Anticipatory Computing: from a High-Level Theory to Hybrid Computing Implementations

Mihai Nadin

Ashbel Smith University Professor,
Director, antE – Institute for Research in Anticipatory Systems
University of Texas at Dallas, AT -10, 800 West Campbell Road
Richardson TX 75080-3021
nadin@utdallas.edu

ABSTRACT

The perspective of natural phenomena as computational expression can help us find ways to carry out anticipatory computing. With this goal in mind, we can reach back to Feynman's attempt to define quantum computation. His understanding that space-time states can be defined not only in reference to the past and the present, but also to the future proves significant for showing how anticipatory processes can be computationally simulated. Anticipatory computing is embodied in adaptive, non-deterministic, and open-ended information processes. Given the realization that failure to acknowledge anticipation results in major breakdowns (such as the current global financial crisis), the need for anticipation-based computational applications is higher than ever. In this article, an anticipatory control mechanism implemented for the automotive industry is presented.

Keywords: adaptive, anticipation, DNA computing, feed forward, quantum computing, self-referential.

The notion of computation can be understood only against the background of its ever-increasing number of embodiments. Although the Turing machine remains the dominant reference, more and more attempts are made to transcend it. By its nature, anticipation is not reducible to Turing computation. A good theoretic foundation of anticipatory computing is the prerequisite for deploying this form of computation for practical purposes.

1. THEORETIC FOUNDATIONS

In a far-reaching attempt to formulate fundamental aspects of quantum computation, David Deutsch elegantly stated [1985]:

> . . .a computing machine is any physical system whose dynamic evolution takes it from one of a set of "input" states to one of a set of "output" states [. . .] the machine is prepared in a state with a given input label and then, following some motion, the output state is measured. For a classical deterministic system the measured output label is a definite function *f* of the prepared input label. . . . [p. 97].

In so many ways, his view resonates Richard Feynman's early predicaments on the subject. Ultimately, we owe to Feynman the realization that the specific forms of computation are best carried our in the medium we are trying to explore. DNA computing can be emulated in silicon; but, as Leonard Adleman [1994; Adleman & Gifford 1994] proved, choosing the difficult Hamiltonian Path Problem, the efficiency reached in working with DNA and not with a digital representation is higher by many order of magnitude [1,2]. Adleman [1998] compared the timescale of ca. hundreds of years of computing "using the best available algorithms and computers" to the week it took him with his "liquid computer [3]." All this is history by now,

because 10-15 years ago is, relatively speaking, a very long time in science. But it is worth mentioning since in the meanwhile, the scientific community came to the realization that, independent of whether we deploy computing machines or not, science is computational in the broadest sense of the word. What has not been very clear, and still causes controversy, is whether all science is reducible to the universal Turing machine. Formulated differently: Are there irreducible aspects – such as quantum phenomena – that in order to be understood would require a quantum-based form of computation? Initially, as Adleman reported, he was inclined to make a DNA computer in the image of a Turing machine, with an enzyme replacing the finite control. (Bennett and Landauer [1985] suggested the same idea when they addressed "The Fundamental Physical Limits of Computation") [6]. The thought is simple: In the view of many scientists, computationally inclined or not, all it takes to compute is a way to store data and methods for operating on the data. The description applies to an abacus as it does to a supercomputer. This structure assumes that there is information we can define unequivocally, and that operating upon information can also be defined in an unambiguous manner. Most of the time, this is the case. Let us quote Adleman [1998, p. 57] once more [3] :

> To get your computer to make Watson Crick complements or play chess, you need only start with the correct input information and apply the right sequence of operations – that is, run a program. DNA is a great way to store information. In fact, the cell has been using this method to store the "blueprint for life" for billions of years. Furthermore, enzymes such as polymerases and ligases have been used to operate on this information.

Adleman connected his understanding of computation to Alan M. Turing, Alonzo Church, S.C. Kleene, that is, the logical foundations. The well-known Church-Turing hypothesis: "Every function which would naturally be regarded as computable can be computed by the universal Turing machine" translates as the equivalent of algorithm and computation. The fact that this foundational hypothesis does not account for the very important role that time plays in both storing information and acting upon it deserves our attention. Indeed, in nature, some phenomena take place very slowly – just think about the formation and deformation of mountains – while others are very fast – think about the rhythm of change in the living. We have very good algorithms for describing physical phenomena, and therefore we can compute descriptions of such phenomena. But we actually have no algorithms for describing protein folding – to name one example – and therefore no computation of this fundamental characteristic of the living.

1.1. Acknowledging the future

Quite at the time when computing with DNA came into existence, a book with an unusual title, *Feynman Lectures on Computation*, was published (September 1996). Feynman's own involvement with computers dated back to Los Alamos (Manhattan Project, 1943-1945); in his biographical notes, the subject is dealt with among so many others. However, one cannot ignore his own surprise at finding out that some digitally computed data were quite different from the results produced when computing the same data in his mind. He did not specifically bring up the difference made by the medium of computation, but awareness of this difference cannot be ignored. In the early 1980s, Feynman, John Hopfield, and Carver Mead offered "The Physics of Computation" course at the California Institute of Technology. Later on, interaction with Gerry Sussman (on sabbatical from the Massachusetts Institute of Technology) helped him develop "Potentialities and Limitations of Computing Machines." Another interaction, with Ed Fredkin, allowed him to understand the problem of reversible computation; and yet another interaction, with Danny Hillis, gave him the opportunity to become involved in parallel computing.

These biographical details – and there are so many more relevant to the depth of Feynman's involvement with the subject of computation – are well documented. (The interested reader should consult the book edited by Anthony Hey [1999].) They are testimony to a creative approach to science as computation rarely

achieved since. I compared this achievement with Leibniz's contribution to the foundation of computer science [Nadin 1991] [31]. The reason I bring up these details is exactly because in my view, science is again challenged to renew itself, to transcend those traditional boundaries within which it proved, quite successfully, to be the agent of change we know it became in modern times.

In an article entitled "Simulating Physics with Computers," Feynman [1982] made relatively clear that he was aware of the distinction between what is represented (Nature – his spelling with a capital N, and nothing else, since physics always laid claim upon it), and the representation (computation) [15]. The physical system can be simulated by a machine, but that does not make the machine the same as what it simulates. Not unlike Deutsch, Feynman focused on states: the space-time view, "imagining that the points of space and time are all laid out, so to speak, ahead of time." The computer operation would be to see how changes in the space-time view take place. This is what dynamics is. His drawing is very intuitive:
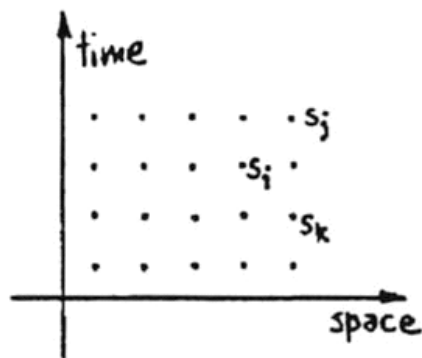

Figure 1. Feynman's [1999] original state diagram

The state s_i at space-time coordinate i is described through a function F (Feynman did not discuss the nature of the function):

$$s_i = F_i(s_j, s_k, \ldots) \qquad (1)$$

The deterministic view – i.e., the past affects the present – would result, as he noticed, in the cellular automaton: "the value of the function at i only involves the points behind in time, earlier than this time i." However – and this is the crux of the matter – "just let's think about a more general kind of computer . . . . whether we could have a wider case of generality, of interconnections . . . . If F depends on *all* the points both in the future and the past, what then?"

Had Feynman posed this rhetoric question within the context of my own research, my answer would be: If indeed F depends on all the points both in the future and the past, then ? Anticipation. Indeed, I define an anticipatory system as one whose current state depends not only on a previous state and the current state, but also on possible future states. Feynman would answer: "That could be the way physics works" (his words in the article cited).

There is no reason to fantasize over a possible dialog – what he would say, his way of speculating (for which he was famous). But there is a lot to consider in regard to his own questions. After all, anticipatory computation, as I see it, states:

1. "If this computer were laid out, is there in fact an organized algorithm by which a solution could be laid out, that is, computed?"

2. "Suppose you know this function F_i and it is a function of the variables in the future as well. How would you lay out numbers so that they automatically satisfy the above equation?"

These are Feynman's words, his own questions. To make it crystal clear: the questions he posed fit the framework of anticipatory computing, but Feynman was not even alluding to a characteristic of a part of Nature – the living – to be affected not only by its past, but also by a possible future realization. Feynman's focus was on quantum computation, and therefore the argument developed around probability configurations. When he wrote about simulating a probabilistic Nature by using a probabilistic computer, he realized that the output of such a machine "is not a unique function of the input," he realized the non-deterministic nature of the computation.

Where Feynman's model and my own considerations on anticipatory computing diverge is not difficult to define. For him, as for all those – from Aristotle to Newton (*Philosophiæ Naturalis Principia Mathematica)* to Einstein – who have made physics the fundamental science it is, there is an all-encompassing Nature, and physics is the science of Nature. For me, Nature, or nature in our understanding of it, is but the contradictory unity between the physical – what used to be called the *inanimate* – and the living – what used to be called the *animate*, which means as much as having life, and being able to influence its own dynamics. This is not the place to reopen the chapter on animism, which seemed tightly closed after reductionism-determinism postulated that whether we deal with life or not, we can reduce all there is to matter and to cause-and-effect. But while such a discussion is in the long run unavoidable, it will not help us define a very distinct form of computation resulting in a more adequate description – i.e., understanding – of how the living "computes" i.e., expresses itself in its behavior.

