AI-Kindergarten: A method for developing biological-like artificial intelligence

Inventor: Danko Nikolić

Sachsenhäuser Landwehrweg 96
60599 Frankfurt am Main
Germany
danko.nikolic@gmail.com
Phone: 069 3535 3253

ABSTRACT

There is a problem of building artificial general intelligence (AGI) that can perform various tasks that are today easy only for humans. An AI is needed that can replace humans in many intellectual, perceptual, decisions making, problems solving and creative processes. This AI would understand the surrounding world and is referred to as strong-AI or artificial general intelligence (AGI). The present method for creating strong-AI is based on a recent theory of organization of biological systems named practopoiesis. The method provides cybernetic knowledge across three adaptive levels of the agent: working knowledge, semantic knowledge, and machine genome. The method follows closely the phylogenesis and ontogenesis of biological intelligence. The present method allows very rapid phylo- and ontogenesis of machines as it combines three accelerators, each offering at least a 1000x acceleration compared to biology. The three accelerators interact recursively and form together AI-Kindergarten—a development system that involves extensive play-like interactions with human trainers. The method allows AI to asymptotically approach human intelligence and yet never fully replace humans and still remain safe and under full control of its human creators.

FIELD OF THE INVENTION

The general field of invention is artificial intelligence (AI). More specifically, this invention pertains to creation of artificial general intelligence (AGI). In particular, this invention offers a method for providing artificial agents with knowledge necessary to achieve human levels of intelligence.

BACKGROUND OF THE INVENTION

A problem of the current AI technologies is that they can do very well some things that require effort for a human (e.g., calculating prime numbers, searching databases) but have difficulties doing things that are for humans easy (e.g., perceiving, walking, navigating through space). This problem may reflect an even deeper issue, and this is the fact that neuroscience does not seem to have a satisfactory theory that would explain how the biological brain achieves its functions in the first place.

For further developments of AI, we are in a need of using the principles of biology to a higher degree than what we have been able to do so far. The present invention is based on a new theoretical approach on how biology and brain work, resulting in a radically new view on what the nature of mental and cognitive operations is. This new theory consists of two theoretical steps. First, a set of general principles has been formulated by which adaptive systems organize, the theory being named *practopoiesis* – which stands for "creation of actions". Second, based on practopoietic principles a theory of the organization of brain and mind is proposed, named *tri-traversal* theory (Nikolić 2015a; Nikolić 2014).

One of the fundamental implications of tri-traversal theory is that efficient cognitive operations cannot be implemented solely by computational mechanisms i.e., by manipulation of symbols. Rather, according to the theory, to make mental operations intellectually powerful—meaning to achieve high levels of intelligence, creativity, and flexibility—mental operations have to employ adaptive mechanisms besides the symbol manipulations (Nikolić 2015a). In other words, in a similar way in which a species adapts to its environment throughout evolution and an organism adapts to its environment throughout lifetime, the theory proposes that thinking occurs while the state of the brain adapts to its environment from moment to moment, each adaptation step resulting with a new content of a thought. The adaptation steps at the level of thought are relatively fast, taking often less than a second. These adaptation steps are what we experience as mental events. Thus, according to the tri-traversal theory, a percept, a decision, attention directed or a new idea popping up in our minds, are all acts of fast such adaptive processes. The difference between an adaptive process and the classical approach to AI based on pure computation is that only the latter can be executed in a closed box (e.g., inside a computer without any further inputs from the outside; mathematically it is described as mapping from one set to another that occurs entirely within the agent) while the former necessarily requires iterations with the environment (each action executed needs feedback from the environment followed up by a corrective action, followed by feedback, etc.; mathematical mapping can describe the process only if the environment is also taken into account).

