An Interview with John Holland[1]

John Holland was born in 1929 in Indiana. After studying physics at MIT, he worked for IBM where he was involved in some of the first research on adaptive artificial neural networks. He went on to the University of Michigan for graduate studies in mathematics and communication sciences and has remained there ever since. He is Professor of Psychology and Professor of Electrical Engineering and Computer Science at Michigan. Among many important contributions to a number of different fields, mostly related to complex adaptive systems, he developed genetic algorithms and learning classifier systems, foundation stones of the field of evolutionary computing. He is the recipient of a MacArthur Fellowship, a fellow of the World Economic Forum and a member of the Board of Trustees and Science Board of the Santa Fe Institute[2].

This is an edited transcript of an interview conducted on the 17th May 2006.

Phil Husbands: Could you start by saying something about your family background?

John Holland: My father's family came from Amsterdam, way back, so the Holland has some relation to my origins. My mother's family originally came from Alsace in France. My father owned several businesses that all had to do with soy bean processing and my mother often worked as his accountant. She was quite adventurous; in her forties she learned to fly.

PH: Were there any particular influences from early school days or from your family that led you to a career in science?

JH: Not particularly, although my parents always encouraged me and supported my interest, from the first chemistry set they bought me, and in those days these were much more explosive than they are now, through to high-school and beyond. But I grew up in a very small town, with a population of less than nine thousand, so there wasn't much in the way of direct encouragement in science.

PH: You went on to study at MIT; can you say a bit about your time there? Were there particular people you came across who influenced the intellectual direction you took?

JH: There was one person who was very important. I was in physics. At MIT in those days, I think it's still true, you had to do a dissertation for your bachelor's degree. I decided that I wanted to do something that was really quite new: work with the first real-time computer, Whirlwind. Work on Whirlwind was largely classified but I knew someone who was involved: Zdenek Kopal. He had taught me a course on what was then called numerical analysis; now it would be called algorithms. He was an astronomer, working in the Electrical Engineering department, so I went and knocked on his door and he agreed to be the director of my dissertation. He helped me get

---

[1] Preprint of: P. Husbands (2008) An Interview with John Holland, in P. Husbands, O. Holland, M.Wheeler (Eds) The Mechanical Mind in History, MIT Press, 383-396.

[2] Since this interview was first published John Holland has died (August 9, 2015)

double the usual number of hours and I wrote a dissertation – using Whirlwind, getting the necessary security clearances and everything – on solving Laplace's equation using Southwell's Relaxation Method. He later took the first chair of astronomy at Manchester University and had a very distinguished career.

PH: What year was this?

JH: This would be 1949.

PH: So very early as far as modern digital computing is concerned.

JH: Indeed. Whirlwind was only recently operational and was, as far as I know, the first computer to run in real-time with a video display. It was being used for such things as air traffic control, or at least that's what we were told, but it was obvious that it was also being used relative to missile detection and all that kind of stuff.

PH: During your undergraduate days did you come across Wiener, McCulloch or any of the other cybernetics people?

JH: Oh yes, and that had an influence, but a rather distant one. Wiener we saw all the time. He was often called Peanuts because he'd walk down the hall flipping peanuts into the air and catching then in his mouth. So there was some influence there, and I also took a course on Bush's Differential Analyzer which of course got me more interested in computers as well.

PH: What happened next? Did you go straight to graduate school?

JH: No. Because of the Whirlwind, I was offered a very interesting position at IBM in what was then their main research lab at Poughkeepsie, New York. The job was in the planning group for their first commercial computer, the 701. They called it the Defense Calculator. I was one of a group of about eight. For such a young guy it was quite an eye opener. We did the logical planning for the organization of the 701. This was in the very early days of commercial computing - the 701 laboratory models had cathode ray tubes for storage and used punched cards for input. The engineers were building the prototype during the day, and we were testing it using our programs at night. There was a rush because Remington-Rand was also racing to produce a commercial programmed computer. Arthur Samuels and I worked coincidentally at night – I was doing neural nets and he was working on his checkers player[1]. A major part of our logical planning was to make sure that the machine was readily programmable in machine language (remember this was before Fortran). Rochester convinced the Lab Director that these unusual programs (the checkers player and the neural net) gave the machine a good workout. The neural net research came about after J.C.R. Licklider, from ARPA, who knew Hebb's theory of adaptation in the nervous system[2] very well, came through and lectured on it at IBM. Nathaniel Rochester, my boss, and I became quite interested in this and did two separate models which were later published in a single paper[3]. We went back and forth to Montreal at least six or seven times to see Hebb at McGill University while we were developing the model.