1.2. Reaction vs. anticipation

At this juncture, that is, after having defined an epistemological context, we need some guiding principles in order to see how anticipatory computing might be simulated (it is performed continuously in the living).They are derived from the knowledge on whose basis society has advanced in the last 400 years.

Principle 1 : The deterministic description we call cause-and-effect (sometimes manifested as action-reaction) describes the physical in a relatively complete and consistent manner.

Principle 2 : Each causal sequence can be described as a change in the state of the system.

Principle 3 : To compute descriptions of causal sequences is to simulate them.

In contrast to the physical, the living is an entity made up of matter and subject to: a) descriptions we call physical laws (gravity, conservation of energy, etc.) expressed as reaction; b) descriptions we call biological expressions (actions, i.e. behaviors, in the first place) expressed as anticipation, i.e., action before, not after, a stimulus, before the apparent cause, although not a causal. That our descriptions of the living account for the fact that possible future states can, and actually do, affect their current state simply says that if we want to compute anticipatory characteristics of the living, we will need to find means for representing past, present, and future information. Regarding past and present information, there are many methods currently in use for achieving this goal. Numerical expression is only one of them. In a study [Nadin 2006], the biologically inspired Evolving Transformation System formalism (ETS) proposed by Lev Goldfarb *et al*. [2008] is used as a substitute for number representation [18,33]. ETS is the mathematical foundation for the structural measurement process. Accounting for future information is more difficult. As

such, this information does not exist as more than possibility. Therefore, the possibility itself needs to be represented, or the dynamics of possibility realization computed. The living generates this information under the guidance of the information regarding the past and present, as acquired through our senses. But it is exactly in this process of generating information that the possible future, or better yet, possible futures, is/ are integrated in the process.

If the question were "How can we simulate anticipatory processes?" the answer, in Feynman's tradition, would be: Let the computer itself be made up of living elements, which account for both the reactive and the anticipatory. That such computer – an anticipatory computer – will produce output – expressed as behavior, not as a unique function of the input – should not surprise anyone.

Thesis 1: Anticipatory processes are non-deterministic.

Thesis 2: Anticipatory processes are self-reflexive.

Thesis 3: Anticipatory processes are open-ended.

With all these considerations in mind, it is time to define the living. We know that the animistic explanation is at best incomplete, and at worst, untenable. But if we want to define a type of computation associated with particular forms of dynamics in nature characteristic of the living, we need to understand the necessary and sufficient conditions for such forms. I am aware of many contributions to the matter. From among all those worth the effort of scrutinizing, I would like to focus on Walter Elsasser's [1998] attempt at a scientific foundation of biology – the science of the living – as well as on Robert Rosen's *Life Itself* [1991], and last on Gregory J. Chaitin, defining life from an informational perspective [13,39]. The intention is to find an identifier of the living on whose basis we can define the characteristics of the particular computation we call anticipatory.

At the foundation of Elsasser's biology lie four principles and a "basic assumption" (as he defines it).

1. *The principle of ordered heterogeneity*. It states that the living consists of structurally different cells. There is order at the cellular level and heterogeneity at the molecular level. Heterogeneity corresponds to individuality, a concept that in the physical (inanimate) world has no meaning.

2. *The principle of creative selection*. This states that, as opposed to the homogenous physical, the variation in structure does no average out. There is always an immense multitude of possible states; selection is the unique realization of one of these possibilities.

3. *The principle of holistic memory*. The creative selection (cf. the second principle) account for two processes: a) homogenous replication (assembly of identical DNA molecules); and b) heterogeneous reproduction (self-generation of similar, though morphologically distinct entities). Replication is a "dynamic process." Replication and reproduction need to be understood in their unity.

4. *The principle of operative symbolism*. Since the discrete genetic message (i.e., genetic information) is represented by a symbol, the question to be answered is "What triggers the generation of genetic message?" We need an answer to this because each new genetic expression – an organism, for instance – is the result of a process involving operative symbols. The biological information is stored as data for the homogenous replication, and as a large array of alternate state choices, from which on will eventually be realized in the heterogeneous reproduction. This implies that biological systems are in part autonomous.

Consequential for defining the specific computation that would qualify as anticipatory is the observation that, "Holistic information transfer involves . . . the reproduction of states of processes that have existed previously in the individual or species as the case may be [Elsasser 1998] [13].

Rosen, with his (M,R)-systems (i.e., characterized by metabolism and self-repair), defined the living as having its own implicit dynamics, while the dynamics of the physical is the result of external forces. He dedicated an entire book to the subject [1991] [39].

In attempting "A Mathematical Definition of Life," Chaitin [1970] realized early on that there is a definite distinction between entities consisting of "independent particles" and entities defined through the "enormous interdependence between their components [7]." He went on to state, "A living being is a unity; it is much simpler to view it as a whole than as the sum of its parts." There are, of course, computational consequences, which we'd better focus on if we want to see what it takes to compute life. Given Chaitin's focus on the size of programs that calculate the dynamics of a system it is not surprising to read the following:

> . . . if we want to compute a complete description of a region of space-time that is a living being, the program will be smaller in size if the calculation is done all together, then if it is done by independently calculating descriptions of parts of the region and then putting them together. What is the complexity of a living being, how can we distinguish primitive life from complex forms?

Chaitin correctly noticed (comparing the cutting of a leg of a table to the same operation performed on an animal) the "interdependence between an animal's past experience and its present behavior; that is to say, it learns." (He missed the self-repair function, which Rosen brought up). A very powerful observation guided his conclusion: "If the whole is very much simpler than the sum of its parts, we have the interdependence that characterizes a living being."

1.3. Complexity threshold: a criterion for defining it.

For the sake of maintaining a clear record of observations, these are principles that ought to guide the definition of anticipatory computing:

1. Holistic

2. Autonomous

3. Non-deterministic

4. Open-ended

5. Capable of learning, adaptive

6. Self-replicating.

This is a high-order set of requirements. There is no need for panic if the state-of-the-art in computation – including quantum computation, on which many hopes are pinned – does not seem to be conducive to the immediate, or even less than immediate, implementation of this kind of computation. Rather, we need to further examine which part of nature actually justifies the effort. Even after summarizing rigorous attempts at defining the living, we must realize that the qualifier *living* is, for all practical purposes, at best undifferentiated. In some ways, stones and stars have a life, which might be different from that of a human being or of a mono-cell, but nevertheless is expressed in a specific dynamics resulting from the natural computation of their respective states. On the other hand, to define the living as above a threshold of complexity does not really help.

One could say: Everyone talks about complexity, but no one really defines it. If we consider the views of Elsasser, Rosen, and Chaitin on the living, we easily identify complexity as the common denominator. But in Elsasser's case, the numbers make the difference: he referred to orders of magnitude of the nature of 10^{30}, for instance. Rosen stated that complexity defined the living. Chaitin, delimiting algorithmic information theory, saw "a law of nature" as a "piece of software, a computer algorithm" and therefore ends up with a measure of complexity expressed in the size of the program (seen as a finite string of bits). My own take is different and goes back to the famous Incompleteness Theorem that Gödel [1931] formulated [17]:

> Any effectively generated theory capable of expressing elementary arithmetic cannot be both consistent and complete. In particular, for any consistent, effectively generate formal theory that proves certain basic arithmetic truths, there is an arithmetical statement that is true, but not provable in theory.

It is necessary to understand that Gödel's argument concerns a standard formal system. It proves that, unless the formal system is inconsistent, there exists a formula in the language, such that neither this formula, nor its negation can be proved. Such formulas are called *undecidable.* A formal system that has undecidable formulas is called *incomplete.* Gödel showed that the scope of the argument was not to cover one specific formal system, but to rule over a wide variety of formal systems, which include all the "mainstream" formal systems that can represent natural numbers.

Observation: Given the specific dynamics of the living, it is conceivable that our ability to fully understand it in a consistent manner is limited. This observation is prompted by all we know from research focused on the living (from molecular biology to genetics to physiology, and so on).

Suggestion for addressing the question, "Why is the complete and consistent description of the living, to date, not granted?" If we take Gödel's theorem pertinent to formal systems and consider that a theory of the living has the living as reference, we can make the following inference: There is an intrinsic characteristic of the living that prevents completeness and consistency of its formal descriptions. Therefore, we infer from the undecidable nature of the formal description to the complexity of the described. This is the criterion for defining the threshold of complexity at which anticipation emerges as a characteristic of the living that I am suggesting. Recently, Jean-Paul Delahaye [2009] alluded to Chaitin's [1974] suggestion that incompleteness is somehow connected to complexity. The nature of this connection cannot be trivialized.

Indeed, there is rich experimental evidence, acquired in a variety of contexts, for anticipatory processes. It turns out that anticipation is a premise for evolution in its broadest sense. In other words, the living is defined by its anticipation. Once anticipation ceases, the living returns to the physical [Nadin 1991]. In other words, below the threshold of complexity reflected in the incompleteness of our knowledge of the living, we are back in the realm of physics. This is the consequence of the criterion submitted for defining the living. The ability to understand the living and the non-living in their unity reflects the specific condition of their respective descriptions, and suggests that anticipatory computing can only be conceived as an open-ended process.

1.4. Impredicativity

From an information viewpoint, Rosen [1966] was fundamentally opposed to von Neumann's understanding of the threshold of complexity [40]. He articulated, quite passionately, bringing up the need to account for the characteristics of the organism as evolvable. Nevertheless, in hindsight we can say that both realized, although in different ways, that if complexity is addressed from an informational perspective, we end up understanding that life is ultimately not describable in algorithmic terms. Non-algorithmic self-assembly (epigenetic progresses) is of such a condition that it does not require either full descriptions of the functions, or of the information involved in living processes.

