Artificial Intelligence Evolved from Random Behaviour: Departure from the State of the Art

Wiesław Pietruszkiewicz and Akira Imada

Abstract. Since John McCarthy at MIT coined the term artificial intelligence in 1956 aiming to make a machine have a human-like intelligence in a visible future, we have had lots of discussions whether it is possible in a true sense, and lots of intelligent machines have been reported. Nowadays, the term is ubiquitous in our community. In this chapter we discuss how those proposed machine intelligences are actually intelligent. Starting with how we define intelligence, how can we measure it, how those measurements really represent intelligence and so on, by surveying the Legg and Hutter's seminal paper on formal definition of machine intelligence, we name a few others, taking a brief look at our own too. We also consider a modern interpretation of the Turing test originally proposed in 1950. Then we argue a benchmark to test how an application is intelligent by means of an algorithm for stock market investment as an example. Finally we take a consideration of how we can achieve a human intelligence in a real sense in a real visible future, including an analysis of IT changes stimulating artificial intelligence development.

1 Introduction

The main mission of this chapter is to evaluate artificial intelligence (AI) by exploring definitions of intelligence and different approaches so far proposed, as well as its resemblance to natural intelligence. In our opinion, the term of intelligence is too

Wiesław Pietruszkiewicz
Faculty of Computer Science and Information Technology, West Pomeranian University of Technology in Szczecin, ul. Żołnierska 49, 71-210 Szczecin, Poland
e-mail: `wieslaw@pietruszkiewicz.com`
web: `http://www.pietruszkiewicz.com`

Akira Imada
Intelligent Information Technologies Department, Brest State Technical University, Moskowskaja Street 267, Brest 224017, Belarus
e-mail: `akira-i@brest-state-tech-univ.org`
web: `http://neuro.bstu.by/ai/akira.html`

X.-S. Yang (Ed.): Artif. Intell., Evol. Comput. and Metaheuristics, SCI 427, pp. 19–41.
springerlink.com

often incorrectly assigned to the simple methods of data processing, thus devaluing the notion of AI.

AI is a Holy Grail for many researchers and for the past half of century it was assumed that humans will be able to create a machine–based resemblance of intelligence. For many years it has been tried by creating sophisticated algorithms, which were supposed to imitate natural processes of intelligence, being formed into arbitrary equations. Unfortunately, none of these research succeeded in something we could consider to be a form of real human–like machine intelligence. However, in the latest years yet another idea to realize AI emerged. It aims at the creation of biologically inspired evolving processes, where simple random–driven algorithms, very often using multiple instances, might be thought to bring us closer to the real artificial intelligence. Hence, we would like to compare these two different ideas of AI and explain their assumptions, applications, advantages and disadvantages.

Finally, we would like to highlight the directions of future development of AI by explaining how new findings in science, improvement of algorithms, and stimulation's in software and hardware industries will lead us to further AI development.

2 Artificial Intelligence vs. Natural Intelligence

An excellent survey of this topic by Legg and Hutter[38] gives us a comprehensive bird's-eye view on what is intelligence, how can it be measured, and so on. We now take a brief look at it in the following three subsections.

2.1 Definition of Human Intelligence

What usually reminds us of, when we say human intelligence, might be IQ test. Standard IQ tests measure levels in various cognitive abilities such as reasoning, association, spatial recognition, pattern identification etc. Statistical correlation of these abilities is called *g-factor*, meaning a factor of general intelligence, coined by Charles Spearman[68]. In a situation in schools indeed, this *g-factor* is quite a good estimation. "She is more intelligent than he is," implies "she has higher *g* value than he has." However, we also say "He is very intelligent," for a football player, a conductor of a symphony orchestra, a chef in a restaurant, etc. Hence a standard IQ test does not represent a general intelligence.

Legg and Hutter [38] collected tens of definitions of human intelligence. Let us quote just one, among others, by Nicer et al. [47].

> *Individuals differ from one another in their ability to understand complex ideas, to adapt effectively to the environment, to learn from experience, to engage in various forms of reasoning, to overcome obstacles by taking thought.*

2.2 Informal Definitions of Machine Intelligence

As informal definitions of machine intelligence, Legg and Hutter quote from Albus[3]:

> *Ability of a system to act appropriately in an uncertain environment, where appropriate action is that which increases the probability of success, and success is the achievement of behavioural subgoals that support the system's ultimate goal.*

to which Legg and Hutter added, *"This is especially similar to ours."* (In the next subsection, we can see how similar it is.) Or, from Gudwin[25]:

> *Intelligent systems are expected to work, and work well, in many different environments. Their property of intelligence allows them to maximize the probability of success even if full knowledge of the situation is not available. Functioning of intelligent systems cannot be considered separately from the environment and the concrete situation including the goal.*

Further, from Poole [56]:

> *An intelligent agent does what is appropriate for its circumstances and its goal, it is flexible to changing environments and changing goals, it learns from experience, and it makes appropriate choices given perceptual limitations and finite computation.*

2.3 Formal Definitions of Machine Intelligence

Legg and Hutter [38] wrote, *"One perspective among psychologists is ... that intelligence is the ability to deal with complexity. ... if we could formally define and measure the complexity of test problems using complexity theory we could construct a formal test of intelligence. The possibility of doing this was perhaps first suggested by Chaitin. ... Essentially this is the approach that we have taken."* In fact, Chaitin [10] suggested a possibility of defining a machine intelligence exploiting Gödel's complexity theory, writing *"Develop formal definitions of intelligence and measures of its various components; apply information theory and complexity theory to AI,"* as one of his directions for future research.

2.3.1 Legg and Hutter's Universal Intelligence of an Agent

Now let us see Legg and Hutter's formal definition a little more in detail, since all other formal definitions mentioned in this section are based on this definition by Legg and Hutter more or less.

We now summarize it by paraphrasing their paper. Legg and Hutter start with an informal definition of intelligence:

> *An ability to achieve goals in a wide range of environments.*

An agent behaves in an environment with a goal. A behaviour starts with an observation o_1, then receives an information of how-good-or-bad-is-current-situation, which is called a reward, r_1, and make an action a_1. Repeating this procedure creates a sequence,

$$o_1 r_1 a_1 o_2 r_2 a_2 o_3 r_3 a_3 \cdots, \tag{1}$$

called a history. Now let's define an agent π. The agent takes the current history as input and decides the next action as output. Thus agent π is formally represented as a probability measure of next action given a history before the action. For example,

$$\pi(a_2|o_1r_1a_1o_2r_2) \tag{2}$$

is a probability measure of the 2nd action of the agent. Further, environment μ is defined as the probability of o_kr_k given the current history

$$o_1r_1a_1o_2r_2a_2\cdots o_{k-1}r_{k-1}a_{k-1}, \tag{3}$$

that is,

$$\mu(o_kr_k|o_1r_1a_1o_2r_2a_2\cdots o_{k-1}r_{k-1}a_{k-1}). \tag{4}$$

If we have a multiple paths to the goal, the simplest one should be preferred, which is sometimes called the principle of Occam's razor. Then, the formal measure of success of an agent π under the environment μ denoted as V_μ^π is defined as the expected value of the sum of rewards that is:

$$V_\mu^\pi = E(\sum_{i=1}^{\infty} r_i). \tag{5}$$

Then the measure of the complexity of environments should be expressed. For the purpose, let's recall that the Kolmogorov complexity of a binary string x is defined as the length of the shortest program that computes x. That is,

$$K(x) = \min_p \{l(p)|U(p) = x\}, \tag{6}$$

where p is a binary string which we call a program, $l(p)$ is the length of this string in bits, and U is a prefix universal Turing machine.

