the cellular level that can be fatal at the organism level); and, at the same time, higher level entities -- like colonies, species, or even ecologies -- compete for resources, reproduce themselves and generate new attributes that lead to new colonies, species, or ecologies. The coexistence of all these processes constrains the structure and direction of each of them.[27]

3.3 Economics, EHO and Abstract AWs

In general, abstract AWs are designed to study processes whereby higher-level structure emerges from lower-level functional interactions. The two abstract AWs described in this paper, FOG and EPD, focus on two different aspects of these processes: the characterization of types of structure that can arise as a function of constraints on allowable interactions; and the dynamics of emergent structure. Clearly, far more exploration of both of these themes, by these and other abstract AWs, must be carried out before we can expect to gain useful insights into the lawfulness of EHO processes. Once obtained, such insights will serve as a background against which it might be possible to understand what is generic and what particular to real-world processes in which these themes appear to play a role.

Here I offer an economic example of such a real-world process: the coming into being of a new industry.[28] This process is central to economic growth and development. The point is not that we can apply Fontana's and Lindgren's investigations to learn anything interesting about this process. Rather, I want to call attention to those of its features that appear to exhibit EHO and to argue that these features are fundamental to understanding what the industry comes to "be" and to "do". Furthermore, the most interesting questions that arise about the process in my description involve precisely the themes that FOG and EPD were designed to investigate.

The emergence of industrial structure

I begin by sketching what I mean by the structure of an industry. An industry can be described in two complementary ways. First, the industry can be identified with the set of products that it produces. These products are related to each

[27] According to Leo Buss (1989), the two phenomena are related: the bursts of structural innovation coincide with the emergence of a new level of entity, which has successfully developed mechanisms that control the selection processes operating on its component entities so that they do not favor variants that are harmful to the larger entity of which they are a part.

[28] The formulation of this process, sketched here, is described in detail in a forthcoming paper by the author, Franco Malerba and Luigi Orsenigo.

other functionally, by the uses to which they can be put, and technologically, through the processes by which they are made. These two kinds of relations induce a structure to the industry's product set.

An industry's product set changes over time, as new products and ways to make them are developed. Since new products may come from the modification of existing ones (or their production processes), products also are related to one another by descent. Descent relations induce a hierarchical structure on the product set, with higher-level "taxa" defined in terms of successively more remote "common ancestors". As is the case in biology, the members of higher-level families of products also may share attributes, for example, functional complementarities (such as computers that share software) or similar production processes (so that expertise accrued in making one of the family carries over to making others).

The second way of describing an industry is as a collection of economic entities or "agents". These entities have a variety of structural relations with one another, all oriented towards developing, making and exchanging products in the set described above. At least six classes of entities enter into these relations: producers, demanders, suppliers, financiers, scientists, and governments. While the industry has an organization induced by the relations between its component entities, these entities themselves (firms, universities, research centers, regulatory agencies) have internal structure as well. Thus, an industry exhibits hierarchical structure. For example, a firm may have subordinate divisions -- marketing, production, R&D -- and may also belong to a superordinate entity like a research consortium or a trade association.

The entities that make up an industry and the kinds of relations between them also change over time, as a result of the interactions between the entities. Thus, the industry's organization is an emergent phenomenon. Consider, for example, the case of biotechnology.[29] By 1975, research funded by NIH and NSF and carried out by scientists working in the biomedical centers of several American universities had resulted in the development of recombinant DNA and hybridoma technologies. With financing obtained initially from venture capitalists (a relatively new kind of financial entity, swollen with profits from prior investments in microelectronics), some of these scientists set up new firms designed to exploit the economic possibilities of the new technologies. There were some formidable obstacles to be overcome, especially in product selection and development and "scaling-up" production volume.

Lured both by the promise of the technologies and their potential competitive threats to existing products and production methods, some older, established firms explored a variety of techniques to acquire proficiency in the new technologies -- ranging from research contracts with individual scientists and their universities or with the new biotech firms, to buying into the new firms, to setting up in-house biotech R&D units. The most active of these established firms were pharmaceutical companies, which had long-standing ties to the research centers where the new ideas originated and thus were well positioned to appreciate their implications; and companies with experience in fermentation techniques, which were crucial to "scaling up". The background and competences of these firms played a key role in reinforcing the orientation of the new technologies towards medically-related products and, later, extending them to agricultural products. By the mid-1980's, the interactions between the new research-oriented firms, the pharmaceutical companies, the chemical companies with expertise in fermentation, the venture capitalists, the universities, and the government regulators had produced a distinctive organization of "biotechnology" entities, with a burgeoning (if still largely prospective) product set.

Connections between entities take many forms. Of course, some of the interactions between entities take place in impersonal markets. But many more involve direct and longer-lasting relationships. Pharmaceutical and chemical companies fund university research, place representatives on the boards of smaller, research-oriented companies, send their in-house researchers to scientific meetings. Producing firms carry out extensive market research into the needs and preferences of current and potential customers and use special price and service incentives to consolidate long-term relationships with suppliers and buyers. Competing firms cooperate in various research initiatives, form consortia to jointly produce particular products, work together through their trade associations to lobby legislatures and develop international markets for their products.

Industry structure is then the totality of the connections between the economic entities that make up the industry. To understand how an industry develops, this structure matters, for at least two reasons:

- Not everyone knows how to do everything. The competence to perform economic tasks is embodied: particular entities have acquired skills, particular ways of doing things, through experience and over time. It is not generally possible to transfer these skills without immersion in the experiences that gave rise to them. To solve new economic tasks, like those that arise in the early days of a new industry, it is necessary to patch together solutions to old problems, as embodied in the entities with the requisite skills. That is, new economic tasks requires new entities, which consist of old entities connected in new ways. For example, the research-oriented biotechnology firms combined the technological skills of the university researchers with business plans put together under the auspices of the venture

[29] For an excellent analytic account of the emergence of the biotechnology "industry" through 1985, see Orsenigo (1989).

capitalists -- and when these firms developed products, they formed partnerships with older firms that embodied competences in production, marketing and regulatory management.

- To decide what to do next -- what new products to make or how to improve production processes -- a producer has to ferret out opportunities, which requires knowledge outside the producer's current competence. That knowledge is embodied somewhere else -- in the tastes or experiences of users of the industry's products, in the theories or experiments of scientific researchers, in the factories or design studios of competitors. And the knowledge can be obtained only through the connections that already exist between the producers and the entities that embody it. Without the mutual experiences that arise from these connections, it is not even possible to conceive of what one needs to know about. So who is connected to whom (and how) determines in part what directions will be explored and how those explorations proceed.

Thus, the process whereby new industries come into being links two interdependent processes, both of which can be viewed as evolutionary in the sense described in Section 3.2. The first takes place in the product set; in it, technological and functional relations between existing products give rise, through the interactions of different kinds of agents, to new products. The other occurs in the set of agents, amongst whom new connections create new structures that embody the solutions to the economic problems posed by developing, making and using the industry's new products. The kinds of structure to which these linked processes can give rise and the dynamics by which they do so ought then to be fundamental objects of economic inquiry. Abstract AWs can provide an important modelling tool in this enterprise, particularly by shedding light on what is peculiarly economic about these evolutionary processes.

4. Classifier Systems: Modelling Agents that Learn

In neoclassical economics, agents are modelled as rational actors. In this section, I consider a different approach to modelling agents, and I describe an AW, John Holland's classifier system, that realizes this approach. I then briefly discuss two ways to use classifier systems to "populate" AWs that are expressly designed to study economic phenomena.

4.1 Rational Actors or Agents Who Learn?

The concept of rationality that underlies neoclassical economics is a particular method for handling the problem of choice. In any given choice situation, rational actors are supposed to know what they want and what it is possible for them to do. While they may be uncertain about what will happen as a result of their possible actions, they know what all the possible consequences are, and they understand how what they get depends on what they do. They are rational, because they choose to do that which gets them (at least in expectation) the most of what they want.

There are of course many ways to criticize this concept and its applicability to "real" economic agents. How can agents know all these things? Is it plausible (one might say, "physiologically justifiable") to suppose that, even if they did have the requisite knowledge, they would have the ability to compute which act has the highest payoff? And even if they could act rationally, is there any evidence that real agents in fact behave in this way?

Here, I want to start with a question more fundamental than any of these: can we really come to understand economic action by examining "choice"? We -- you and I and economic agents -- are immersed in a continuous, ever-changing stream of information, partly received as signals from the outside world by our sensory apparatus, partly generated internally (and recursively) in response to this stream. Before we can "choose", we have to select out a small part of all this information and "attend" to it. Then we have to recognize, on the basis of the information to which we attend, that we face a choice situation. Finally, we have to formulate all the ingredients that choice situations require: what we want to have happen, what options we have, who the other relevant actors are, what consequences we can expect. Only at this point does a methodology for handling choice -- rational or otherwise -- become relevant.

Thus, our actions, even those that are based upon choice, depend upon acts (of attention, category formation and conceptual organization) that logically precede choice and cannot, without creating infinite self-referential loops, be subsumed under any choice-based theory of action. To found economic theory on a choice-based theory of action implies that the processes that produce "pre-choice" acts are irrelevant to what happens when agents actually get down to the business of making their choices. Or to put it more precisely, it implies that the actions that economists wish to study will be the same, however (and by whomever, the modeller or the "real" agent) these unmodelled processes are carried out.

Suppose, on the contrary, that these processes matter, in the sense that the kind of economic behaviors in which agents engage depend upon the way in which they learn to recognize and structure choice situations -- or even that, through these processes, agents come to develop certain behavioral repertoires (for example, "organizational routines", as in Nelson and Winter, 1982), without benefit of "choice", in contexts that neoclassical economists simply misidentify as "choice situations". Then, the descriptive and explanatory power of economic theory would be seriously compromised by the very definition of the nature of the agents it takes as its subjects of analysis.

An alternative is to base economics on a learning-based theory of action. An agent in such a theory lives in an "environment", which might of course include other agents. Structurally, an agent can be thought of as a set of sensors, a processor, and a set of effectors. The sensors determine which states of the environment the agent is able to perceive, and the effectors which actions into the environment the agent can perform. The sensors transmit their perceptions to the processor; on the basis of these, and a set of internal states that it maintains, the processor sends instructions to the effectors that result in actions.

The key to learning is the notion of a "reward", which the agent receives intermittently from the environment. The "aim" of the agent is to act in such a way to receive an increasing quantity of this reward. The agent accomplishes this aim by building up and refining a repertoire of actions that tend to lead to reward. The instructions for these actions are coded in the agent's processor as particular sequences of transitions of internal states, triggered by particular patterns of perceived environmental states. A learning-based theory of action describes how this coding takes place and how the code is stored and executed.

In contrast to a choice-based theory of action, a learning-based theory directly models the transformation from information-stream to actions. That is, all the mechanisms that process the information stream on the basis of which the agent is assumed to act are handled internally to the theory. In principle, agents in such a theory could learn to "choose" -- but the theory would be responsible for describing how the agents identify situations in which they regard choice as appropriate, how they organize what they perceive about the environment into the ingredients of a problem of choice, and how they develop the methodology that they apply when they go about the act of choosing. In other words, "choice" might arise as an emergent property in a world (Artificial or not!) populated by learning agents.