Given the implications of this observation, we need to give it a bit more attention. Along the line of the Church-Turing thesis – i.e., that every physically realizable process is computable – von Neumann went on a limb and stated, "You insist that there is something a machine cannot do. If you will tell me precisely what it is that a machine cannot do, I can always make a machine which will do just that," [von Neumann 1963] [35]. If von Neumann was convinced that telling *precisely* what it is a machine cannot do – emphasis on precisely – is a given, he was not yet disclosing that *telling precisely* might after all require infinite strings, and thus make the computation to be driven by such a description impossible. Actually, von Neumann should have automatically thought of Gödel in realizing that a complete description, which would have to be non-contradictory, would be impossible.

Before concluding this theoretical section of the article, we shall return to the state-based model of computation that Deutsch suggested. If we want to account for both the reactive (physics) and the anticipatory as definitory, in their complementarity, of the living, we need to realize the structure of the processes involved. Within physics-based explanations, the current state of a system is determined by its past and is deterministically well defined, i.e., non-ambiguous. An anticipatory system is a system whose current state depends not only on previous states, and eventually its current states, but also upon possible future states.
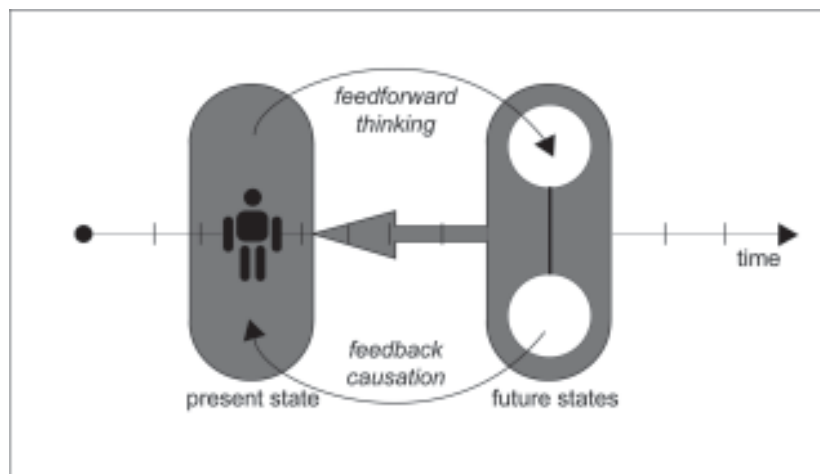

Figure 2. Feedback and feedforward.

This dependence on possible future states is non-deterministic, ill-defined, i.e., ambiguous. (This should not be confused with the indeterminate in quantum mechanics.) The definition suggests that the dynamics of an anticipatory system involves the future as a realization in the vast domain of the possible. There are goals in view, directions for the future, a vector of change. Life is process, more precisely, non-deterministic process. Therefore, in addressing causality in respect to the living, we need to consider past and present (cause-effect, and the associated reaction), both of which are well defined, in conjunction with a possible future realization that is ill defined and ambiguous. When we have to account for higher complexity – the threshold beyond which reaction alone can no longer explain the dynamics – the anticipatory component must be integrated in our understanding. In logic [Kleene, 1950] an impredicative definition is one in which the definition of an entity depends on some of the properties of the entities described [2,4]. The definition of life is an example of impredicativity; that is, it is characterized by complexity, which in turn is understood as a threshold for the living. Impredicative definitions are circular. This is the reason for my attempt to introduce a criterion for defining complexity. Kercel [2007] noticed that ambiguity is an observable signature of complexity. He goes on to connect this to the issue of prediction: "ambiguity of complexity shows that

the 'unpredictable' behaviors of complex systems are not random, but are causally determined in a way that we (having no largest model of them) cannot consistently predict [22]."

Between memory in the physical realm and living memory, the distinction is similar to that between storing a picture and storing the instructions for making the picture, actually many pictures. The physical embodiment of the living (not unlike stones, or pieces of metal, etc.) stores information as matter is literally changed. A wound is the vivid memory of a physical injury. The living learns – this is what brain science defines as brain plasticity, i.e., the capability to adjust, re-organize. The living, within anticipation dynamics, heals itself (the self-repair function). As a physical entity, the living has a degree of complexity fully characterized by reactive behavior described in the laws of physics. When we have to account for higher complexity – the threshold beyond which reaction alone can no longer explain the dynamics – the anticipatory component is integrated in our understanding. There is no doubt that the living, as a complex system, allows for partial descriptions that conform to the reaction model. For example: we can attach to the living a number that defines its current weight, or height, or temperature. But there is no way we can build from such partial descriptions – no matter how many – a holistic entity as complex as life. From quantified discrete aspects of the living, we cannot infer to the complementarity of reaction and anticipation as the origin of its dynamics. To simulate life with computers, we would have to learn to compute in the medium of the living itself. Alternatively, we could proceed with integrating silicon-based computation (for the physics of the living) and the living itself. This hybrid form of computation will be defined in an example.

1.5. Why should anticipatory computing interest us?

When Feynman, and others who addressed the issue of quantum computation, made a case for its possibility, and indeed for its necessity, of course they argued for better ways to acquire knowledge, i.e., to practice science. But they also argued for applications inconceivable with other forms of computation, including the classical computer, as they called it. Among the subjects that could be investigated was anticipation itself. Although Feynman did not use the concept, he defined the subject in the closing of his "Simulating Physics with Computers [1982] [15]:

I mentioned something about the possibility of time – of things being affected not just by the past, but also by the future, and therefore that our probabilities are in some sense "illusory." We only have the information from the past, and we try to predict the next step, but in reality it depends upon the near future which we can't get at . . . .

Since I am writing this article in the context of a worldwide crisis that was caused exactly because of the lack of any anticipation, Feynman's words seem almost prophetic. Indeed, the probabilities that drove the risk models practiced in securitizing subprime mortgages were illusory, and so were the probabilities used in assessing joint defaults (the famous Gaussian copula function). Moreover, the black box speculative schemes on the world stock markets are part of the same broad context of reaction-based trading, to the detriment of any anticipatory consideration. The crisis also reflects our new questions regarding the environment and sustainability. What I am trying to say is that the challenge of anticipatory computing, performed by the living as a condition of its existence, transcends interest in theory, no matter how promising it is for science. As a matter of fact, it is a challenge of enormous practical consequences. In the following considerations, I will argue that while quantum computation, for example, could increase the efficiency of encryption by many orders of magnitude, anticipatory computing could help us address some of the great challenges of computing, as acknowledged by the scientific community.

Arguing from a formal system (the Turing machine, the von Neumann sequential computer, algorithmic or non-algorithmic computation, etc.) to reality is quite different from arguing from a characteristic of the

living (in particular, brain functioning) to formalism. Libet's [et al.1983] readiness potential (i.e., the time before an action, signaled through neurological activity, actually takes place) is an expression of anticipation [28]. It was and continues to be measured, i.e., quantified in various cognitive studies and in brain research. The area of inquiry extends from the anticipation of moving stimuli (vision) to synchronization mechanisms, medicine, genetics, motion planning, and design, among others. (For more details, check www.anteinstitute.org.) Inferring from this very rapidly increasing body of data, two distinct formal definitions of anticipatory systems emerge:

1. An anticipatory system is a system whose current state depends no only upon a previous state, but also upon a future state. (This was already mentioned above.)

2. An anticipatory system is a system that contains a model of itself that unfolds in faster than real time (the Rosen and Nadin models).

3. An anticipatory system is a self-adaptive, non-deterministic system.

These definitions can serve as a basis for conceiving, designing and implementing anticipatory computing pertinent to some critical activities in which humankind is currently engaged.

The more constrained a mechanism, the more programmable it is. Reaction can be programmed, though this is not always a trivial task, even without computers. Although there is anticipation of a sort in the airbag and the anti-lock braking system in cars, these remain expressions of pre-defined reactions to extreme situations. In programming reaction, we infer from probabilities (a shock will deploy the airbag, sometimes without justification), always defined after the fact (collisions result in mechanical shock). They capture what different experiences have in common, i.e., the degree of homogeneity. Proactive behavior can to some extent be modeled or simulated. If we want to support proactive behavior, for instance prevention, we need to define a space of possibilities and to deal with variability. We need to make possible interpretations (e.g., a shock that does not require the airbag should be distinguished from a collision). To infer from the combined possibility-probability mapping of the information process describing the dynamics of reality to anticipation means to acknowledge that deterministic and non-deterministic processes are complementary. This is especially relevant to information security (and to security in general). It is not in the nature of the computer – a homogenous physical entity – to cause security breakdowns, but rather in the nature of those involved in the ever-expanding network of human interactions – a heterogenous entity of extreme variability.

Given the nature of computation, it is quite possible either

1. to achieve effective pseudo-anticipation performance within the forms of computation currently practiced; or

2. to develop hybrid computational mechanisms that integrate physical and living components with the aim of achieving effective anticipatory properties.

These are two distinct research themes within the emerging notion of anticipatory computing.

Information Security and Assurance will become an ever more elusive target within the reactive mode of computation, as it is practiced today. Every step towards higher security and assurance only prompts the escalation of the problem that gave rise to such steps in the first place. In order to break this cycle, one has to conceive, design, implement, and deploy anticipatory computing that replaces the reactive model (such as virus detection) with a dynamic stealth ubiquitous proactive process distributed over networks. Anticipatory computation, inspired by anticipation processes in the living, implies a self-repair component. It also involves learning, not only in reaction to a problem, but as a goal-action oriented activity. The human

immune system, which is anticipatory in nature, is a good analogy for what has to be done. In some ways, anticipatory processes are reverse-computations. Therefore, an area of anticipatory computing research will involve experiments with reverse computation (cf. Nadin, 2010). (limited, of course, by the physical substratum of the computation process, i.e., by the laws of thermodynamics), either through quantum computation implementations or through hybrid computers (with a living component).

