Witold Marciszewski

UNDECIDABILITY AND INTRACTABILITY IN SOCIAL SCIENCES

Motto 1: *There are actually lots of threads that led to computer technology, which come from mathematical logic and from philosophical questions about the limits and the power of mathematics.*
Greg Chaitin[1]

Motto 2: *Computer simulations are extremely useful in the social sciences. It provides a laboratory in which qualitative ideas about social and economic interactions can be tested. This brings a new dimension to the social sciences where 'explanations' abound, but are rarely subject to much experimental testing.*
Richard J. Gaylor (University of Illinois), Louis J. D'Andria (Wolfram Research, Inc.)[2]

Introduction

In scientific practice a computer has become so indispensable a tool as a sheet of paper, pen or laboratory instruments used to be before it was discovered.[3] Hence a question regarding a range of its possibilities occupying the minds of the philosophers of science and methodologists. In the first place, one has to consider astonishing results of the logical and mathematical research – like the fact that not every problem of arithmetic can be solved with the help of a computer algorithm. There is no algorithm to decide the following: is there a solving algorithm of any mathematical problem? And even if such algorithms do exist, their use often requires so huge time or space (memory) resources that practically one can hardly expect them to solve a problem. This is an example of algorithm intractability.

Is this limitation only an internal business of pure mathematics? Or perhaps it also refers to empirical sciences (social sciences being among them) which mathematics provides with algorithms to model the reality computationally?

A positive answer to the last of the above-asked questions is not given a priori. Here perhaps some amount of philosophical faith in good demon would not be absurd – the demon would have created the empirical world in such a way that there would be only relations

[1] "A Century of Controversy over the Foundations of Mathematics" in: C. Claude and G. Paun, *Finite versus Infinite,* Springer-Verlag London 2000, pp. 75-100. Due to his original and influential ideas and results on range and limits of algorithmical methods, Chaitin, the IBM mathematician, has become a classic of both computer science and philosophy of science. In this respect, he is an eminent continuator of Gödel and Turing's thought. He is much acknowledged for the discovery of the incalculable Omega number defined as a probability that a computer program will end after a random binary sequence has been entered. For a comprehensible study of the definition see Paul Davies, *The Mind of God,* Chapter 5, Part "Unknowable", pp. 128-134 (Simon and Schuster, New York, 1992).

[2] From the introduction to the book entitled *Simulating Society: A Mathematical Toolkit for Modeling Socioeconomic Behavior*, Springer Verlag, 1998. See www.telospub.com/catalog/FINANCEEON/SimSoc.html

[3] This paper develops ideas underlying the research project "Nierozstrzygalność i algorytmiczna niedostępność w naukach społecznych" [Undecidability and Intractability in Social Sciences], no. 2 H01A 030 25, supported by the Polish State Committee for Research and Development, in years 2003-2006. It is a revised version of the Polish paper entitled "Nierozstrzygalność i algorytmiczna niedostępność w naukach społecznych" (likewise the title of the project in question) which appeared in the Polish quarterly "Filozofia Nauki" [Philosophy of Science], 2004, vol. 12, no. 3-4, pp. 5-31.

represented by computable functions. What is more, such functions would always allow to grasp problems algorithmically.

What we already know about natural and social problems does not confirm that faith. However, an awareness of this fact does not easily reach researchers, especially those dealing with social sciences. Some of them seem to advocate a view that issues of undecidability remain in an exotic sanctuary of pure mathematics, being away from empirical facts. Computer and logical researches putting that view into question are relatively fresh. Hence a need to provide necessary information along with some reflection what is to be done in such a situation.

However, this is not the end of astonishing messages. Physicists claim that certain problems solved with the help of algorithms or software (on condition that a parallel development of hardware is equal) may remain unsolved due to the developmental limits on the part of physics. Miniaturization, constantly enlarging its computational power, is in the long run limited by material granularity, whereas the development of computers towards a bigger capacity will finally result into a slowdown in signal transmission, obviously unable to exceed the velocity of light.

On the other hand, physics along with some philosophical reflection bring good news, too. In Nature, especially in the brain, there are some computing processes that have not been proved to undergo the restrictions that the universal Turing machine, or in other words, a digital computer, is subjected to. There is a significant group of physicists who see the source of superiority of the human brain over algorithm in the fact that the brain belongs to the sphere governed by quantum mechanics. To explain its superiority, they highlight the ability of the brain (not to say "of the mind") to recognize the correctness of Gödel's statement, discover axioms and invent algorithms. Since none of these things can be gained from a mechanical procedure, their source must be hidden somewhere in organic nature (or in its surroundings penetrated by philosophy).

1. THE STATE OF THE PROBLEM AND THE WAYS HOW TO SOLVE IT

1.1. By the end of the 20th century **the theory of computational complexity** had appeared in computer studies. It continued the main thread of logic starting from the point marked by Gödel, Turing, Church, Post and Tarski (just to mention the main leading figures), who followed Hilbert's thought. At that point logic surprised the world of science by the discovery of the issue of undecidability both in mathematics and in logic itself. After a short time there appeared a recognition of the fact that in the sphere of decidable problems there are some problems whose solutions can never be found even if we were to employ as many supercomputers as a number of electrons in the space, giving them as many years to count as the age of the world's history. This practical insolvability of the problems that are "solvable" themselves has been called *intractability,* sometimes preceded by the adjective *computational* or *algorithmic*.

> This term should not be understood as a lack of the proper algorithm to solve a problem but rather as an implication of the fact that a solution expected from such an algorithm is inaccessible (because of the reasons which will be discussed later – like a lack of time or memory resources). Analogically, when it is said that a problem is computationally intractable, it does not imply the impossibility of the computational process that would lead towards a solution (it is possible when a solution is a computable number). Rather, it implies the fact that apart from making calculation, the solution will not be reached. In this expression the adjective "algorithmic" has been used because the term "algorithm" is better known than the term "computing" in the technical sense of Turing [1936], which is different from colloquial.

> Colloquially, one can say, “a jumper has calculated the distance well” even though this process does not require any number operations (as it is in case of Turing).

The sources of algorithmic intractability, tackled by the theory of computational complexity, will be discussed later. At this stage, all we need is to take note of it just to recognize the following methodological issues relating to social sciences.

– [1] Can algorithms, used to model and simulate social phenomena, have the complexity that would lead towards indecidability or intractability of the problem?
– [2] If so, are there any methods to transform the problem so that it is no longer intractable and at the same time its solution would be close enough to the solution of the original problem?
– [3] If so, what are these methods?
– [4] Are there statements or other elements of the theory whose acceptance is not justified by any of the algorithms applied in the given theory?

The answer to the question [4] comes immediately both for deductive sciences (logic and mathematics) and for empirical ones, social sciences being among them. Certainly, such statements exist. In the deductive theory these are axioms and rules whereas in the empirical theory they are observational statements as well as meaning postulates in the Carnapian sense [1956], also called language postulates by Ajdukiewicz [1965]. Even though the answer to question [4] is obvious, the question needs to be asked for it reveals the following:

– [5] On which basis are statements of a particular theory accepted when none of the algorithms justifies it?

If we were to address questions [1], [2] and [3] to physicists, we would receive answers illustrated by an example list of the problems of physics – undecidable, intractable or having only approximate solutions. Physics has at its disposal a significant collection of such limitation results, dealing with the limitations of solvability.

Stephen Wolfram’s text entitled *Undecidability and Intractability in Theoretical Physics* [1985][4] makes a good starting point to consider the problems in view. The author gives examples of undecidable and computationally inaccessible problems in physics employing the notion of *reducibility,* which also reveals the term *compressibility* in the Kolmogorov-Chaitain algorithmic information theory. A lack of this feature consists in the fact that an algorithm, simulating the process under examination, has to reproduce it step by step (explicit simulation), having no possibility to reduce it. Because of its length, unreducible computation is subjected to face a lack of time or memory resources or, in other words, it will prove to be intractable. Wolfram finds examples of unreducability in certain processes taking place in cellular automata, electric circuits, nets of chemical reactions, etc. In this collection some problems are undecidable and other are intractable. In the end the author concludes that such situations are not exceptional. Instead, they are common.[5] The title of Wolfram’s article has served as a model for formulating the topic of this work. Since he has been largely quoted, Wolfram can make a widely accepted model of problems of certain type. The question, which

[4]Wolfram is widely known due to his works on cellular automata collected in the book [1994]. He is also known as the author of the software “Mathematica” meant for computing and programming in scientific research. His monumental book [2002], which states that cellular automata of a certain type constitute an adequate model of the physical world, was the scientific bestseller of the year.

