Genetic Programming and Agent-Based Computational Economics: From Autonomous Agents to Product Innovation

Shu-Heng Chen

Abstract Despite their great development over the last decade, most ACE (agent-based computational economics) models have been generally weak in demonstrating discovery or novelty-generation processes. In this sense, they are not very distinct from their counterparts in neo-classical economics. One way to make progress is to enable autonomous agents to discover the modular structure of their surroundings, and hence they can adapt by using modules. This is almost equivalent to causing their "brain" or "mind" to be designed in a modular way. By this standard, simple genetic programming is not an adequate design for autonomous agents; however, augmenting it with automatic defined terminals (ADTs) may do the job. This paper provides initial research with evidence showing the results of using ADTs to design autonomous agents.

1 Introduction

GP maintains a unique position when compared with other computational intelligence tools in modeling autonomous agents. Basically, there are two distinguishing features of using GP in modeling autonomous agents. First, in a sense, GP provides agents with a larger degree of autonomy. Second, it provides us with a concrete picture to visualize the learning process or the discovery process as a growing process, i.e., that of growing the evolving hierarchies of building blocks (subroutines) from an immense space of subroutines.

Shu-Heng Chen
AI-ECON Research Center, Department of Economics, National Chengchi University, Taipei, Taiwan, e-mail: chchen@nccu.edu.tw

1.1 Autonomy

The first feature, a larger degree of autonomy, has two implications. First, it lessens the burden of model-builders in their intervention or supervisory efforts over these agents. Second, it implies a larger degree of freedom left for agents to explore the environment around them, and a better chance for us to watch how they adapt and what they learn.

The first implication is important when model-builders themselves know very little about the structure of the environment in which their agents are placed, and hence they do not even know how to supervise these agents in a well-defined manner; in particular, they do not want to misinform these agents with biased information. The second implication is even more important because what they learn or discover may be non-trivial for us. In this case, we are taking lessons from them. Alternatively, it makes us able to have the novelties- or surprises-generating processes, an essential element of any complex adaptive system. By observing and making sense of what agents learned, we as outsiders are also able to learn.

1.2 Learning

The second feature is also appealing because it enables us to give an alternative interpretation of what we mean by *learning*. Learning is a highly interdisciplinary concept, which concerns many disciplines, ranging from psychology, education, neural sciences, cognitive sciences, mathematics and statistics, to information sciences.

Its meaning in economics also varies. In some situations, it is very trivial and means nothing more than *making a choice* repeatedly under the same or a very similar environment with the same options. There are a number of learning algorithms corresponding to this simple case. The most famous one is *reinforcement learning*, and the other equally familiar and related one is the *discrete choice model* associated with the Boltzmann-Gibbs distribution. These learning algorithms only involve a very simple stimulus-reaction mechanism, and the development of sophisticated reasoning is not required, at least, not explicitly.

In some other situations, learning means the attempt to find out the law between the causes and the effects, the mapping between the inputs and outputs, and the underlying mechanism by which observations are generated. It is more like a scientific learning. The feedforward neural networks (FNNs) represent such a kind of learning. Numerous mathematical analyses of neural networks show that FNNs are universal function approximators, even though to build such an approximation process is another issue.

However, these two kinds of learning, the stimulus-reaction learning and the scientific learning, may cover only a very limited part of what we generally experience about learning. What has been missing is the idea of the *building block*, which connects what we have learned before to what we are learning now or what we will learn in the near future. In considering the learning of mathematics as an example, we cannot study differential equations without having calculus as the prerequisite. If we perceive learning as a walk along a *ladder* which makes us move higher and become more experienced at each step, then the kind of learning which we are interested in is developmental learning, and genetic programming is one of the learning algorithms which are able to demonstrate this feature.