Anticipatory computing is indeed a grand challenge. The ALife community could not deliver this kind of solution because it failed to acknowledge the role of anticipation. The current efforts of leading scientists and research centers (e.g., Intel's research in proactive computing, the work of the Department of Energy's Sandia Laboratory, IBM's interest in anticipation) support the claims I made in 1998 – anticipation is the new frontier in science – and in 2000 – anticipation is the second Cartesian revolution. A science of anticipation will be dedicated to the description of even more complex forms of causality than those associated with determinism and reductionism. Social expectations, expressed in the notion of trust – itself an anticipatory entity – are such that this research will eventually become mainstream.

2. ANTICIPATORY CONTROL

Research in anticipation (going back about 30 years) is becoming more widespread as anticipation is discovered in various fields of science (physiology, neurology, cognitive science, biology, animal behavior, human behavior, cell behavior, to name a few). Many publications, covering cognitive studies, brain imaging, medicine, and computers, identify anticipation as such. Usually, it is subsumed – as in swarm behavior, human dynamics, for example – or presented under a different name. Nevertheless, the impressive increase in a large variety of experiments actually based on anticipation continues to produce rich data in the life sciences, mathematics, computer science, artificial intelligence, and the medical fields, for example.

Most encouraging is the emergence of new ways of thinking about computation beyond the use of the classical van Neumann type of machine. Still there is a definite reluctance to embrace anticipatory computation, since the obsession with what the industry calls "the killer app", i.e., the fast money-earning application, prevents many computer scientists from realizing that the action-reaction paradigm reached its limits.

The example to follow represents research conducted initially at the Computational Design Program at the University of Wuppertal (Germany), and subsequently within the framework of the antÉ – Institute for Research in Anticipatory Systems (currently housed at the University of Texas at Dallas). The major reason for the applied research is to provide examples of how anticipation-informed engineering can address problems critical to society. In parallel to this project, a number of other endeavors were pursued in the area of anticipation and risk assessment, human-robot interaction, conception of new tools and products with embedded anticipatory characteristics, and development of interactive games to stimulate anticipatory performance. The project described below (in abridged form) prompted useful interactions with major car manufacturers (Daimler AG, Audi), as well as with researchers in the automotive industry (Mercedes-Benz Research & Development North America in Palo Alto).

As already stated, in an anticipatory system the current state S(t) depends on previous states, its current state and future states. For convenience this can be represented as

$$S(t)=F[S(t-1), S(t), S(t+1), p] \qquad (2)$$

The project emphasizes the implementation of anticipatory computing in relation to possibilistic models. The goal is to make possible the management of complex systems that can be controlled by a computer in coordination with a human user. The project focuses on real-time control, i.e., adjustment. The solution

sought must correspond to the time expectations of control systems, which are expected to function with great precision, but which should also be capable of adjustments to circumstances. The research was conceived in a manner that leads to a solution that facilitates applications for various machines (automobiles, airplanes, boats and ships, and other control systems). A possible extension in the area of human-robot interaction is of extreme interest (and continues to preoccupy us).

2.1. Data integration

So-called hybrid control mechanisms, i.e., involving computers and users (there are also other definitions, such as the integration of analog and digital) still represent a relatively new area of research. User-independent control mechanisms facilitate the automation of certain production procedures. In a hybrid context, autonomous control should complement user control. In addition, the control mechanism should support the user in making control decisions. Control mechanisms are usually reactive: something happens and the control mechanism reacts to this (to a disturbance, for example). In an anticipatory system, anticipatory procedures will be facilitated in such a way that knowledge related to the controlled process is integrated in the decision process. Major research in this direction has been performed in the area of intelligent agents, fuzzy modeling, autonomous control, etc. **[**Zadeh 1972; Kohonen 1982; Nauck & Kruse 1993; Isidori 1995**;** Nürnberger *et al*. 1999; Zimmermann 1995; Michels *et al*. 2002] [20,25,29,34,37,45]. In recent years, Balkenius [2007, 2008], Johansson [2007], Pezzulo [2008], Castelfranchi [2008], among others specifically dealt with anticipatory aspects in their respective research [5,21]. Risk and anticipation is the subject of an entire issue if the journal Risk and Decision Analysis (Ndin, 2009). A comprehensive review of the role of anticipation in a variety of applications is made available in Nadin (2010).

.

In our days, an increasing number of control and automation mechanisms have been implemented as hybrid digital applications. A data bus serves as a unified conduit for relevant information regarding the processes subject to computer-assisted control or automation. The data accumulated with the help of an array of various sensors can be used for communication (information made available locally or at remote locations), and for optimizing the functioning of the system in question. The data can also be fed into integrated
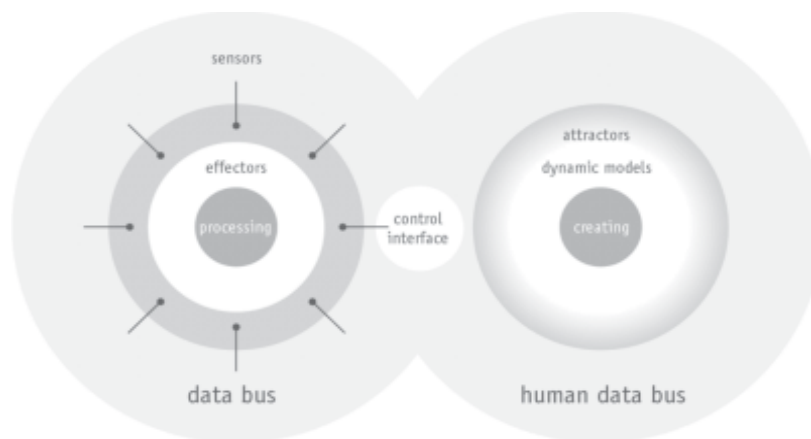

Figure 3. Integration of data bus and human originated data.

automation programs of various degrees of intelligence, which in turn can be networked. In order for an integrated hybrid control mechanism to function, the data pertinent to human actions is subject to a second bus – the living bus – and the two (data bus and living bus) are integrated.

The most optimistic technological view is that eventually, no matter how complex a task or operation is, it can be fully entrusted to a control mechanism programmed to ensure optimal functioning of the system. An intelligent control mechanism would make the system autonomous (self-controlled). The least optimistic view ascertains that for a broad range of applications, the expectation of intelligent self-control is justified. However, for some control systems, of high complexity or of a dynamics scientists still do not fully understand, self-control is unattainable. Using knowledge describing the system controlled, and performing abductive reasoning, we could assess the likelihood of previously minimized events. To this category of events belongs the operation of a system outside its assigned function. Let us take the following examples: The purpose of a car, or the purpose of an airplane, is mobility, i.e., transporting people and goods over long distances. Their purpose is not to ram into buildings during terrorist attacks. But the latter, as deviant as it is, is a possible purpose. Using anticipatory computation, the use of a car or an airplane can be monitored in order to distinguish among possible events, even if they are highly improbable (possibility vs. probability), and to prevent dangerous situations.

2.1.1. *Hybrid control mechanisms*

As various scientists have reported, there are some activities for which the only solution is a hybrid, i.e., a combination of digital control mechanisms complemented by human control, even though this human involvement is increasingly supported by encompassing digital mechanisms: sensor-based data collection, data evaluation, digitally driven controls through effectors, for example. In other cases, the human being is the beneficiary: cars to be driven, boats to be steered, airplanes to be piloted. Among the examples often given in respect to desired human involvement are driving (e.g., cars, trucks); flying (the complexities above and beyond the automatic pilot procedures); navigation (in various situations such as difficult terrain and taxing weather conditions); interaction with robots. The interface to the human operator (driver, pilot, engineer, etc.) should adapt itself to the needs of the operator as these change when the context changes. Example: reduced visibility during a storm or darkness, or blinding sun reflection. These, and other changes, are to be compensated by the dynamic interface. Anticipation can support "form-fitting" interfaces, and thus improve the control mechanisms of the system in question. Obviously, in fully automated production lines, or in dedicated robotic implementations, the issue of a complementary human control capability is eliminated. In human-robot interactions – i.e., the robot expecting something from the human – they are highly relevant. Moreover, in certain kinds of production situations – those that cannot be standardized given the nature of the production process and the many variables involved – complementary human control, supporting enhanced human performance, remains the last option. Example: regardless of how sophisticated drones ("flying robots") are, complementary human control, usually from remote locations, is decisive in increasing their effective deployment (for reconnaissance assignments or for other functions, some related to terror control). Despite the fact that progress has been made in representing the intelligence appropriate to complex control mechanism and expressing it in artificial intelligence (AI) procedures of all kinds – from the expert systems of the past to the neural networks and genetic programs of our days – human control is in some cases the solution of or the newer intelligent multiagent societies last resort. This is related to the fact that the living is capable of anticipatory inferences in ill defined situations (ambiguous information or insufficient information).

2.2. An example: automobile control

Based on experience acquired to date, I have been able to advance the notion of anticipatory computing as part of improved control mechanisms for extremely complex situations (research project advanced at DC RTNA, Palo Alto, CA, May 2002). Let's take the automobile as an example.
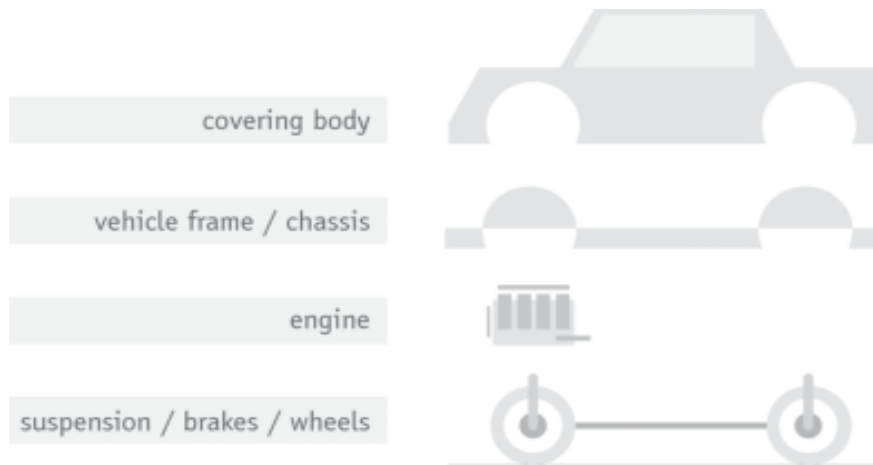
Figure 4. Automobile components.