[5] Wolfram mentions numerous examples of undecidability and intractability in physics: “Quantum and statistical mechanics involve sums of over possibly infinite sets of configurations in systems. To derive finite formulas one must use finite specifications for these sets. But it may be undecidable whether two finite specifications yield equivalent configurations. So, for example, it is undecidable whether two finitely specified four-manifolds or solutions to the Einstein equations are equivalent (under coordinate reparametrization).”

in Wolfram's article refers to physics, can as well be asked in relation to any science or a group of sciences. Therefore, let us ask this question in relation to social sciences.

1.2. To state how researchers into social phenomena and theorists of the issue of computational complexity relate to questions [1]-[5], let us consider them once again.

An awareness of the types of the employed algorithms' complexity would be a rare thing to meet while searching for the answer to question [1] in sociological and economic literature, giving an account of the use of mathematical models. However, if computational complexity was to be found in the works of theorists, there would easily appear a pile of limitation results similar to the ones present in Wolfram's work, on which the present paper has been based. In the light of these results the answer to [1] is positive.

Therefore, there appears a need to confront both trends: the one using algorithms of the empirical studies with the logical studies of these algorithms. Bearing in mind the fact that the problem of possible limitations of the computational power of algorithms is often ignored in some empirical studies, this need becomes even more important. In this case, the methodologically important questions [2] and [3] have no chance to appear.

A procedure that does not take any account of algorithm limitations is only justified when it is known that a problem under examination is simple enough to be solved by the algorithm employed. However, something different is being observed. Ignorance of the questions regarding complexity of the problem takes place while considering highly difficult problems, with the highest level of complexity. Three examples of this type are given below (Ex 1 - Ex 3).

> Ex 1 – Strong artificial intelligence (AI). It is a highly ambitious project for it aims at the entire (having no difference from the original) simulation of the most complicated product of Nature, that is, the human brain. The realization of this project would have a significant influence on social sciences, enabling them to gain knowledge how to create optimum social configurations with the help of simulation through artificial societies. Artificial societies (AS) are the societies where every member is represented by a subprogram defining its (artificial) mind as well as its interactions with the surroundings (this branch of computer science falls under the definition of "multi-agent simulation"). Not only is AS conditioned by the prior creation of the AI, but it is also a necessary condition of the advanced AI since the development of intelligence requires adequate social surroundings. Apart from this complexity, intensified by the feedback AI-AS, the research into AI does not offer any reports on limitation results. However, there are promises of an upcoming success.

> Ex 2 – Central socialist planning supported by computers is a concept propagated by Oscar Lange in the polemics with the Austrian Economic School (von Mises, Hayek). According to that School, socialist planners were doomed to face a failure due to their disability to show the complexity of economic facts. At the beginning of the 60s Lange fought that blame back arguing that the creation of the computer made it possible what had seemed impossible before. Among a group of leftist economists the concept remains fertile till today. Due to the fact that economic phenomena are relatively easy to measure, it must be possible to state an order of magnitude when it comes to the size of input data. Also, there are proposals of some mathematical models of (for example) the market equilibrium (Pareto, Lange, etc). As a result, there are good reasons for estimating the complexity of the algorithms necessary for an effective socialist calculation. A search query must be held to find calculations of this type in the already existing literature. In case of its failure, one has to estimate the complexity of the problem on his own.

> Ex 3 – Referring to the professional computer simulation (conducted in MIT), by the end of the 20th century a report of the Club of Rome foresaw a total worldwide economic and ecological failure. It is obvious that such a complicated enterprise cannot be achieved unless

the model is simplified. There are simplifications that do not deform the reality making the study easier and quicker, instead. On the other hand, there are simplifications that lead towards a picture that is completely different from the real world. A similar simplification of the report of the Club of Rome was the omission of the human creativeness factors as well as people's ability to exert creativeness when being under threat (similarly, this factor must be omitted in the concept of central economic planning for it is impossible to plan discoveries). As a result, the report produced fictitious prognosis. A lack of knowledge anticipating what may happen in the mind of scientists, discoverers, reformers, etc. makes a considerable difficulty here. But even if such knowledge were to be revealed by Laplace's demon, in the face of the infinite complexity of creative thinking one can hardly expect this process to be simulated by the algorithms within a digital computer's reach.

What may the best reaction towards ignoring the real complexity in social studies be? A fatalistic reaction would depend on the assumption that the evaluation of the model complexity is impossible whereas the gap in the knowledge about the model would be recompensed by the faith that it could match the reality. In other words, it would be able to explain and foresee real processes. Perhaps it would be acceptable to agree with such a reaction if the evaluations of the model complexity in social sciences were beyond researchers' possibilities. Fortunately, the truth is different. To prove it, it is enough to consider any widely used model of social phenomena and mention the studies of this model conducted within the theory of computational complexity.

1.3. Prisoner's Dilemma, which is a standard model for a wide class of social interactions, makes a perfect case for this aim. The popularity of this model is confirmed by the fact that at the beginning of 2003 Google showed over 800 Internet sites offering links related to the topic. This example enables one to observe how an increase of the number of the output data (number of social actors or players, number of strategies, number of rounds to play) in some cases leads to undecidability or intractability.

The name "Prisoner's Dilemma" is related to the story illustrating a problem of two prisoners. Being suspected of robbery, they face a dilemma (this is the plot of the dilemma stated in 1940 for the first time). To get evidence, an investigating magistrate offers each prisoner a deal so that they think they have been treated individually. The conditions of the deal are as follows. If two prisoners acknowledge their guilt, they will get a fifteen-year prison sentence. If they do not admit their fault so that it is impossible to prove the robbery, they will be sentenced only for three years (for a different crime). If one of the prisoners acknowledges his fault, he will be released (as the one who has contributed to the investigation a lot) whereas the other will get a twenty-year prison sentence.

The question what pays off – to cooperate together loyally or to pursue one's selfish interest even at the partner's expense when it is not known what he is going to do makes the root of the dilemma. In other words, the unawareness of the partner's intentions is the heart of the problem. Different ways of behavior to choose from are called strategies of the game. They are defined respectively as *cooperative strategy* (in the example of the prisoners this is abstaining from burdening themselves with evidence) and *competitive strategy* (aiming at one's profit without taking into consideration a loss of the partner).

Undoubtedly, the prisoners' example perfectly highlights the drama of the dilemma. On the other hand, a situation when the members of the game can play it repeatedly is not illustrated well enough. This situation is important for the explanation of the evolutionary processes. Consisting of a few component games called rounds, such *iterated* games give its players a chance to learn by getting to know a partner better and possibly harmonizing their actions. Let us illustrate this general situation by a game with the following rules.

If A and B are willing to cooperate, each will get 3 gold coins;
If A chooses cooperation with B while B chooses competitiveness – B gets 5 gold coins and A gets nothing;
When both choose competitiveness, each gets one gold coin.[6]

In the one-round game, when uncertainty as to the other member' behavior is marked as 0,5 probability for each option, a competitive strategy would be more beneficial. If a game consists of many rounds, the case gets more complicated when, for example, each player may anticipate the other player's strategy. In the situation when each side gets the certainty about the partner's willingness to cooperate, each partner stands a chance of winning 3 gold coins whereas their choice to compete would result in only one-coin profit for each. Here we have a certain model of social evolution directed at the increase of cooperation.

A natural computational model in the research on decidability of the prisoner's problem is provided by cellular automata.[7]

Among others, there has been considered an arrangement where a cell has eight direct neighbors as its partners. The strategy, being the state of the cell, can be (a) purely competitive, (b) purely cooperative, (c) mixed – "tit for tat" – when a cooperative move is followed by cooperative and a competitive one by competitive, and (d) – the strategy in which a player observes the strategies of his/her neighbor and chooses the one which has proved to be more beneficial. In the course of the game certain strategies appear more often and become predominant, which can be observed as a change of configuration on a two-dimensional panel where the evolution of automata is being played.