We express μ_i as a binary string by a simple encoding algorithm. Then, the complexity of μ_i is $K(\mu_i)$. To formalize above mentioned Occam's razor we use this in the form of probability distribution $2^{-K(\mu)}$. Let E be the space of all environments under consideration. Thus, the expected performance of agent π with respect to the universal distribution $2^{-K(\mu)}$ over the space of all environments E is:

$$\gamma(\pi) = \sum_{\mu \in E} 2^{-K(\mu)} \cdot V_\mu^\pi. \tag{7}$$

In other words, weighted sum of the formal measure of success in all environments where the weight is determined by the Kolmogorov complexity of each environment.

We now recall the starting informal definition: *'an ability to achieve goals in a wide range of environments.'* In the above equation, *'the agent's ability to achieve'* is represented by V_μ^π, and *'a wide range of environments,'* by E – all well defined environment in which reward can be summed. Occam's razor is given by the factor $2^{-K(\mu)}$. Thus the authors called this the *universal intelligence* of agent π.

It is concluded that *"Essentially, an agent's universal intelligence is a weighted sum of its performance over the space of all environments. Thus, we could randomly generate programs that describe environmental probability measures and then test the agent's performance against each of these environments. After sampling sufficiently many environments the agent's approximate universal intelligence would be computed by weighting its score in each environment according to the complexity of the environment as given by the length of its program. Finally, the formal definition places no limits on the internal workings of the agent. Thus, we can apply the definition to any system that is able to receive and generate information with view to achieving goals."*

2.3.2 Other Formal Definitions of Machine Intelligence

Legg and Hutter survey Smith's proposal [66] as *"another complexity based formal definition of intelligence that appeared recently in an unpublished report."* It uses polynomial time reentrant algorithm called *problem generator* which uses random bits and spits out an infinite sequence of output bitstrings called *problem* and also spits out a second bitstring called the *secret answer*. It also uses algorithm called *solution checker* which reads the *problem* and *secret answer* spit out by the *problem generator*. Thus the *entity under test* which allowed to see problem and solve it, is tested by the *solution checker*. Author wrote, *"Both Hutter's and this development exhibit some striking similarities, but we had both different attitudes and different terminology and in some cases investigated different topics or reached differing conclusions."*

Recently, Hernandez-Orallo and Dowe also proposed a modified version of Legg and Hutter's measure [31]. Much more recently, Hibbard proposed yet another approach to define and measure machine intelligence [32, 33] in which intelligence measure is defined in both Turing machine and finite state machine models. This is also principally based on Legg and Hutter's definition. We will return to this model by Hibbard more in detail in later section.

3 A Thought on Artificial Intelligence So-Far-Proposed

We have had a plenty of propositions each of which claims a realization of machine intelligence more or less. Hence, despite Legg and Hutter [38] wrote *"Intelligence is not simply the ability to perform well at a narrowly defined task,"* we are sometimes curious to know whether those machine intelligences reported so far, which are not universal at all but very domain-specific though, are really intelligent or not, or if so, how intelligent. From this perspective, we want to try to remove summation over different environment form their formal definition of intelligence. That is, we measure the intelligence of agent π for the specific task μ simply by V_{μ}^{π}.

In addition, some of what they call an intelligent machine may indeed perform the given task much more efficiently, effectively, or precisely than human, while we human are not usually very efficient, effective nor precise, but rather

spontaneous, flexible, unpredictable, or even erroneous sometime. When we address a human-like intelligence, we expect somewhat of a different behaviour even when we come across a same situation again than the one as we behaved before, not exactly the same one as before. We don't necessarily expect artificial intelligence to be as efficient, but sometimes expect its flexibility, spontaneity, or unpredictability. Frosini [18] wrote *"... contradiction can be seen as a virtue rather than as a defect. Furthermore, the constant presence of inconsistencies in our thoughts leads us to the following natural question: is contradiction accidental or is it the necessary companion of intelligence?"* Or, as we will mention in a later section *"Intelligence might be well demonstrated by concealing it,"* which Michie described in [44] about Turing's suggestion of machine's deliberate mistakes encouraged in order for the machine to pass the Turing test [72]. From this view point, we want to add:

> *Performance should be different more or less than previous one even when the agent comes across the same situation as before,*

to the Legg and Hutter's informal definition. Note that the above mentioned measure of intelligence V_μ^π does not reflect such a flexibility of human intelligence, but only an efficiency. Therefore, a reformalization of Legg and Hutter's formal definition will be quite a new challenging task, which we have not yet succeeded. The other question is, can we evolve a huge population of random binary string, assuming they can represent π, eventually into an intelligent one with fitness being such an intelligence measure?

4 Artificial Intelligence Evolved from Randomness

Our natural intelligence is a result of a tremendously long time of evolution starting with just a tiny simple mechanism which gave just random movements. Then why not trying a creation of artificial intelligence by an evolution from randomness?

4.1 Machiavellian Intelligence

Machiavellian intelligence (see, e.g., [8]), named after Niccolo Machiavelli - medieval Italian politician, is an intelligence which enables individuals to pursue particular goals by means of social manipulation. Miller [45] wrote, *"Machiavellian intelligence evolves because it lets primates predict and manipulate each other's behaviour,"* and went on *"predictive capacities tend to select for unpredictability in counter-strategies, ... For example, prey animals often evolve 'protean' (adaptively unpredictable) evasion behaviour to foil the predictive pursuit tactics used by their predators,"* and concluded *"sexual selection through mate choice results in adaptations like bird song, whale song, and courtship dances, which could have elaborated primate social proteanism into human creative intelligence."*

This model in which protean behaviour - being unpredictable to evade predator - assumed to be the origin of human intelligence might give us a good motivation to simulate predator-prey games as a meaningful step, not just a toy example.

4.2 Hibbard's Formal Definition Revisited

In this subsection we want to revisit the formal definition of machine intelligence by Hibbard [33]. One reason is, he employed a predator and prey model. The other is, both the agent and environment are represented by finite state machine, which will give us a very appropriate method to simulate the pursuit and evasion game. We now take a brief look at how Hibbard defined a machine intelligence.

In the process of defining a formal definition of machine intelligence, Hibbard modelled predictors and evaders as finite state machines as a more realistic models than Turing machine.

An evader e has a state set S_e, an initial state I_e, and a mapping

$$M_e = B \times S_e \to S_e \times B, \tag{8}$$

where B is a binary alphabet.

Similarly for predictor p, state set S_p, initial state I_p, and mapping

$$M_p = B \times S_p \to S_p \times B \tag{9}$$

are specified. Evader e creates a finite binary sequence $x_1 x_2 x_3 \cdots$, and predictor p creates also a finite binary sequence $y_1 y_2 y_3 \cdots$. A pair of evader e and predictor p interacts where e produces the sequence according to

$$x_{n+1} = e(y_1 y_2 y_3 \cdots y_n), \tag{10}$$

and p produces the sequence according to

$$y_{n+1} = p(x_1 x_2 x_3 \cdots x_n). \tag{11}$$

Then predictor p wins round $n+1$ if $y_{n+1} = x_{n+1}$ and evader e wins if $y_{n+1} \neq x_{n+1}$.

4.3 Avidian

Recently, a self-replicating synthetic life was artificially created as a world's first synthetic form of life. They inserted synthetic DNA into Mycoplasma capricolum cells and found those cells had grown into colonies [20].