The digital data bus (implemented in various ways by different manufacturers) makes possible a level of control never before possible. It provides real-time information and supports efficient control mechanisms, including advanced navigation (GPS based).

2.2.1. *Real-time monitoring*

Basically, every component of the complex machine called a car can be monitored in real time. What results is a digital map of this machine's functioning. From this digital map, the user/driver or the maintenance program can infer to the condition of the machine and even prevent malfunctioning or breakdown. As a result, cars endowed with a digital bus and the appropriate mechanisms for control reach, among other things: a) higher security levels (for the driver and traffic participants); b) lower energy consumption; c) lower emission levels; d) higher performance; e) lower cost per mile/kilometer; f) lower maintenance costs.

In fact, a digital bus and the appropriate control mechanisms are the means for reactive and proactive control procedures that are used for: a) accumulating data from the various components; b) facilitating control of individual components or combinations thereof; c) facilitating integration, and automation; d) facilitating generation of a performance map; e) facilitating interface to the outside world (GPS, telemetry, etc.); f) facilitating effective control mechanisms. In the economic equation of mobility, the research and development (R&D) costs for such systems, production costs, and maintenance exceed until recently the immediate benefits to the driver. The secondary benefits (e.g., lower pollution, less dependency on oil, increased security) are usually difficult to assess. But in our times they are more critical than ever.
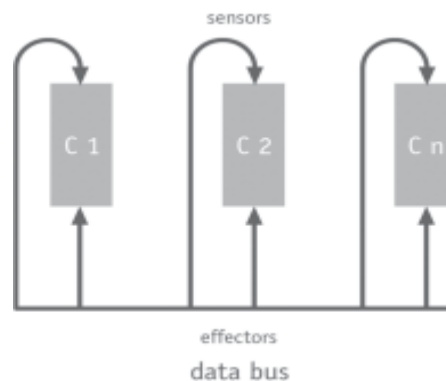

Figure 5. Sensor based reactive control.

2.3. A generalized problem: definition

Similar to the car, any other machine can be considered. It consists of interrelated components, integrated functions, and it provides a quantifiable output.

Any machine (car, airplane, production line, etc.) can be subject to the reactive model of control. Sensors provide data from components. Data are transmitted to the digital bus and eventually made available through some interface with the environment in which the machine operates. In other words, the physics of the

Figure 6. Optical data bus in automobiles.

automobile (and of any machine) is reflected in the digital model; and the control mechanism is focused on the high number of various cause-and-effect sequences describing the deterministic properties of the integrated system making up the machine (in particular, the car).

A very good example of how successful such reactive control procedures are is the integration in the hybrid car of two different modes of operation: electric and combustion based locomotion.

2.4. Specific anticipation aspects related to the problem at hand

Anticipation is usually associated with the mind. It makes possible the performance of actions for which scientists do not have a good deterministic (i.e., cause-and-effect sequence) description. Davidsson *et al.* [1994] provide several examples [9] :

A tennis player has to anticipate the trajectory of the ball in order to make a good hit. A stockbroker makes forecasts of stock prices in order to make a good deal. In short, they use knowledge of future states to guide their current behavior. To do this they make use of an internal model of the particular phenomenon. The experienced tennis player's use of his model is probably on an unconscious level and has been learned through tedious sensorimotor training. A novice, on the other hand, has to use his model of the ball's trajectory in a more conscious manner. Similarly, the stockbroker's model is probably on a conscious level, learned through theoretical studies and experience of previous stock prices. (p. 1427).

Such examples were documented by other authors as well [Wolpert 1998; Dubois, 1999] [12,44].

In respect to the subject of the research herewith described, it is important to notice that in driving, navigation-related activities, orientation, control of extremely complex machinery, etc. anticipatory instances have also been recorded in a wide variety of quantitative assessments. Davidsson *et al.* [1994] continue their assessment by confirming the evaluation of the situation that I described above:

> . . .anticipatory reasoning has not been sufficiently studied and, as a consequence, is not well understood within the field of intelligent autonomous systems. This is probably due to the strong influence from the more traditional sciences, e.g., physics, that have essentially been limited to the study of causal systems which, in contrast to anticipatory systems, do not take knowledge of future states into account. We believe that autonomous systems with the ability to anticipate as described above would exhibit novel, interesting and possibly unexpected properties that might enhance the capacity of autonomous systems. (p. 1428)

With all this in mind, I will proceed in providing a framework for the understanding of the research, insisting on the above-defined elements.

2.5. Anticipation and agent control technology

2.5.1. *Traditional agents*

Growing interest among computer scientists in the field of autonomous agents is probably due to the emergence of a promising model for digitally supported interaction, quite different from the traditional models practiced so far. I refer here to agents technology, which follow an anthropomorphic view of the world, that is, the world seen through the eyes of the human being. Agents are often referred to as reactive, or behavior-based programs endowed with capabilities pertinent to a certain activity or to an array of activities. Early work in agents development [Nilson, 1969]) advanced an approach based on two main components: 1) the world model, i.e. a description of the agent's environment; 2) a planner of a sort. The functioning of this early generation of agents (also known as traditional agents) can be described through the sequence [36]:

Sensory acquisition of data ⇒ world model ⇒ planner based outline of actions ⇒ action cycle   (3)

From sensors ("sense the environment and produce sensor-data," cf. work on *Shakey*, a mobile automaton [Nilson, 1969]), information is derived to update the world model. In turn, this is "used" (i.e., accessed) by the planner in order to "decide" (e.g., decision tree procedures) which actions to take; that is, which output will drive what kind of choice from among those made available in the program. These decisions serve as input to the effectors that actually carry out the actions (i.e., are activated).

2.5.2. Deliberation and action

Writing or automatically generating the software that is expressed in these kinds of agents prompted computer scientists and AI professionals to observe that even if they were able to perform some relatively advanced cognitive tasks, such as planning and problem solving, agents had problems with more elementary tasks, such as routine reaction—something happened that was not predicted in regard to the agent. The cat that jumps in front of a car can be accounted for through the planner, but a "dead pixel" on a scene analyzer might appear as an undecidable bit of information that will confuse the system [Daimler Chrysler 1994]. These occurrences sometimes require fast action but no extensive deliberation. Think about a traffic situation, or a flight control episode (near-collision situation) in order to realize that after a perfect performance, the agent can fail due to a trivial occurrence.

2.5.3. Behavior-based agents

As a premise for the new concept stands the idea, derived from cognitive studies, that most of our

daily activities consist of routine actions rather than being the result of abstract reasoning. In some ways, the behavioral premise was reactivated and tested in a variety of new situations. Instead of ambitious world modeling and difficult planning capabilities – all with high computational cost – the agents are endowed with a finite collection of simple action-reaction cycles, i.e., behaviors. In retrospect, we know that some of the most influential solutions in the form of such agents are Brooks' robots, based on a subsumption architecture. They belong to the "intelligence without representation" paradigm. Noteworthy are also the so-called Pengi implementations [Agre and Chapman 1987] [4] – based on activity theory – and situated automata, with an epistemological substratum (Rosenschein and Kaelbling [1987]) [4,41]. In defining this stage in the development of agent technology, we can join the many authors who make the following observation:

> Probably the most controversial element of this new approach concerns the representation of knowledge. Brooks, in particular, argues that explicit representations of the world are not only unnecessary but also get in the way when implementing actual agents. Instead the agent should use the world as its own model, continuously referring to its sensors rather than to an internal world model [9]. [Davidsson *et al*. 1994, p. 1439]

Behavior-based agents have been shown to perform better in situations in which the first-generation agents fail: accomplish a limited number of simple tasks in real-world domains. However, in addition to not being particularly versatile, they have problems with handling tasks that require knowledge about the world that must be obtained by reasoning or from memory, rather than perception—a notion to which I shall return. According to Kirsh [1991], some possible candidates for such tasks are activities that require a) response to events beyond the agents current sensory limits; b) some amount of problem solving; c) understanding a situation from an objective perspective, or prediction of other agents' behavior [23].

2.6. Synergy of inputs – Multimodal processing architecture

From the perspective of biologically inspired computation, in particular evolutionary models, the main objective to the behavioral reaction is that the living can be described, from the viewpoint of information processing architecture, as a multimodal system. No reaction is reducible to one specific sensory channel. Rather, a combination (prompting the notion of synergy) of inputs, through a number of different channels, explains both precision and richness of human action. This (multimodal) combination allows us to understand that agents reduced to the behavioral model cannot reach a similar precision and richness [Sjölander 1994] [42]. The human being is capable of generalization and abstraction, due to a sort of "central representations" that other species, superceded by human beings in evolution, do not have. As computer scientists working on evolutionary models know very well, evolutionary advantages are relative; they need to be understood in the context of the actions which we try to emulate through agents or other intelligent programs, including programs with learning capabilities. To go from monosensorially governed constructions of several internal representations to a centralized intermodal representation is probably one of the most important aspects in the evolution of mind [Nadin 1991] [31].