To some extent it is possible to anticipate the direction of this evolution. Consequently, there appears a question: is there an algorithm able to decide for each case whether *a given configuration of the strategies will gain a permanent predominance over the rest*? Grimm [1997 as well as the already quoted WWW text] has given evidence that this problem is undecidable in the classical Gödelian sense, which means that such an algorithm does not exist.[8]

A research on the connections between computational complexity of a game and its players' rationality makes another important trend. The theory of complexity should help the theory of games realize an empirical factor that cooperative strategies can be beneficial for both sides. Paradoxically, the theory of games itself implies that when the players do not change a strategy (which is known as the state of equilibrium), it is beneficial for both sides to stick to the competitive strategy. The consequence, which takes place in case of giving the players unlimited means to solve problems (the so-called unlimited rationality), stops to be obligatory for example, in cases of limiting the storage capacity; memory can be measured by

---

[6] This version of the game can be practised at: serendip.brynmawr.edu/playground/pd.html.

[7] This is a summary of the material presented at:
www.sunysb.edu/philosophy/faculty/pgrim/SPATIALP.HTM. This is Patrick Grim's article entitled "Undecidability in the Spatialized Prisoner's Dilemma: Some Philosophical Implications". The author presents an interesting account of the problem emphasizing the significance of the Prisoner's Dilemma as a model of social processes: "This simple game-theoretic model seems to capture in miniature something of the tensions between individual acquisitiveness and the goals of collective cooperation. That is of course precisely why it has become a major focus of modeling within theoretical sociology, theoretical biology, and economics. […] It is no simplification to say that our strongest and simplest models of the evolution of biological and sociological cooperation – and in that respect our strongest and simplest models of important aspects of ourselves as biological and social organisms – are written in terms of the Iterated Prisoner's Dilemma.

[8] See Grim [WWW, op. cit]."There is no general algorithm […] which will in each case tell us whether or not a given configuration of Prisoner's Dilemma strategies embedded in a uniform background will result in progressive conquest. Despite the fact that it is one of the simplest models available for basic elements of biological and social interaction, the Spatialized Prisoner's Dilemma proves formally undecidable in the classical Gödelian sense."

a number of the states of finite automaton realizing a particular strategy in the game with *n* rounds. A. Neymann [1985], a pioneer of the concept of limited rationality, stated that when the limitation of memory of both players consists in its capacity at the interval $[n^{1/k}, n^k]$, where n is a number of rounds and $k > 1$, cooperation becomes the most beneficial strategy for both players. It has also been proved that cooperation dominates competitiveness in the games with an infinite number of rounds as well as in the games with a finite but unknown number of rounds (see Papadimitriou and Yannakakis) [1991]).

These and other numerous results dealing with a particularly useful model of interaction - the Prisoner's Dilemma - illustrate how the methodology of social sciences can benefit from the theory of computational complexity. However, there appears a problem similar to the one about its practical use in industry and economy. An initial discernment shows that research practice of social sciences has made little use of this theoretical background. This initial discernment should be put under a systematic verification and in case of its confirmation, a question regarding its reasons should be asked. Are these reasons objective, based on the fact that the results of the theory of complexity are too subtle or abstract for the real needs of the research practice? Or perhaps there are subjective reasons, resulting from the researchers' disability to follow the progress of the theory of complexity? Indeed, these questions are worth asking in the research conducted within the methodology of science and the study of the progress of human knowledge.

1.4. The above-mentioned Grim's research on decidability of one of the problems within Prisoner's Dilemma (compare footnote 8) appears to be of great use when it comes to the question which was formulated earlier (point 1.1, question [4]), namely, are there statements or other elements of the theory whose acceptance is not based on any of the algorithms applied in the given theory? Indeed, a positive answer is obvious since in the empirical theory neither observation statements nor meaning postulates give evidence as to their derivation from the algorithm. However, results similar to those of Grim indicate statements capable of being axioms (at the stage of axiomating a given theory)

> To specify this conclusion, let us refer to the clue that is hidden directly in Grim's research. The question that, according to this research, is undecidable can be answered with a hypothesis of an intuitive character drawn from certain social phenomena. For instance, the fact that civil wars (well-situated within the scheme of Prisoner's Dilemma) often reach a compromise after the two sides "have lost blood" (after many rounds of the iterated game). It means that a cooperative strategy wins for both sides, which results in a certain state of equilibrium. Its approach, as it is indicated by Grim's limitative result, cannot be concluded from the axioms of the theory of games supplying here the model required. Therefore, it is a clue that our hypothesis is likely to take a position in a group of axioms.

The question [4] produces another question, namely [5]: *on which basis are statements accepted in a specified theory when no algorithm supports it?*

The following circumstances may constitute such a basis: statements, accepted in a given theory without any proof, may have proof in another theory that is worth accepting (we deal with a certain algorithm in case when a proof has been formalized). This is an example of a current procedure in science. A supposition that some algorithms "exist" beyond theories, supplying certain theories with propositions worth accepting as theorems, is less obvious.

The admission of such a supposition (in this context) comes from the need to begin a dialogue with the theory called strong artificial intelligence. It helps to articulate the thought of the theory that the statements of the class represented by the Gödelian statement are also produced by some algorithm. Having formulated it in such a way, there appears a following question: to which theory does an algorithm belong if it does not belong (for example) to mathematics? This marks the next move in the discussion as, for example, a statement that the algorithm must be beyond any theory – let us say – functioning somewhere in Gödel's brain. It would mean another statement that this brain has computational power equal to the power of the Turing machine, which would be another step forward prejudging on whom onus probandi rests.

Remaining central in the discussions about artificial intelligence, this problem has been outlined to be looked at from a methodological point of view. From this point it is important that we deal with the statements which in a given theory neither derive from any other statements nor belong to its axioms. What is more, they are not observational statements, registering only sensory perception of something taking place here and now. **A priori** statements (or propositions) would be a suitable term for them.[9]

To justify this term, let us highlight the following. The Gödelian statement is a prototype of the statements deserving to be regarded as true although they are neither axioms of a given theory nor they are derived algorithmically from the axioms. The above-mentioned metatheoretical research shows that such statements also appear in those empirical theories (social theories being among them) that use mathematical models. This is a methodologically important category that has to be distinguished by a special term. Such statements deserve the name of potential axioms. However, apart from the aspect of their possible purpose, there is a need to hint at the aspect of their origin. The traditional expression "a priori", used in philosophy for years, reveals this genetic aspect.

A view on social theory as including a priori statements corresponds well with the view of one of the classics of methodology of social sciences – Ludwig von Mises. In the work entitled *Human Action* [1966] he characterized economy as a highly general science about human action that he called praxeology. From a methodological point of view it was described by him as follows:

> "Praxeology is a theoretical and systematic science. […] It aims at knowledge valid for all instances in which the conditions exactly correspond to those implied in its assumptions and inferences. […] Its statements are, like those of logic and mathematics, a priori." [p. 32]
> "The fact that man does not have the creative power to imagine categories [of thought, action, etc] at variance with the fundamental logical relations and with the principles of casualty and teleology enjoins upon us what may be called *methodological apriorism*." [p. 35, italics LvM.]

The table of content of the work *Human Action* illustrates which social regularities, according to von Mises, are experienced in such an a priori way. Typical topics of economy such as account, market, price, credit, work, the role of government, taxes, etc can be found there. Indeed, a view that economic rights of the above-mentioned things are as a priori as mathematical statements are sounds radical. However, it is possible to make it less radical by adopting the idea of W. V. Quine regarding the levels of apriority. Its highest level is to be revealed by logic and mathematics (hence a complementary element – empirical content – can only be found in them to a little extent). It is further divided in different rations. In this hierarchy economy could be placed relatively close to the top of apriority. Indeed, such principles as a principle of expected utility, rights of supply and demand, or an informative function of prices (F. Hayek's idea) are highly a priori.

[9] If this category seems incomprehensible or even mystical, supporters of the "scientific sobriety" may prefer the algorithms that remain beyond any theory. They can be hardly rejected by those who represent the "only scientific" view that the brain is the Turing machine, producing both axioms and any other apriority according to the axioms unknown even by itself.