To provide an adequate basis for economic theory, a learning-based theory of action should provide a representation of agents and environments rich enough to support such characteristic features of economic behavior as the following:

- The theory should be able to model complicated, changing environments, since real economic agents engage in complex interactions, exchanging and transforming many kinds of commodities (and information). In the course of these interactions, the agents can perceive much more than they can "cognize". The more restricted is the information that a learning-based theory allows its agents to perceive in their environment, the less possibility there is for understanding the processes whereby economic agents actually come to know their complex worlds and their relation to it.[30]

- It should be possible to interpret the internal states of agents in the theory so that the agents seem to progressively "model" their world: that is, to generate broad categories that describe the world, to develop plausible hypotheses about the relationships between these categories (in particular, those that suggest actions likely to produce reward), and to refine these categories and hypotheses on the basis of increasing experience.

- Agents should be able to build up behavioral repertoires that include chains of actions that are initiated long before the agent obtains the reward they eventually yield. The capacity to build up these chains and then to act on them in the appropriate circumstances gives an "outside view" meaning to strategic behavior, since that is how the resulting behaviors might appear to an outside observer, regardless of the internal process by which the agent acquired them.

- Agents should be able to develop the capacity to plan future actions on the basis of their expectations of what the consequences of these actions will be. This capacity provides an "inside view" meaning to strategic behavior. If a theory admits the possibility of strategizing, without building it in, then it can be used to explore the very interesting questions of when agents actually engage in strategic action and how they come to do it, especially in comparison to the answers given by economic theories of choice.

4.2 Classifier Systems: An Introduction

In a classifier system, the agent is essentially just a collection of basic cognitive units, called classifiers.[31] Each classifier integrates perception, categorization and action. A classifier monitors the world, on the watch for a particular constellation of perceptible features. When this

[30] While Bayesian decision theory can be interpreted as a learning-based theory of action, from this point of view it is quite restricted, since Bayesian agents can only process environmental information about which they have already "cognized" their opinions (as to its form and probability). Moreover, Bayesian decision theory requires that the categories that agents use to construct their world, their prior opinions about these categories, and their procedures for changing opinion and taking action on the basis of opinion be "hard-wired" into the model. As a learning theory in the sense described here -- as opposed to a theory of choice -- this "hard-wiring" has no prescriptive (and certainly no descriptive!) justification.

[31] See Section 4.3 for a formal definition of a classifier -- the functional "definition" given in this paragraph (as a circumstance-specific behavioral propensity) is sufficient for the remainder of this section .

constellation is perceived, the classifier "proposes" that the agent take a particular action.

There are no consistency requirements on the classifiers of which an agent is comprised. Thus, propensities to act in different, even contradictory, ways, can coexist inside an agent. The notion of an agent as a bundle of possibly inconsistent behavioral propensities is a far cry from the rational prototype of economic theory, whose internal consistency is guaranteed by a probability distribution over all possible states of the world, well-defined preferences encoded in a utility function, and a single principle of action: maximize expected utility.

There are, however, some advantages to a conception of agents as inherently inconsistent. First, it is clear that requiring consistency imposes great computational costs on a system, as it entails a lot of internal structure and frequent consistency checking amongst different structural components. Second, since the world is always more complicated than our personal experience, maintaining consistency in an agent's behavioral or conceptual system almost necessarily requires a reduction in the agent's range of possible action, in particular in response to novel situations. Finally, there is overwhelming evidence that we humans do in fact maintain overlapping and inconsistent conceptual systems and associated behavioral propensities[32] -- perhaps because we are the products of an evolutionary process that rewards behavioral flexibility and is constrained by computational cost.

An agent that maintains inconsistent behavioral propensities has to have some mechanism that determines on which of these propensities it will actually act. After all, the world itself provides certain kinds of consistency conditions for behavior: you cannot move forward and backward at the same time. In classifier systems, this mechanism depends on a number that is associated with each of the agent's classifiers, its strength, which registers the "memory" of how well the classifier has served in the past in the agent's quest for reward. When different classifiers propose contradictory actions in the same circumstances, the agent tends to act upon the one that has the greatest strength.

So far, I have described an agent in a classifier system statically, as it exists at a particular point in time. But agents learn, and learning means changing. Classifier system agents learn in two ways: the strength associated with each classifier changes with experience, and old classifiers with low strength are replaced by new ones.[33]

[32] See, for example, Lakoff (1987), especially chapter 18, and Holland, Holyoak, Nisbett and Thagard (1986), which presents the learning-based theory of action underlying classifier systems, along with supporting arguments from psychology and philosophy.

[33] It is worth noting that "learning" in classifier systems has a quite different meaning than it does in rationalistic theories like Bayesian learning theory. In the rationalistic view, the world is composed of definite objects, properties and relations, and "learning" is the process whereby an agent forms a mental model of the world that correctly describes these features. Learning in classifier systems is about acquiring circumstance-specific behavioral propensities that function together to produce reward. That is, the agent is learning how to act in the world, rather than how to describe it. In the process, the agent may or may not develop descriptive categories, causal theories and so forth; and even if he does, there is no presumption that these categories and theories match some objective features "out there", nor would their worth to the agent depend on whether or not they did so. See Winograd and Flores (1986) for an extended critique of rationalistic learning and decision theories.

In order that useful classifiers increase their strength over time, the mechanism that changes classifier strength must in effect identify actions that lead to reward -- not just those that produce reward directly, but also those that "set the stage." Holland introduced his "bucket brigade" algorithm to solve this problem. The bucket brigade changes classifier strengths in two ways. First, any classifier whose action is implemented passes some of its strength to its immediate predecessors -- that is, the classifiers that proposed actions immediately preceding its own, which helped produce the constellation of features that triggered the classifier to propose its action. Second, the strength of classifiers whose action is implemented when the agent receives reward is increased as a function of the reward received. In this way, chains of action that culminate in reward can in principle build up: initially, the last classifier in the chain gains strength with the reward, which is passed back link by link as the chain is repeatedly executed. And as each classifier in the chain augments its strength, the sequence of actions proposed by the chain as a whole becomes more and more likely to be executed in the appropriate circumstances -- see Section 4.4 below.

Two kinds of mechanisms are required to carry out the operation of replacing old classifiers with new ones. The first determines when replacements take place. It is desirable for mechanisms of this type to recognize situations in which the agent "needs" new classifiers. For example, some mechanisms proposed in the literature trigger replacement when the world presents features that no existing classifier recognizes, while others introduce new classifiers that serve to link the actions of pairs of classifiers that have been activated in sequence.[34]

The second type of mechanism constructs the new classifiers, and here it would be desirable for the new classifiers to plausibly improve the prospects for the agent to obtain reward. For this purpose, Holland proposes the use of genetic algorithms, which build new classifiers by

[34] The intuition behind this so-called Triggered Chaining Operator is that, logical fallacy to one side, sometimes "post hoc" is "trying" to imply "propter hoc"!

combining parts of existing high-strength classifiers.[35] The idea, based upon an analogy with the success of sex (that is, meiotic genetic recombination) in biological evolution, is that useful classifiers work because they are composed of good "building blocks", either in the features of the world that trigger them or in the actions they recommend -- and that trying out new combinations of these building blocks is more likely to produce useful new classifiers than is any kind of random search through the space of possible classifiers.

At first sight, an agent in a classifier system seems quite disagreggated. One might well wonder whether such an entity could possibly display attributes that we usually associate with the human beings or institutions that function as agents in an economy. For example, will a classifier system agent appear to outside observers to have an "identity" -- that is, to manifest predictable behavioral regularities over broad categories of circumstances? Can a classifier system agent develop "points of view" -- internal models of the world in which it functions? If the answer to either of these questions is "yes",[36] then it would be reasonable to regard these properties as emergent phenomena in classifier systems, driven by the ability of agents' learning mechanisms to induce a "match" between the agents and their world that endows agents with a coherence that is in no sense "built-in".

Indeed, a classifier system can be interpreted as an evolutionary process, with classifiers as replicators. The replicator dynamics are given by the bucket brigade, with relative strength representing relative frequency of replicators of each classifier type, while the genetic algorithms function as variation mechanisms. In this view, the agent is an evolving population of replicators -- but selection of replicators is a function of their joint effects, through actions carried out at the agent level. In this respect, the classifier system agent is similar to the population of strategies in Lindgren's EPD, and it is perhaps then not so surprising that structure and coherence at the level of the agent should evolve, or that they should manifest themselves in the agent's behavior.

In section 4.4 below, I will review some evidence that, at least in relatively simple instances, classifier systems can exhibit such emergent phenomena as chains of linked behaviors and "mental models" that categorize and provide causal explanations for features that appear in the agent's world. Indeed, with a little additional structure, classifier system agents may even engage in a form of strategic planning.

[35] See Goldberg (1989) and Booker, Goldberg and Holland (1989), both of which provide good introductions to the literature on genetic algorithms.

[36] See Section 4.4, where I argue that both these questions may be answered affirmatively.

Like the λ-calculus, classifier systems are computationally complete. In addition, they have two particularly desirable computational efficiency properties:

- None of the processing algorithms -- classifier activation, bucket brigade and replacement algorithms -- impose heavy memory requirements on the system. All the information that has to be retained about classifiers are included in their representations and their strengths. It is not necessary to "remember" any descriptions of the circumstances in which a classifier proposed action or what happened as a result; in particular, no information about the joint actions of classifiers is maintained by the system.

- Much of the information processing in all the processing algorithms can be carried out in parallel. For example, in classifier activation and the bucket brigade, each classifier acts as its own "processor", to determine whether the feature constellation it monitors obtains and to pass on strength to its predecessors respectively. Similarly, in the principal genetic algorithm, pairs of classifiers undergo recombination independently of one another.

4.3 Specifying a Classifier System

Formally, a classifier system is a discrete-time AW that models a learning agent and the environment in which the agent lives. The current state of the environment is represented by a vector, one component of which registers whether or not the agent receives any reward in the current period -- and if so, how much. The other components code for various features of the agent's world, as perceived by the agent. Since the environment is represented as the agent perceives it, the agent's sensors are implicitly modelled in terms of the features registered in the environmental state vector.[37]

The state of the environment changes according to a specified dynamic, which may depend upon current and past environmental states and the agent's current action. Since this dynamic describes how the agent's actions change the environmental state, the agent's effectors are also implicitly modelled, through their actual effects.

The agent's processor is represented by two structural features: a *behavioral repertoire*, which determines the set of possible actions the agent can take, and a *message board*, which records the agent's current internal state in the form of a list of messages. The behavioral repertoire consists of the set of all the MEs in the classifier system. These MEs are called *classifiers*. Each classifier consists of two

[37] The agent is presumed to have access to the current state of the environmental vector. As a result, the question of the fidelity of the agent's perception does not arise here. Of course, this issue has to be confronted when the modeller constructs the environmental state dynamic.

symbol strings, the *condition* and *action* strings (say A and B respectively), along with a label identifying the classifier and the value of a numerical attribute, the classifier's *strength*. All classifiers have the same number of symbols in their representation, say n.[38]

The classifier (A,B) is interpreted as a behavioral rule: IF the conditions specified by A are satisfied by the current state of the environment and at least one of the messages currently on the message board,[39] THEN take the actions specified by B. There are two kinds of action: external actions, which change the state of the environment, and internal actions, which send a message to be posted on the message board.[40]

The contents of the message board in any given period consist of a list of messages sent to it in the previous period, along with the label of the classifier that sent the message. Each message on the list either does or does not satisfy any particular classifier's condition string. It is both computationally necessary and makes good modelling sense to assume that there is an upper bound to the number of messages that can be posted in any period on the message board.