2 Genetic Programming and Economics

Genetic programming is a methodological innovation in economic. It is so because it captures three essential elements in the making of economics. The three elements are *constant changes* from inner nature to outer forms, *evolving populations* of decision rules, and *modularity*. These three elements have been initiated by three prominent economists at different times. Two of them, Herbert Simon and Robert Lucas, are Nobel Laureates, and the one, who is not, died in 1924 when the Nobel Prize had not yet existed, but who is generally regarded as the father of the neo-classical economics. In what follows, we shall go through them in chronological order.

2.1 Alfred Marshall

The first connection between GP and economics is the idea of *constant change*. Its origin can be traced back to the late 19th century. Alfred Marshall [20] wrote:

> Economics, like biology, deals with a matter, of which the inner nature and constitution, as well as outer form, are constantly changing. (Ibid, p. 772)

He also wrote

> The Mecca of the economists lies in economic biology rather than in economic dynamics. (Ibid, p. xiv)

Alfred Marshall is regarded as a pioneer in starting the dialogue between economics and biology, whose legacy has been further pursued in a branch of economics, referred to as *Evolutionary Economics*. To have an idea of the *constant change* of the inner nature, the constitution, and the outer form of a matter, one can think of the evolution of technology, from its primitive form to its state of the art.[1] Nevertheless, this picture of constant change has not been demonstrated in any model known to economists before the advent of GP. Even the leading economists in Evolutionary Economics did not provide us with a tool to simulate this developmental-biology-like process.

[1] For example, see [3], in particular, Figures 1.3 and 1.4.

2.2 Robert Lucas

The second connection between GP and economics is the idea of *evolving populations.* [19] provided a notion of an economic agent.

> In general terms, we view or model an individual as *a collection of decision rules* (rules that dictate the action to be taken in given situations) and a set of preferences used to evaluate the outcomes arising from particular situation-action combinations. (Ibid, p.217, Italics Added.)

Immediately after the *static description* of the economic agent, Lucas continued to add an *adaptive* (*evolutionary*) version of it.

> These decision rules are continuously under review and revision: new decision rules are tried and tested against experience, and rules that produce desirable outcomes supplant those that do not. (Ibid, p.217).

So, according to Lucas, the essence of an economic agent is *a collection of decision rules which are adapting (evolving) based on a set of preferences.* In brief, it is an idea of an *evolving population.*

If we suppose that an evolving population is the essence of the economic agent, then it seems important to know whether we economists know any operational procedure to substantiate this essence. Back in 1986, the answer was absolutely *no.* That certainly does not mean that we did not know anything about evolving one *decision rule.* On the contrary, since the late 1970s, the literature related to bounded rationality in macroeconomics has introduced a number of techniques to evolve a single decision rule (a single equation or a single system of equations): recursive regression, Kalman filtering, and Bayesian updating, to name a few. [25] made an extensive survey of this subject. However, these techniques shed little light on how to build a Lucasian agent, especially since what we wanted to evolve was not a single decision rule but a population of decision rules.

In fact, it may sound a little surprising that economists in those days rarely considered an individual as a population of decision rules, not to mention attending to the details of its evolution. Therefore, all the basic issues pertaining to models of the evolving population received little, if any, attention. For example, how does the agent *initialize* a population of decision rules? Once the agent has a population of decision rules, which one should they follow? Furthermore, in what ways should this population of decision rules "be continuously under review and revision"? Should we review and revise them one by one because they are independent, or modify them together because they may be correlated with each other? Moreover, if there are some "new decision rules to be tried," how do we generate (or find) these new rules? What are the relationships between these new rules and the old ones? Finally, it is also not clear how "rules that produce desirable outcomes should supplant those that do not."

2.2.1 John Holland

There is one way to explain why economists are not interested in, and hence not good at, dealing with a population of decision rules: economists used to derive the decision rule for the agent *deductively*, and the deductive approach usually led to only one solution (decision rule), which is the *optimal* one. There was simply no need for a population of decision rules.