These considerations encourage us to adopt von Mises's concept of **methodological apriorism**, which reveals the fact that numerous laws of social sciences behave like mathematical or logical axioms. Therefore **meaning postulates** – the term introduced in 1.1. – would be suitable for them. It includes axioms but refers to the class that is more capacious, holding the statements that share only axiom apriorism without their role of being first premises of the system.

Meaning postulates are not a product of the algorithm acting within the theory. It may be believed that some algorithm is produced beyond the theory, functioning in our brains to produce a priori principles. They can also be produced as a result of some non-algorithmic process. In any case, they are unlikely to become a pain in the neck for the researchers because of their excessive algorithmic complexity. A problem of the algorithmic procedure in relation to them does not appear at all.

How should such a situation be approached in social sciences? What is a cognitive value of these a priori principles, having a kind of privilege, for they are subjected neither to empirical verification nor to control as to their consistency with other statements of the system? This is a significant research problem. On the one hand, the recognition of such an important role of the a priori element disturbs a widely accepted empirical paradigm. On the other hand, a practice of mathematical modeling (when mathematical equations become statements of an empirical theory) introduces a significant element of apriorism. The necessity of meaning postulates as constituting the language of the theory makes another element of this kind.

This problem can be successfully attacked with the help of the following strategy. In social sciences the performance of such fertile and widely accepted models as the theory of games or cellular automata should be put under a methodological analysis. In the course of its application, a mathematical model, as expressing certain assumptions of the rationality (intelligence) of the actions, is being confronted with the observations that do not always confirm such rationality (often they lead towards its negation). What rule of preference should a researcher choose in this situation? Should he change the a priori assumptions or should he leave them and interpret the observation data in their light? There is no ready answer. Each case should be analyzed separately and a final verdict is bound to appear in the developmental course of science when a solution will be accepted or rejected after some time, remaining a mere exhibit in the records of the abandoned theories.

1.5. This passage is a kind of an erudite annex. Offering neither problems nor hypotheses, it only comments upon a few terms which were mentioned above (the passage is dedicated to those who feel a need to have a closer look at them).

The term of decidability is derived from logic where it appeared explicitly in the context of Hilbert's Program with the conviction that every well-formulated mathematical problem is decidable (see Hilbert and Ackermann [1928]). The evidence of the hypothesis is defined as positive whereas its refutation as a negative solution of the problem of decidabiliy (*Entscheidungsproblem*). In the original, historically close formulation of Hilbert and Ackermann [1928, p. 73], it is put as follows (italics by H and I).

> *Das Entscheidungsproblem ist gelöst, wenn man ein Verfahren kennt, das bei einem vorgelegten logischen Ausdruck durch endlich viele operationen die Entscheidung über die Allgemeinheit bzw. Erfüllbarkeit erlaubt.*
> *Die Lösung des Entscheidungsproblem ist für die Theorie aller Gebiete, deren Säztze überhaupt einer logischen Entwickelbarkeit aus endlich vielen Axiomen fähig sind, von grundsätzlicher Wichtigkeit.*

Apart from relativizing to particular axiomatics and a specified group of formal rules of inference, the concepts of procedure appearing here (Verfahren) has a scale as wide as it is in case of the terms of algorithm or computer program. Being a computational operation, every step of the algorithm execution conducted by a computer is at the same time an inference from arithmetical axioms, done according to the rules of logic (the implication form of these rules allows for their use as the rules of inference).

A positive solution to the problem of decidability for logic would allow for such an algorithm when the correctness of every step would be indicated by referring to a proper logical formula whose validity (*Allgemeinheit*) could always be demonstrated due to decidability of logic. Such an algorithm would ensure decidability of every axiomatized theory, additionally formalised by the laws of logic.

When Turing [1936] and Church [1936] demonstrated undecidability of logic or, in other words, proved the negative solution to the problem of decidability, there appeared reasons for asking more questions concerning algorithms, which gave birth to a new theory. Called **computational complexity**, this theory is considered to be a section of computer studies. The history of the problem shows that it is an area bordering computer studies and logic, remaining a point of interest for both of them.

The following terminological nuances deserve a special attention. The term "algorithmic" is being used interchangeably with "computational" in the contexts of the expressions "algorithmic intractability" and "computational intractability". A similar interchangeability has not been adopted with "complexity". "Algorithmic complexity" has a meaning that is different from "computational complexity". The former relates to the measure of complexity (defined independently by Kołmogorow, Chaitin, and Solonoff) regarding a relation between the length of symbol sequence produced by the algorithm and the length of the algorithm in question (see Chaitain [2002]). The last relates to memory and time resources necessary to solve a given problem by the algorithm. The complexity is measured with the size of the resources required. (Hartmanis and Stearns [1965] were pioneers in this field).

Algorithmic tractability or intractability (as well as decidability or undecidability) is predicated of problems. The problem is **algorithmically tractable** when it is decidable, and it is not too computationally complex for a solving algorithm (program) to be conducted with space (memory) and time resources within the computer users' reach (definitely, this conception of reach is not precise but this imprecision is not really harmful). Chapter 3 will discuss how the complexity of the problem is connected with the limits of both space (memory) and time of the algorithm execution.

2. THE IDEA OF RATIONALITY AND THE NOTION OF INTELLIGENCE IN SOCIAL SCIENCES

2.1. It was on purpose that in the title the word rationality was used in the context of the word "idea" while the word "intelligence" was linked with "notion". Out of these two words it is the word "idea" which reveals a bigger amount of normative or axiological charge (surely because of its nearness to the meaning of the word "ideal"), which makes the difference between these close meanings.

A person who is capable of solving his/her problems successfully and possibly with a little expenditure, being able to differentiate between the levels of problems' importance, is called intelligent. More or less the same applies to the definition of rationality. Therefore, the difference lies not in any radically different substance but rather in association, stress and context. This justifies a connection of the two topics, which should lead towards a mutual complement of the two spheres of our consideration. An example of the social problem, which is going to be presented in the second part of this chapter, can be discussed in the category of rationality as well as in the category of intelligence.

The notion of rationality is inseparable from the standard one in social sciences of the game models. The aim of the game is to win. Therefore, it is only natural to define the behavior that results in profit as being rational whereas the behavior that brings losses is defined as irrational. In this context a similar thought will be expressed when the word “rational” will be replaced by the word “intelligent”. However, apart from the interchangeability, this theory reveals its ability to enrich one content with another. The problems of the issues of artificial intelligence connect intelligence with computational power – one of the main topics of the theory of computational complexity. Hence, as it is seen from the texts mentioned in 1.3., the authors often refer to a limitation of computational power and, consequently, intelligence as to bounded rationality.

In this manner the two notions start converging into one notion, which also leads towards a tie-up between social sciences and the theory of intelligence. A few directions of this tie-up are worth mentioning. Among others, the following facts are concerned:

It was in the early 90s that an intensive process of the tie-up between social sciences and Artificial Intelligence (AI) started. The progress of artificial intelligence led towards the programs that enabled an interaction between artificial brains represented by appropriate programs. It was called **distributed AI**. The appearance of the net interactions (Internet, etc) made another step possible – at that stage an interaction between the programs functioning in different computers was within reach. The sides of such interactions were called “agents”, hence the appearance of the term “multi-agent models”. Due to those results, a new research direction appeared – **Artificial Society** (AS) – a continuation of AI in the direction of social sciences. As a result, the programs functioning as artificial brains are used in the computational models for computer simulations of social phenomena.

The theory of cellular automata (started by John von Neumann and Stanisław Ulam) is a rich source of computational models for (among others) the processes taking place in different societies. A cellular automaton is a collection of objects, situated in the space that is regularly divided into cells. The states of these objects change according to where and which objects appear in their close neighborhood. It delivers models of different social interactions such as gossip expansion or an appearance of isolated ethnic groups. Simple rules, specifying dependences, often lead towards highly complex and unforeseeable processes, which result in undecidable or algorithmically intractable problems.

The AI section, which constructs learning machines, delivers computational models meant not only for the observation of the evolution of the individual brains but also for the observation of the evolution of social structures. Its ability to adapt to the new conditions is a typical example, illustrating that it is capable of learning. Therefore, learning machines make another computational model for social simulations.