Much earlier, in 1990's, we had a digital version of this experiment in computer, called Avidian. Inspired by Ray's Tierra [58], a population of self-replicating computer programs, called *digital organisms*, in a computational environment in which the population evolves as the organisms replicate, mutate and compete for resources in the environment [1, 2, 39, 49, 52]. Instructions that made up digital organisms are designed to be robust to mutations so that any program will be syntactically legal when mutated [48]. The world is a discrete two-dimensional grid of cells in which at most one organism may occupy. The genome is a circular list of program instructions that resemble assembly language, that runs in a virtual central processing unit. When an organism replicates, its offspring is placed into a random grid cell, and either the offspring and previously occupied organism survives in the cell. Thus, the

organisms compete for the limited set of grid cells, and organisms that are able to replicate more quickly will more likely to have a greater proportion of descendants within the population.

Under this circumstance, Grabowski tried to model gradient following behaviour of *E. coli* [24]. Grabowski made the other experiments expecting an evolution of simple intelligence and found digital organisms evolved to exploit memory [23, 22].

5 A Modern Interpretation of Turing Test

In 1950, Turing [72] posed a question *"Can machines think?"* and proposed a test which is now called Turing Test. Turing test is a test if a computer can pass then we should grant it is intelligent thereby, or equivalently, a test to see if a computer can cheat a human via a chat with teletype that it is a human. It was originally proposed as the *Imitation game*, in which a man and a woman are in two separate rooms and communicate with an interviewer[1] outside only via a teletype, and the interviewer should identify which is the man by asking a series of questions. The man tries to make the interviewer believe he is the woman while the woman tries to make the interviewer believe she is woman. Later the man is replaced by a machine. If the interviewer cannot tell the machine from the person, then it passes the test and we can say machine is intelligent. Note that the test only gives us a sufficient condition for intelligence. We now briefly see a chronicle of reflections on the Turing Test.

5.1 During 50 Years Since the Proposal

Not a few discussions - some positive, some negative - have taken place since Turing proposed the test [72]. Let's name a few.

Gunderson [26] asked "Can rocks imitate?" by showing a modified Turing's imitation game as follows. A man and a woman are in a room. There is a small opening at the bottom of the wall through which the interviewer can place his toe. The interviewer must determine which of the two in the room is the woman just by observing the way in which his toe is stepped on. Then a rock given an electric eye is replaced with the man in the room, and the rock can put itself softly on the interviewer's toe placed in the opening of the wall. Even if the rock plays this toe-stepping game very well it would not be acceptable that the rock imitates.

Gunderson pose another scenario also in [26]. A vacuum cleaner salesman visited a house and recommended a housewife to buy his vacuum cleaner claiming this is *'all purpose'* by demonstrating how it can suck up bits of dust. The housewife asked, "What else? Isn't it all-purpose? What about bits of paper or straw or mud? I thought sucking up bits of dust was an example of what it does." The salesman failed to show more than one example of what it does." Gunderson thought that the term "thinking" in the Turing test is used to represent more than one capability.

[1] In Turing's original paper the term "interrogator" was used instead of "interviewer."

Yet another argument to pose a doubt for Turing Test is the *Seagull Test* by French [16]. One day in an isolate island, where the only flying animals known to the inhabitants are seagulls, two resident philosophers discuss what flying is all about. After arguing about a pebble tossed from the beach into the ocean, clouds in the sky, balloons, kite, and penguins, one asked the other to assume someone invented a machine that can fly. And they hit upon a test with two 3-D radars one of which tracks a seagull the other tracks the putative flying machine. They concluded the machine will be said to have passed the seagull test for flight if both philosophers are indefinitely unable to distinguish the seagull from the machine.

Purtill [57] denied the Turing's imitation game as a piece of science fiction. Hayes and Ford [30] criticized the Turing Test even as harmful for artificial intelligence to be developed.

Probably one of the most famous criticism is the Chinese Room argument [65] posed by John Searl, philosopher, which conclusively asserts that it is impossible for computers to understand language or think. Suppose now a person who knows only English has a computer program that enables an intelligent conversation in written Chinese by manipulating symbol strings with syntactic rules without understanding semantics, or like a perfect version of Weizenbaum's ELIZA [74], if any. Searl called it *Chinese subsystem*. Then the interviewer outside the room sends a question in Chinese. The people in the room can pass the Turing Test for understanding Chinese while he does not understand any word of Chinese. Similarly the program would not understand the conversation either. Searl wrote, *"Whereas the English subsystem knows that "hamburgers" refers to hamburgers, the Chinese subsystem knows only that "squiggle squiggle" is followed by "squoggle squoggle."*

Harnad also doubted the Turing Test as *Simulating Simulation* and claimed that what is important is not a simulation but an implementation [27]. He denied Searl's claim too. He insisted on removing the wall between the both ends of the teletype link from the interviewer to the machine to be tested. He wrote, *"... mental semantics must be grounded"* [27], which implies the meanings in mind should be derived from interactions with environment. He went on, *"It is like a learning Chinese only with a Chinese-Chinese dictionary, and the trip through the dictionary would amount to a merry-go-round, passing endlessly from one meaningless symbol to another, never coming to a halt on what anything meant."* Thus he extended the Turing test to what he called Total Turing Test in which target machine is a robot with sensorimotors. In this robotic upgrade of the Turing Test the interviewer can visually assessed the machine to be tested, instead of with just a verbal communication via teletype.

In addition to the above mentioned Harnad's Total Turing Test, some researchers also proposed new tests by modifying the original Turing Test such as Harnad's yet another Total Total Turing Test [29], Schweizer's Truly Total Turing Test [64] or Watt's Inverted Turing Test [73]. These are sometimes abbreviated to TTT, TTTT, TRTTT, and ITT, respectively, besides TT to the Turing Test. It might be interesting to see a series of severe discussions after Harnad's refute. For example, Searl's

rebuff and the response by Harnad [28], or other arguments such as Bringsjord[2] vs. Harnad[3]. As for a story from TT to TTT and TTTT, see a review by Fetzer [14]. For a more exhaustive survey on Turing Test, see, e.g., Saygin et al. [62], or French [17]. As a survey positive for the original Turing Test proposed by Turing himself, it might be interesting to read a witty essay recently written by LaCurts [36].

5.1.1 Loebner Prize

We have a contest organized by Hugh Loebner who will pledge $100,000 to the program that succeeds in passing the Turing Test if appeared[4]. The contest started in 1990. Four human judges sit at computer terminals with which the judges can talk both to the program and to the human who tries to mimic computer. Both are in another room and after, say, 5 minutes the judge must decide which is the person and which is the computer. The first computer program that judges cannot tell which is which will be given the award, and then this competition will end. Although a minor award is given every year to the program which responds in most human-like way, as of 2011 the contest has not ended yet, and the contest in 2012 will be held at Bletchley Park, UK.