PH: Did you interact much with Samuels while you were at IBM?

JH: Yes I did. We met with him regularly at lunch and once every other week we met at his house to play, in rotation, poker, Kriegspiel, and Go.

That time at IBM was obviously an influence on me. I worked for them for eighteen months and then decided I really did want to go to graduate school. IBM was good enough to offer me a consulting contract to help pay my way for four years of graduate school. I would go to school in the winter and go to IBM in the summer. So I came to the University of Michigan, which had one of the best math departments in the country; they had a couple of members of the Bourbaki group and things of that sort. Also, and not totally incidentally, they had a lot of co-eds.

So anyhow I did math and I had actually started writing a dissertation in mathematics – on cylindrical algebras, algebras that extended Boolean algebras to predicate logic with quantifiers – when I met Art Burks. He is certainly one of the big influences in my life. He and others were starting a new programme called Communication Sciences which went all the way from language and information theory through to the architecture of computers. Both MIT and Michigan had Communication Sciences programmes and in both cases they later became Computer Science departments. Art convinced me that this was of great interest to me, and indeed it was, and so I stopped writing my maths dissertation and took another year of courses in areas such as psychology and language, and then did my dissertation within the Communication Sciences programme.

PH: What was the topic of that thesis?

JH: It was called *Cycles in Logical Nets*. Burks and others had set up a kind of abstract logical network, related to McCulloch and Pitts networks, and I wanted to see if I could characterise the kinds of changes you got if you allowed the network to contain cycles, feedback in other words. The thesis was finished in 1959.

PH: Who were the people you interacted with during that time, apart from Burks?

JH: There were quite a few. Someone who was in the same cohort as me was Bill Wang who later went to Berkeley and became a world renowned linguist. Actually quite recently, within the last four or five years, Bill and I have got back together again to build agent-based models of language acquisition; so that was a kind of long-range boomerang. Lakoff, the linguist who's done work on metaphor at the logical level, among other things, was here at Michigan. Gunnar Hok, a man who is not so well known but wrote an important book on information theory, was also someone I interacted with. Anatol Rapoport, well known in game theory and several other areas, was also here at that time. So there was a good spread of people with a real knowledge of many aspects of what we would now probably call complexity.

PH: Yes, and it sounds as if there was also quite a strong flavour of what would become cognitive science.

JH: Oh yes, definitely. Anatol Rapoport, especially, was developing ideas in that direction.

PH: During this period did the group at Michigan interact much with other groups in the USA, for instance at MIT?

JH: Yes. We had summer courses in what was called automata theory; after the first year I directed them. Herb Simon, Al Newell, Marvin Minsky, John McCarthy, they all came and lectured on them, so there was quite a bit of interaction. John McCarthy was also at IBM during the same summer periods as me, so we got to know each other pretty well. In fact he taught me how to play Go. That would be about 1954 or 1955. John was editing the *Automata Studies*[4] book with Shannon at that time.

PH: Do you remember what the spirit was like at the time? What were the expectations of people working in your area?

JH: There were already differences in expectations. But two things that I remember are that there was a fair amount of camaraderie and excitement, and also a bit of challenge back and forth between us. Who knows how much this is coloured by memory, but I think of it as a heady time. I enjoyed it.

PH: These were heady times, as you said, but do you remember if people's expectations were naïve, at least in hindsight, or if the difficulties of the problems were appreciated from the start?

JH: Let me make some observations. By this time, the mid 1950s, there was already a strong belief that you could programme intelligence into a computer. There was already an interesting nascent division, which later became much more prevalent, between what came to be known as the Neats and the Scruffies. The Scruffies were on the East Coast, strangely enough since we tend to think of people from there as pretty neat, and they were going to hack it all in. The Neats were on the West Coast and they wanted to do it by logic – the logic of common sense and all that -- and make it provably correct. The Scruffies didn't believe the problem was tractable using logic alone and were happy to put together partly ad hoc systems. To some extent this was a split in approaches between John McCarthy, in the West, and Marvin Minsky in the East. Interestingly enough, as this unfolded there was very little interest in learning. In my honest opinion, this held up AI in quite a few ways. It would have been much better if Rosenblatt's Perceptron work[5], or in particular Samuels' checkers playing system[1], or some of the other early machine learning work, had had more of an impact. In particular, I think there would have been less of this notion that you can just put it all in as expertise.