Finally, let us note the benefits brought by the evolution of notions to the interpretation of traditional sociological problems. Although rationality of social structures (eg. a certain type of civilization) makes the main point for Max Weber, a classic of sociology, the term “intelligence” has not been adopted in that context. Recognition of these two guises of the same notion will enable us to use the theory of intelligence and its methods for modeling the already mentioned structures.

2.2. The notion of intelligence or rationality as stated in relation to some social structure has its historical exemplification that reveals a methodological role of the notions of decidability and algorithmic tractability. That was a famous dispute initiated by Ludvig von Mises in the 20s of the last century. It considered a possibility of calculation in socialist calculation (see von Mises [1966]). The dispute reached its climax in the 30s when Friedrich Hayek and Oskar

Lange joined it. The same polemicists continued it after the War till the death of Lange (1965). Today Lange's ideas used in the context of informatics are resumed by some authors (e.g. Cottrell and Cockshott [1993]).

Lange argued with Friedrich A. Hayek (1899-1992) about the most intelligent economic regulator. The question was whether it was free market or central planning. The appearance of computers made Lange believe in his final victory for he treated free market only as a computational instrument for calculation of the proper prices, namely such prices that could ensure the equilibrium between supply and demand. He did not deny the fact that market was performing that duty but according to him, the manner of that performance was too slow and full of mistakes. He believed that it would take a second for the Central Planning Commission computer to calculate perfectly something that would take a long time for the market to count (bearing in mind its slowliness).

On the other hand, being aware of the fact that computational power of machines could not always meet economic complexity, Lange allowed for an auxiliary role of the market as an instrument of central economic control while controlling economy in short distances. Central planners' absolute superiority, according to him, consisted in solving long-term problems of the economic growth. Because market could only currently regulate the economic equilibrium, it was not capable of marking any long-term goals of its development.

Hayek opposed those views on the ground of the considerations about information processing. His thought could be expressed in a short way with the help of the modern terminology of computational complexity. The thesis about the realizability of central planning with the use of computers implies that the algorithms used for this aim are quick enough not to wait for the calculation results for many years. The complexity becomes even more monstrous bearing in mind the fact that a central planner would need the complete data of the whole country regarding supply, demand, etc, in relation to every product so that on that basis he could calculate an optimum price. Next, if necessary, an hour by hour he would need to bring it up to date. On the other hand, a computational system, which is free market, radically limits this flood of data in two ways. Every member of the market game needs only data regarding the price of the product and only these products that remain within his activity. It is similar to the data processing that is parallel, dispersed and, at the same time, it enjoys the merits of analogue computation.

Hayek justified those statements intuitively. Today a notional apparatus of the theory of complexity allows for their precise confirmation. One of the procedures could be as follows. Since there are active advocates of Lange's view who operate the informatics notions, they are expected to offer a proof that economic problems of central planning are solved with the help of algorithms working at polynomial rather than at exponential time and that there exists sufficient memory resources which are also measured polynomially, etc. On the other hand, the analysis of the behavior of the market members should indicate whether the problems they solve can be modeled with the help of some decidable theory and if so, whether they are more algorithmically tractable than the problems faced by a central planner.

The dispute cannot be treated only as a historical tale. Nowadays it gains a new interest for two reasons. Equally important, they come from different directions. Due to the growing wave of socialistic tendencies, which has been observed worldwide, a political reason can be indicated. These tendencies can be observed in the impetuous antiglobal movement or in some standards of "political correctness". Hence, there arises a need of a possibly most precise comparative analysis of both socialist economy and free market. According to Lange and Hayek, the theory of computational complexity in its present state of advancement makes an ideal tool. Even if it was not for a practical demand, the advanced state of the tools encourages us to test them in such an interesting theoretical field. Being theoretical and mutually reinforced by the practical one, it makes the second reason to continue the dispute.

3. ADDRESSING COMLEXITY

3.1. There are many ways to deal with the complexity of natural, mental and social processes: to simplify problems, to reinforce computational means, to go beyond the Turing machine, and to create an interaction between intuition and algorithm. The information civilization consists in a growing ability to deal with complexity in these various ways. A struggle against complexity takes place at least on two fronts. The first front is operated by the theory of deterministic chaos (of the unstable dynamic systems) whereas the second one is dominated by the theory of complexity - the topic of our considerations.

First, it is recognized what problems are within algorithm's reach. Undecidable problems as well as those potentially decidable but depending on inaccessible time and space resources remain out of reach. Time is a number of steps necessary to solve a problem. Space is a memory capacity that could be insufficient when confronted with a huge number of input data (other resources may be involved - for example, a number of sharing processors - but the former two are considered most often). Differentiation between two time categories – polynomial and exponential – makes a demarcation line dividing the sphere of what is algorithmically impossible from what remains within reach.

Polynomial time is exemplified by function n^3 whereas function 2^n (where n is a number of input data) exemplifies **exponential time**. Let the polynomial $7n^3 + 5n^2 + 27$ (with n of input data) define a maximum number of steps. To evaluate the complexity of a polynomial algorithm, it is enough to consider its highest exponent omitting the coefficient as a negligible quantity (as 7 in $7n^3$). This distinguished exponent defines the order of the algorithm complexity. It is said that a given algorithm requires, for example, time $0(n^3)$ (notation with 0 indicates the limitation to the order of magnitude with the omission of the negligible quantities). A sorting algorithm with the order of complexity of $0 = n \log n$ (hence it is less than $0 (n^2)$) is an example of the algorithm working at polynomial time.

A problem of satisfiability of the formula of the propositional calculus (abbreviated as SAT) belongs to the class of the problems demanding exponential time. Having a given formula of the propositional calculus in, for example, the conjunctive normal form (that is, the conjunction of alternatives), it is necessary to identify whether there exists such a configuration of logical values assigned to the variables, which makes the formula true. Suppose, the formula has 300 variables. In the least favorable case (when, for example, only one assignment makes the formula true and it is found in the end) the solution will take 2^{300} steps.

The traveling salesman problem makes another example of an unimaginably big request for time that, being factorial, is even greater than exponential. Having data of the whereabouts of n towns, his task is to visit all of them using the shortest route and visiting each town only once. Let him give 20 towns to visit (a starting place does not count). Then a number of routes is 20! since this is the number of possible arrangements in the set of 20 elements. An algorithm, which consists of all its possible arrangements, summing the length of the sections of each combination and a final recognition of the smallest sum, has not been found yet. In the light of the fact that

$$20! = 2,432,902,008,176,640,000$$

it is possible to imagine what algorithmic intractability depends on. If a computer is capable of checking a million combinations during a second, it will take 77.000 years to check all of them. Suppose we add a few more towns and the calculation would last longer than the age of the universe. This is an example of the algorithm that uses the so-called brute force; that is to say, it is based on the mechanical realization of all possibilities. There is no quicker algorithm for this problem, capable of giving an equally certain and precise result. However, in case of our acceptance of approximate results, the time of solving the problem of the traveling salesman can be shortened to a large extent.

3.2. Usually, the class of the problems solved by polynomial algorithms is marked with the letter P. This concise notation simplifies a consideration that has become the source of imposing results in the theory of complexity. Having marked another class of problems by the abbreviation NP, there arises a fundamental question – are these classes equal: P=NP?

This consideration has its roots in the observation that there are polynomial algorithms that answer the questions regarding the decision (the answer "yes" or "no") by giving certificate then and only then when the answer "yes" is true. The question whether a given formula of propositional logic is satisfiable (if it is really satisfiable) will be answered positively, for example, by a polynomial algorithm. Likewise it is in case of the traveling salesman when the question is: is the length of a given route not bigger than such-and-such number?