We do not know exactly when or how the idea of the *evolving population of decision rules* began to attract economists, but John Holland's contribution to *genetic algorithms* definitely exerted a great influence. In 1991, John Holland and John Miller published a sketch of the *artificial adaptive agent* [16], where they stated

> ...an agent may be represented by a single string, or it may consist of *a set of strings* corresponding to *a range of potential behaviors*. For example, a string that determines an oligopolist's production decision could either represent a single firm operating in a population of other firms, or it could represent *one of many possible decision rules* for a given firm. (Ibid, p. 367; Italics added.)

Now, formally, each decision rule is represented by a *string*, and, at each point in time, agents may have a set of strings characterizing a range of potential behaviors. In this sense, the agents' behavior is no longer deterministic; instead there are many decision rules competing before the final one is chosen.[2]

2.2.2 John Koza

It is interesting to note that the (binary) strings initiated by Holland were originally motivated by an analogy to machine codes. After decoding, they can be computer programs written in a specific language, say, LISP or FORTRAN. Therefore, when a GA is used to evolve a population of binary strings, it behaves as if it were used to evolve *a population of computer programs*. If a decision rule is explicit enough not to cause any confusion in implementation, then one should be able to write it in a computer program. It is the *population of computer programs* (or their machine codes) which provides the most general representation of the *population of decision rules*. However, the equivalence between computer programs and machine codes *breaks down* when what is coded consists of the parameters of decision rules rather than the decision rules (programs) themselves, as we often see in economic applications with GAs. The original meaning of evolving binary strings as evolving computer programs is lost.

The loss of the original function of GAs has finally been noticed by John Koza. He chose the language LISP as the medium for the programs created by genetic programming (GP) because the syntax of LISP allows computer programs to be manipulated easily like the bitstrings in GAs, so that the same genetic operations

[2] Whether or not the mind of an agent can simultaneously have many different competing ideas or solutions is certainly an issue not in the realm of conventional economics, but a subject long studied in psychology, neuroscience, and the philosophy of the mind. See also [21].

used on bitstrings in GAs can also be applied to GP. Genetic programming simulates the biological evolution of *a society of computer programs*. Each of these computer programs can be matched to a solution to a problem. This structure provides us with an operational procedure of the Lucasian agent. First, a collection of decision rules is now represented by a society of computer programs. Second, the review and revision process is implemented as a process of natural selection when the genetic operators are applied to evolve the society of computer programs.

2.3 Herbert Simon

The third connection of GP to economics is the idea of complexity, in particular, the Simonian notion of complexity [26], i.e., *hierarchy*. Herbert Simon viewed hierarchy as a general principle of complex structures. Hierarchy, he argued, emerges almost inevitably through a wide variety of evolutionary processes, for the simple reason that hierarchical structures are *stable*. To demonstrate the importance of a *hierarchical structure* or *modular structure* in production, Simon offered his well-known story about a competition between Hora and Tempus, two imaginary watchmakers. In this story, Hora prospered because he used the modular structure in his design of watches, whereas Tempus failed to prosper because his design was not modular. Therefore, the story is mainly about a lesson: the advantage of using a modular design in production.

Modularity is becoming more important today because of the increased complexity of modern technology. Using the computer industry as an example, [2] shows that the industry has experienced previously unimaginable levels of innovation and growth because it embraced the concept of modularity. [17] also asserts that embracing the principle of modular design can enable organizations to respond rapidly to market needs and allow the changes to take place in a cost-effective manner.

3 What is Missing in ACE?

The three ideas individually already had an impact on the later development of economics. For example, after Marshall, through the additional efforts made by Thorstein Veblen, Armen Alchian, Richard Nelson, Sidney Winter, and many others, the ideas of evolution have been brought into the modeling of economics. Recently, much of this progress has been further made in *agent-based computational economics* (ACE), where we can see how the Lucasian agent has been brought into evolutionary economics via genetic programming [9, 10, 11, 8, 7, 12].