2.7. High-level reasoning/low-level reactive capabilities

The suggestion is that as things stand now, agents based on the reactive paradigm will not reach human-level performance except for locally defined tasks. Hendler [1990], Mitchell [1990], Kuokka [1991], Lewis [1991], Ferguson [1992], and especially Zadeh [1996] have convincingly shown that an intelligent agent must have both high-level reasoning and low-level reactive capabilities [14,19,26,27,30,46]. Partially, this is a task related to anticipation and anticipatory characteristics supported by digital processes.

The rationale behind the hybrid approach – machine and the living – is to integrate the reaction ability of agents (quite appropriate for routine tasks) with the human faculty of planning, as this proved to be an

essential ingredient in approaching advanced tasks (driving in a city situation, navigation under less than standard conditions, integration of tasks, etc.). In line with this model, several attempts have been made to integrate anticipation in such systems (e.g., Tsoukalas *et al*. [1989], Davidsson *et al*. [1994], Nadin [1991, 1999] [9,31,32,43]. It should be pointed out here that solutions based on soft computing [Zadeh, 1996] have a different premise: The human being operates on the basis of incomplete, imprecise, and limited information. Still, performance is higher than that of systems designed to reach spectacular levels of precision and that operate on huge amounts of data. Again, my suggestion is that the trade-off between the two computational solutions is due to anticipation. This suggestion is the core of the implementation that makes up the subject of this research proposal.

2.8. Endowing agents with anticipatory characteristics

The system to be designed and implemented is based on an anticipatory computational model that complements the reactive model. An anticipatory model in a hybrid implementation of a control mechanism should make possible performance above and beyond that afforded by control mechanisms based only on a reactive model of the process /machine/ system subject to automation. The extent of the increase in performance will depend on many factors and cannot be predicted by using any of the currently known predictive methods. That an increase in performance can be expected is based on research reported at DARPATech 2002 (Anaheim, California, July 2002), by the ISPI (International Society for Performance Improvement) and by the preliminary reports submitted for Human Performance 2003 (a NASA Advanced Technology Integration conference). Especially relevant to the present application is the emphasis on Non-Invasive Measuring and Monitoring of individual performance.
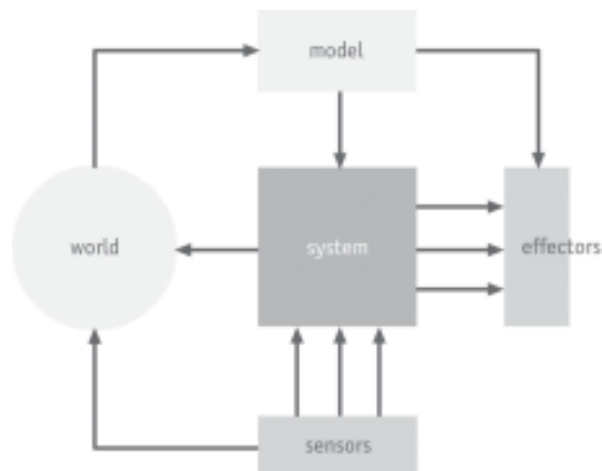

Figure 7. Rosen's model.

2.9. Robert Rosen's model of anticipation

Robert Rosen [1985] set forth one of the first effective definitions [38]:

[...] a system containing a predictive model of itself and/or of its environment, which allows it to change state at an instant in accord with the model's predictions pertaining to a later instant. [p. 339]

In this diagram, the system subject to control S, and the context model W (sometimes called "world") are connected; M is a model of the system operating in faster than real time; sensors (for measurement purposes) and effectors, complete the model. Of special interest here is the Module M, because through this model, predictions are turned into actions. The model affects the system. Its predictions change the

dynamic characteristic of S. The model can also control the Effectors and thus trigger some actions. Here we actually have two concurrent processes: 1) a reactive process at the object level (system controlled); 2) a predictive – anticipatory process – at the meta-level, i.e., the level pertaining to the model of the controlled system.

Their integration is by no means computationally trivial.

2.10. An alternative model

If instead of conceiving a module M – which is only a model S unfolding in faster than real time (such as a simulation, or flight control program) – we implement an architecture of choices, the complexity increases, but so does the ability to avoid a control system than is rather rigid. Indeed, M in the architecture described above is only appropriate if we know that the system has a limited degree of freedom. But if the functioning of a system is by its nature subject to more parameters and to many control paths (driving a car involves the coordination of engine performance, brakes, navigation, etc.) – in other words, if we deal with a non-linear modus operandi (many possibilities), we need an alternative model.

Moreover, if alternative predictions/anticipations are generated, we reach yet another level: Among the various possibilities generated "on the fly," for each new situation, there will be competition. We can thus implement an architecture that makes the following available:

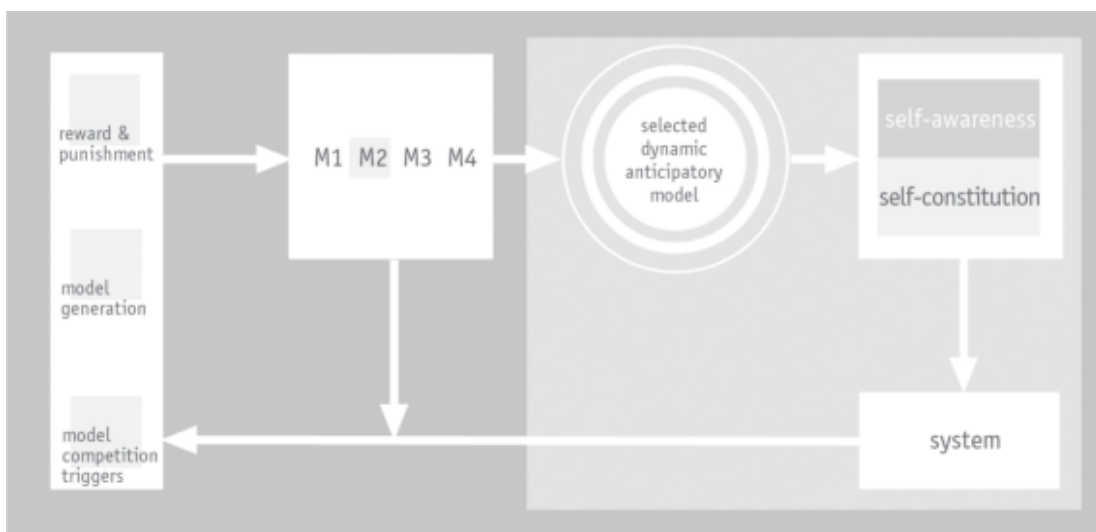- a space of possibilities (in Zadeh's sense [2003]);

Figure 8. Nadin: Competition among models and reward mechanism.

- conflicting possibilities—for each situation, a certain possibility is relatively better than the rest;
- a selection and reward mechanism, much like in cognitive processes of anticipation – computational resources are allocated to the "winning" mechanism.

The following diagram suggests an alternative architecture:

At this moment, the description of processes with anticipatory characteristics is relatively rudimentary:

$$\text{Control} = \text{function of (past state, current state, future state) of system} \qquad (4)$$

In this case, knowledge of future states is a matter of possibilistic distributions.

3. IMPLEMENTATION ASPECTS

A trade-off between how detailed the model M of S and the model of the world and the accuracy of predictions made using these models is from the beginning accepted as unavoidable. From all we know about such computations, it is clear that the higher the level of abstraction (in the description on which the models are based), the better the expected predictions (within the decidable aspect of the task).

More specifically: The model of the machine (car, airplane, boat, robot, etc.) as such and the model of the world in which the machine in question operates (highways, cities, skies, rivers, oceans, etc.) are conceptually quite different. One is a state-machine of a sort expressed in an autopoietic (self-generating) map. The other involves the description of the world in which the machine operates. From the driving simulators in operation today (Berlin, Detroit, Toulouse, and others), we know that this world must have sufficient detail, but still be reduced to a level of generality that makes it simulatable. Situation-dependent knowledge will serve as input to feedforward mechanisms. ("If driving up a hill, higher engine performance expected" is a limited example.) Feedforward mechanisms are also efficient in dealing with disturbances.

3.1. Performance measure

Tsoukalas *et al*. [1989, p. 281] introduced the so-called "anticipated performance." I shall not go into the details of their presentation. What interest us in this respect is the following [43]:

Sensors bring information regarding a hill; the system should anticipate the performance of the car based on the current condition (state of the engine, fuel supply, driver's condition, car load, weather, transmission condition, tire condition, brake condition, etc.). If the state of component C_n is defined as $State_{current}C_n$, this definition is a good measure of the performance of that component. Furthermore, if we know that

$$\text{Performance} = \text{function of State } C_n \tag{5}$$

we can say that given another situation, during which $State_{current}C_n$ becomes $State_{future}C_n$, a new $Performance_{future} = \text{function of } State_{future}C_n$ is definable.

Regardless of whether we can describe it more precisely, or even compute it, there is a relation between the two performances. The question to be addressed is to what extent we can calculate or evaluate in some ways $Performance_{future}$ based on $Performance_{current}$. In terms of soft computing, we can work on membership functions, and focus on "will-be-adequate" or not, leaving the control mechanism to operate on fuzzy descriptions.

3.2. Real-time aspects.

Components 1, 2, ...n can be further distinguished as functionally time independent and time sensitive. The functioning of components over time is adequately described by conditional probabilities: $p(y_j|x_i)$. They form a symmetric matrix [Tsouklas *et al*. 1989, p. 281] capturing experience and variation (as measured by sensors) [43]. If a component performs adequately at time t, it does not automatically mean that at $t = t+ dt$ it will also perform adequately. But adequate itself is hard to define in the performance of complex systems. This is why an idealized model and a state variable distribution had to be developed in order to serve as a reference.