5.2 An Easy Way to Cheat Human?

One of the easiest ways to make the interviewer believe that the machine is a human, might be a deliberate mistake from time to time pretending not to be too precise to be a human. Turing wrote in [72]:

> *It is claimed that the interrogator could distinguish the machine from the man simply by setting them a number of problems in arithmetic. The machine would be unmasked because of its deadly accuracy. The reply to this is simple. The machine (programmed for playing the game) would not attempt to give the right answers to the arithmetic problems. It would deliberately introduce mistakes in a manner calculated to confuse the interrogator.*

5.3 Turing Test These Days

It had been a long time dream to create a machine which can play chess like human. See, e.g., the book about an eighteen-century chess-playing machine by Standage [69]. The real chess match between a human world champion and a computer - the then world champion Garry Kasparov vs. IBM's Deep Blue - was held in 1996. In a six-game match Deep Blue won one game, tied two, and lost three. Deep Blue was defeated. The next year, Deep Blue again challenged Kasparov also in a six-game match. Kasparov had won the 1st game, lost the 2nd, tied 3rd, 4th and 5th, then lost

---

2 `http://philpapers.org/rec/BRIPAI`
3 `http://www.archipel.uqam.ca/144/2/r-brings.htm`
4 `http://www.loebner.net/Prizef/loebner-prize.html`

the 6th[5]. Thus, finally Deep Blue beat the world champion. Now we know, however, that the Deep Blue won by a brute force rather than with an intelligent strategy.

Turing wrote the first chess computer program, which was called the paper machine because it was before computers even existed. Precedent of his 1950 version, the game was with a mathematician A who operates the paper machine, and two chess player B and C. C plays chess with either A or B both of whom are in the separate room, and C should guess whether he is playing with human or the paper machine [71].

Now it might be easy to imagine a scenario in which A is IBM.'s Deep Blue, B is Kasparov, and C is the current world chess champion. The Deep Blue would be sure to pass the test. See also comments by Crol [35] on Deep Blue vs. Kasparov.

In mid February in 2011, IBM's room size supercomputer called Watson challenged 'Jeopardy' - America's favourite quiz show on TV. In Jeopardy, normally three human contestants fight to answer questions over various topics, with penalties for the wrong answer. The questions are like "Who is the 19th-century painter whose name means police officer?" or "What is the city in US whose largest airport is named for a World War II hero; and its second largest for a World War II battle." [6].

The contest was held over three days with Watson being one of the three contestant and the other two being the ex-champions of Jeopardy - Ken Jennings and Brad Rutter. As Watson cannot see or hear, questions were shown as a text file at the same moment when they were revealed to the two human contestants. By the end of the third day, Watson got $77,147 while Jennings got $24,000 and Rutter $21,600. Watson beat the two human ex-champions. If we set up an appropriate scenario, Watson could pass the Turing Test.

Turing Test is, to simply put, a test to know whether computer can fool human that 'I am a human not a computer!' Nowadays we have a very practical program called CAPTCHA in order to prove 'I'm not a computer but a human.' Actually it stands for 'Completely Automated Public Turing Test to tell Computers and Humans Apart.' This is an acronym based on the English word 'capture.' This is sometimes called a reverse Turing Test. CAPTCHA is exploited by computer with a target being human while Turing test is supposed to be exploited by human with a target being a computer. Nowadays, the original Turing Test is not only of theoretical interest but also as practical as CAPTCHA[7]. For example, a poker playing robot must cheat a web casino site to play there as human. Actually Hingston [34] proposed a new test as follows:

> *Suppose you are playing an interactive video game with some entity. Could you tell, solely from the conduct of the game, whether the other entity was a human player or a bot? If not, then the bot is deemed to have passed the test.*

[5] See, e.g., "Human-computer chess matches" From Wikipedia. `http://en.wikipedia.org/wiki/Human-computer_chess_matches`

[6] This is from the article in New York Times by John Markoff entitled "*Creating Artificial Intelligence Based on the Real Thing*" on 17 February 2011.

[7] Weak CAPTCHAs are possible to be broken by machines using OCR mechanisms. Therefore, creators of CAPTCHAs introduce noise and blurred or distorted text to make this task harder.

6 Biologically Inspired Artificial Intelligence

The creation of man–made intelligent systems or units has two distinctive parts. The first one is a traditional approach, assuming that the secrets of intelligence could be revealed, converted to sets of equations and later programmed. While the second approach, inspired by biology, assumes that the self–adaptive capabilities of flexible structures will allow to adapt themselves to selected problems and to find expected solutions. In this way we don't have to find exact formulas defining behaviour of intelligent system in particular situation, instead giving them a chance to find solution by a partially random behaviour. The term biologically inspired artificial intelligence relate to a wide range of AI algorithms introduced as resemblances of natural processes observed in biological environment. The main groups of Bio–AI include algorithms and systems such as [15]:

- neural – being networks or circuits of information processing units being resemblances of biological neurons, interconnected in organised structures, cooperating in complex information processing tasks. Information flows from one node (or a layer) to another one, being transformed by operations done by previously passed neurons.
- cellular – assuming that multicellular structure will have capabilities unexpected from the isolated units, and this is not a simple effect of scaling–up,
- collective – synergistic interaction of individuals, is done for a common good, e.g. to find food or a better route. In this variant, collective systems (artificial as well as natural ones) perform as one superorganism, more qualified than the sum of its parts' qualifications.
- immune – as living creatures are threatened by pathogens, being external exploiters, they developed protecting immune systems. In their artificial version they protect against external attacks, internal faults and to be used in various information processing tasks perceived from the perspective of system protection.
- evolutionary – where the best (fitted to environment) individuals have a chance to survive and have more offspring. The genes of next generations contain information from their parent, partially modified by a random process of mutation. However, there exists a difference between natural and artificial evolution. In nature, evolution create a vast diversity of creatures (at a certain moment becoming different species), while artificial evolution helps us to produce population satisfying our predefined problems. Therefore, the overall aim of artificial evolution is more similar to e.g. dog breeding, than a random natural process with unexpected outcomes, giving species special abilities helping them to survive.

This chapter focuses on the evolutionary systems, providing a simple benchmark to evaluate their performance in comparison to human intelligence and fully random process. More information about these kinds of algorithm and their usage can be found in [15], [43], [12], [5] or [6]. It must be noted that important difference between the traditional and bio–inspired AI, is the number of elements involved in these processes. The traditional AI usually involves small number of elements (often even one), where each element is expected to perform as good as it could be done.

The bio–inspired AI is built over a large number of elements, where only a subset of them will provide meaningful results. This is an exact situation found in nature, where redundancy and large populations are typical, and the progress of populations development is being driven by small subsets of best individuals.

7 A Benchmark to Evaluate Artificial Intelligence

In his book "A Random Walk Down Wall Street," Malkiel wrote *"a blindfolded monkey throwing darts at a newspaper's financial pages could select a portfolio that would do just as well as one carefully selected by experts"* [41]. Can we evolve this random strategy to an intelligent strategy? For example, Lipinski proposed *a decision support system for stock market trading, which is based on an evolution strategy algorithm applied to construct an efficient stock market trading expert* [40]. This is just one among many such proposals. Then those strategies can be called intelligent? Or they pass the Turing Test?

In this section we would like to investigate the evolutionary algorithms applied for a problem of financial investments – done in form of stock portfolio, creating an optimal structure of financial assets. This is a very well–known task with detailed description presented in many books and papers (to find more about its financial meaning see, e.g. [59] or [13]). These factors caused that portfolio selection was a subject of many research, including experiments performed different evolutionary algorithms, including genetic ones. Among many papers about this area we can mention [50], [67], [7], [21] or [4]. Another important factor that caused we decided to use this task as a benchmark for natural and artificial intelligence comparison is the algorithmic characteristics of portfolio selection. Finally, in the financial practice, an optimal selection of portfolio is a very significant task of financial investment. Therefore, we will examine the relation for natural intelligence of stock investors and evolutionary intelligence.