However, the central element on *constant change in the inner nature and outer form* has largely been missing in this literature. As we have seen above, Simon's work on modularity also concerns evolution. How Simon's view of evolution in terms of *modularity* can be related to Marshall's view of evolution in terms of *con-*

stant change is also missing in the literature, even though a reflection of human history does indicate that our economy evolves toward higher and higher degree of complexity and novelty. The idea of *hierarchical modularity* should then play a central role as the economy evolves with these features. Nevertheless, not many ACE models are able to deliver this feature, including those so-called *agent-based economic models of innovation*.[3] To fill the void, there are a number of research questions that need to be addressed. One of these is an in-depth investigation of the relationship between complexity and diversity. The other issue, as a continuation of what has been said in Section 1.2, concerns a learning algorithm enabling our autonomous agents to learn in a developmental or accumulation process through which unsupervised discovering can be expected.

3.1 Complexity and Diversity

The diversity which we discuss in this section is restricted to the production side, in particular, the *product diversity*. It could more broadly include other related kinds of diversity, such as process diversity, organizational diversity, and job diversity, but it is still restricted to the production aspect. This restriction may drive our attention away from other important diversity issues which may appear in the context of, for example, biodiversity, cultural diversity, anthropological diversity, etc.[23, 27]. The reason for making such a restriction is to have a sharp focus on modularity.

Like complexity, diversity is involved because it is an important feature observed in the evolutionary process. Studies have shown that the development of our economy is accompanied by constant increases in product diversity.[4] However, in addition to that, what concerns us more is that the two ideas, diversity and complexity, may not be disentangled. Intuitively speaking, the more diversified an economy is, the more complex it becomes.[5] Assume that without being able to manage the level of complexity required to match a certain level of diversity, the further pursuit of diversity is technologically infeasible; in other words, the incompetence to cope with increasing complexity can be a potential barrier to the realization of a greater diversity. Then the following issue becomes important: if complexity is an inevitable consequence of diversity, and diversity is welfare-enhancing, how can the economy manage its complexity while enjoying the fruits of diversity? Simon already gave us the key for the solution, i.e., using modular design. However, what is lacking is a demonstration of how this modular design can emerge from the economy.

---

[3] For a survey on this literature, see [14].

[4] According to an EPA (Environmental Protection Agency) study conducted in conjunction with the U.N. Task Force On Global Developmental Impact, consumer-product diversity now exceeds biodiversity. See *Onion*, October 21, 1998, Issue 34-12. http://www.theonion.com/content/node/38901

[5] Of course, this statement can not be made seriously without a clear notion of complexity. What we, therefore, propose here is something similar to algorithmic complexity, while with a modification in order to take cognitive constraints of human agents into account.

3.2 Learning of Hierarchical Modularity

One key element to see the emergence of modular design is to have autonomous agents so that they can constantly discover useful modules (building blocks). The next question is how such autonomous agents can be designed. This leads us to some further thinking on learning, given what we have already discussed in Section 1.2.

What do we mean that we *learned*? How do we make sense of what we learn? How do we know or feel confident that we are *learning*? Must sensible learning be *incremental* (i.e., in a developmental process)? If sensible learning is incremental, then how do we compare learning at different stages? What is the role of building blocks or functional modularity in this learning process? How do building blocks or modules help agents to learn and hence to manage the complexity given their severe cognitive constraints?

4 Toward a New Design of Autonomous Agents

4.1 Gram-Schmidt Orthogonalization Process

The *Gram-Schmidt orthogonalization process*, well taught in linear algebra or functional analysis, provides us with a kind of developmental learning. In fact, mathematicians also use the term "innovation" for the orthogonal elements (residuals) extracted from projections. This is because, along this process, each innovation implies the discovery of a new *basis*, which is equivalent to the discovery of a new space. The basis may be taken as a kind of building block. The developmental learning defined by the Gram-Schmidt orthogonalization process can, therefore, be used to think of how to construct a similar discovery or learning process driven by GP.

4.2 Automatically Defined Terminals

Although GP can have a hierarchical modular structure, the simple genetic programming is not good at using the modular structure. The standard crossover and mutation can easily destroy the already established structure, which may cause the whole discovery or learning process to be non-incremental and non-progressive. This problem is well-known in the GP literature, and has been extensively studied with various treatments [1, 15, 18, 24]. Motivated by these earlier studies, [6] proposes *automatically defined terminals* (ADTs) as a way to enhance GP to find structured solutions.