3.3. Learning

Functioning under continuously changing conditions means that control mechanisms have to reflect this dynamic situation. I mention learning here, although the research could not include a learning component

(limited resources, mainly effective know-how). If the project advances the way I defined it so far, it will become possible and necessary to continue work by implementing a learning module. Anticipation always involves learning.

3.4. The Human Bus – Customizing Control

The notion of profiling remains controversial. Still, given the implications of system security, it gained fast acceptance in the IT community. Numerous applications of profiling (user profiles as participants in e-commerce, e-banking, e-learning, etc.) are made available commercially. Such profiles are a digital portrait of the individual: patterns of behavior (mainly decision-making) are extracted from information resulting from transactions performed on-line and off-line. Profiling received much attention after the world became aware of the dangers of identity theft, and even more after terrorist acts and threats (although in respect to this it is rather approached in a defensive manner, i.e., last resort solutions).

In view of all what has been mentioned up to this point, the purpose of creating a human control-data bus (complementing the digital) of a control profile of the individual who drives a car, pilots an airplane, steers a boat, controls a production system, interacts with a robot, etc. is almost reducible to the purpose of generating a profile of the person involved in the action for which anticipatory control and commands is developed. This profile reflects the individual's characteristics as they change over time (including aging).

3.5. Influences on human performance

Human performance is subject to many influencing factors: health, psychological state, environment, interaction elements (other people, interfaces of all kind, etc.), weather, time, season, ageing, and many others. No list can be as detailed as to capture all the aspects involved. Human performance is affected by awareness of being observed, whether the observation is obtrusive or non-obtrusive. Over the time of interaction with a machine, human performance changes. The interesting thing is that machines are conceived as ageless—they should maintain their performance – while aging of the human is almost never accounted for in the design and manufacture of machines, although it affects the performance of control and monitoring.

3.5.1. *Profiles*

With all these aspects in mind, the project speaks for the digital portrait/profile of the user. The way to generate such a profile—probably through neural network digital methods – is to accumulate human performance data, and subject it to data-mining procedures. Depending upon the nature of human performance – different in driving a car from piloting an airplane or using a production system – a profile will capture behavior features. Let's take patterns of braking at a red light as an example.

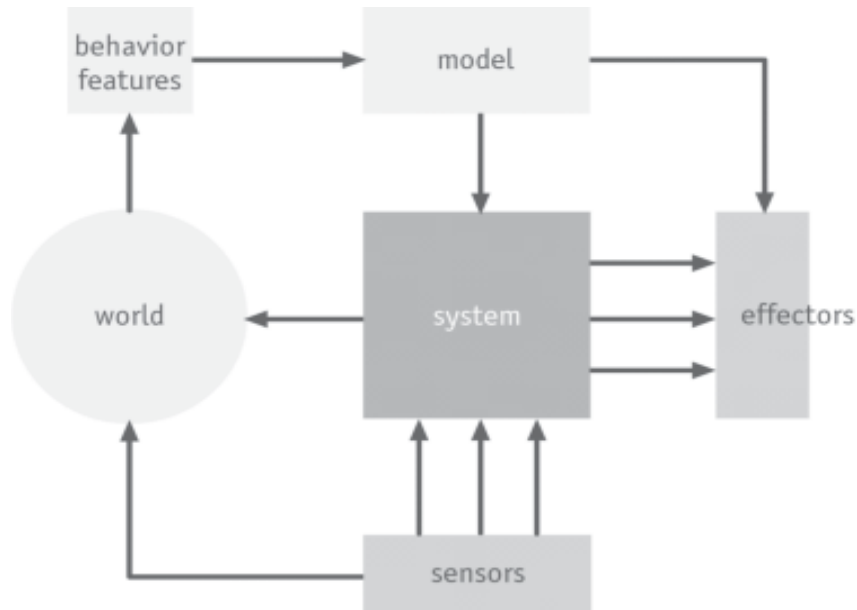**Figure 9. Age as a factor in anticipatory control.**

Figure 10. Integration of behavior features.

These patterns depend on the time (day, night, twilight), traffic conditions, weather, parallel activities (talking to a passenger, listening to music or a report, consulting a navigation system for directions, being on the phone, messaging, etc.). Behavior features are indexed to such parameters (sensors report all the aspects mentioned) and to time. What results is a new situation:

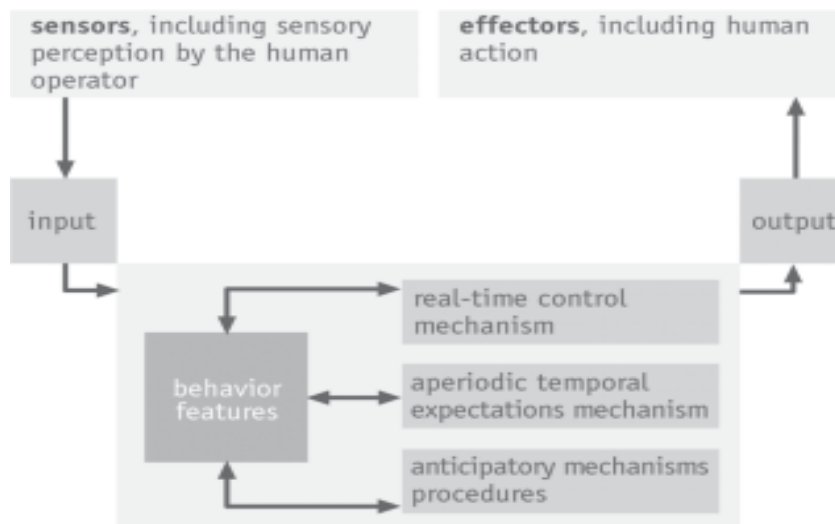As these are checked against each other, a new architecture becomes possible:

Figure 11. Integration of data.

In this new architecture, the patterns of human control constitute a new module. These patterns continue to be influenced by the dynamic properties of the context in which the action subjected to monitoring and control takes place.

Figure 12. Control dynamics.

Figure 13. Phase sequence.

3.6. Architecture of an integrated control

This system should integrate the digital control bus and the human data bus (as expressed through the map of behavior features):
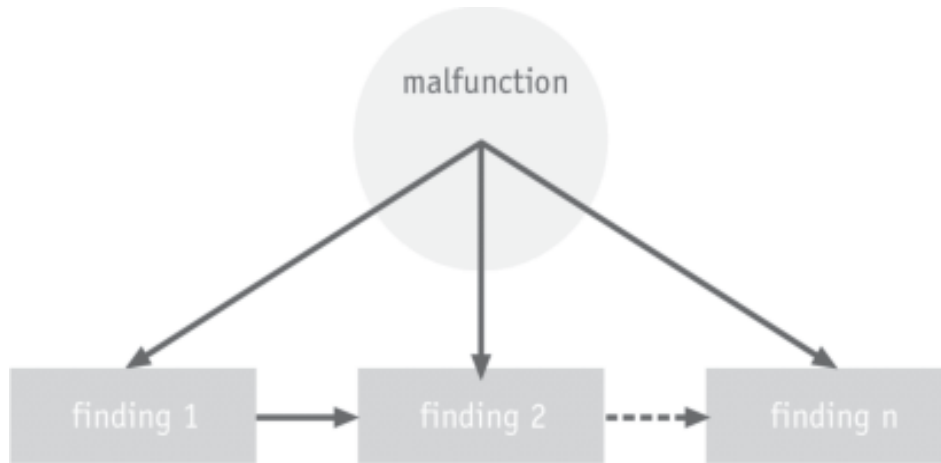
Figure 14. Diagnosing structure.

3.6.1. *Formalization*

Control = Aggregate(Internal proactive agency, Selection mechanisms, Experience) (6)

What we have here, after all, is the following cybernetic control dynamics:

In a more detailed manner, this appears to us as a phase sequence:

The algorithm corresponding to this phase sequence can be formulated as follows:

Read incoming data values (perceptions) from the context (Read function), add values to the knowledge base maintaining their time identification (when), select appropriate action (reaction or anticipation, i.e., pre-computed prediction of future state). If not enough time is available for the computation, trigger reaction.

3.7. Implementation of the anticipatory control system

The actual control is performed over simulated processes. It is too early to select which such processes are more adequate for evaluating performance. In fact, we shall allow for the interaction between a digital data driven anticipatory mechanism and a control profile of the human operator.

The simple network structure or diagnosing a system is sufficient for pointing out the complexities of the implementation, but not the specific ways in which one or the other mechanism needs to be designed or an effective computational implementation.

4. COMPUTING WITH PERCEPTIONS

In his Foreword to my book, *Anticipation –The End Is Where We Start From*, Lotfi A. Zadeh [2003] made the following remark:

Returning to the point I made earlier, my suggested modification of Professor Nadin's definition of Anticipation leads to the concept of what may be called perception-based anticipation. The marriage of anticipation and perception has important implications. First, it highlights that all living organisms, including humans, employ perception-based anticipative control to guide decision-making on goal-oriented stage decision processes. More specifically, if at a stage of a decision process I have n alternatives, $a_1, \ldots, a_n$, to choose from, then using a perception-based model of the underlying system, I form a perception of the

next state and next output, and choose that a_1 that brings me closer to the goal. As a simple example, this is what we do when we drive a car or balance a pile.

More generally, perception-based anticipation is what makes it possible for humans to perform a wide variety of physical and mental tasks without any measurements and any computations. It is this remarkable capability that machines do not have.