> To have a closer look at the example of decidability concerning the property of being prime, let me make use of the following recollection. Once Andrzej Mostowski, a widely known researcher of the problem of decidability, gave me a funny and simple task to solve. One day, while arranging a meeting, he asked me about my address. When he heard that the number of the flat was 917, he was quick to ask whether it was a prime number. Surprised, I thought that he should have known the answer (although I did not). Now I am sure that the question had a didactic character – Professor Mostowski was testing the way I would cope with it. Although on a late reflection, I would cope with it in a typically non-deterministic way by giving, for a start, the first answer that comes to my mind after the elimination of those that are certainly erroneous. After calculating that 9 +1 + 7 cannot be divided by 3, I eliminate even numbers and number 3. Then I eliminate 5 for 917 ends neither with 0 nor with 5. So I approach 7, ready to consider the following candidates (11, 13, etc). A strategy has been chosen but I am ready to leave it and look for another one any time it fails. Therefore, I am using a simple polynomial algorithm where a number of steps (execution time) has a linear dependence on the length of the number sequence indicating the divided number. I am lucky - the first attempt gives the result without a remainder of the division – 131. Consequently, the solution is as follows – 917 is *not* a prime number. In this manner, it is possible to guess any big numbers - it is a matter of calculating ability. For instance, it would take a short time for a particularly talented person to guess that 226107 could be divided by 777 without a remainder (the result is 291).

This kind of polynomial algorithms is described as non-deterministic because they aim at the verification of the statements whose acquisition is not determined by any procedure (they should not be mistaken for probability algorithms). "Non" gives the letter N in the abbreviation NP. If the problem does not fall into the NP class, even a solution limited to such a confirmation may appear to be highly difficult. An assumption that the right solution is always within reach is a fiction, impossible to be realized by any machine. On the other hand, this fiction is very useful for it allows for the already mentioned question (P=NP?).

The class P is included in NP in the sense that if there is a polynomial algorithm to solve a problem in a general way, it will also serve in all those cases that are reduced to the question whether such-and-such solution is correct? On the other hand, the thesis that P includes NP has not been proved. For if such a thesis existed, the fact that a certificate problem in case of the traveling salesman is polynomial would result in the conclusion that the traveling salesman problem is polynomial in general or, in other words, it belongs to P. A hypothesis that the answer to the question $NP (P$? is negative is widely accepted. Therefore, $NP \neq P$.

The NP class contains a subset of problems which are NP-complete. The problem is said to be **NP-complete** when in belongs to NP and has a following property: if there is a polynomial algorithm for some NP-complete problem, there is a polynomial algorithm for each problem in NP. It results in the conclusion that if a polynomial algorithm could solve at least one problem of this class, the same would apply to the rest (bearing in mind their mutual conversion). Therefore, the equality P= NP would hold. Algorithm conversions of this type take place at polynomial time, which makes it tractable.

To summarize, relations between the classes of complexity under consideration are as follows. Both P and the class of NP-complete are proper subsets of the NP class. They are also thought to be disjointed.

A problem of satisfiability remains in the NP-complete class. It was the first to be recognized as revealing that quality. It also has become the measure of tractability for the remaining points of the NP class; if it could be solved polynomially, it would apply to the whole NP class. What is more, apart form the traveling salesman problem, the NP-complete class also encloses many problems from different fields – graph theory, operational research, cryptography, theory of games, and theory of social choice.

3.3. The disability to solve NP problems results in the search for their approximate solution. Stratification of this class is conducted according to the complexity level, which is more complicated than the above-mentioned basic distinctions. The NP-complete theory has been developed in the direction of various issues of approximation. For this aim, approximation algorithms are being created. A below-given quotation, taken from the study [Impag-WWW, p.2], shows that this research must be based on a subtler theory.

> "Define SNP to be the class of properties expressible by a series of second order existential quantifiers, followed by a series of first order universal quantifiers, followed by a basic formula (a boolean combination of input and quantified relations applied to the quantified element variables). This class is considered for studying approximability of optimization problems". The authors refer to Papadimitriou and Yannakakis [1991].

Another example how to deal with planning complexity involves coding a plan in the propositional calculus. After assigning logical values to variables, it is finally translated into the original planning problem. This method is described by Ernst et al [1997] at the beginning of his paper:

> "Recent work by Kautz et al. [1992] provides tantalizing evidence that large, classical planning problems may be efficiently solved by translating them into propositional satisfiability problems, using scholastic search techniques, and translating and resulting truths assignments back into plans for original problems."

The same paper shows (it has also been stated experimentally) that by using the above-mentioned method it is possible to reduce a number of variables by half whereas the formula length can be reduced by 80%. Consequently, planning problems, reconstructed as a result of their decoding from the simplified formulae, are made considerably easier.

The above examples fall within a general strategy of approximation and simplification, where the following directions may appear:

When, under a given mathematical model, a problem is too complicated, we simplify the model in question. However, care should be taken to make sure it still remains an approximation of the reality that is close enough not to ruin the accuracy of predictions.

If we do not decide to simplify a model, we have to be satisfied with what is only an approximate solution. This approach is successfully represented by genetic algorithms that is, imitating the process of the Darvin evolution in a specified population (e.g. of mathematical formulae or programs) with its laws of natural selection, heredity, struggle for survival (individuals who do not fulfill the criteria die) and chance mutation (which is reinforced when leading towards the criteria fulfillment). For instance, genetic algorithms cope well with the traveling salesman problem.

We look for an accurate solution but without being certain of getting it. In this case, there appears a necessity to accept a quite high level of the solution accuracy, which requires certain methods of the accuracy estimation.

Limitations of the results connected with these methods do not have to do much cognitive harm. In science, just as in everyday life, we are forced to simplify and approximate. Why should it be different in the sphere of the research that operates with algorithms? It is superior to the traditional research methods because it allows to estimate both precision diversion and their cognitive consequences.

3.4. In the process of dealing with complexity there exists a reverse direction, opposite to that advocating a deliberate agreement with the restrictions. As a starting point it also recognizes restrictions but only one of them is being considered. The fundamental limitation which has to be acknowledged consists in incompletability of arithmetic as well as undecidability of logic. It is connected with the qualification of the range of algorithm power. In this respect, the orthodox position in computer science amounts to the Church-Turing thesis. That is, to the claim that (roughly speaking) any device capable of algorithmic solving problems equals the computational power of the Universal Turing Machine (UTM).

Such a clear statement of limitations shows a field on which it would be possible to aim at getting superiority over UTM. To wit, there arises the following question whether the same problems, solved by UTM during too long a period of time (and therefore intractable), could be solved in a considerably quicker manner? The answer is positive. Some computational systems have appeared, due to which it is possible to cope with the complexity of different problems better. Here is their exemplifying overview.

Parallel computing takes place when a certain set of processors executes one task that is divided among them. **Distributed computing** appears when a computational process is divided among the computers remaining in the net; the data is being shared among them. Although each case involves different elements of the set (in the first case these are processors of the same computer whereas in the second one these are independent computers), a certain analogy takes place between the systems, which found its way in the title of the electronic magazine entitled *Journal of Parallel and Distributed Computing*. Both systems accelerate computing processes noticeably. Distributed computing deserves a special research aiming at its verification as a free market model, imitating the aspect which Friedrich Hayek called distributed (or local) economic knowledge as differentiated from centralized knowledge required by central planning system.

Interactive computing consists in the interaction between the system and the environment, as a result of which the system learns. The fact that there is no need to equip the system with highly complex algorithms, ready to face different circumstances, makes the essence of the obtained improvement. Instead, it is equipped with the program, controlling its learning on the basis of the information obtained from the environment, which makes the strategy incomparably more economic. A self-steering missile is an example of such a system.

It behaves respectively to the obtained observation. The ability to react according to the environment requires a set to be equipped with suitable devices (input, output devices).

Cellular automata (CA) (mentioned earlier in the section regarding Prisoner's Dilemma) are called so because they consist of simple objects located in the cells, reminding of a pattern of the chessboard. Each object has a certain number of possible states (e.g. alive or dead; or white, black or other color, etc). Because these objects change their states, they undergo some evolution. It is done according to the imposed rules that condition a choice of such-and-such state, depending on the situation in the environment (e.g. an object disappears after it has been squeezed by the presence of other individuals around). Self-reproduction is one of the processes taking place in the objects. A construction of automata, capable of self-reproduction with the help of the materials located in their environment, was a primary intention of von Neumann while his works on the construction of CA. Apart from the simplicity of the rules, underlying the CA behavior, it can often become unforeseeable. It allows them to be used to simulate unstable systems (chaotic), particularly studied by Stephen Wolfram. It has also been confirmed that CA has the power of UMT (the same range of the solvable problems). However, it is accompanied by an incomparably bigger efficiency.