An ADT, as shown in Fig. 1, is very similar to the automatically defined function (ADF) [18]. It itself has a fixed structure, in this case, a tree with a depth of two. The root of an ADT can be any function from the primitives (function set), while its

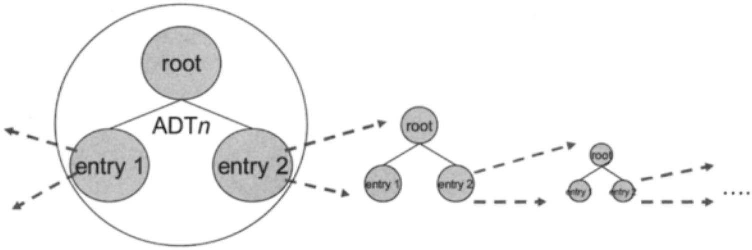

Fig. 1 Automatically defined terminals.

leaf can be either a terminal from the primitives (terminal set) or can be any existing ADTs. In this way, it shares the same spirit as an ADT, namely, *simplification*, *reuse*, and *encapsulation*. The last item is particularly important because it means that whatever is inside an ADT will not be further interrupted by crossover and mutation.

In this way, ADTs can be considered to be the part of learning in which we have great confidence, and which leaves no room for doubt. Through ADTs we distinguish what is considered to be *knowledge* from what is still in a trial-and-error process. Only the former can then be taken as the building blocks (modules), but not the latter.[6] Without ADTs or equivalents, simple genetic programming is essentially not designed to develop building blocks; therefore, it is not very good at finding the modular structure inherent in the problem.

4.3 Modular Economy

[6] tested the idea of augmented GP (augmented with ADTs) in a *modular economy*. The modular economy, which is first proposed in [5], is an economy whose demand side and supply side both have a *decomposable* structure. The decomposability of the supply side, i.e., production, has already received intensive treatments in the literature (See Section 2.3). On the demand side, the modular economy implies a market composed of a set of consumers with *modular preference*. Therefore, it is based on a crucial assumption that *the preference of consumers can be decomposable*. This is, indeed, a big assumption, since its validity has received very little attention in the literature. The closest study which may shed light on this assumption is that the study of *neurocognitive modularity*. The recent progress in neuroscience has allowed us to identify a number of brain modules at various levels of granularity. In addition, various hypotheses regarding *the modularity of mind* also exist, such as the famous *massive modularity hypothesis* [28, 13]. Nevertheless, whether or not one can build preference modules upon the brain/mind modules is still an open issue.

[6] One criterion for modules is their persistence as identifiable units for long enough time spans or generations [22].

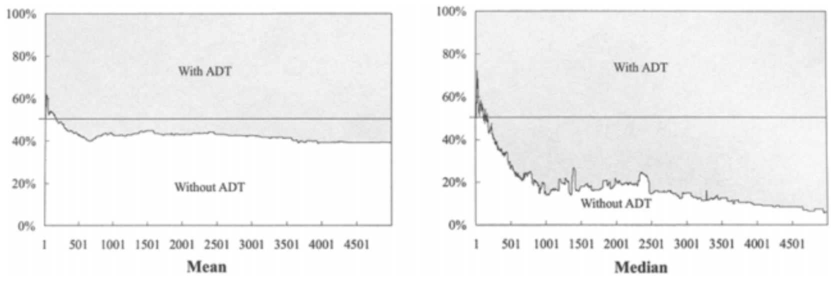

Fig. 2 Modularity and competitiveness.