In my recent writings, I mentioned a theory, referred to as the computational theory of perceptions (CTP). In this theory, perceptions are dealt with through their descriptions in a natural language, e.g., traffic is heavy, Robert is very honest, speed is high, etc. The use of CTP opens the door to adding to machines the capability to operate on perception-based information expressed in a natural language. In particular, it makes it possible to train a neural network to produce perceptions in response to measurements. Such networks may be said to be *neuroperceptive*. Neuroperceptive networks may find important applications in automation of processes in which the output is a human assessment of, say, food or, more generally, of sensory perceptions [p. 3].

This is a promising avenue. If nothing else, this might become the continuation of the applied research I described above.

5. CONCLUSION

A *living* computer is more difficult to program, given that the living comes genetically pre-programmed. However, in our days, genetic engineering is practiced as a matter of routine. In order to achieve the desired anticipatory performance, we would have to further resolve the issue of representing information pertinent to the future. The mathematical theory of possibilities is one available avenue. Implementations in the form of human-machine applications have the advantage of interfacing naturally embodied anticipatory computing with deterministic forms of computation.

References

Adleman, Leonard and Gifford, David K. On the path to computation with DNA, *Science*, 266, pp. 993-994, 1994.

Adleman, Leonard. Molecular Computation of Solutions to Combinatorial Problems, *Science*, 266, pp. 1021-1024, 1994.

Adleman, Leonard. Computing with DNA, *Scientific American*, August, pp. 54-61, 1998.

Agre, P.E., Chapman, D. Pengi: an implementation of a theory activity, in *Annals of the Association for Artificial Intelligence,* pp. 268-272, 1987.

Balkenius, C., Förster, A., Johansson, B., and Thorsteinsdottir, V. Anticipation in Attention, *The Challenge of Anticipation*. (Pezzulo, G., Butz, M. V., Castelfranchi, C., and Falcone, R., Eds.) LNAI, 5225. Berlin: Springer. 2008.

Bennett, Charles H. and Landauer, Rolf. The Fundamental Physical Limits of Computation, *Scientific American*, 253 (1) July, pp. 48-56, 1985.

Chaitin, Gregory J. A Mathematical Definition of Life, *ACM SICACT News*, 4 (January), pp. 12-18, 1970.

Chaitin, Gregory J. 1974. Information-theoretic computational complexity, *IEEE Transactions on Information Theory* IT-20, pp. 10-15.

Davidsson, P., Astor, E. and Ekdahl, B. A framework for autonomous agents based on the concept of anticipatory systems, *Cybernetics and Systems*. Singapore: World Scientific, pp. 1427-1439, 1994.

Delayhaye, Jean-Paul. *Pour la science*. January (375), pp. 88-93, 2009.

Deutsch, David. Quantum theory, the Church-Turing Principle and the universal quantum computer, *Proc. R. Soc. Lond. A*, 400, pp. 97-117, 1985.

Dubois, Daniel M., Hyperincursive McCulloch and Pitts Neurons for Designing a Computing Flip-Flop Memory, *Computing Anticipatory Systems, CASYS'98—Second International Conference* (D.M. Dubois, Ed.). New York: AIP Conference Proceedings, 465, pp. 3-21, 1998.

Elsasser, Walter M. *Reflections on a Theory of Organisms*. Baltimore: Johns Hopkins University Press. (Originally published as *Reflections on a Theory of Organisms. Holism in Biology*, Frelighsberg, Quebec: Orbis Publishing.) 1998.

Ferguson, I.A. Touring machines. Autonomous agents with attitude, I*EEE Computer*, 25(5), pp. 51-55, 1992.

Feynman, Richard P. Simulating Physics with Computers, *International Journal of Theoretical Physics*, 21 (6/7), pp. 467-488, 1982.

Feynman, Richard P. *Feynman and Computation* (Anthony J.G. Hey, Ed.). Reading MA: Perseus Books, 1999.

Gödel, Kurt. Über formal unenscheidbare sätze der Principa Mathematica und verwandte systeme, *Monatshefte für Mathematik und Physik*, 38, pp. 173-198. The first incompleteness theorem originally appeared as Theorem VI, 1931.

Goldfarb, Lev, Gay, D., Golubitsky, O, Korkin, D. Scrimger, I. 2008. What is a structural representation? A proposal for an event-based representational formalism. Available at http://www.cs.unb.ca/~goldfarb/ ETSbook/ETS6.pdf.

Hendler, J. Abstraction and reaction, *AAAI Spring Workshop on Planning in Uncertain, Unpredictable or Changing Environments* (J. Hendler, Ed.). Stanford: Stanford University Press, 1990.

Isidori, A. *Nonlinear Control Systems* (3rd edition), Berlin: Springer Verlag, 1995.

Johansson, B. and Balkenius, C. An Experimental Study of Anticipation in Simple Robot Navigation, *Anticipatory Behavior in Adaptive Learning Systems: From Brains to Individual and Social Behavior* (Butz, M. *et al*., Eds.). LNAI, 4520. Berlin: Springer, pp. 365-378, 2007.

Kercel, S. Entailment of Ambiguity. *Chemistry and Biodiversity*, 4 (10), 2369–2385, 2007.

Kirsh, D. Today the earwig, tomorrow man? *Artificial Intelligence*, 47(1), pp. 139-159, 1991.

Kleene, S.C. *Introduction to Metamathematics*. Princeton: van Nostrand, 1950.

Kohonen, T. Self-organized Formation of Topological Correct Feature Maps, Biological Cybernetics 43, pp. 59-69, 1982.

Kuokka, D.R. MAX : A meta-reasoning architecture for "X," *SIGART* 2(4), pp. 93-97, 1991.

Lewis, L.M. A time-ordered architecture for integrating reflexive and deliberative behavior, *SIGART* 2(4), pp. 110-114, 1991.

Libet, B., Gleason, C. A., Wright, E. W., and Pearl, D. K. Time of conscious intention to act in relation to onset of cerebral activity (readiness–potential). The unconscious initiation of a freely voluntary act. *Brain*, 106:623-642, 1983.

Michels, K., Klawonn, F., Kruse, R. and Nürnberger, A. *Fuzzy-Regelungen: Grundlagen, Entwurf, Synthese*. Heidelberg: Springer Verlag, 2002.

Mitchell, T.M. Becoming increasingly reactive, *AAAI-90*, pp. 1051-1058, 1990.

Nadin, Mihai. *Mind – Anticipation and Chaos* (from the series Milestones in Thought and Discovery). Stuttgart: Belser Presse, 1991.

Nadin, Mihai. Anticipation: A Spooky Computation, *CASYS, International Journal of Computing Anticipatory Systems* (D.M. Dubois, Ed.), Partial Proceedings of CASYS 99, Liege: CHAOS, 6, pp. 3-47, 1999.

Nadin, Mihai. The Anticipatory Profile: ETS, pattern recognition, and dynamic representation. (unpublished manuscript submitted to L. Goldfarb, Ed., *Pattern Recognition*, Amsterdam: Elsevier) 2006.

Nadin, Mihai (Ed.), Anticipation and Risk Assessment, in *Risk and Decision Anlaysis,* vol.1, Number 2, 2009

Nadin, Mihai. *Anticipation and Dynamics: Rosen's anticipation in the perspective of time,* in International Journal of General Systems, vol. 39, No. 1. 2010, pp. 3-33

Nadin, Mihai. *Annotated Bibliography. Anticipation.* , in International Journal of General Systems, vol. 39, No. 1. 2010, pp. 35-133.

Nauck, D. and Kruse, R. A fuzzy neural network learning fuzzy control rules and membership functions by fuzzy error backpropagation. *IEEE Int. Conf. on Neural Networks,* 2, pp.1022–1027, 1993.

Neumann, John von. The general and logical theory of automata. (From the 1948 Hixson Lecture) Taub, A.H., Ed. *John von Neumann Collected Works*, 5. New York: Pergamon, 1963.

Nilson, N.J. A mobile automaton: An application of AI techniques, *IJCAI-69*, pp. 509-520, 1969.

Nürnberger, Andreas, Nauck, Detlef, and Kruse, Rudolf. Neuro-Fuzzy Control Based on the NEFCON-Model, *Soft Computing*, 2 (4), Berlin: Springer, pp.182-186, 1999.

Rosen, Robert. *Anticipatory Systems*. New York: Pergamon Press, 1985.

Rosen, Robert. *Life Itsel*f. A Comprehensive Inquiry into the Nature, Origin, and Fabrication of Life. New York: Columbia University Press, 1991.

Rosen, Robert. *Theory of self-reproducing automata*. A.W. Burks, ed. Urbana: University of Illinois Press, 1966.

Rosenschein, S., L. Kaelbling. The synthesis of digital machines with provable epistemic properties, *The 1986 Conference on Theoretical Aspects of Reasoning and Knowledge* (J.F. Halpern, Ed.). New York/Amsterdam: Springer, pp. 85-90, 1987.

Sjölander, S. Some cognitive breakthroughs in the evolution of cognition and consciousness and their impact on the biology of language, *Evolution and Cognition*, 3, pp. 1-10, 1994.

Tsoukalas, L.H., Lee, G. W., and Ragheb, M. Anticipatory Monitoring and Control in a Process Environment, Proceedings *of The Second International Conference on Industrial and Engineering Applications of Artificial Intelligence & Expert Systems IEA/AIE-89*, ACM/SIGART, Tullahoma, Tennessee, June 6-9, 1989.

Wolpert, Daniel M., Mial, Chris R., Kawato, Mitsuo. Internal Models in the Cerebellum, *Trends in Cognitive Sciences*, 2 (9), September, pp. 338-347, 1998.

Zadeh, L.A. A Rational for Fuzzy-Control, *Journal of Dynamic Systems, Measurements and Control*, 34, pp. 3-4, 1972.

Zadeh, L.A. The role of fuzzy logic and soft computing in the conception, design, and deployment of intelligent systems, *British Telecom Journal*, 14 (4), 1996.