To create biological-like AI, the present invention presumes that it is necessary to mimic biology in respect to the levels of organization at which a biological agent adapts to its environment. However, it is not necessary to mimic any other details of biological implementations. Each adaptive level can be simulated by a machine even if the machine is starkly different from the biological implementation. There is no need to simulate biological neurons or synapses in order to achieve the same type of intelligence as that of human.

An agent created on the basis of tri-traversal theory needs to possess three separate sets of mechanisms that execute adaptive operations; we can refer to them as policies (Nikolić 2015b). In that case, each policy is located at a different level of adaptive organization with an important hierarchical organization: policies lower on the hierarchy act on those higher on the hierarchy. Only the policy on the top of the hierarchy directs its actions towards the outside of the agent (on the external world), whereas all other policies operate internally, making changes to the agent itself i.e., its other policies. Only the policy at the bottom of the adaptive hierarchy is a fixed one, not being changed during the lifetime of the agent.

The three types of policies are respectively referred to as: *Working knowledge* (at the top), Semantic knowledge or *Ideatheca* (in the middle), and *Machine genome* (at the bottom). The actions of Machine genome make changes to Ideatheca. Ideatheca corresponds to what is in psychology known as long-term memory. The actions of Ideatheca make changes to Working knowledge. Working knowledge corresponds to what is in psychology known as short-term memory or working memory. Finally, the actions of Working knowledge produce behavior of the agent. In Nikolić (2015b) it has been argued that such a hierarchy of policies can produce a much higher variety of behavior than can an agent with a single policy for the same amount of total resources.

In practopoietic systems, each policy receives input from the environment and the purpose of these inputs is to trigger actions and provide feedback on the success of those actions. The consequence is that Ideatheca serves a function equivalent to a large repository of learning rules being applied to adjust Working knowledge. Similarly Machine genome provides a depository of learning rules on how to change Ideatheca. By that token, Machine genome enables the agent to *learn how to learn* i.e., how to apply Ideatheca to create new Working knowledge.

Given that there are in total three policies, we refer to this organization as tri-traversal or for short T_3. As defined in Nikolić (2015a), a traverse is a process by which general cybernetic knowledge existing at a lower level of practopoietic hierarchy (e.g., at Machine genome) gets instantiated in a more specific form at a higher level of the hierarchy (e.g., at Ideatheca). Thus, a traverse is like a process of extracting specific cybernetic knowledge from the environment by applying general cybernetic knowledge stored at a lower level of organization (Nikolić 2015a). This general-specific relation of knowledge is closely related to the problem of inductive bias in machine learning (Mitchell 1980), stating that the kind of knowledge that can be learned by an agent is limited by the learning rules that the agent begins the learning process with.

Generality-specificity relation between different knowledge levels i.e., across policies at different levels of the adaptive hierarchy, is an important feature enabling the AI to create intelligent behavior. Much of the capability of an agent to behave intelligently rests on having appropriate learning rules at the bottom of the adaptive hierarchy. If the general knowledge of those low-level learning rules is not designed to extract a proper specific knowledge for a given situation, there is no possibility for the agent to ever act intelligently in that situation.

Advantages of T_3-organization

The advantage of a T_3-organization over T_2 or T_1 stems from the general advantages of having additional adaptive levels. To produce intelligent behavior an agent needs sufficient resources i.e., the bigger its brain, the more knowledge it can store and the more intelligent it can possibly be. The amount of resources that are on the disposal of an agent should match the demands that the environment poses—the more complex the environment, the more resources are needed. This fact stems from the Ashby's (1947) requisite variety theorem: To be a successful regulator, an agent (i.e., agent's brain) has to be able to produce at least as much variety as the system that is being regulator (i.e., the environment in which the agent operates).

The problem is then how to store sufficient knowledge that generates sufficient amounts of variety given a relatively limited storage size (e.g., a brain, computer memory) compared to the vast variety that the real-life living conditions that may occur and may need to be acted upon on planet earth. How can two-kilograms of brain tissue store sufficient information to address the complexities of real human life? To answer that question, one may begin by asking: How much total memory would be required to store all the necessary knowledge needed to execute human-like intelligent behavior?