Which actions the agent actually takes in a given period depends upon the current contents of both the behavioral repertoire and the message board, as follows. First, each classifier's condition string is checked against the current environmental state vector and the messages currently on the message board. Note that this checking operation can be carried out in parallel, with each classifier "processing" each of the available messages (including the "message" coded into the environmental state vector) and determining by which, if any, of these it is satisfied. The classifiers that are satisfied by at least one of the messages are then eligible for activation (see footnote 39).

Next, a subset of the eligible classifiers are selected for activation. It may not be possible to activate all of the eligible classifiers, for two reasons. First, the action strings of different classifiers may dictate changes to the environmental state vector that are mutually inconsistent. This inconsistency has to be resolved in some way in order to assign a definite value to the environmental state vector for

[38] Generally, the bits represent values of the sensors, internal states and effectors. If these values are binary, the symbols come from the set {0,1,#}, where # in a condition is interpreted as "don't care" -- that is, disregard the feature represented by any bit whose value is #. The *specificity* of a condition string is the number of non-# symbols it contains.

[39] In most implementations, each classifier actually has two condition strings, and both must be satisfied before the classifier's action is eligible for execution. In this way, internal and environmental conditions may interact to trigger particular behavioral responses (hence the conjunction "and" in the text).

[40] A single classifier may produce both types of action.

the next period. Second, if more classifiers send messages than the message board can hold, some mechanism must determine which of these messages get posted. In either case, a competition between the relevant set of eligible classifiers determines which of them actually execute their actions. The rules of the competition vary from one implementation of classifier systems to another, but the basic idea is always the same: the greater is the strength of a competing classifier, the more likely it is that its action will be executed.[41]

The strength of a classifier whose action is executed can change in three different ways. First, it must pay for this right with a (fixed) fraction of its strength. This payment is distributed amongst those classifiers whose posted messages in the previous period satisfied the conditions of the winning classifier. Second, if after all the acts are executed, the environmental state indicates that the agent has obtained a reward, then this reward is shared out to increase the strength of each of the winning classifiers.[42] Third, if classifier posted a message on the message board, it has the chance to gain strength in the following period, in the form of payments from those classifiers whose conditions its message turns out to satisfy (and which themselves win the right to execute their actions). In addition to these "bucket brigade" strength changes, some implementations of classifier systems impose a small strength tax every period on all classifiers in the behavioral repertoire, in order to expedite the replacement of useless classifiers.

Implementations of classifier systems use a variety of different mechanisms to replace low strength classifiers. Most systems replace a fixed fraction of classifiers at regular intervals. Generally, classifiers are deleted with probability an inverse function of current strength. In addition, as described in section 4.2, it is possible to add event-dependent triggering conditions and algorithms for new

[41] Typically, the competition is probabilistic, with the probability that a given eligible classifier will be selected proportional to some increasing function of its strength. Sometimes, other attributes of the classifiers besides their strength affect their probability of selection. These include specificity (the number of different features of the environment or internal state that are "checked" by the condition string) and support (the number of different messages on the board that satisfy the classifier's condition string). Both of these measure, though in different ways, the extent to which a particular classifier is "tuned" or adapted to the particular circumstances of the agent and the environment. The more specific is the satisfied condition, the more the classifier "exactly fits" the particular situation; while high support indicates a fit of the classifier to other behavioral elements in the agent's repertoire.

[42] Note that each classifier whose action is executed receives a share of the reward, whether or not its action had anything to do with the agent's obtaining the reward. The sorting out of "causal" from noncausal actions takes place statistically, over time.

classifiers tailored to the triggering events. For example, when no classifier's condition is satisfied by the current set of messages, the so-called Cover Detector Operator constructs a new classifier whose condition string is satisfied and whose action string is chosen in some random way.

The most important general algorithm for constructing new classifiers is recombination, a genetic algorithm. To implement this algorithm, select two "parents" from the current behavioral repertoire, with selection probability proportional to strength. To produce two "daughter" classifiers, first copy each parent. Next, choose two position indices (integers between 1 and n inclusive), and exchange the symbols in the copied classifiers between these two positions. The two classifiers that result from this operation are the daughters. One of the two daughters is then chosen as the replacement classifier, with strength initialized as some function of the strengths of its parents.

In summary, to construct a classifier system, one must specify the following ingredients:

- symbol string representations for the environmental state vector and for classifiers;
- a dynamic for the environmental state vector;
- versions of the activation, bucket brigade, and replacement algorithms;
- an initial population of classifiers.

4.4 Classifier Systems: Emergent Properties

At the end of Section 4.1, I listed some criteria for an economically useful learning-based theory of action. In particular, the agents in such a theory ought to be able to construct "mental models" of their world, to build up repertoires of temporally-linked behaviors that culminate in reward, and to plan future actions based upon expectations of consequences. In this section, I discuss some work that suggests that classifier systems can produce emergent properties satisfying each of these criteria.

Categories and default hierarchies

The world presents itself to us as a ceaseless succession of sensory stimuli. To form our mental models of the world, we have to endow it with a set of objects, properties and relations, in terms of which we reason, develop causal hypotheses, plan our actions. The process whereby we construct this set, from the raw material of sensory stimuli and the changes in our subcognitive "internal states" they trigger (and, recursively, from the elements we have already constructed), is a process of category formation.

How we form categories, and what structure the resulting categories come to have, are difficult and important psychological and philosophical problems.[43] At first sight,[44] it might appear that the categories we use to describe the world are "natural" -- that is, they merely reflect the structure of the world itself and hence can be defined as a set of necessary and sufficient conditions on perceived "states of the world".[45] However, there are deep reasons why the idea that there can be a simple isomorphism between the structure of the world and our mental models of it fails[46] -- and with it, the classical conception of categories as mirrors-of-the-world. In addition, there is abundant evidence that the categories we use to describe the world are not reducible to sets of "states of the world" and in fact exhibit complex internal structure.[47] As we shall see, these features also characterize the principal structures that represent categories in classifier systems, default hierarchies.

A default hierarchy (DH) is a set of classifiers, whose condition strings differ in their specificity.[48] The most general classifiers in the hierarchy establish "default" values for the category that the DH represents. These values may be overruled or modified by some of the more specific classifiers at the next level of the hierarchy -- and so on, down the hierarchy.

For example, consider a category that we could call "things to avoid". A DH representing this category might include a

[43] See Lakoff (1987) for a stimulating survey of recent research on the process of category formation and its profound psychological and philosophical implications, many of which challenge the foundations of neoclassical economics. Lakoff stresses the importance of the experiential and biological bases of categorization. His analysis is supported and extended by the evolutionary and neurophysiological arguments of Gerald Edelman (see Edelman, 1992, for an introduction and references).

[44] Which, in the history of philosophy, lasted a long time -- from Aristotle to Wittgenstein!

[45] Note that if categories actually had this structure, then any category X could be expressed as a simple disjunction (over different states of world) of rules of the form "If [state A] then [category X]" -- and thus would be directly expressible in a classifier system.

[46] See, for example, Lakoff (1987), chapter 15, for a discussion of Putnam's Theorem, which establishes the internal inconsistency of objectivist semantics.

[47] For example, our categories typically display prototype effects -- that is, some instances of a category are consistently regarded as more "typical" or "central" than others (for example, a robin is a more central member of the bird category than is a penguin; and blue is a more central color than is violet). Prototype effects are inconsistent with the classical conception of a category as a set of objects with membership criteria defined by necessary and sufficient conditions on some attribute set.

[48] That is, the number of non-# symbols in their condition-strings; see footnote 38.

general classifier of the form "IF a large vehicle is moving towards you THEN turn and move quickly out of its path".[49] This classifier establishes a good general policy for avoiding traffic accidents on city streets. However, if you want to travel by bus, it might be good to have an exception rule of the form "IF you are waiting at a bus stop and a bus moves towards you THEN wait where you are". These classifiers might in turn be supplemented by the even lower-level exception rule "IF you are waiting at a bus stop and a bus moves towards you and fails to slow down THEN turn and move quickly out of its path."

In a default hierarchy, even though the component classifiers may contradict one another, they actually work together to define complex categories efficiently. Good general classifiers benefit the system of which they are a part, because they cover many possible situations and produce an appropriate action for most of them. On the other hand, more specific exception classifiers, which generate better actions in the situations their condition-strings match, can accumulate high strength. As a result, they tend to win bidding competitions in these situations against the general classifier whose action theirs contradicts, particularly when bidding rules favor classifiers with higher specificity. This does two good things for the system. First, it leads to appropriate actions in these situations. Second, it protects the valuable general classifier from losing strength, by preventing it from winning bidding competitions and consequently paying out strength, in situations where it is unlikely to gain strength by producing an action that results in reward. As a result, it is possible for all the members of a DH to maintain relatively high strength values, which increases the probability that the DH will persist inside the classifier system. Thus, for example, without having to maintain a lot of specific rules that cover every imaginable interaction between you and large vehicles, a two-rule default hierarchy will still allow you to avoid being run over and to catch the bus when you need to.

Notice that DHs represent categories implicitly: the "meaning" of the category is distributed among all the classifiers that make up the DH representing it. This fact has two important consequences. First, it is possible for a category to "function" but have no "name" -- that is, no way to refer to it inside the classifier system. Second, the "meaning" of categories can change over time. New classifiers are always being generated by the system's replacement operators, and some of these new classifiers will function interactively (competitively or cooperatively) with classifiers in the DH. As a result, the "meaning" of the category represented by the DH will change, in the sense that new situations will be recognized as instances of the category or new actions will be taken when certain instances are recognized. Of course, such changes in the "meaning" of categories corresponds to experientially-based learning by the agent that the classifier system represents.[50]

Categories can also be referred to explicitly in classifier systems, through the use of tags. A tag just corresponds to a particular symbol substring, for example "001" occurring at the 6th through 8th position of a condition or action string. Tags can name categories. Using tags, the system can support classifiers that recognize the category named by a tag, say B (IF state A THEN #B#, where #B# is a string with the tag B and all other positions "don't care"), and others that take appropriate action when the system has recognized the category (IF aBc THEN action C, where aBc is satisfied if some B-recognizing classifier has posted its message the previous period, and perhaps some additional conditions, represented by a and c, are met). In this way, categories can link directly to other categories, and "abstract" mental models can be represented in the classifier system. Because they require specialized subsets of classifiers for recognition and response, tagged categories have more complex structure than their untagged counterparts, but DHs are equally suitable for both representation tasks.

So far, I have described how categories can be represented in classifier systems. The question of real interest, of course, is different: will DHs that represent categories actually emerge? In general, this is hard to prove: not only must the classifier system produce the DH -- but we, as observers, have to recognize that it did so! In one of the most impressive of the relatively few studies addressing this critical question, Riolo (1989a) provided a strong case that, in some circumstances, DHs in fact emerged in a particular classifier system.

Riolo investigated the performance of a classifier system that detects 8-bit binary vectors and must determine which of four categories they belong to. The highly nonlinear function that determines the "real" categories is, of course, unknown to the system. The system is rewarded whenever it

[49] Note that this classifier links the recognition of a category instance to an appropriate behavior. This pragmatic orientation is a general feature of classifier system categories. A classifier system supports categories not just to "name the world", but because it has to act in it. Those categories that help the system obtain reward are the ones that are reinforced and consequently persist and ramify.