Neural networks are highly simplified, physical (hardware) or logical (software) imitations of the nervous system. Their ability to learn gives them a basic advantage over UTM. Another difference is the fact that their actions remain only partly digital and partly analogue (which imitates analogue chemical conditions of the body, e.g. performance of the neuron transmitter). When it comes to the comparison with the UTM computational power, an orthodox view maintains that it is not bigger. On the other hand, the speed and ability to solve problems (to compute) are bigger. However, there is a group of non-conformists who firmly believe in the superiority of the networks, or in other words, in their ability to solve the problems which cannot be solved by UTM.

3.5. Together with this controversy let us consider the last point of the attacks on complexity. It should also be pointed out that a couple of other types of data processing (including quantum computation) will be omitted for the sake of a shortened character of the author's reasoning. The question whether an attack, frontal enough to go beyond a possibility limit of the Turing machine, can appear makes the last point to tackle. Going beyond a possibility limit has become known as **hypercomputation**.

The problem of hypercomputation ramifies into two issues. Although not under this name, one of them followed Gödel's thesis as regards the undecidability of arithmetic, and after the appearance of the undecidability results in logic in 1936. The question was whether the human brain was superior over the Turing machine. Gödel used to give a positive answer to that question whereas Turing (starting from 1950) used to answer it negatively. Apart from the Gödelian version of the negative answer, where the brain is comprehended independently from physics, there is a physical version, developed by Penrose [1989, 1994].

Penrose's view has been made radical in the ideas and projects of the researchers among whom Jack Copeland is acknowledged as taking a remarkable research and writing initiative. The notion of hypercomputaion functions within that group. While Penrose defends the hypothesis that in Nature there are systems capable of solving the problems unsolvable by the Turing machine (meaning brains), hypercomputationalists go further, maintaining a possibility of imitating Nature. According to them, this could be done by the construction of technical devices that are superior over the Turing machine in terms of computational power. It also implies their advantage over equivalent systems, such as digital computers.

Paradoxically, hypercomputation makes attempts to refer to Turing (his surname, accompanied by his photo, was taken as a name of the hypercomputationists' Internet site). This reference deserves a special attention for it allows to discuss an important and, at the same time, mysterious notion of oracle introduced by Turing in his work [1938].[10] Because the work was his Doctor's thesis, supervised by Alonzo Church, it revealed two great names of the discoverers of the problem of computability. That is why historians of logic take a great interest in it.

Since the issue has been a subject of interpretative arguments, it cannot be presented without giving some specific interpretation (as long as it is not restricted only to quotations). The author of this work shares the view of A. Hodges, the author of the famous biography of Turing [1993], who returned to the issue in his lecture [2002] while his polemics with Copeland. Turing [1938] had made an attempt to formalize the notion of intuition understood as a factor that enables one to recognize the truth of the Gödelian statement. Without going deeper in the overview of the attempt's success and results, it is enough to note that Turing had introduced the notion of oracle as something that could compute incomputable functions. It is necessary to add here that he did not treat it as a hypothesis regarding the existence of such a real object (which, according to Copeland, he did) but rather as a fiction helpful for theoretical considerations. In this light, there is no basis to proclaim Turing a patron of superhypercomputationalists. What is more, there is no need to "reconcile" Turing [1938] with Turing [1950].

The success of the hypercomputationalist project depends on the truthfulness of the philosophical view stating that continuous or non-discrete processes (which include analogue computing) do not just appear as real but they really exist. Such "discrete" physicists as Ed Fredkin and Frank Tipler oppose this view. However, the project does not have to be defended on the basis of mere philosophical premises. It is the philosophy of continuity that can gain an experimental confirmation provided an analogue device (if proved to be predominant over the Turing machine in terms of its computational power) is constructed. The Havy Siegelman system [1999] described by its author in the book entitled *Neural Networks and Analog Computing: Beyond the Turing Limit* claims to achieve that aim. A summary of these ideas, included in the lecture (Siegelman [2001]), presents a perfectly described alternative as regards the disability to cross the Turing limit. The author puts it like this:

> "[…] The theory of computational complexity requires the assumption of discrete computation and does not allow for other types of computational paradigms.
>
> We consider a basic neutral network model: finite number of neurons, recurrent interconnection, continuous activation function, and real numbers as weight. This model is considered "analog" for both the use of real numbers as weights that makes the phase space continuous, and the continuity of its activation function. In computational terms, this type of continuous flow (due to its activation function) is definitely a restriction in our model. There are no discrete flow commands such as "If z is greater than 0, compute one thing, otherwise continue in another computation path.
>
> We show that the network […] can compute anything that a digital machine does. Furthermore, due to its real-valued weights, it is inherently richer than the digital computational model and can compute functions that no digital computer can. The network, although transcending the Turing model, is still sensitive to resource constraints and still computes only a small subclass of all possible functions. It thus constitutes a well-defined super-Turing computational model."

[10] A caution against ambiguity of the term "oracle" present in literature must be taken. Apart from the original meaning, referring to Turing [1938], another meaning has appeared. It appears in the NP problem context, where a metaphor of the result guessing has also been called "oracle" (meaning something which is capable of guessing things hidden for others). "Oracle" as used in the second context has nothing to do with uncomputable numbers.

As it is seen from the above-quoted extract, the computational power of the Siegelman network seems to be located somewhere between UTM and oracle in the Turingian sense [1938]. The question whether the network could be defined well enough as a theoretical object and whether, being a physical device, it will experimentally prove its superiority over UTM, remains open (a discussion of this topic taking place in the field allows one to note that).

3.6. Before the above-mentioned argument has been settled, we may rely on the knowledge about people as individuals as confronted with the knowledge about computers. Two human dispositions deserve a special attention here – our ability to accept axioms and the ability to ask questions. The use of these two dispositions along with our commission to perform digital tasks by machines (when they are better than people) will advance the research results to the maximum. Figuratively, if these two dispositions were to fall under the function performed by oracle in the Turingian sense, then an optimum research strategy would consist in a positive feedback between oracle and the machine.

When it comes to the ability to find axioms and recognize their truth, it is necessary to highlight that, apart from few marginal references, this issue is notoriously being neglected in discussions about AI. This fact seems to indicate that the most eager defenders of the claim to reduce human brains to UTM at this stage cannot imagine such an algorithmic procedure that would lead towards, for instance, the discovery of the axiom of choice.[11] This issue is also omitted by the advocates of strong AI although they seem to be burdened with onus probandi that every axiom must be produced by the Turing machine. Though it is possible to imagine a production of axioms by genetic algorithms through the application of a consistency requirement as a probabilistic evolution criterion. In the long run, axioms are expected to be true, while the recognition of truth is a cognitive domain of the human brain.

A construction of the machine capable of making questions would mean another exam passed positively by AI. Among researchers, this issue is covered by silence as it is with the case of arriving at axioms. The silence is much too strange bearing in mind the fact that in the light of the popularity of the Turing thesis, many researchers should be expected to come up with its development when the nearness of the machine and brain is not measured only (as it was proposed by Turing) by the similarity of the answers to given questions but also as to justification, originality and penetration of the questions being asked.

As long as such a machine does not exist, equality between artificial intelligence and natural one remains beyond the dream frontiers. To give answers it is enough to have a database put into memory and a program of their logical processing. To ask questions it is necessary, as it was noted by Ch. S. Peirce, to possess an ability to get irritated or annoyed. Irritation appears due to one's lack of knowledge, ambiguity of notions or some contradictions. Moreover (let us develop Peirce's thought) there are situations in which questions derive from curiosity. Either is a state of the mind that is not separable from a certain emotional state. As far as it is known now, this state can be achieved only by a living

[11] A certain detail from the famous letter of Gödel to von Neumann written in 1956 gives evidence how the attention of logic and the theory of complexity concentrates on the problems of axiom demonstration and not on the search for axioms, which was precursory in relation to the theory of computational complexity and "P=NP?". Gödel asks von Neumann what he thinks about a certain algorithm which, according to Gödel, can fully help a mathematician (as quoted from the text translated from German [Buss-WWW]): "It would obviously mean that in spite of the undecidability of Entscheidungsproblem, the mental work of a mathematician concerning Yes-No questions could be completely[2] replaced by a machine." Footnote 2 after *completely* goes (italics – mine): *except for the setting up of axioms*. The fact that the matter deserves only a footnote in the letter allows us to think that the impossibility of the axiom acquisition on the basis of algorithms remained beyond discussion both for Gödel as well as (in Gödel's opinion) for the addressee. A complete text accompanied by a historical commentary is given by Hartmanis [1989].

protein and not by pieces of silicon or other material carriers of artificial intelligence (von Neumann [1958] claimed that a nervous system has a logic of its own, other than that of digital machine; see also Marciszewski's commentary [1996b]).