In the modular economy, the assumption of modular preference is made as a dual relation to the assumption of modular production. Nevertheless, whether in reality the two can have a nice mapping, e.g., a one-to-one relation, is an issue related to the distinction between *structural modularity* and *functional modularity*. While in the literature, this distinction has been well noticed and discussed, "recent progress in developmental genetics has led to remarkable insights into the molecular mechanisms of morphogenesis, but has at the same time blurred the clear distinction between structure and function." ([4], p. 10)

The modular economy initiated by [5] does not distinguish two kinds of modularity, and they are assumed to be the same. One may argue that the notion of modularity suitable for preference is structural, i.e., *what it is*, whereas the one suitable for production is process, i.e., *what is does*. However, this understanding may be partial. Using the LISP parse-tree representation, [5] actually integrated the two kinds of modularity. Therefore, consider drinking coffee with sugar as an example. Coffee and sugar are modules for both production and consumption. Nevertheless, for the former, producers *add* sugar to coffee to deliver the final product, whereas for the latter, the consumers drink the mixture by knowing the existence of both components or by "seeing" the development of the product.

Within this modular economy, [6] considered an economy with two oligopolistic firms. While both of these firms are autonomous, they are designed differently. One firm is designed with simple GP (SGP), whereas the other firm is designed with augmented GP (AGP). These two different designs match the two watchmakers considered by [26]. The modular preferences of consumers not only define the search space for firms, but a search space with different hierarchies. While it is easier to meet consumers' needs with very low-end products, the resultant profits are negligible. To gain higher profits, firms have to satisfy consumers up to higher hierarchies. However, consumers become more and more heterogeneous when their preference are compared at higher and higher hierarchies, which calls for a greater diversity of products.[7]

[7] If the consumers' preferences are randomly generated, then it is easy to see this property through the combinatoric mathematics. On the other hand, in the parlance of economics, moving along the hierarchical preferences means traveling through different regimes, from a primitive manufacturing

The figures show the simulation results of the competing firms in the modular economy based on 100 runs. The main statistics displayed are the mean and median market shares of two competing firms. It can be seen that the AGP firm (the firm using modular design, ADTs) performs better than the SGP firm (the firm not using modular design), as Simon predicted.

5 Concluding Remarks

The design of autonomous agents plays a pivotal role in the further development of agent-based models in economics. The essence of autonomous agents is to own the automatic-discovery capability. This leads us to have a more fundamental thinking of *what to learn* and *how to learn* in light of the evolution of the real economy, in particular, the constant change of the production economy, the product, the technology and the organization. This paper has shown that Simon's notion of *near decomposability* provides an important direction for us to work with, i.e., a *modular economy*. Needless to say, the empirical content and operational details of the proposed modular economy need to be further addressed. Nevertheless, the modular economy guides us to grasping the key to the promising design of autonomous agents. In this paper, we suggest the use of *automatic defined terminals* in GP to design autonomous agents. The agent-based economic models composed of these autonomous agents can, therefore, feature a process of constant change with incessant novelty-findings, which is what the history of our human economy has evidenced.

Acknowledgements An earlier draft of this paper was first prepared as an invited talk delivered at the Argentina Academy of Sciences on May 9, 2007, and was then delivered as a keynote speech at *The 5th International Workshop on Agent-based Approaches in Economic and Social Complex Systems* (**AESCS'07**), Waseda University, Tokyo, Japan, on August 29-30, 2007. The author is grateful to Prof. Ana Marostica, Daniel Heymann, Julio Olivera, Hajime Kita and Takao Terano for their superb arrangements regarding the invitation. With great sadness, I learn that Prof. Ana Marostica passed away in Dec. 2007. This paper is, therefore, written in memory of her, in particular her enthusiastic devotion to academia and to friends.

References

1. Angeline P, Pollack J (1993) Evolutionary module acquisition. Proceedings of the 2nd Annual Conference on Evolutionary Programming. MIT Press, Cambridge, 154–163
2. Baldwin C, Clark K (2000) Design rules: The power of modularity, Vol. 1. MIT Press, Cambridge
3. Basalla G (1988) *The evolution of technology*. Cambridge University Press, Cambridge

---

economy to a quality service economy, from mass production of homogeneous goods to limited production of massive heterogeneous customized products.