The most fundamental and widely used approach to optimal selection of financial assets constituting portfolio is the MPT (Modern Portfolio Theory) introduced by Harry Markowitz [42]. In language of mathematics selection of best (optimal) portfolio is a task involving an analysis of expected portfolio efficiency and risk. The most common measure of efficiency is expected return:

$$E\left(R_p\right) = \sum_i w_i E\left(R_i\right), \tag{12}$$

where: R_p is the return of portfolio, R_i is the return on asset i and w_i is the weight of assets.

The standard measure of risk is portfolio volatility calculated as the standard deviation of return:

$$\delta_p^2 = \sqrt{\sum_i \sum_j w_i w_j \delta_i \delta_j \rho_{ij}}, \tag{13}$$

where: δ_i is the standard deviation of returns for i assets and ρ_{ij} is correlation coefficient of returns for i and j assets.

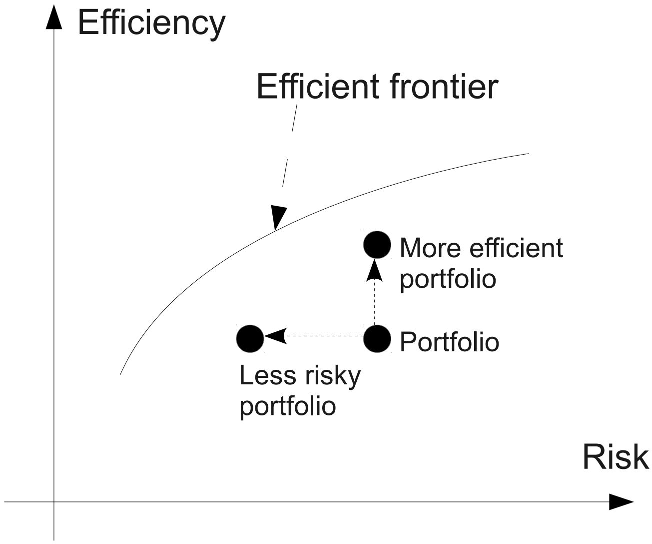


Fig. 1 The schema of risk–efficiency map for portfolio selection

To briefly explain the meaning of this task, let's analyse Figure 1. As we can notice it contains a plane, where X-axis denotes risk and Y-efficiency (both measured according to the rules introduced above). This chart is called a risk–efficiency map, as it provides comparable information about these two parameters. Contrary to the incorrect common opinion, we cannot find an optimal portfolio by searching for one with greatest efficiency or minimal risk, as looking for global extrema will cause irrational selection. This problem requires us to analyse both of these parameters. Indeed, in the two most common scenarios, portfolio optimality is understood as an extreme value of risk or efficiency for portfolios having particular efficiency or risk (accordingly). There exists an efficient frontier for this selection, being a subset of portfolios, each with the highest efficiency for fixed risk or the lowest risk for fixed efficiency.

To examine evolutionary (genetic algorithms) approach to portfolio selection, we have decided to compare three types of portfolios:

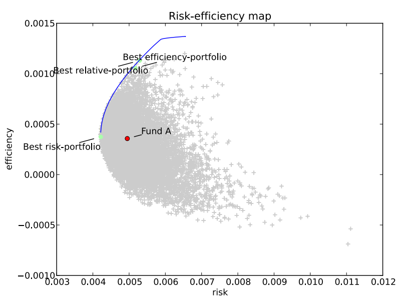
- Portfolios of two financial investment funds – we consider them to be the outcomes of natural intelligence, as they are created by financial experts managing these funds.
- Randomly generated portfolios – randomly distributed are the worst–case scenario, as if portfolio construction is a knowledge requiring task, random selection should give significantly worse results.
- Portfolios generated using genetic algorithms – an evolutionary simple approach, where randomness is driven by algorithm of selection allowing the best portfolios in one generation to be potentially improved in following generations.

The analysis was done for Warsaw Stock Exchange during year 2010, being a quite stable year for WSE (and other markets), as a few previous years were very nervous for stock markets around the world. Thus, they cannot be expected to provide simple and easily understandable evaluation for computational methods of intelligence, as well as for human knowledge.

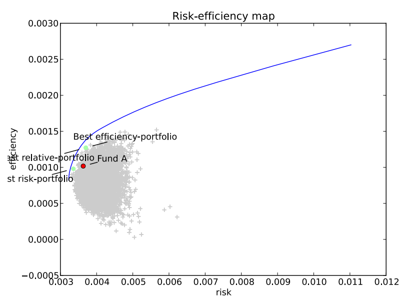
We investigated three scenarios:

1. minimal risk portfolio – where we minimise risk value for portfolios with efficiency equal to a certain value,
2. maximal efficiency portfolio – where we maximize efficiency value for portfolios with a certain risk,
3. minimal risk–to–efficiency ratio portfolio – where the average risk per unit of efficiency should be minimal.

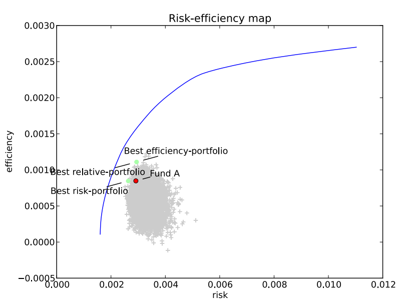
In all cases, we have assumed that the all weights of assets in portfolio sum to one i.e., $\sum_i w_i = 1$ and the short–selling is not allowed i.e., $w_i \geq 0$. The genetic algorithms were used in a variant with roulette-wheel selection (for explanation see [12]), and real–valued N chromosomes used in fitness function (N was a number of stocks) representing weights of particular stocks in portfolio. The fitness function was



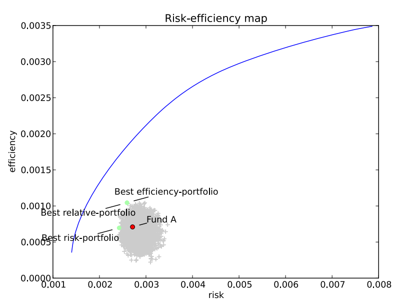
(a) Fund A – 5 stocks



(b) Fund A – 10 stocks



(c) Fund A – 20 stocks



(d) Fund A – all stocks

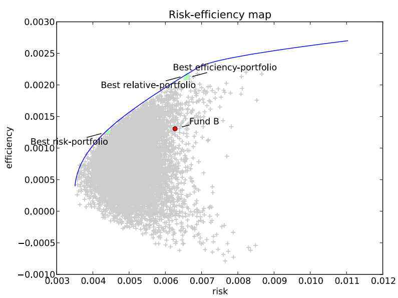
Fig. 2 (a), (b), (c) and (d)

selected to match expected optimality of portfolio (minimal risk, maximal efficiency or minimal risk–to–efficiency).

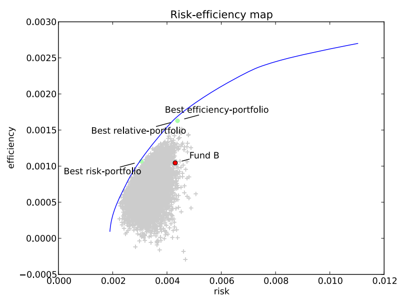
We have performed two stages of experiments for two investment funds (denoted as Fund A and Fund B) oriented on investing in WSE–noted stocks, by examining their portfolios, more precisely subsets WSE stocks (small parts of portfolios was invested on other markets or in bonds). The results of these experiments can be found in Figures 2 and 3. Both Figures contain four subfigures presenting analysis done for 5, 10, 20 and all stocks. They contain frontiers, random portfolios (grey areas) and genetic portfolios.