PH: The alternative to that, adaptive systems, seem to have been the focus of your attention right from the start of your career. Is that right?

JH: Yes, certainly. A major influence on me in that respect was Fisher's book *On The Genetical Theory of Natural Selection*[6]. That was the first time I really realised that you could do mathematics in the area of biological adaptation.

PH: Was that the starting point for genetic algorithms?

JH: Yes. I came across the book when I was browsing in the open stacks of the Math library. That must have been somewhere around 1955 or 1956. Computer programming was already second nature to me by that time, so once I saw his mathematical work it was pretty clear immediately that that was programmable.

PH: Were you initially thinking in terms of computational modelling of the biology or in terms of more abstract adaptive systems?

JH: Well, probably because of exposure to Rapoport and others, I began to think of selection in relation to solving problems as well as the straight biological side of it. In fact by the time I was doing the final writing up of my thesis I had already gone heavily in the direction of thinking about genetics and adaptive system. So the thesis became pretty boring to me and I wanted to move on to the new stuff.

PH: Once you'd finished your thesis how did you get to start work on what became genetic algorithms, were you given a post-doc position or something?

JH: Well this is where I had a great piece of luck and Art Burks was just superb. The stuff I wanted to do was not terribly popular – the typical comment you'd get was 'Why would want to use evolution to try and solve problems; it's so slow.' – but Art always stood up for me and said 'This is interesting work. Let him get on with it.' So I got a job where I was teaching a couple of courses – logic for the philosophy department and so on – and doing my research. Within a year they made me an assistant professor and in those days you got promoted pretty rapidly so things went on very quickly and I settled in at Michigan.

PH: Almost hidden away in some of the cybernetics writing of the 1940s and 1950s there are several, usually fairly vague, mentions of the use of artificial evolution. For instance Turing in his 1950 *Mind* paper[7]. So the idea was floating around to some extent. Were you aware of any of these? Were they a kind of background influence?

JH: Oh yes. One thing that I came across in retrospect and under analysis from others, at IBM actually, was Friedberg's work on evolving programs[8]. This was a really important piece of work, but it was flawed. One of the people in his own group, Dunham I think, later wrote a paper with him showing that this evolutionary process `was slower than random search[9]. Still, the idea was there; Friedberg was a smart guy. That was of great interest to me because you could see why it didn't work.

PH: Did you come across this after you'd started work on developing your evolutionary approach?

JH: Yes, I'd already read Fisher and gotten interested. A bit later Fogel, Owens and Walsh wrote their book on using evolutionary techniques to define finite state machines for simple predictive behaviours[10]. Again you could easily see why it could go wrong, but it was influential in helping to show the way. So there was something in the wind at the time.

PH: It took a long time for genetic algorithms[11], which developed into the field of evolutionary computing, to become mainstream. When it did, were you surprised? What were your feelings when, in about 1990, it suddenly became enormous?

JH: Yes, it did seem almost explosive at that time. It was surprising. By that time I'd had a lot of graduate students who had finished their degrees with me, so there was a local sphere of influence and we knew there were kinds of problems that could be solved with evolutionary methods that couldn't be solved easily in other ways. But I think that the tipping point, as we'd call it nowadays, was when it became more and more obvious that the kinds of expert systems that were being built in standard AI were very brittle. Our work offered a way around that. So suddenly, partly because people were looking elsewhere for alternatives, and because some of my students had became reasonably well known by then, the whole thing just took off.

PH: That must have been gratifying.

JH: Yes, but there were pluses and minuses to it. It was nice to see after all that time, but on the other hand you begin to get too many phone calls!

PH: Rewinding back to the 1950s, your name is mentioned on the proposal for funding for the Dartmouth Conference as someone who would be invited. But did you actually go?

JH: No I did not. At that time I had heavy commitments at Michigan. I can't remember why I didn't go, because I planned to. But I did not and that was my great loss, because that was a very important meeting.

PH: Yes, and of course it was very influential in advocating what became known as symbolic AI. That seems to go against the grain of the kind of work you have always been involved in. Were you aware at the time that the tide was turning in that direction?