[50] A third consequence of the distributed "meaning" of categories represented by DHs is that they share many of the attributes that recent psychological research has established for our categories. For example, not all instances of a category represented by a DH have the same membership status, since different instances trigger different classifiers in the representing DH, with different ensuing xding competitions and outcomes. As a result, for example, some kinds of instances may always be recognized as belonging to a category, while others may be accorded membership sometimes and sometimes not. Hence, classifier system categories give rise to prototype effects -- see footnote 46. For many more examples, see Holland, Holyoak, Nisbett and Thagard, 1986.

achieves a correct classification.[51] Starting with a random set of 100 classifiers, Riolo's system was able after 30,000 trials to correctly classify the input vector over half the time. By "interpreting" the high strength classifiers in the system and analyzing their interactions, Riolo was able to show that the classification was accomplished by means of categories represented by DHs. It is worth noting that some of the interactions between the general rules and their exceptions in some of these DHs were subtle; expressing the categories represented by these DHs in English -- clearly an inappropriate language for this world -- would be quite difficult!

Riolo's work not only provides strong evidence that DHs representing categories can in fact emerge in classifier systems, but it also gives some insight into conditions under which this is likely to happen. For example, bidding rules that favor more specific classifiers turn out to be necessary to maintain DHs in the system (Riolo, 1987b), as do provisions that prevent the random removal of high strength classifiers (Riolo, 1989a). Whether DHs will emerge in more complex classifier systems like those that might be used to model economic agents -- and whether we will be able to interpret the categories they represent if they do -- remain questions for future research.

<u>Classifier chains and strategic action</u>

Viewed from the outside, agents -- firms, chess players, urban racoons foraging for food -- appear to act strategically when they carry out a sequence of separate but linked actions that culminate in a favorable outcome. Seen once, the action sequence might appear coincidental. The more it recurs in circumstances which turn out similarly well for the agent, the more we would tend to regard the agent and the behavioral sequence as "strategic" -- particularly if we had seen the agent initially respond in different ways to the kinds of situations that later trigger the sequence, then engage in bits and pieces of the sequence, and finally put it all together and repeatedly act it out in the appropriate circumstances.

Classifier system agents are capable of this kind of strategic action. Linked chains of classifiers can emerge, such that each successive classifier acts to bring the system closer to reward and also sends a message that triggers the action of the next classifier in the chain. The system can maintain these chains -- and they can be assimilated into longer sequences, that either begin temporally even further from the eventual reward, or end by producing even more reward for the system (see Wilson, 1985, Riolo, 1987a, Robertson and Riolo, 1988, Riolo, 1989b).

For such chains to emerge, the classifier system must be equipped with special bidding rules and replacement operators. In the next several paragraphs, I describe how links between classifiers are accomplished in classifier chains and review some of what is known about the conditions under which classifier chains can emerge.

The links in a chain of classifiers are forged by means of tags. To see why tags are necessary, consider the following example. Suppose A_1, A_2, A_3 are classifiers of the form "IF the world is in state S_i THEN change it to S_{i+1}" ($i = 1,2,3$). Moreover, suppose that the classifier system gains reward when the world is in S_4. Then, if the world starts out in S_1, and our three classifiers fire sequentially, the world ends up in S_4 and the system gets reward, which all goes to A_3. What about the other two classifiers, who set the stage for A_3? So far, they get nothing. In fact, they lose, because in order to execute their actions, they have to win bidding competitions and pay out their winning bids to their "suppliers".[52] Thus, this "chain" (so far unlinked except through its "function") will not last very long.

Now suppose we modify these classifiers in two ways. First, suppose A_1 and A_2, in addition to changing the state of the world through the system's effectors, also post messages in the form of tags, say B_1 and B_2 respectively. Second, suppose that A_2 has the tag B_1 and A_3 the tag B_2 in their condition strings.[53] Through these tags, A_1 is now a "supplier" of A_2 and A_2 of A_3 -- and as a result, strength is paid out down the chain, from A_3 to A_2 and A_2 to A_1. In this way, reward won by A_3 will eventually result in an increase to the strengths of both its predecessors.[54]

[51] Note that a perfect solution to this problem is possible -- obviously, with 256 separate classifiers, one for each "state of the world"; not so obviously, with only 17 classifiers, by taking advantage of the structure of the encoding function. As such, it is a very different world than the one we -- or interesting modelled economic agents! -- inhabit. Clearly, the larger is the set of classifiers in Riolo's system, the less incentive there is to achieve a "compact" categorization by maintaining general rules in a DH. Conversely, the smaller the classifier set, the more "pressure" on the system to organize its categories efficiently -- and so the more likely it is that DHs might emerge.

[52] In this simple case, their suppliers are just the detectors, which posted messages identifying the successive states of the world. It turns out in general a bad idea to pay out bids to detectors, for reasons perhaps clear from this trivial example!

[53] Formally, this requires that they have two condition strings, as described in footnote 39 above.

[54] Higher-level representations of chains can also be achieved by tagging an entire chain, in addition to the link-by-link tags described here. This sort of tagging essentially makes the chain into a category -- organized diachronically, in contrast to the synchronic categories described above. In this way, the system can refer to the entire chain, "mobilizing" it into action as a unit -- or terminating its execution before completion, should circumstances warrant. Research has not yet been carried out on the conditions under which such higher-level structures can emerge.

Evidence suggests that tagged links between classifiers are unlikely to develop by chance -- at least at a fast enough rate to generate functionally useful classifier chains (Robertson and Riolo, 1988). As a result, "strategic" classifier systems require a special mechanism, called a Triggered Chaining Operator (TCO), to bind together classifiers that plausibly might function usefully as part of a classifier chain. The TCO creates a pair of coupled classifiers in the following way. First, it selects a classifier B that was active and experienced a net strength gain in period t -- and another classifier, A, that was active in period t-1. So far, there is no relation between A and B; the idea, however, is that A's action in period t-1 <u>might</u> have helped set the stage for B's strength gain in the next period. So the TCO creates two new classifiers, **A** and **B**, which are joined by a tag (in the action string of **A** and the condition string of **B**), but otherwise identical to A and B. Through the tag, **A** triggers **B**.

If there is any benefit to the system in the connection between A and B, A is essentially volunteering its services -- while **A** is paid for its efforts by **B**'s bid. As a result, **A** has a much better chance of staying around, and the system does better as a result. On the other hand, if there is no benefit to the connection, **B**, with its additional triggering requirement of **A**'s tag, will not fare well relative to B and will soon disappear from the system -- and without its income from **B**, so will **A**.

The story is not complete yet: why should **B** prosper relative to B? After all, B is triggered whenever **B** is -- and sometimes when it is not. What if there are other favorable situations for B -- or if B is only active after A? In either case, **B** has no obvious advantage over B. If the action of A or some other stage-setter is necessary for B's success, then the overall system suffers if **B** (and other linked versions of B) cannot supplant B. B is free-riding on the stage-setters it requires for its reward, and as their strength declines, so does the system's opportunities for reward -- not to mention B's as well.

This is a serious problem, because unlinked classifiers like B have to precede linked classifiers like **B** in the system -- they are the building blocks out of which long chains are constructed. Thus, by the time the linked versions come onto the scene, via TCO, the unlinked versions are already established and tend to have relatively high strength. Unless classifiers like **B** have some bidding advantage over classifiers like B, the prospects for building up long action chains is not bright. Several solutions to this problem have been proposed in the literature. For example, it is possible to bias bidding competitions towards classifiers that appear to fit the general context better -- that is, are supported by a greater number of the messages currently on the message board (see footnote 41). This gives **B** an advantage over B whenever **A** is active, since **A** supports **B** but not B.

Riolo (1989b) presents striking evidence that these mechanisms work. He worked with a classifier system designed to negotiate its way through a 4-level, 16-state feed-forward network, with reward possible only when the system entered one of the four possible fourth-level states. Without TCO, the system learned how to proceed from the third-level states to the fourth-level reward state, but essentially nothing more. With TCO and two mechanisms designed to solve the "free-rider" problem described in the last paragraph,[55] the system performance greatly improved, through the emergence and maintenance of effective classifier chains that guided the system from each first-level state of the network to the payoff state.[56]

<u>Lookahead and strategic planning</u>

Suppose some classifier system agents actually had the capabilities described in the last two subsections. That is, they could form categories that described useful features of their world, and they could generate chains of actions that tend to culminate in reward. Would such agents count as rational actors, in the sense described in Section 4.1?

To answer this question, first recall how rational actors decide what to do. They begin by recognizing that they are in a choice situation. Next, they determine their possible courses of action and forecast the consequences of each of them. Finally, they choose the course of action that promises them the most favorable outcome.

Now, how might our idealized classifier system agents behave? With their ability to categorize, they could come to recognize situations in which it would be appropriate to activate one of a number of their chains of actions. Moreover, they would "choose" which of these chains to activate on the basis of the relative strengths of the

[55] One was the inclusion of support in bidding, as described in the text. The other was a modification in the way the genetic operators deleted classifiers when adding new ones to the system -- only classifiers that had submitted a bid in the current period were eligible for deletion. This modification served to limit the number of copies of unlinked "free-riding" classifiers, which otherwise tended to swamp the newer linked versions in bidding competitions.

[56] Negotiating the network with transitions chosen at random yields an average score of 150 per three-transition trial; perfect performance yields the maximum payoff per trial of 1000. With TCO, the system achieves an average score of around 400 after about 3000 trials, and does not improve much thereafter. With the modified bidding and genetic rules described in the text and in footnote 55, the average score jumped to over 600 after 3000 trials and then continued to increase, reaching nearly 800 after 12000 trials. One additional modification, designed to "encourage" the formation of chains that traverse less frequently encountered paths achieved further improvement, producing an average score over 900 after 12000 trials. This last modification extends the idea of the Cover Detector Operator, described in Section 4.3.

chains' constituent classifiers, which of course reflect the agents' experience of the chains' relative benefits-- and hence their "expectations" of the chains' benefits in the current situation.

These two descriptions sound quite similar. However, they mask a key distinction in the way in which the two kinds of agents form their expectations of benefit from a particular course of action. Classifier agents look backwards, since they base their expectations on classifier strengths, which aggregate over every past experience with the relevant action. In contrast, rational actors look forward: starting from the current state of the world, they envision what consequences will follow if they take the action under consideration -- until they reach some end-state whose value can be determined. Typically, they have to consider more than one end-state for each possible action, since the future depends on other contingencies than the current state and the rational actors' actions. When this is the case, they evaluate the benefits of taking a particular action by averaging over the benefits of the different possible consequences of that action.

This might not seem to be such a big difference, since, after all, rational actors have to base their scenarios for the future on what they have learned from the past. But this is misleading: the difference between forward- and backward-looking strategies is indeed profound. In particular, the forward-looking strategy requires two capabilities that our classifier system agents so far lack:

- Rational actors need to generate explicit predictions of future states of the world, and to do so they must have methods for storing information about how these states have changed in the past in response to their own actions. In contrast, classifier system agents do not predict future states of the world explicitly. Moreover, it is not clear how classifiers that were designed to predict would survive, since a classifier's strength accrues only with respect to its action's usefulness in obtaining reward, not with respect to how well it predicts changes in the state of the world.

- Rational actors have the capability of operating in a "putative mode", in which they run their predictive models counterfactually to discover what the future may hold for them (that is: IF the world were in state A THEN take act B and the world would enter state C; IF the world were in state C THEN ...). In contrast, in classifier systems there is no putative mode: every action always results in a real change, either to the state of the world or to the systems' internal state; and any action changes the strength of the classifier that proposes it, through the system's bidding rules.