The above-given remarks are to reinforce the claim that an optimum research strategy of the future will not consist in successive replacing people by computers but in the parallel increase of their potentiality. Computing abilities of a computer increase together with the creation of quicker algorithms and perfection of equipment. It will result in the growth of knowledge that, in turn, will increase the rise of new questions and new axioms. This is bound to bring new research tasks for humans and computers, and so on, infinitely – provided that both the living intelligence and the mechanical intelligence are to exist in infinitum.

Bibliography

Ajdukiewicz, Kazimierz, "Pragmatical Logic", (trans.) Olgierd Wojtasiewicz, Reidel, Dortrecht, PWN, Warszawa 1974.

[Buss-WWW] Buss, R. Samuel, "On Gödel's Theorem on the Lengths of Proofs II: Lower Bounds for Recognizing Symbol Provability", displayed at:
math.ucsd.edu/ sbuss/ResearchWeb/publ.ps

Carnap, Rudolf, "Meaning and Necessity", University of Chicago Press, 1958.

Chaitin, Gregory J., "The Limits of Mathematics, The Unknowable, Exploring Randomness, Conversations with a Mathematician", Springer-Verlag, Berlin etc. 1998, 1999, 2001, 2002.

Church, Alonzo, *An unsolvable problem of elementary number theory*, in "Am. J .Math." 58, 345-363, 1936.

Church, Alonzo, *A note on the Entscheidungsproblem*, in "J. of Symbolic Logic" 1, 40-41 and 101-102, 1936.

Copeland, Jack and Proudfoot, Diane, "Introduction to Hypercomputation: Computing the Uncomputable", Copyright by the authors, July 2000.
www.alanturing.net/turing archive/pages/Reference

Copeland. Jack, "Turing's O-Machines of 1938", May 2000.
www.alanturing.net/turing archive/pages/Reference%20Articles/hypercomputation/

Allin Cottrell and W. Paul Cockshott. "Calculation, Complexity and Planning: The Socialist Calculation Debate once again".
/www.dcs.gla.ac.uk/ wpc/reports/
Compare the same authors "Socialist Planning after the Collapse of the Soviet Union", Revue European des Sciences Sociales, 167-186. Publisher: Academic Press, 1993.

Davies, Paul, "The Mind of God", (Chapter 5, Part "Unknowable", 128-134), Simon and Schuster, New York 1992.

Grim, Patrick "The Undecidability of the Spatialized Prisoner's Dilemma, Theory and Decision" 42, 53-80, 1997.
See also: www.sunysb.edu/philosophy/faculty/pgrim/SPATIALP

Ernst, Michael D.; Millstein Todd D.; Weld, Daniel S., "Automatic SAT-Compilation of Planning Problems", Proc. the 15th Intern. Joint Confer. on Art. Intell., Nagoya, Japan, August 23-29, 1169-1176, 1997.
pag.lcs.mit.edu/ mernst/pubs/satcompile-ijcai97-abstract.html

Gödel, Kurt, *Die Vollständigkeit der Axiome des logischen Funktionenkalküls*, in "Monatshefte für Mathematik und Physik" 37, 349-360, 1930

Gödel, Kurt, *Über formal unentscheibare Sätze der Principia Mathematica und verwandter Sys-teme – I*, in "Monatshefte für Mathematik und Physik" 38, 173-198, 1931.

Gödel, Kurt, "Über die Länge der Beweisen, Ergebnisse eines mathematischen Kolloquiums Heft 7", Franz Deuticke, Leipzig und Wien 1936.

Hartmanis J. and Stearns R., "On the computational complexity of algorithms", Transactions of the AMS 117, 285-306, 1965.

Hartmanis J., "Gödel, von Neumann and the P=?NP problem", Bulletin of the European Association for Theoretical Computer Science 38, 101-107, 1989.

Heller, Michał, "Wszechświat u schyłku stulecia", [The Universe in the Close of the Century], Znak, Kraków, 1994.

Hilbert D. und Ackermann W., "Grundzüge der theoretischen Logik", Springer, Berlin 1928.

Hodges, Andrew, "What would Alan Turing have done after 1954?", Lecture at the Turing Day, Lausanne, 2002.
www.turing.org.uk/philosophy/lausanne2.html

Impag-WWW] Impagliazzo, Russell; Paturi, Ramamohan, and Zane, Francis, "Which Problems have Strongly Exponential Complexity?" (Research supported by a NFI grant), cm.bell-labs.com/cm/ms/who/francis/ papers/focs98-subexp.pdf

Kautz H. and Selman B, "Planning as Satisfiability", Proc. 10th Eur. Conf. on AI, 359-363, Wiley, Vienna 1992.

Marciszewski, Witold, Murawski, Roman, "Mechanization of Reasoning in a Historical Perspective", Editions Rodopi, Amsterdam 1995.

Marciszewski, Witold, *Leibniz's Two Legacies. Their Implications for Knowledge Engineering*, in "Knowledge Organization" vol. 23, no. 2, 1996-a.

Marciszewski, Witold, *Hätte Leibniz's von Neumanns logischen Physikalismus geteilt?*, in "Beiträge zur Geschichte der Sprachwissenschaft", 6.2, 1996-b.

Neyman, A., *Bounded Complexity Justifies Cooperation in the Finitely Repeated Prisoners' Dilemma*, in "Economic Letters" 19, pp. 227-229, 1985.

Papadimitriou C. and Yannakakis M., *Optimization, Approximation and Complexity Classes*, in "Journal of Computer and System Sciences", 43, 425-440, 1991.

Penrose, Roger. "The Emperor's New Mind: Concerning Computers, Minds, and the Laws of Physics", Oxford Univ. Press, Oxford etc. 1989.

Penrose, Roger, "Shadows of the Mind", Oxford University Press, New York 1994.

Siegelmann, Hava T., "Neural Networks and Analog Computation: Beyond the Turing Limit", 1999.

Siegelman, Hava T., "Computing Beyond Digital Computers by Neural Models", a lecture at the conference "Hypercomputation Workshop", 2000, University College, London. www.alanturing.net/turing archive/conference/hyper/abstracts.html

[Sirridney-WWW] http://www.sirrodney.nu/public/english/Misc

Tarski, Alfred; Mostowski, Andrzej; Robinson, Raphael M., "Undecidable Theories", North Holland, Amsterdam, 1968.

Turing, Alan, "On Computable Numbers, with an Application to the Entscheidungsproblem", Proc. of the London Math. Society, Series 2, 42, pp. 230-265, 1936.

Turing, Alan, "Systems of Logic Based on Ordinals", Proc. of the London Math. Society, Series 2, 45, pp. 161-228, 1939. quoted as Turing [1938].

Turing, Alan, *Intelligent machinery: Raport, National Physics Laboratory*, in B. Meltzer and D. Michie (eds), "Machine Intelligence" 5, Edinburgh Univ. Press, Edinburgh 1969.

Turing, Alan, *Computing Machinery and Intelligence*, in "Mind" 59, 433-460, 1950.

von Mises, Ludwig, "Human Action. A Treatise on Economics", Contemporary Books, Chicago, 1966.

von Neumann, John, "The Computer and the Brain", Yale Univ. Press, New Haven 1958.

Wolfram, Stephen, *Undecidability and Intractability in Theoretical Physics*, in "Physical Review Letters" 54, pp. 735-738, 1985. Displayed at:
www.stephenwolfram.com/publications/articles/physics/85-undecidability/index.html

Wolfram, Stephen, "Cellular Automata and Complexity", Addison-Wesley, 1994.

Wolfram, Stephen, "A New Kind of Science", Wolfram Media, Inc., 2002.