JH: I would say within the year I was aware of it. Not immediately, but fairly quickly, because, even though I wasn't there, there was a lot of back and forth. Notions like adaptation simply got shoved off to one side so any conversation I had along those lines was sort of by-passed. My work and, for instance, Oliver Selfridge's work on Pandemonium[12], although often cited, no longer had much to do with the on-going structure of the area.

PH: Why do you think that happened? Why was the work on adaptive systems and learning sidelined?

JH: John McCarthy and Marvin Minsky are both very articulate and they both strongly believed in their approach. That was part of it. Herb Simon and Al Newell had worked on their Logic Machine[13], so they were oriented in that direction and they were influential. McCulloch and Pitts' network model, even though it was connected to neural networks, was itself highly logical – Pitts was a brilliant logician. This was the time when symbolic logic had spread from philosophy to many other fields and there was great interest in it. But even so, it's still not absolutely clear to me why the other approaches fell away. Perhaps there was no forceful advocate.

PH: Putting yourself back into the shoes of the graduate student of the 1950s, are you surprised how far things have come, or haven't come?

JH: Well, let's see. If I look back and think of expectation from that time, I am surprised. I really believed that by now we would be much better at things like pattern recognition or language translation, although I didn't think we'd get there the logic way. Partly because I had worked with Art Samuels, and had great respect for him, I really believed that taking his approach to playing games, developing it and spreading it into things that were game-like, would make tremendous advances within a decade or two. But what we have today is Deep Blue[14], which doesn't use pattern recognition at all, and we still don't have a decent Go playing programme.

PH: Why do you think that is? Because the difficulty of the problems was underestimated?

JH: I think that's part of it. In my opinion, those problems can't be solved without something that looks roughly like the human ability to recognise patterns and learn from them.

PH: We've already discussed the sudden popularity of genetic algorithms, but a lot of other related topics came to the fore in the late 1980s and early 1990s. The rise of artificial life, complex systems theory, nouvelle AI and the resurgence of neural networks all happened at about that time, and there was the founding of the Santa Fe Institute in the mid-1980s. You were involved in most of those things. At least in AI, the switch from the mainstream to topics that had been regarded as fringe for a long time seemed quite sudden. Was there a shift in scientific politics at this time, or some successful lobbying? Or something else?

JH: I think the Santa Fe Institute is a good way to look at this tipping point; I think its founding says a lot about what was happening. Let me make a comparison. Just before World War II, there was this really exceptional school in logic in Poland, the Lwów-Warsaw School of Logic, and many of the best logicians in the world came out of there. The Santa Fe Institute seemed similar to me in that it depended a lot on a very few people. George Cowan, a nuclear chemist from Los Alamos, had the idea to set up the institute. He thought there were a group of very important problems, that weren't being solved, which required an interdisciplinary approach – what we would now call complexity. He recruited Murray Gell-Mann and together they brought in three other Nobel laureates and they decided they should start an institute that wasn't directly connected to Los Alamos so there would be no classification and security problems. It was originally called the Rio Grande Institute. About a year and a half later they decided it should be located in Santa Fe and renamed it the Santa Fe Institute. That really did start something. As I often say to graduate students, research at the Santa Fe Institute was how I imagined research would be when I was a young assistant professor just starting out, but it wasn't until the institute was set up that I really got engaged in the way that I had dreamed of as a young guy. It still is an extremely exciting place. The first major impact we had was when we got a group of people together to discuss how we might change economics. The group included people like John Reed, who was the CEO of Citicorp, Ken Arrow, Nobel laureate in economics, Phil Anderson, Nobel laureate in theoretical physics, who had a real interest in economics, a bunch of computer scientists and some others. We got together for a week and produced some interesting ideas about viewing the economy as a complex system.

PH: Did you have much of an interest in economics before that? You've done quite a bit of work in that area since.

JH: I got interested in economics as an undergraduate at MIT where I took the first course that Paul Samuelson offered in the subject. Samuelson was a great teacher – his textbook became a huge classic – as well as a great economist (he went on to get a Nobel Prize). But I hadn't really done any work in the area until the Santa Fe meeting. Ken Arrow was a great influence on me – the Arrow-Debreu model is the basis of so much of modern economics but Ken was ready to change it. He said, 'Look, there's this wrong and this wrong.' Interacting with him was really good. Anyway, I think the energy and intellectual excitement of the Santa Fe Institute, which involved some highly regarded and influential people, played an important part in shifting opinion and helped to catalyse changes in outlook in other areas. That was a very exciting period.