Both Holland (1990) and Riolo (1990) have recently shown how to design classifier systems that can operate in the putative mode. Such systems require surprisingly few modifications of the basic classifier system architecture. Moreover, all of the required modifications can be carried out without violating the fundamental principles of classifier design, including parallel processing and local memory storage.

In order to express what happens in the putative mode, two representation problems have to be solved. First, it is necessary to distinguish between "putative" and "real" states of the world. Riolo solves this problem efficiently, with a single-bit tag attached to all messages that refer to states of the world; the value of the bit establishes in which of the two modes, putative or real, the state is meant to obtain. Second, the system must support classifiers that express predictions of future states of the world. Here, Riolo uses classifiers of the following "P+A" (prediction + action) form: IF the world is (or were -- depending on the current mode!) in state X THEN take action Y and the world will be in state Z. In the real mode, this P+A classifier specifies an action (Y) and predicts that the next state of the world will be C, which may or may not turn out to be correct. In the putative mode, however, the classifier generates a putative action (Y again) -- and determines the next putative state of the world (C).[57]

Next, the system needs rules that determine when it enters and leaves the putative mode. For example, Riolo's system enters the putative mode in response to messages about the state of the world received from its detectors,[58] and it leaves it either when some action reaches a threshhold level of support or when "too much" time has elapsed (in which case its next "real mode" act is determined by a bidding competition). In particular, if a sufficiently well-supported action is available, the system simply acts -- otherwise, it goes into the putative mode to "decide" what to do.

Once in the putative mode, the system can use P+A classifiers to explore the future consequences of its currently available "real mode" acts. Here, the crucial design question is how to choose between these acts on the basis of this exploration. The difficulty is that the world does not provide reward in the putative mode. There are a variety of possible solutions to this problem. For example, the system could predict in which states of the world it will obtain reward, and then use a simulated reward as a basis for its choice. The system must then of course distinguish

[57] Clearly, it is important to supplement the system with replacement algorithms that generate a sufficient supply of such structured classifiers when needed. See Riolo (1990) for examples.

[58] And recognized by special classifiers that post messages declaring "putative" states of the world, in response to messages from detectors or from "prediction" classifiers. The preceding footnote also holds for this type of classifier.

between classifier strength that reflects *real* reward and a local, putative mode strength augmented by simulated reward.[59]

The final key design problem is how to make the system predict accurately. For example, Riolo's system must keep track not only of how useful P+A classifiers are, but how well they predict what happens next. Strength serves for the first of these tasks, but for the second, Riolo introduced yet another quantity associated with each classifier. This quantity, which measures the classifier's predictive effectiveness, only pays attention to the prediction part of a P+A classifier's action. It is updated every time the classifier is activated in the real mode, by averaging its past value with an indicator variable that is 1 if the prediction is correct, 0 otherwise. A P+A classifier's bid (in either mode) is an increasing function of *both* its strength and its predictive effectiveness.

Riolo (1990) provides convincing evidence that these modifications work. In three different task domains, the modified system comes to learn enough about how its world behaves to predict the consequences of its actions and to plan its actions accordingly -- and, finally, to achieve a high level of task performance. Thus, at least with respect to problems of the complexity of maze-learning and navigating around obstacles, classifier system agents appear capable of strategic planning -- and hence rational action. Moreover, in Riolo's experiment, some interesting cognitive features emerge. For example, the more the system masters a particular task, the less time it needs to spend planning how to carry it out. That is, in these systems, planning is a response to unfamiliarity and its consequent uncertainty. Given a sufficiently regular world, the nëed for planning is self-limiting.[60]

[59] Riolo uses a version of this idea, in which the predicted simulated reward is represented implicitly. He associates a second, "local" strength with each classifer. In contrast to "real" strength, "local" strength changes in the putative as well as in the real mode; its value in a particular execution cycle of the putative mode reflects the "real" strengths of the classifiers it activates, directly or indirectly. The support that determines which "real" action to take depends on the "local" strength of the classifiers associated with each of the possible actions. In the putative mode, then, "local" strength is passed from classifier to classifier by a variant of the usual bucket brigade algorithm. In addition, the "local" strength of unused classifiers is adjusted so that it comes to reflect the classifier's "actual" strength.

[60] This is consistent with a large body of psychological research contrasting the performance of human experts with novices. For example, expert clinicians typically generate *fewer*, not more, conjectures in the course of a diagnostic consultation than do medical students or residents, and they use less data to distinguish among the conjectures they do generate. From this point of view, the function of the "putative mode" activity of strategic planning is to facilitate the process whereby agents come to generate the categories and associated chains of action that guide their "real mode" behaviors. When the

As both Holland and Riolo acknowledge, there is a lot of room for improvement in their system designs. For example, the systems' predictive capabilities do not yet have much statistical sophistication. In particular, in Riolo's system, every prediction is about the "entire" state of the world and is either exactly right or exactly wrong -- neither very useful features in a complicated world. In addition, it would be very desirable to construct a classifier system that could function in more than one task domain and develop predictive models based on appropriate categories for each of them. Again, particularly for economic modelling, there ought to be more than one kind of "reward" that agents can obtain from their environment, and it would be good if they could develop the ability to establish preferences amongst these different rewards. But these opportunities for improvement should not obscure what has already been accomplished: classifier systems represent an approach to modelling agents in which agent "identity" and even rationality can reasonably be regarded as emergent properties.

4.5 Economic Modelling with Classifier Systems

In this section, I will briefly describe two AWs that apply classifier systems to economic problems. The microentities of both of these AWs are classifier systems, which represent economic agents. Thus, these AWs have a built-in hierarchical structure, since their microentities are themselves AWs. Interesting properties emerge at both levels in this hierarchy: as a result of their interactions with each other, the individual classifier system agents come to take on coherent "identities"; and an ecology of agents forms, with aggregate-level pattern and structure.

Repeated Wicksell triangles and the emergence of money

Marimon, McGratton and Sargent (1990) constructed an AW based on classifier system agents, which implements a multi-period Wicksell triangle economic environment introduced by Kiyotaki and Wright (1989). This environment is populated by three different types of agents, who produce, exchange and consume three different types of goods. Each of the agent types can produce exactly one type of good (different for the different types) and gains positive utility only by consuming a single, different type of good (again different for different types). Thus, trade is necessary to satisfy wants in this environment.

At the beginning of each period, each agent holds one good. Agents are randomly paired with one another, and each agent must make two choices. First, he must decide whether or not to exchange his good with the one held by the agent with whom he is paired. If both agents decide to trade, an exchange takes place. At this point, each agent decides whether or

process is successful, explicit planning is no longer necessary -- "plans" are implicit in organized actions.

not to consume the good he now holds. If an agent consumes his good, he immediately (without cost) produces a "replacement", whose type of course depends on his type's production capability. Thus, regardless of whether or not he decided to consume, at the end of the period each agent again holds one good. Before the next period begins, the agents must pay a storage fee, which depends on the type of good they are holding.

The accounting that underlies decision-making in this model trades off utility gains from consumption against storage costs.[61] Agents clearly might want to trade either for the type of good they like to consume or for goods with lower storage costs than the one they currently hold. In addition, they might conceivably want to speculate by obtaining and holding a good that they hope will bring them a "profit" in subsequent period trades that is sufficient to offset a high storage cost in the present period. What strategies agents will actually pursue -- either rational agents trading in equilibrium or classifier agents learning how to act -- depends on a complete specification of the parameters of the environment: the production functions (that is, which agent types produce which good types), the agent type-specific utility functions, the relative storage costs of the different good types, and the proportion of each type of agent in the environment.

Kiyotaki and Wright assumed that the agents who populate this environment satisfy the usual rationality assumptions, and they calculated Markovian Nash equilibria for a number of particular specifications of the environment.[62] In contrast, Marimon, McGratton and Sargent (hereafter MMS) use classifier systems to represent their agents. Their primary interest was to discover whether these classifier system agents would learn their way into the Kiyotaki-Wright equilibria -- and, in situations in which there was more than equilibrium, which one would the classifier agents prefer. Their hope was that models with classifier system agents could support and even extend standard neoclassical theory, by providing a mechanism for arriving at equilibria, a tool for finding equilibria when direct calculation is intractable, and a way of

[61] Of course, this accounting is explicit in the Kiyotaki and Wright models, but only implicit (and distributed amongst all the classifiers that constitute a particular agent!) in the Marimon, McGrattan and Sargent AW. The description of strength changes in the text below shows how these quantities affect the strengths of individual classifiers.

[62] These equilibria of course depend on the model specifications. In particular, for some specifications, Kiyotaki and Wright found equilibria in which the good with lowest storage cost served as "money" (that is, in every exchange, each participant obtained either a good he wanted to consume -- or the "money" good). Other specifications produced equilibria that included some speculative exchanges, as described in the previous paragraph of the text. And still other specifications supported more than one equilibrium.

distinguishing between multiple equilibria. As we shall see, this hope was not realized.

With agents restricted to Markovian strategies, the Kiyotaki-Wright environment is a very simple world. "Reasonable" agents really only have one choice to make: they have to decide which of the two types of goods to which they assign zero utility they prefer to store between periods![63] Of course, the problem is harder for classifier system agents, because they are not a priori reasonable -- nor do they even have a priori knowledge of their utility functions or the storage costs that will be imposed on them.

MMS use two linked sets of classifiers to represent their agents. One of these sets codes rules for exchange, the other rules for consumption. Exchange rules are of the form: IF you hold good type in set A and your trading partner holds good type in B THEN C; where A and B are subsets of {1,2,3} and C is either "trade" or "don't trade". Consumption rules have the form: IF you hold good type in A THEN D; where D is either "consume" or "store".[64]

To decide whether or not to exchange and then whether or not to consume in period t, each agent holds successive bidding competitions in its two classifier sets. The classifiers that submit the highest bid in their respective competitions win -- call these winning classifiers E_t and C_t respectively. Of course, the consumption competition occurs after the action specified by E_t has been carried out.

Agents learn through changes in strength to the classifiers that win the competitions in each period.[65] As usual, winners lose strength when they pay out their bids, and they gain strength from the payments of the bids of the winners of the "next" competition (C_t pays E_t, and E_{t+1} will in turn pay C_t).[66]

[63] I am supposing that agents know which type of good has positive utility for them. "Reasonable" agents will always exchange for this good and then consume it (assuming, of course, that the utility they derive from consumption is greater than the storage cost of the good they produce as a replacement). And since agents do not produce the good they like, the only way they can end up with the good they neither like nor produce is to exchange for it (if they don't start with it) or keep it (if they do) -- and then not consume.

[64] Note that all actions are "external", so the system does not use the standard classifier message list. Instead, its short-term memory only records which classifier of each type won the last competition.

[65] Because the possible choices for agents are so limited -- even without assuming "reasonableness" -- MMS could represent all possible strategies with computationally tractable classifier sets. They also considered variant models with some standard replacement operators (Cover Detectors, Cover Effectors, specification mutations and recombination).