Analysing these Figures, we have made observations for:

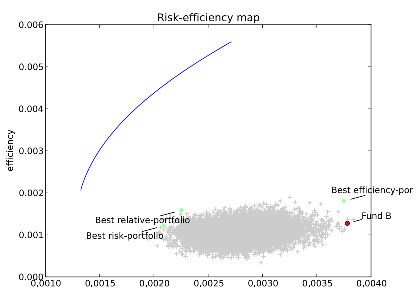
- Funds – being not so optimal as it was expected in terms of MPT. However, we do not want to criticise human experts as their selection might have been a result of analysis beyond MPT theory.
- Random portfolios – random selection of portfolios, as it was expected, resulted in mean results. Interesting observation is that with increasing number of stocks, random areas were more distant from efficient frontier and it could mean that optimal selection is harder to perform (small changes in the weights of portfolio result in its suboptimality).



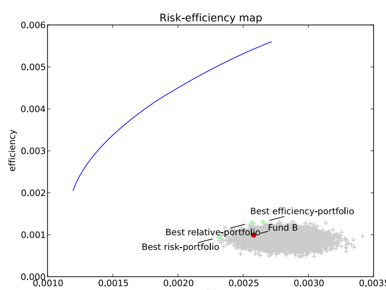
(a) Fund B – 5 stocks



(b) Fund B – 10 stocks



(c) Fund B – 20 stocks



(d) Fund B – all stocks

Fig. 3 (a), (b), (c) and (d)

- Genetic portfolios – achieved better (according to the MPT theory) results for all three scenarios in both cases (Fund A and B). However, we have observed their weak spots too, i.e. a number of adjustable parameters and core algorithm – both to be selected and tuned by human operator, and larger demand on computer resources. This last problem, in the context of processing limits, will be discussed in the following section and we must remember about significant risk of sub–optimality for classic optimisation algorithms. The partially positive influence of random component in the optimisation of complex functions was discussed in [55].

8 To Aim a Real Human-Like Machine Intelligence

In this section we will discuss ideas, research and technological changes influencing further development of artificial intelligence. Together with reorientation of AI on bio-inspired algorithms they might cause that the term of machine intelligence will become more realistic.

8.1 Huge Number of Neurons–From Emulation to Simulation

Recently, IBM's researchers unveiled a project called SyNAPSE (Systems of Neuromorphic Adaptive Plastic Scalable Electronics) in which experimental computer chips which emulate the brain was awarded 21 million US dollars from the Defense Advanced Research Projects Agency (DARPA). Currently prototype contains 256 neurons and 262,144 programmable synapses and 65,536 synapses for learning[8].

On the other hand, simulating brain by a program, instead of emulating brain by hardware, also has attracted, and still attracts, researchers. One such idea is evolving artificial neural networks. Direct encoding of artificial neural networks, where structure and/or all the synaptic strengths are directly encoded to genes, is not practical because it is computationally very expensive, and as such, lots of indirect encoding methods have been proposed. Hypercube-based Neuroevolution of Augmenting Topologies (HyperNEAT) [70] is one of them. Infact Gauci [19] wrote, *"Although HyperNEAT generates ANNs with millions of connections, such ANNs can run in real time on most modern hardware. Using a 2.0GHz processor, an eight million connection networks takes on average 3.2 minutes to create, but only 0.09 seconds to process a single trial."* Clune [11] applied it to design a neural network that controls a quadruple leg robot.

Although once Frederic Jelinek, a pioneer in speech recognition, put it in the debates with the linguists, *"airplanes don't flap their wings to fly like birds"*[9], the most likely candidate of artificial intelligence might employ real biologically

[8] This is from the article in New York Times by Steve Lohr entitled "Creating Artificial Intelligence Based on the Real Thing" on 6 December 2011.

[9] This is from the article "Computer scientist of Czech-Jewish origin Jelinek dies in USA," in The Prague Daily Monitor on 27 September 2010,

plausible artificial neurons to think like human brain. An example would be, evolving trillions of spiking neurons with a fitness of how intelligent, assuming we have a good measure of machine intelligence mentioned in the previous section. Let us quote Sandberg and Bostrom's paper "Whole Brain Emulation: A Roadmap" [60]. *"The so far (2006) largest simulation of a full Hodgkin Huxley neuron network was performed on the IBM Watson Research Blue Gene supercomputer using the simulator SPLIT. It was a model of cortical minicolumns, consisting of 22 million 6-compartment neurons with 11 billion synapses, with spatial delays corresponding to a 16 cm^2 cortex surface and a simulation length of one second real time. Most of the computational load was due to the synapses, each holding 3 state variables. The overall nominal computational capacity used was 11.5 TFLOPS, giving 0.5 MFLOPS per neuron or 1045 FLOPS per synapse. Simulating one second of neural activity took 5,942 sec. The simulation showed linear scaling in performance with the number of processors up to 4,096 but began to show some (23%) overhead for 8,192 processors."* See also Cattell and Parkers paper [9] on this topic.

8.2 Toward Real AI by Parallelism

From computational point of view, when we compare the processing power of human brain with the power of machines the main differences relate to:

- power – computers are more powerful in specific tasks, allowing them to perform faster calculation or analysis of structured data, while the power of total human brain is still exceeding its machine counterpart (see [46]) and cause that we are able to see, hear or speak (not to mention about thinking).
- parallelism – in this case human brain is parallel biological computer, while machines are much more sequential.

Therefore, it is expected that increased parallelism will be a significant factor influencing further development of AI. As we can notice, the most of currently introduced or investigated AI algorithms is based on multiple instances of simple mechanisms (including neural systems or swarm intelligence) comparing to sophisticated algorithms typical for the traditional approach to AI. As we – researchers – haven't succeeded in reimplementing the nature using machine–based tools (algorithms and programming languages) we should aim at creation of self–adaptive resemblance of nature (brain) in large scale and expect that process of evolution will also work in this case.

However, considering further increase of computational power of machine–based intelligence, we think that the next obstacle we should overcome is the processing limits related to all computer system. As it was stated by Pietruszkiewicz (see [53] - for other factors important in AI applications see [54]) they relate to: algorithms, software, hardware and even human operators. These limits can be eased by many means, especially by increased parallelism, available in different forms including:

- Multi-core processors – is possible to implement fully parallel data processing on a single–chip machines and deploy task parallelism in systems. Furthermore, the current versions of processors available on market compete with number of cores, as increase of power of a single-cores is limited by quantum effects[10];
- GPU-enabled processing – allows us to deploy cost and energy effective GPU (Graphical Processing Unit) cards to problems where parallel data processing significantly reduced time of processing – described as GPGPUs applications (General Purpose Graphical Processing Unit). The power of even mid–range GPU cards, being multi–core RISC processors, is at a few rank over power of CPUs. The processing based on GPU fits very well to algorithms of AI, where tasks could be divided into interdependent parts, e.g. neural networks, evolutionary algorithms or swarm intelligence. The success of these systems could not have been achieved without supporting software technologies, like CUDA or OpenCL (see [61] or [63]), allowing one to easily build and deploy GPGPU applications;
- Distributed processing – transforming computers in network into metacomputers, where the clusters of distributed or co–located machines could be used in various tasks offering their resources. This solution also could not succeed without appropriate software technologies allowing developers to build distributed systems over software layers responsible for management of distributed systems (e.g. controlling them and performing tasks management). One of the most popular distributed data processing technologies is Apache Hadoop (see [37]) and its application to intelligent problems led to the development of Apache Mahout, build over Hadoop to perform data mining tasks (for more information about Mahout see [51]).