PH: In my opinion, the spirit of your work has always seemed close to that of some parts of cybernetics, perhaps not surprisingly, given when you started. It often reminds me of the work of people like Ross Ashby. Is that a fair link?

JH: Yes. I certainly read his books avidly and there was a group of people including Ashby, Bertalanffy and the General Systems theorists, Rapoport, Rashevsky, who was here for a while – not to mention von Neumann and Art Burks – who created a whole line of thought that was influential for me. Art Burks actually edited von Neumann's papers on cellular automata so we were seeing that stuff before it was published.

PH: Those are some of the main names we would associate with the beginnings of systems theory and complexity.

JH: That's right. Someone else I should mention is Stan Ulam, the great mathematician who invented the Monte Carlo method among many other things. At the time the Santa Fe Institute was founded he was still alive, but he died soon thereafter and his wife donated his library to the institute. For a while all his books were collected together on a few shelves so you could go in and pick them out. He had a habit of making notes in the margins. This was about the first time I've been able to almost see into someone's mind, following the way it works. Ulam was just exceptional.

PH: Let's concentrate on the present for the final part of this interview. What do you think are the most import problems in evolutionary computing today?

JH: Well I think a really deep and important problem is what has come to be called evo-devo: evolutionary development. I think a lot of the framework we have is relevant to that problem in biology. At the moment most of the discussion on evo-devo is sort of like evolutionary biology pre-Fisher – a broad framework, some useful fact, but nothing like Fisher's mathematical framework.

PH: That's very interesting. So you think there is a bigger role for evolutionary computing in theoretical biology?

JH: I think it's quite possible, especially when combined with agent-based modelling of complex adaptive systems. A major effort at the Santa Fe Institute, and one I am involved in, is developing those kinds of studies of complex adaptive systems involving multiple agents that learn. Evo-devo has got to be heavily related to that. You can really think of developmental processes, where the cells in the body modify themselves and so on, as a complex adaptive system where agents are interacting – some agents stop others from reproducing and things like that. It seems to be a natural framework for development.

PH: Extrapolating a bit, do you think that if we are going to use evolutionary methods to develop machine intelligence, development will have to be taken seriously? That it will be an important part of the story?

JH: Yes I do, very much so. A nice basic project in that direction might be to try and develop a seed machine – a self-replicating machine out of which more complex systems could develop – or at least the theory for one. NASA have already put a lot of money into this kind of thing, and it won't be easy or happen quickly but I think it should be doable and would be a good goal to set up in looking at evo-devo.

PH: Related to this, more generally what do you think the relationship between computer science and biology should be? Should they get closer or be wary of that?

JH: I'm a very strong advocate of cross-disciplinary research. My own idiosyncratic view is that the reason many scientist burn out early is that they dig very deep in one area and then they've gone as far as it's humanly possible at that time and then can't easily cross over into other areas. I think at the heart of most creative science are well thought-out metaphors, and cross-disciplinary work is a rich source of metaphor. Although you've got to be careful; metaphors can be over-hyped.

PH: A slightly different angle, particularly in relation to AI, is the notion that the only way we are ever going to make significant progress is to learn from biological systems.

JH: In a way I do agree with that. If we go back to when Ashby and Grey Walter, Wiener, Selfridge and all the others were looking at these problems, they used biological metaphor in a rich and careful way. I do not think that simply making a long list of what people know and then putting it into a computer is going to get us anywhere near to real intelligence. So then you have to ask yourself, what are the alternatives? Artificial neural nets are one possibility, and another is try to work with a mix of cognitive science and agent-based modelling which could be very fruitful for AI and computer science in general. This allows you to work at a more abstract level than trying to reverse engineer biology as some people, I think wrongly, advocate. I become very cautious when I hear people claiming they are going to use evolution and they're going to down-load human brains into computers within twenty years. That seems to me to be at least as far-fetched as some of the early claims in AI. There are many rungs to that ladder and each of them looks pretty shaky!

PH: Without imposing any timescales, how do you see the prospects for AI? Where is it going?

JH: As I mentioned before, it seems to me that very central to this is what we loosely call pattern recognition, and also building analogies and metaphors. My views on this owe a big debt to Hebb. I think that we can, and must, get a better grasp on these rather broad, vague things. My personal view on how to go about this is through agent-based modelling. We have some of the pieces but we need to understand how to take things further. I think Melanie Mitchell's work with Doug Hofstadter on Copycat[15,16] points the way to a much different approach to notions such as analogy. Central is the need to get much better at recognising patterns and structures that repeat at various levels. One thing that we haven't done much with so far is tiered models where the models have various layers. I think all of these things fall roughly under the large rubric of complex adaptive systems

PH: Looking back at all the work you have been involved in, is there one piece that stands out?