[66] Note that the action of E_t leaves the agent with the good that matches the condition of Ct, whose action in turn leaves the agent with the good that matches the condition of E_{t+1}. So this is just the usual "payment to supplier" idea, without the usual message list. MMS do not use the standard bucket brigade to determine the magnitude of strength changes. With their strength updating algorithm, strength reflects

Reward comes from consumption. If C_t's action is "consume", it is credited with the utility gain the agent experiences from consumption, and its strength increases as a result. In addition, storage costs have to be paid. Since C_t's action determines which type of good the system holds between periods t and t+1, its strength is reduced as a function of the cost of storing this good.[67]

What happens in the MMS AW? First, the agents develop coherent behavior patterns: they trade to obtain goods they like to consume or to lower their storage costs, and they consume for the same reasons. As a result, "money" emerges in this AW: in every transaction, an agent either obtains a good he wants to consume or "money", that is, the good with lowest storage cost. Thus, organization occurs at both levels of this hierarchical AW: the agents develop coherent economic identities, and the economy they form is characterized by structured patterns of trade.

Second, the stable trading structure that emerges in this AW does not necessarily correspond to a Markovian Nash equilibrium. In particular, classifier system agents are reluctant to speculate -- that is, to hold a good with high storage costs in the hopes of trading it in the next period for a desired consumption good -- even when it is "rational" for them to do so. Thus, the classifier system agents do not organize themselves into an equilibrium trading pattern in Kiyotaki-Wright environments that support only speculative equilibria. Rather, they "prefer" to trade only for immediate consumption or "cash".[68] As a result, the idea to use classifier system agents in a mere supporting role in equilibrium theory seems a dead end.

average payoff per activation, rather than total payoff as in Holland's system. For a justification of this change in terms of convergence properties, see Arthur (1990).

[67] A peculiar feature of the MMS dynamics is that they impose the same classifier systems on every agent of a given type. This means that when strengths are modified as a result of an interaction between any two agents, changes are made to all agents of the same type as the two interactors. MMS justify this imposition of "representative agents" in terms of savings in computer time and space, but it violates the spirit of AW modelling. In particular, it means that MMS could not probe the extent to which (path-dependent) heterogeneity between initially homogeneous agents can arise in their economic environment.

[68] Of course, it is not particularly surprising that MMS agents do not speculate, since to do so, they would have to form linked chains of actions -- and as we saw in Section 4.4, these are unlikely to emerge without system rules that promote them, like Triggered Chaining Operators. In addition, the "representation agent" constraint described in the previous footnote makes it impossible to explore variant within-type agent behavior. With within-type agent heterogeneity and some provision for differential replication rates for agents with different behaviors, perhaps more interesting behaviors might arise in a Kiyotaki-Wright environment.

An artificial stock market

The second example of an economic AW with classifier system agents is currently being developed by Brian Arthur, John Holland, Richard Palmer and Paul Tayler (hereafter AHPT).[69] Their AW models a simple stock market, in which a single security is traded. This security pays a dividend, which varies stochastically with respect to the outside interest rate. Each period, agents can place orders to buy or sell a single unit of stock, or they can do nothing. They base their decisions just on the information contained in the time series of past prices, dividends and interest rates. When the buy and sell orders are in, a specialist fulfills all trades and uses an algorithm based on current price and excess demand to declare a new price for the next period.

An AHPT agent consists of a set of classifiers that code for predictions of future price movements, based on past stock prices and returns. These predictions have forms like: "IF last period price exceeds twice fundamental value (dividend/interest rate) THEN price will go down" or "IF the average price of the last five periods exceeds the average of the last 50 periods THEN price will go up". At the beginning of each period, a bidding competition amongst each agent's matched classifers determines the agent's prediction for the next period's price. If the prediction is that prices will rise next period, the agent places a buy order; if the prediction is that prices will fall, the agent sells; otherwise, he holds. Winning classifiers are rewarded on the basis of the (one period) profits that result from the transaction they initiate. Current versions of the AW consist of 100 agents, each with 60 predictor classifiers. Genetic algorithms periodically generate new predictors for each agent.

In the AHPT world, stock price is determined each period on the basis of the action of all the agents, which in turn reflect complicated interactions between their constituent predictors. How well any given predictor functions depends in turn on the market's overall price dynamics. As a result of this complexity of interaction and feedback between levels, the behavior that emerges in this system, both at the level of the individual agents and at the level of the market's price dynamics, is very rich. According to Arthur's summary account of experiments with AHPT, price begins by fluctuating around fundamental value. But then "mutually reinforcing trend-following or technical-analysis-like rules" establish themselves in the predictor populations. Later, other phenomena, such as speculative bubbles and crashes, can be observed to occur. Moreover, the market does not seem to settle down to any stationary state, as in the MMS AW. AHPT

[69] Unfortunately, there is not yet any detailed description of the AHPT AW in print. My account is based primarily on preliminary material given in Arthur (1992) and personal conversations with AHPT. As a result, it is even more sketchy than the other model descriptions reported in this paper.

test for this by cloning and "freezing" successful agents and then reintroducing them much later into the system, where they turn out to perform poorly, since they are no longer adapted to the behavior of the other agents in the market. This may happen despite the fact that the price series itself appears stationary to an "outside observer".

AHPT's primary purpose in designing their artificial stock market was to gain insight into the reasons why real-world traders perceive their markets as they do. According to Arthur, "traders talk about the 'mood of the market', its 'nervousness', or 'confidence' for example; they take technical trading rules or 'chartism' seriously; they see temporary surges and crashes as more than random fluctuations." None of this makes sense from the point of view of neoclassical economics. AHPT's hope was to design a system in which such features arise as the result of the interactions amongst heterogeneous agents, each capable of learning about the world their joint actions are creating, but exploiting different frames of reference that generate different "local" opportunities for successful action. They seem to be succeeding in this enterprise. The next step is to figure out how to carry out experiments with their AW that will shed some light on how and under what circumstances these kinds of phenomena emerge.

5. Artificial Economies and the Problem of Coordination

Perhaps the most surprising thing about an economy is that there is such a "thing" at all. From one point of view, an economy appears totally disaggregated: every firm separately decides what to produce, every consumer what to buy, and all these decisions are based just on agent-specific needs, interests and information. Why should anything coherent result from such a process? Yet it does: an economy exhibits large-scale structure -- with organized markets, mutually dependent but distinct industries, trade associations, labor unions and so on -- and relatively stable macroeconomic descriptors that vary slowly compared to the rate of change of the underlying microeconomic decisions over whose consequences they aggregate. The problem of coordination is: where does this order come from? That is, what are the mechanisms whereby Adam Smith's Invisible Hand accomplishes its task?

5.1 Two Approaches to the Problem of Coordination

Neoclassical economic theory places at the center of its account of coordination a single concept: Walrasian price equilibrium.[70] In a Walrasian equilibrium, the market efficiently coordinates the actions of agents, through the prices it assigns to the various commodities. The great triumphs of neoclassical theory are mathematical theorems that guarantee the existence of Walrasian equilibria, when competition is "perfect" and agents "rational",[71] and algorithms for computing equilibria, given assumptions about production capabilities and agents' endowments and values.

On the other hand, there are several unsatisfactory aspects of the equilibrium account of coordination. First, it does not explain how the market arrives at a Walrasian price equilibrium. Quite the contrary: attempts to provide a dynamic account of out-of-equilibrium price formation have yielded more and more general counter-examples to the proposition that price adjustments to over-supply or over-demand result in convergence to equilibrium prices. Second, it is based on unrealistic assumptions about agent behavior and market conditions. Firms and consumers do not form expectations about the future or decide what to do in the rational manner that the theory posits, and competition is frequently far from perfect. Third, and most important, many of the most striking aspects of real-world economic coordination play no role at all in the theory. In particular, the real economy is constantly changing: new kinds of commodities are developed, then produced and traded, and richer institutional linkages connect agents over wider and wider geographical areas. It is hard to see how the problem of the coherence of such a system can be addressed with a theory that fails to assign a central role to processes of innovation and change, in what economic agents can do and the structures through which they act.

Artificial Worlds provide another approach to the problem of coordination. In this approach, economic coordination is regarded as a special kind of EHO, and the central question is to find the attributes of specifically economic objects and interactions that result in specifically economic forms of aggregate pattern and structure. Artificial Economies are Artificial Worlds whose microentities represent economic agents and products. Interactions between these microentities model fundamental economic activities -- production, exchange and consumption. Unlike the Artificial World models described in Section 4.5, Artificial Economies are meant to represent "entire" economies. Thus, they have certain closure properties: for example, what consumers

[70] A Walrasian equilibrium is a system of prices for commodities such that, if agents exchange freely at these prices, (i) each agent will obtain a set of commodities that provides him with the maximum attainable value (given his initial endowment), and (ii) all markets will clear (that is, the supply for each commodity will exactly equal the demand for that commodity).

[71] In perfect competition, agents may take prices as given when they decide what to buy and sell, ignoring the effects of their actions on the prices that obtain. Rational agents are able to form rational expectations about future contingencies and they always act so as to maximize their own expected utility. Even with these assumptions, the "equilibrium guarantee" only covers certain conditions on production functions and agent's values.

spend in one market of an Artificial Economy, they are paid in another.

When an Artificial Economy is populated with an initial set of agents endowed with an inventory of products, and the values of its system parameters have been fixed,[72] the economy can be "run". Under some conditions on population and parameters, the resulting economy exhibits such emergent features as stable growth paths for the Artificial Economy's analog of GDP, Pareto-law distributions for firm size, and characteristic product life-cycle curves -- under others, no sustained growth or orderly industrial structure at all takes place. The purpose of Artificial Economy experimentation is to discover what kinds of structured economic regimes can occur and to see how they depend on system parameters and the characteristics of the constituent agents.

In contrast to General Equilibrium models, Artificial Economies are inherently dynamic.[73] While General Equilibrium modellers start by assuming a desired outcome state (Walrasian equilibrium), the designer of an Artificial Economy is first of all concerned to model how economic agents interact -- the institutional arrangements through which interactions take place, as well as the ways in which agents take advantage of the opportunities these arrangements afford. The more plausible are the assumptions about agents and institutions built into an Artificial Economy, the better:[74] the argument that emergent aggregate regularities in the Artificial Economy are causally related to observable macrofeatures of real economies depends on the match between the characteristics of the microinteractions built into the Artificial Economy and those that actually take place in real economies.

Artificial Economy have to be "playable" -- and so a lot of institutional details have to be explicitly specified. For example, events have to be scheduled to occur in a logically meaningful and physically realizable order -- a firm cannot produce until it has hired the workers it will use to do so. Also, market rules have to spell out how prices are formed and who ends up trading with whom, as a function of the allowable actions of the agents who trade in the market. And

[72] For example, in the Artificial Economy described in section 5.2 below, most of these parameters control features of the economic environment that affect innovation, such as technological opportunity, degree of appropriability of new technologies, cumulativeness of research, and extent of learning-by-using.

[73] By "dynamic", I do not mean merely "time-indexed", as the term is used in the literature on intertemporal equilibria. Rather, I mean that the model specifies transition laws that govern how its state at time t transforms into its state at time t+1, and these laws are not a function of future states (as they are in rational expectations theory).

[74] In particular, agents in an Artificial Economy are constrained to make their decisions in a psychologically plausible way, in the face of future contingencies about which they can be no better equipped to form rational expectations than is the designer himself!

if firms can borrow, some form of bankruptcy law has to be implemented, since a firm might find itself unable to make good on the terms of its loans. One of the most interesting experiences in designing and experimenting with an Artificial Economy is coming to realize that these institutional details matter -- and that any theory that leaves them out is sweeping something crucial under the rug.