Therefore, as we can see all people involved in IT industry – researchers, developers, IT companies – are oriented onto increased availability of parallelism in computer systems at level of processor, machine or networks. Due to this observation and preliminary research done for AI using these technologies, we claim that this approach has a great potential to bring us closer to . To conclude – an urban expression *dumb as a bag of hammers* has a special meaning for AI. Human brain being also *“a bag of neurons”*, where a single neuron is not as bright as we could expect, is still the greatest intelligent system we could observe. Maybe a large number of parallel neurons will bring the man–made machines to this biological excellence. Additionally, this approach suits very well to the idea of bio–inspired AI, including evolutionary intelligence.

9 Conclusions

In this chapter we have analysed the most popular or influencing definitions of intelligence for natural and man–made systems. We have investigated two different

[10] The quantum computers for many years are considered to have a great potential and are expected to cause a technological revolution. However, they are still in a research stage, far from maturity and being market–ready.

approaches to artificial intelligence, the traditional and evolutionary one. Both of these approaches have many theories, methods and implementations and they were introduced and discussed herein.

To examine behaviour of evolutionary intelligence and compare it with natural intelligence we have performed an evaluating experiment. The introduced benchmark, being an AI–based solution to one of the most popular financial problems – resulted in evolutionary intelligence outperformed results of human experts. Additionally, it revealed the difference between the popular theory (which should be taken into account by investors) and business practice.

The last part of this chapter contained an analysis of technological changes that could support further development of intelligent systems. In our opinion one of significant technological changes taking place, with a great potential for AI, is a move towards parallelism in both – hardware and software.

In our final words, we would like to point that we are aware, that in the near future researchers community will still be discussing the definition of AI and what should be considered as a man–made fully intelligent system. However, we shouldn't forget that people behaviour involving knowledge and intelligence is not always as bright and clever as we expect. In situation where the genetic algorithms performed better than financial experts, which group should be consider to be intelligent? Or maybe we should start to think about AI in the same way we think about some people with great minds, allowing them to deal with complex tasks much better than with daily routines. Who will perform better at the Turing test – *the Rain Man* or a well designed chat–bot?

References

1. Adami, C., Brown, C.T.: Evolutionary learning in the 2d artificial life systems avida. In: Proceedings of Artificial Life, vol. IV, pp. 377–381. MIT Press (1994)
2. Adami, C., Ofria, C.A., Collier, T.C.: Evolution of biological complexity. Proceedings of the National Academy of Science 97, 4463–4468 (2000)
3. Albus, J.S.: Outline for a theory of intelligence. IEEE Transactions Systems, Man and Cybernetics 21(3), 473–509 (1991)
4. Anagnostopoulos, K.P., Mamanis, G.: The mean variance cardinality constrained portfolio optimization problem: An experimental evaluation of five multiobjective evolutionary algorithms. Expert Systems with Applications 38(11), 14208–14217 (2011), `http://www.sciencedirect.com/science/article/pii/S0957417411007603`, doi:10.1016/j.eswa.2011.04.233
5. Back, T., Fogel, D., Michalewicz, Z. (eds.): Evolutionary Computation 1: Basic Algorithms and Operators. Taylor & Francis (2000)
6. Back, T., Fogel, D., Michalewicz, Z. (eds.): Evolutionary Computation 2: Advanced Algorithms and Operations. Taylor & Francis (2000)
7. Bermúdez, J., Segura, J., Vercher, E.: A multi-objective genetic algorithm for cardinality constrained fuzzy portfolio selection. Fuzzy Sets and Systems 188(1), 16–26 (2012), `http://www.sciencedirect.com/science/article/pii/S0165011411002387`, doi:10.1016/j.fss.2011.05.013
8. Byrne, R.W., Whiten, A.: Machiavellian intelligence: Social expertise and the evolution of intellect in monkeys, apes and humans. Clarendon Press, Oxford (1988)

9. Cattell, R., Parker, A.: Challenges for brain emulation: Why is building a brain so difficult? (2011), http://synapticlink.org/Brain%20Emulation%20Challenges.pdf
10. Chaitin, G.J.: Godel's theorem and information. Theoretical Physics 21(12), 941–954 (1982)
11. Clune, J., Beckmann, B.E., Ofria, C., Pennock, R.T.: Evolving coordinated quadruped gaits with the hyperneat generative encoding. In: Proceedings of Congress on Evolutionary Computation, pp. 2764–2771 (2009)
12. Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. Springer (2007)
13. Fabozzi, F.J., Peterson, P.P.: Financial Management and Analysis. John Wiley & Sons (2003)
14. Fetzer, J.H.: Minds and machines: Behaviorism, dualism and beyond. Stanford Electronic Humanities Review 4(2) (1995)
15. Floreano, D., Mattiussi, C.: Bio–Inspired Artificial Intelligence: Theories, Methods, and Technologies. MIT Press (2008)
16. French, R.M.: Subcognition and the limits of the turing test. Mind 99, 53–65 (1990)
17. French, R.M.: The turing test: The first fifty years. Trends in Cognitive Sciences 4(3), 115–121 (2000)
18. Frosini, P.: Does intelligence imply contradiction? Cognitive Systems Research 10(4), 297–315 (2009)
19. Gauch, J., Stanley, K.: Generating large-scale neural networks through discovering geometric regularities. In: Proceedings of the Conference on Genetic and Evolutionary Computation, pp. 997–1004 (2007)
20. Gibson, D.G.: His 23 colleagues: Creation of a bacterial cell controlled by a chemically synthesized. Science 329(5987), 52–54 (2010)
21. Gorgulho, A., Neves, R., Horta, N.: Applying a GA kernel on optimizing technical analysis rules for stock picking and portfolio composition. Expert Systems with Applications 38(11), 14,072–14,085 (2011), http://www.sciencedirect.com/science/article/pii/S0957417411007433, doi:10.1016/j.eswa.2011.04.216
22. Grabowski, L.M., Bryson, D.M., Dyer, F.C., Ofria, C., Pennock, R.T.: Early evolution of memory usage in digital organisms. In: Proceedings of the International Conference on Artificial Life, pp. 224–231 (2010)
23. Grabowski, L.M., Bryson, D.M., Dyer, F.C., Pennock, R.T., Ofria, C.: Clever creatures: Case studies of evolved digital organisms. In: Proceedings of the European Conference on Artificial Life, pp. 276–283 (2011)
24. Grabowski, L.M., Elsberry, W.R., Ofriam, C., Pennock, R.T.: On the evolution of motility and intelligent tactic response. In: Proceedings of Genetic and Evolutionary Computation Conference, pp. 209–216 (2008)
25. Gudwin, R.R.: Evaluating intelligence: A computational semiotics perspective. In: IEEE International Conference on Systems, Man and Cybernetics, pp. 2080–2085 (2000)
26. Gunderson, K.: The imitation game. Mind 73, 234–245 (1964)
27. Harnad, S.: The symbol grounding problem. Physica D 42, 335–346 (1990)
28. Harnad, S.: Grounding symbols in the analog world with neural nets. Think (Special Issue on Connectionism versus Symbolism) 2, 68–73 (1993)
29. Harnad, S.: Turing indistinguishability and the blind watchmaker. Evolving Consciousness Amsterdam, pp. 3–18 (2002)
30. Hayes, P., Ford, K.: Turing test considered harmful. In: Proceedings of the International Joint Conference on Artificial Intelligence, vol. 1, pp. 972–977 (1995)