JH: I guess I really feel good about the mixture of rule-based systems and genetic algorithms that I called classifier systems[17,18]. That injection of flexibility into rule-based systems was something that really appealed to me at the time. Many of the people working with production systems, and rule-based systems in general, knew that they were brittle. The notion that you could take rules but make them less brittle, able to adapt to changes, was very pleasing. In a way classifier systems were the genesis of the agent-based modelling work at Santa Fe. When we started on the economic modelling work, economists like Brian Arthur and Tom Sargent started using classifier systems. I tried to collect many of these ideas, in a form available to the general, science-interested reader in the inaugural set of Ulam Lectures, published as *Hidden Order*[19].

[1] Samuels, AL (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research & Development*, **3**(3), 210-229
This is a landmark paper in machine learning and adaptive approaches to game playing systems.

[2] Hebb, D.O. (1949), *The Organization of Behavior*, John Wiley & Sons, New York. This highly influential book introduced, among many other things, Hebb's description of a fundamental adaptive process postulated to occur in the nervous system: connections between neurons increase in efficacy in proportion to the degree of correlation between pre- and post-synaptic activity.

[3] Rochester, N., Holland, J., Haibt, L., and Duda, W. (1956). Tests on a cell assembly theory of the action of the brain, using a large scale digital computer. *IRE Transactions of Information Theory*, IT-**2**:80-93.
This was one of the very first papers on an artificial neural network, simulated on a computer, that incorporated a form of Hebbian learning.

[4] Shannon, C.E. and McCarthy,J. (1956) *Automata Studies* , Princeton University Press.

[5] Rosenblatt, F. (1958) The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain, *Psychological Review*, **65**(6), 386-408.

[6] Fisher, R.A. (1930) *On The Genetical Theory of Natural Selection*, Clarendon Press.

[7] Turing, A.M. (1950) Computing Machinery and Intelligence, *Mind* **49**:433-460.

[8] Friedberg, R. M. (l958). A learning machine: Part I. *IBM Journal of Research and Development*, **2**(1) 2-13,

[9] Friedberg, R. M., Dunham, B., and North, J. H. l959. A learning machine: Part II. *IBM Journal of Research and Development*, **3**(3) 282-287.

[10] Fogel, L., Owens, A. and Walsh,M. (1966) Artificial Intelligence through Simulated Evolution. John Wiley, New York.

[11] Holland, J. H (1975). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. University of Michigan Press. (2nd edition, MIT Press, 1992)
This seminal work on genetic algorithms was the culmination of more than a decade of research.

[12] Selfridge, O.G. (1959), Pandemonium: A paradigm for learning. In Blake, D., Uttley, A., (Eds),*The Mechanisation of Thought Processes. Volume 10 of National Physical Laboratory Symposia*. Her Majesty's Stationary Office, London, 511-529.

[13] A. Newell and HA Simon. The logic theory machine. In IRE Transactions on Information Theory IT-2, pages 61--79, 1956.

[14] Campbell,M., Hoane, A. and Hsu, F. (2002) .Deep Blue, *Artificial Intelligence,* **134**(1-2), 57-83.

[15] Mitchell, M. (1993). *Analogy-Making as Perception: A Computer Model*, MIT Press.

[16] Hofstadter, D. R. and Mitchell, M. (1995). The copycat project: A model of mental fluidity and analogy-making. In Hofstadter, D. and the Fluid Analogies Research group, *Fluid Concepts and Creative Analogies*. Basic Books. Chapter 5: 205-267.

[17] Holland, J.H. and Reitman,J.S. (1978). Cognitive systems based on adaptive algorithms. In D. A. Waterman and F. Hayes-Roth (Eds), *Pattern-directed Inference Systems*. New York: Academic Press, 313-329.

[18] Holland, J.H., Holyoak, K.J., Nisbett, R.E. and Thagard,P.R. (1986) *Induction: Processes of Inference, Learning, and Discovery*. MIT Press.

[19] Holland, J.H. (1995) *Hidden Order: How Adaptation Builds Complexity*. Addison-Wesley, Redwood City, CA .