5.2 An Artificial Economy

In this section, I describe an Artificial Economy developed by myself in collaboration with Giovanni Dosi, Marco Lippi, Jim Pelkey and Paul Tayler (hereafter DLLPT).[75] This model extends work reported in Chiaromonte, Dosi and Orsenigo (1992), which in turn was inspired by models in Nelson and Winter (1982). Perhaps the most ambitious and well-documented model that could be considered an Artificial Economy is the MOSES model of the Swedish economy described in Eliasson (1985, 1989). All of these models (again in contrast with equilibrium theory) are inspired by Schumpeterian insights into the disequilibrating effects of competition, and assign a central role to processes of innovation -- both successful and "mistaken" -- in their accounts of economic coordination.

Here are brief descriptions of the microentities in the DLLPT Artificial Economy and their principal modes of interaction:

- Microentities: There are five types of agents (Sector 1 and Sector 2 firms, a bank, researchers, and laborers) and two types of products (machines and consumer good). Sector 1 firms hire laborers to produce machines.[76] In addition, they hire researchers to develop new types of machines. Sector 2 firms buy machines from Sector 1 firms and use them, together with labor, to produce a consumer good. Researchers and laborers use their wages to purchase this good, which they then consume. The bank pays interest on savings from firms and workers, lends to firms at an interest rate that it sets, and funds the formation of new firms.

[75] I chose this model to summarize because I am most familiar with it -- and because its inferential difficulties, discussed in the next section, are generic. A program implementing this model was written by Francesca Chiaromonte at the Santa Fe Institute.

[76] All their other production inputs are free.

- Innovation: Sector 1 research is designed to discover machine types that can perform better than the machines the firm is currently capable of producing. Each machine type is described by two positive numbers: one measures how much it costs to produce, the other its efficiency in production. The Artificial Economy's model of technological innovation then corresponds to a controlled stochastic process moving through R^{2+}, where the control variables are the amounts a firm invests in each of three types of innovative activities.

This stochastic process specifies, for each type of innovative activity, the probability of successful outcome of a given research project and the performance indices of the machine type that results from a successful project. The probability of success depends on: how much money the firm invests in the project; system parameters that measure the degree of appropriability of innovations and the technological opportunity for each type of innovative activity; and firm-specific parameters that reflect cumulated research know-how from prior investment.

The three types of research activity in which Sector 1 firms can choose to engage are radical innovation, incremental innovation and imitation. Radical innovation, if successful, produces a machine type belonging to a new family or "technological trajectory" of machine types. Successful incremental innovation leads to a new machine in the same "technological trajectory" as one the firm currently knows how to produce. For the same investment, radical innovation has a lower probability of success than incremental innovation; but when radical search succeeds, it typically produces a machine type whose performance characteristics represent a greater advance than what is achieved from an incremental innovation. Imitative search targets a machine type currently produced by a competing Sector 1 firm. The search succeeds if the imitating firm learns how to produce a machine type on the same "technological trajectory" as the targetted type (though not necessarily as efficient as the target).

- Market Rules: There are three markets in the Artificial Economy: the market in which Sector 2 firms buy machines produced by Sector 1; the market in which workers buy the consumer good produced by Sector 2; and the labor market, in which Sector 1 and Sector 2 firms hire laborers and researchers.

The machine market features production to order. Sector 1 firms issue catalogues listing the machine types they produce along with machine prices, and Sector 2 firms place orders for the machines they wish to purchase. Machines are payable on delivery, so a Sector 1 firm may receive more orders than they can fill, in which case they accept orders on a first-come, first-served basis. Sector 2 firms whose orders are not accepted may place orders for their second and third choices before the market closes.

Consumer goods are sold at a fair held at the end of the production year. Each firm determines how much of the good to bring to the fair and what prices to charge for what it brings. Workers have full access to the prices asked by each seller, and they buy, first-come first-served, from the cheapest to the more expensive. The fair closes when all the good is sold or there is no more consumer demand.

The labor market operates as a hiring hall, at the beginning of each year. Each firm determines the wage rates they are willing to pay (one for researchers and one for laborers), and the workers sign up on a first-come first-served basis, from the highest- to lowest-paying firms. Firms may fail to hire their desired quota of workers, in which case they invest the unspent research or production dollars in the bank; or there may be unemployment, in which case unemployed workers forego all consumption for the year.

Note that in none of these three markets are prices negotiated. Markets that allow price negotiations require more complicated inputs from participants: in addition to quantity and price inputs, negotiating strategies have to be supplied.

- Banking Rules: The Artificial Economy has only a rudimentary banking sector. There is a single bank that sets an interest rate for savings and another for loans. Firms must exhaust savings before they can borrow. Loans are issued for fixed periods that depend on the borrowing firm. Each firm faces a credit cap that depends on its previous year's net turnover. Annual interest and principal payments are due at the end of every year.

Firms that cannot meet the required annual payments are allowed to postpone payment for one year. During this year, the bank will issue an emergency loan covering some production costs. This loan must be paid back over the next two years and no other emergency loans will be granted during this period. Two successive failures to repay all outstanding loans force a firm into bankruptcy. The assets of a bankrupt firm are scrapped.

- Firm Decision-Making -- Sector 1: Firms make their decisions according to "organizational routines" that are modelled as particular forms of decision rules. These rules are typically functions of three types of arguments: adjustable environmental parameters, firm-specific parameters that describe aspects of the firm's "psychology" of decision-making (risk-aversion, time-discounting, etc.), and past observables such as the firm's previous period sales, degree of labor rationing experienced, orders received and so forth. The rules are based either on the empirical literature on firm decision-making or on approximations to "optimal decision-making under uncertainty", with heuristic methods for forming expectations of future aggregate quantities.

Sector 1 firms decide how much to pay workers, which machines to list in their catalogues and how much to charge for them, how many research workers to hire and what tasks to assign them, how many laborers to hire, how much to borrow, and how much to produce of which type of machine.

How much to pay workers: Firms of both sectors try to maintain their position in the labor markets, so if they paid a bonus in excess of the average overall wage rates in the previous period, they offer at least the same bonus in the following period. They raise their bonus only if they experienced labor rationing in the previous period, in which case the increase is a fixed function of the extent of that rationing.

How much to charge for listed machines: A new machine is priced according to a mark-up rule: that is, it is priced at a firm-specific multiplier times the machine's production cost. For previously listed machines, the firms adjust the previous period price in response to three factors: change in production costs (that is, wage rates), change in sales, and the quantity of unfilled orders (which represents a kind of backlog of demand for the machine). The adjustment rule depends on firm-specific "reactivity" parameters. If the adjustment rule determines a price below a firm-specific minimum acceptable mark-up rate, the firm ignores the rule and uses this minimal mark-up rate to set its price.

Which machines to list: Firms want to produce machines that will be attractive to their customers. To Sector 2 firms, evaluating a machine involves trading off between its price and its productivity in use. The Sector 2 firms accomplish this trade-off through a payback period criterion, in which they estimate the profit they will obtain by operating the machine for a firm-specific length of time. The longer this length, the more they are willing to pay to obtain a more productive machine. Sector 1 firms then use the same evaluation functional to decide whether to go into production with new machine types their researchers have designed; they use pay-back periods that are reported to them by their own customers. (This allows a certain amount of "market segmentation" to emerge: some Sector 1 firms producing low-cost, low-productivity machines for their customers who favor short pay-back periods, while others produce high-cost, high-productivity machines for a different set of customers.)

How many research workers to hire and what tasks to assign them: Firms invest a firm-specific proportion of their previous year's net turnover in research. Total research investment is allocated to the three types of search activity according to a formula determined by two firm-specific parameters. Which machine types to incrementally improve or imitate are determined by calculating expected returns for the investment, where the expectations are based on the firm's experience with previous incremental and imitative research projects.

How many laborers to hire, how much to borrow and how much to produce of what: All orders are filled up to the firm's credit ceiling. This determines how much the firm borrows and how many laborers it can hire. If it receives more orders than it can fill, it produces the most profitable machines first.

- Firm Decision-Making -- Sector 2: Like Sector 1 firms, Sector 2 firms have to decide what wages to offer to laborers and how much to borrow from the bank. In addition, they have to decide how much to produce and what prices they will charge, which of their current stock of machines to use in production, how many and which new machines to order, and which machines to scrap.

Determining prices: First, the firm calculates the costs of production with each machine in stock and those it considers purchasing. Next, using a statistical procedure together with data from previous years, it forecasts the highest price that will paid at this year's consumer good fair. It then decides compares production costs with anticipated sales, machine by machine, to determine which machines to use in production. Finally, it sets its prices by reducing its estimated cutoff price by a safety margin that depends on a firm-specific "timidity factor". This process simultaneously determines price and selects which machines to employ in production.

Investing in new machines: After deciding (as above) which of its current machines to use in production, the firm calculates how much cash and credit it could apply to expanding its machine stock. It evaluates machines offered for sale according to the payback period criterion already described, and it orders the available machines for which it projects a profit over its payback period. If it can afford its first choice, it orders it; else it orders its second choice if it can afford that one. It then iterates this process until it can no longer afford any desired machine.

Scrapping machines: Any machine that is not used in production in three successive periods is removed from the firm's capital stock.

- Agent Demography: The number of researchers and laborers grow exponentially. New firms are created according to several different schemes; the rate of creation depends on average profitability rates in the two sectors. New firms are funded by the bank for a fixed period to engage in product research (Sector 1) and to purchase capital stock (Sector 2). Their attributes are selected according to the empirical distribution of currently existing firms. Firms die when they no longer generate positive net turnover and cannot qualify for bank loans.

In all, there are 15 system parameters in the Artificial Economy. In addition, the behavior of each Sector 1 firm is determined by 8 firm-specific parameters, while 3 suffice for each Sector 2 firm. Sensible initialization of such a large parameter set, while difficult, is made easier by the fact that almost all of these parameters have a direct economic or psychological interpretation, and it is possible to define clusters of related parameter values that form "natural" metaparameterizations of "economic regimes".

5.3 Some Difficulties with Artificial Economies

As the last section makes clear, there really is not such a thing as a short summary description of an Artificial Economy . A lot of details are required to specify "playable" institutional arrangements and the agents who have to operate in them. Clearly, there is more than one way to specify any of these details. In the DLLPT Artificial Economy, for example, why organize the consumer good market as a fair instead of an auction? Does it matter that machine manufacturers produce to order instead of building up inventories? Why should firms use payback-period accounting and mark-up pricing rules? And so on, on and on. As a result, any Artificial Economy is open to criticism on the grounds that its design is arbitrary.

A first response might be that it is necessary to start somewhere: if you can show that "macroeconomic" stability can emerge from the dynamics of "microeconomic" interaction in any recognizably "economic" environment,[77] then you have increased the plausibility of the proposition that real-world economic coordination is an instance of EHO.[78] Unfortunately, this response does not go very far. Artificial Economies, unlike the Arrow-Debreu model, lack the virtues associated with a high level of abstraction -- simplicity and mathematical tractability. This produces a real barrier to their social extension. The more richly detailed a model is, the more intriguing it is to its designers -- but the less likely it is to capture anyone else's imagination or interest, which flags at the first ad hoc and unshared assumption. Without mastering the microlevel details built into an Artificial World, it is simply impossible to come to a reasoned judgement on whether an observed aggregate-level property is in fact emergent -- or merely a consequence easily derived from the superposition of some particular microlevel features. And without this judgement, the whole point of the Artificial World is lost.