31. Hernandez-Orallo, J., Dowe, D.L.: Measuring universal intelligence: Towards an anytime intelligence test. Artificial Intelligence 178(18), 1508–1539 (2010)
32. Hibbard, B.: Bias and no free lunch in formal measures of intelligence. In: Artificial General Intelligence, vol. 1, pp. 54–61 (2009)
33. Hibbard, B.: Measuring Agent Intelligence via Hierarchies of Environments. In: Schmidhuber, J., Thórisson, K.R., Looks, M. (eds.) AGI 2011. LNCS, vol. 6830, pp. 303–308. Springer, Heidelberg (2011)
34. Hingston, P.: A turing test for computer game bots. IEEE Transactions on Computational Intelligence and AI in Games 1(3), 169–186 (2009)
35. Krol, M.: Have we witnessed a real-life turing test? Computer 32(3), 27–30 (1999)
36. LaCurts, K.: Criticisms of the turing test and why you should ignore (most of) them. Official Blog of MIT's Course: Philosophy and Theoretical Computer Science (2011)
37. Lam, C.: Hadoop in Action. Manning Publications (2010)
38. Legg, S., Hutter, M.: A collection of definitions of intelligence. Advances in artificial general intelligence: Concepts, architectures and algorithms. Frontiers in Artificial Intelligence and Applications 157, 17–24 (2007)
39. Lenski, R.E., Ofria, C., Pennock, R.T., Adami, C.: The evolutionary origin of complex features. Nature 423, 139–144 (2003)
40. Lipinski, P.: Evolutionary Decision Support System for Stock Market Trading. In: Dochev, D., Pistore, M., Traverso, P. (eds.) AIMSA 2008. LNCS (LNAI), vol. 5253, pp. 405–409. Springer, Heidelberg (2008)
41. Malkiel, B.G.: A Random Walk Down Wall Street: The Time-Tested Strategy for Successful Investing. W. W. Norton & Company, New York (2007)
42. Markowitz, H.M.: Portfolio selection. The Journal of Finance 7(1), 77–91 (1952)
43. Melanie, M.: An Introduction to Genetic Algorithms. MIT Press (1999)
44. Michie, D.: Turing's test and conscious thought. Artificial Intelligence 60, 1–22 (1993)
45. Miller, G.F.: Protean primates: The evolution of adaptive unpredictability in competition and courtship. In: Machiavellian Intelligence II: Extensions and Evaluations, pp. 312–340 (1997)
46. Moravec, H.: When will computer hardware match the human brain? Journal of Evolution and Technology 1 (1998)
47. Neisser, U., Boodoo, G., Bouchard, T.J., Boykin, A.W., Brody, N., Ceci, S.J., Halpern, D.F., Loehlin, J.C., Perloff, R., Sternberg, R.J., Urbina, S.: Intelligence: Knowns and unknowns. American Psychologist 51(2), 77–101 (1996)
48. Ofria, C., Adami, C., Collier, T.C.: Design of evolvable computer languages. IEEE Transactions in Evolutionary Computation 17, 528–532 (2002)
49. Ofria, C., Bryson, D.M., Wilke, C.D.: Avida: A Software Platform for Research in Computational Evolutionary Biology. In: Artificial Life Models in Software. Advances in Artificial Life, 2nd edn., pp. 3–36 (2009)
50. Oh, K.J., Kim, T.Y., Min, S.: Using genetic algorithm to support portfolio optimization for index fund management. Expert Systems with Applications 28(2), 371–379 (2005), `http://www.sciencedirect.com/science/article/pii/S0957417404001356`, doi:10.1016/j.eswa.2004.10.014
51. Owen, S., Anil, R., Dunning, T., Friedman, E.: Mahout in Action. Manning Publications (2011)
52. Pennock, R.T.: Models, simulations, instantiations, and evidence: the case of digital evolution. Experimental and Theoretical Artificial Intelligence 19(1), 29–42 (2007)
53. Pietruszkiewicz, W.: Practical Evaluation, Issues and Enhancements of Applied Data Mining. In: Abd Manaf, A., Zeki, A., Zamani, M., Chuprat, S., El-Qawasmeh, E. (eds.) ICIEIS 2011. CCIS, vol. 252, pp. 717–731. Springer, Heidelberg (2011), `http://www.springerlink.com/content/h077486038250q07/`

54. Pietruszkiewicz, W., Dzega, D.: The Large Scale Artificial Intelligence Applications – An Analysis of AI-Supported Estimation of OS Software Projects. In: Konstantopoulos, S., Perantonis, S., Karkaletsis, V., Spyropoulos, C.D., Vouros, G. (eds.) SETN 2010. LNCS(LNAI), vol. 6040, pp. 223–232. Springer, Heidelberg (2010)
55. Pietruszkiewicz, W., Twardochleb, M., Roszkowski, M.: Hybrid approach to supporting decision making processes in companies. Control and Cybernetics 40(1), 125–143 (2011)
56. Poole, D., Mackworth, A., Goebel, R.: Computational intelligence: A logical approach. Oxford University Press (1998)
57. Purtill, R.L.: Beating the imitation game. Mind 80, 290–294 (1971)
58. Ray, T.S.: An approach to the synthesis of life. Santa Fe Institute Studies in the Science of Complexity X, 371–408 (1992)
59. Reilly, F.K., Brown, K.C.: Investment Analysis and Portfolio Management. South-Western College Pub. (2008)
60. Sandberg, A., Bostrom, N.: Whole brain emulation: A roadmap. Tech. rep., Future of Humanity Institute, Oxford University (2008)
61. Sanders, J., Kandrot, E.: CUDA by Example: An Introduction to General-Purpose GPU Programming. Pearson (2010)
62. Saygini, A.P., Cicekliz, I., Aknan, V.: Turing test: 50 years later. Minds and Machines 10, 463–518 (2000)
63. Scarpino, M.: OpenCL in Action: How to Accelerate Graphics and Computations. Manning Publications (2011)
64. Schweizer, P.: The truly total turing test. Minds & Machines 8, 263–272 (1998)
65. Searle, J.R.: Minds, brains and programs. Behavioral and Brain Sciences 3, 417–424 (1980)
66. Smith, W.D.: Mathematical definition of intelligence (and consequences) (2006), http://math.temple.edu/wds/homepage/works.html
67. Soleimani, H., Golmakani, H.R., Salimi, M.H.: Markowitz–based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm. Expert Systems with Applications 36(3, Part 1), 5058–5063 (2009), http://www.sciencedirect.com/science/article/pii/S095741740800328X, doi:10.1016/j.eswa.2008.06.007
68. Spearman, C.E.: The abilities of man, their nature and measurement. Macmillan (1927)
69. Standage, T.: The Turk: The life and Times of the famous eighteenth-century chess-playing machine. Walker & Company (2002)
70. Stanley, K.O., Ambrosio, D.B., Gauci, J.: A hypercube-based indirect encoding for evolving large-scale neural networks. Artificial Life, 185–212 (2009)
71. Turing, A.M.: Intelligent machinery (1948)
72. Turing, A.M.: Computing machinery and intelligence. Mind 59(236), 433–460 (1950)
73. Watt, S.: Naive psychology and the inverted turing test (1996)
74. Weizenbaum, J.: Eliza: A computer program for the study of natural language communication between men and machines. Communications of the ACM 9, 36–45 (1966)