4. Callebaut W (2005) The ubiquity of modularity. In: Callebaut W, Rasskin-Gutman D (eds) Understanding the development and evolution of natural complex systems, MIT Press, Cambridage
5. Chen S.-H, Chie B.-T (2004) Agent-based economic modeling of the evolution of technology: The relevance of functional modularity and genetic programming. International Journal of Modern Physics B 18(17-19): 2376-2386
6. Chen S.-H, Chie B.-T (2007) Modularity, product innovation, and consumer satisfaction: An agent-based approach. In: Yin H, Tino P, Corchado E, Byrne W, Yao X (eds.), Intelligent Data Engineering and Automated Learning, Lecture Notes in Computer Science (LNCS 4881), Springer, 1053-1062.
7. Chen S.-H, Liao C.-C (2005) Agent-based computational modeling of the stock price-volume relation. Information Sciences 170:75–100
8. Chen S.-H, Tai C.-C (2003) Trading restrictions, price dynamics, and allocative efficiency in double auction markets: Analysis based on agent-based modeling and simulations. Advances in Complex Systems 6(3):283–302
9. Chen S.-H, Yeh C.-H (1996) Genetic programming learning and the cobweb model. In: Angeline P. (ed.), Advances in Genetic Programming, Vol. 2, MIT Press, Cambridge, Chap. 22, 443–466
10. Chen S.-H, Yeh C.-H (2001) Evolving traders and the business school with genetic programming: A new architecture of the agent-based artificial stock market. Journal of Economic Dynamics and Control 25: 363–393
11. Chen S.-H, Yeh C.-H (2002) On the emergent properties of artificial stock markets: The efficient market hypothesis and the rational expectations hypothesis. Journal of Economic Behavior and Organization 49(2): 217–239.
12. Chen S.-H, Liao C.-C, Chou P.-J (2008) On the plausibility of sunspot equilibria: Simulations based on agent-based artificial stock markets. Journal of Economic Interaction and Coordination 3(1):25-41
13. Dawkins R (1976) The selfish gene. Oxford University Press, Oxford.
14. Dawid H (2006) Agent-based models of innovation and technological change. In: Tesfatsion L, Judd K (eds.), Handbook of computational economics, Vol. 2, North Holland, Amsterdam. 1187-1233
15. Hoang T.-H, Daryl E, McKay R, Nguyen, X.H. (2007) Developmental evaluation in genetic programming: the tAG-based framework. International Journal of Knowledge-based and Intelligent Engineering Systems 12(1):69-82
16. Holland J, Miller J (1991) Artificial adaptive agents in economic theory. American Economic Review 81(2):365-370
17. Kamrani A (2002) Product design for modularity, Springer
18. Koza J (1994) Genetic programming II–Automatic discovery of reusable programs. The MIT Press, Cambridge
19. Lucas R (1986) Adaptive behaviour and economic theory. In: Hogarth R, Reder M (eds) Rational choice: The contrast between economics and psychology. University of Chicago Press, Chicago, 217-242
20. Marshall A (1924) Principles of economics, MacMillan, New York
21. Minsky M (1988) Society of mind, Simon and Schuster, New York
22. Muller G, Newman S eds (2003) Origination of organismal form: Beyond the gene in developmental and evolutionary biology. MIT Press, Cambridge.
23. Page S (2007) The difference: How the power of diversity creates better groups, firms, schools, and societies, Princeton University Press, Princeton
24. Rosca J, Ballard D (1994) Hierarchical self-organizaion in genetic programming. In: Rouveirol C, Sebag M. (eds.), Proceedings of the Eleventh International Conference on Machine Learning. Morgan Kaufmann, San Fransisco
25. Sargent T (1993) Bounded rationality in macroeconomics. Oxford Press, Oxford.
26. Simon H (1965) The architecture of complexity. General systems 10: 63-76
27. Spradlin L, Parsons R (2007) Diversity matters: Understanding diversity in schools, Wadsworth Publishing
28. Williams G (1966) Adaptation and natural selection. Princeton University Press, Princeton