If arbitrariness cannot be abstracted away, what then? Here are two complementary research strategies for coping with the problem. First, building on ideas already in the economic "mechanism design" literature, one could at least delimit degrees of arbitrariness by developing functional taxonomies of the various institutional arrangements that have to be introduced into an Artificial Economy. For example, is there a minimal characterization of the functionally different types of market rules or bankruptcy laws? Similarly, one might try to construct a typology of "nonoptimizing" decision-making strategies (or perhaps orientations) that are economically relevant. Such taxonomies would help to define the set of Artificial Economies that have to be investigated to make general assertions about EHO in real economies.

Second, one could change the notion of what an Artificial Economy is, away from the idea of a single parameterized model that specifies a priori its institutional forms and agent behaviors. Instead, imagine an Artificial Economy as an experimental environment in which users can easily tailor models designed to suit their own particular research agendas. Object-oriented programming techniques can be used to construct such an environment, which would consist of a library of different kinds of modelled institutions and agent types, together with an interface that makes it easy for users to combine different items from this library to make particular experimental economies. The interface might also feature statistical and graphical features that aid in the discovery of emergent properties in these experiments -- and procedures for summarizing experimental designs and relevant results in a way that they can be assimilated into a database that all users could access and analyze.[79] With such a tool, assuming a sufficient number of users found it attractive, Artificial Economy research might become better characterized as diversified than arbitrary.

There is another, more serious, difficulty with current Artificial Economies. They offer only very limited scope to the emergence of new structures -- and, so far, none at all to the emergence of higher-level entities. What do emerge are patterns -- in macroeconomic variables like GDP, in aggregate descriptors of industrial organization (like firm size) and innovative demography (like innovation rate as a function of age and size distributions of firms), and in product life-cycles. But no Artificial Economy yet has a way of representing the kinds of innovations in entity structure at the level of the firm and of the industry that are sketched in Section 3.3. In fact, even the entities that the current Artificial Economies do represent are not capable of much change in what they do or how they do it.[80] Nor do any

[77] More particularly: in one that does not assume away the question by invoking "representative agents" (see Kirman, 1992).

[78] This "existence proof" justification is similar to that frequently given for taking the Arrow-Debreu model seriously: the Arrow-Debreu model shows that at least one kind of "economic" environment (surely as remote from a real economy as the Artificial Economy described in the last section) supports Walrasian equilibrium.

[79] A prototype that implements these ideas is currently under development.

[80] MOSES firms can engage in new kinds of activities (for them), but they do not develop novel ways of carrying these out. Firms represented by parameterized behavioral rules for their "behavioral routines" hardly change at all -- at most, their parameters may respond adaptively to the firm's experiences in the marketplace.

of the Artificial Economies represent products in such a way that new "kinds of" commodities (as opposed to the sort of technological development described in the previous section) arise endogenously; to do so would require even more elaborate representations for agents, since really new products must coevolve with "tastes" for them. Thus, Artificial Economies are not open-ended computationally, even to the extent of the other principal Artificial Worlds reviewed in Sections 3 and 4. Though there are some obvious ways to improve this situation -- for example, using adaptive representations for agents, like classifier systems or even neural nets -- real progress will require new insights in economics about the nature of the structures that need to be represented as well as new computational techniques for the representations themselves.

6. An Afterword

As we have seen, Artificial Worlds differ substantially from the kind of "minimalist" models on which much of neoclassical economic theory is based. Structurally, Artificial Worlds are populated with a variety of heterogeneous microlevel "agents" who enter into complex interactions with one another. The "agents" must respond to an environment that is formed in part as a result of the collective history of their interactions. Their response potential is programmed into the Artificial World, but for the World to work, there must be some degree of open-endedness in the way this potential manifests itself. From a computational point of view, this is the great challenge in designing Artificial Worlds, and we have seen a variety of approaches to this problem, from FOG's intensional functional representation, to the genetic operators used in EPD, through the whole gamut of replacement operators and "bidding competition" structures in classifier systems.[81]

Of course, the difference between Artificial Worlds and most neoclassical economic models is more than structural: they are designed to explore different kinds of questions. Artificial Worlds are about EHO, and if EHO is an important kind of phenomenon in real economies, then Artificial Worlds will have a place in economic theory. I have argued that there is a variety of economic phenomena that seem to manifest the characteristic features of EHO, from the processes through which individual agents learn how to act in new situations, through the coevolution of new products and industrial structure, to the emergence of "herd behavior" in markets and macroeconomic metastability. I have also described some existing Artificial Worlds that, suitably modified or extended, have the potential go give some insights into these economic processes. The match between problems and methods is not yet very good. The purpose of this paper is to promote such a match, by pointing to a promising direction for workers with the requisite familiarity with economic institutions and behaviors who might not have considered whit alternative modelling style.

[81] I think we can expect more and more approaches to the problem of designing open-ended computer programs to emerge from the computer science community -- see, for example, Forrest (1990) and Huberman (1988). In addition, object-oriented programming, with its emphases on modularity and extendability, provides a natural environment for building programs of sufficient complexity that they can manifest EHO.

REFERENCES

Arthur, W.B. (1990), A learning algorithm that mimics human learning, Santa Fe Institute Working Paper 90-026.

Arthur, W.B. (1991), Designing economic agents that act like human agents: a behavioral approach to bounded rationality. American Economic Review, 81, (May 1991).

Arthur, W.B. (1992), On learning and adaptation in the economy, Santa Fe Institute Paper 92-07-038.

Barendregt, H. (1984), The Lambda Calculus: Its Syntax and Semantics (New York: North-Holland).

Bedau, M. and N. Packard (1992), Measurement of evolutionary activity, teleology, Artificial Life II, 431-462.

Booker, L., D. Goldberg and J. Holland (1989), Classifier systems and genetic algorithms, in J. Carbonell, ed., Machine Learning: Paradigms and Methods (Cambridge, MIT Press),235-282.

Buss, L. (1989), The Evolution of Individuality (Princeton: Princeton University Press).

Buss, L. and W. Fontana (1992), Algebraic replicators and units of selection, forthcoming.

Chiaromonte, F., G. Dosi and L. Orsenigo (1992), to appear in R. Thomson, ed., Learning and Technological Change (London: MacMillan).

Edelman, G. (1992), Bright Air, Brilliant Fire: On the Matter of the Mind (New York: BasicBooks).

Eldredge, N. (1985), Unfinished Synthesis (New York: Oxford University Press).

Eldredge, N. and S. Gould (1972), Punctuated equilibria: an alternative to phyletic gradualism, in T. Schopf, ed., Models in Paleobiology (San Francisco: Freeman, Cooper), 305-332.

Eliasson, G. (1985), The Firm and Financial Market in the Swedish Micro to Macro Model -- Theory, Model and Verification (Stockholm: IUI).

Eliasson, G. (1989), Modelling the experimentally organized economy: Overview of the MOSES model, Chapter 1 of J.Albrecht, F. Bergholm, G. Eliasson, K. Hanson, C. Hartler, M. Heiman, T. Lindberg and E. Olavi, MOSES Code (Stockholm: IUI), 7-65.

Fontana, W. (1992), Algorithmic chemistry, in Langton et al. eds., Artificial Life II, 159-210.

Forrest, S. (1990), ed., Emergent Computation: Self-Organizing, Collective and Cooperative Computing Networks (Cambridge: MIT Press).

Goldberg, D. (1989), Genetic Algorithms in Search, Optimization & Machine Learning (Reading, MA: Addison-Wesley).

Gould, S. (1989), Wonderful Life: The Burgess Shale and the Nature of History (New York: W.W. Norton)

Holland, J. (1990), Concerning the emergence of tag-mediated lookahead in classifier systems, Physica D, 42, 307-317.

Holland, J., K. Holyoak, R. Nisbett and P. Thagard (1986), Induction: Processes of Inference, Learning, and Discovery (Cambridge: MIT Press).

Huberman, B. (1988), ed., The Ecology of Computation (Amsterdam: North-Holland).

Hull, D. (1988), Science as a Process: An Evolutionary Account of the Social and Conceptual Development of Science (Chicago: University of Chicago Press).

Hull, D. (1989), The Metaphysics of Evolution (Albany: State University of New York Press)

Kauffman, S. (1990), Requirements for evolvability in complex systems: orderly dynamics and frozen components, Physica D, 42, 1-11.

Kauffman, S. and S. Johnson (1992), Co-evolution to the edge of chaos: Coupled fitness landscapes, poised states, and co-evolutionary avalanches, in Langton et al. eds., Artificial Life II, 325-370.

Kirman, A. (1992), Whom or what does the representative individual represent?, Journal of Economic Perspectives, 6, 117-136.

Kiyotaki, N. and R. Wright (1989), On money as a medium of exchange, Journal of Political Economy, 97, 927-954.

Lakoff, G. (1987), Women, Fire and Dangerous Things: What Categories Reveal about the Mind (Chicago: University of Chicago Press).

Langton, C. (1992), Life at the edge of chaos, in Langton et al. eds., Artificial Life II, 41-92

Langton, C., C. Taylor, J.D. Farmer, and S. Rasmussen, eds. (1992), Artificial Life II: Proceedings of the Workshop on Artificial Life Held February, 1990 in Santa Fe, New Mexico (Redwood City, CA: Addison-Wesley)

Lindgren, K. (1992), Evolutionary phenomena in simple dynamics, in Langton et al. eds., Artificial Life II, 295-312.

Marimon, R., E. McGrattan and T. Sargent (1990), Money as a medium of exchange in an economy with artificially intelligent agents, Journal of Economic Dynamics and Control, 14, 329-373.

Nelson, R. and S. Winter (1982), An Evolutionary Theory of Economic Change (Cambridge: Harvard University Press).

Orsenigo, L. (1989), The Emergence of Biotechnology: Institutions and Markets in Industrial Innovation (London: Pinter)

Rasmussen, S., C. Knudsen, R. Feldberg and M. Hindsholm (1990), The Coreworld: emergence and evolution of cooperative structures in a computational chemistry, Physica D., 42, 111-134.

Ray, T. (1992), An approach to the synthesis of life, in Langton et al. eds., Artificial Life II, 371-408.

Riolo, R. (1987a), Bucket brigade performance I: Long sequences of classifiers, in J. Greffenstette, ed., Proceedings of the Second International Conference on Genetic Algorithms and Their Applications, 184-195.

Riolo, R. (1987b), Bucket brigade performance II: Simple default hierarchies, in J. Greffenstette, ed., Proceedings of the Second International Conference on Genetic Algorithms and Their Applications, 196-201.

Riolo, R. (1989a), The emergence of default hierarchies in learning classifier systems, in J. Schaeffer, ed., Proceedings of the Third International Conference on Genetic Algorithms, 322-326.

Riolo, R. (1989b), The emergence of coupled sequences of classifiers, in J. Schaeffer, ed., Proceedings of the Third International Conference on Genetic Algorithms,

Riolo, R. (1990), Lookahead planning and latent learning in a classifier system, in J.-A. Meyer and S. Wilson, eds., Proceedings of the Conference on Simulation of Animal Behavior: From Animals to Animats, Paris, September 1990 (Cambridge: MIT Press), preprint.

Robertson, G. and R. Riolo (1988), A tale of two classifier systems, Machine Learning, 3, 139-159.

Salthe, S. (1985), Evolving Hierarchical Systems: Their Structure and Representation (New York: Columbia University Press)

Somit, A. and S. Peterson (1992), The Dynamics of Evolution: The Punctuated Equilibrium Debate in the Natural and Social Sciences (Ithaca, NY: Cornell University Press).

Winograd, T. and F. Flores (1986), Understanding Computers and Cognition (Reading, MA: Addison-Wesley).