It turns out that the answer to that question depends highly on the number of the levels of organization within the agent. If the agent has only one policy and must store all of its knowledge at that single level of organization, the increase in the demands on recourse (memory, synapses, etc.) is a linear function of the requisite variety (Nikolić 2015b). For producing twice as many different responses (e.g., for distinguishing twice as many different perceptual objects), twice as large a brain is needed.

However, if the bulk of knowledge is stored not in a policy that directly produces output towards outside but in another policy (one level lower) that acts within the agent and changes its properties as a function of feedback obtained from the environment, just a small increase in size of the lower policy can make the agent as a whole capable of increasing multifold the total variety of its behavior. In case of practopoietic hierarchy, variety does not increase by a linear addition of the memory, but multiplicatively (Nikolić 2015b): The total number of theoretically possible states generated by the agent having two policies is a product of the number of states in those two policies.

The operations of a *lower policy* serve a function similar to "loading" new knowledge into *a higher policy*. In other words, by help of a suitable lower policy, higher policy can flexibly re-learn new knowledge, as required. This is similar to deleting the contents of a limited memory and replacing it with new contents as needed (or replacing a processor or a computing module with the one needed at the moment).

An important feature of adaptive intelligent behavior is that there is no alternative storage where the previously deleted information from higher policy could be stored as an exact copy such that it can be easily reloaded later (unlike a computer, which can free RAM by saving an exact copy of information on a larger hard disk, and reload this information later). Instead, an adaptive intelligent agent reloads knowledge by re-learning it (or re-creating that knowledge—known as *anapoiesis*, Nikolić 2015a). This means that the agent only stores a few clues on how to extract the needed information from the environment i.e., it stores a learning rule for that particular content. For example, learning to recognize a chair means then learning the rules on how to "re-learn" quickly to use any particular chair—whenever a chair is being encountered in agent's environment.

Thus, in practopoietic systems much of the information storage is left to the environment and this information, with the help of sensory inputs, is "re-loaded" into a higher policy by the operations of a lower policy. For example, Ideatheca creates Working knowledge.

While creating a lower policy, an agent should optimize its organization to the maximal possible degree in order to use information available from outside as efficiently as possible. As lower policy is acting essentially as a set of learning rules, a consequence is that the knowledge stored in that lower policy has a much more general form than that in the higher policy (Nikolić 2015a).

In fact, calculations in Nikolić (2015b) show that if our own brains would not be tri-traversal but would have only two traverses—i.e., what is traditionally referred to as a single policy acquired through a simple learning rule—our brains could not possibly cope with the actual variety demands of real-life—even if everything in our brains is maximally optimized.

The total number of combinations in which words can be arranged to form meaningful sentences or objects to form meaningful spatial situations are very large. When combined with the total variety of the forms of sensory inputs in which these sentences or spatial situations can come, the resulting numbers of combinations exceed by far the total variety that a single-policy human brain could generate given its total number of neurons and synapses (for details see Nikolić 2015b). To answer to real-life demands and operate only as a T_2-agent, our brain would have to be many orders of magnitude bigger than what it really is.

This brain-size advantage of T_3-agents over T_2-agents is closely related to another one: The learning time, i.e., the time needed to acquire the necessary knowledge. As any lower level of organization stores knowledge in a more generalized form that the higher one, it also stores it in a more condense form. This means that this knowledge can be learned much quicker provided appropriate learning samples. In other words, a T_3-agent deals with details of a specific situation or stimulus only later when/if such stimulus or situation occurs. In contrast, a T_2-agent must pre-learn all of those details irrespective of whether it will ever need them later. This pre-learning requires also larger resources on the learning set and on time needed to acquire knowledge.

In conclusion, as there is not enough memory in the human brain to store human-level intelligence as a T_2-agent, there is also not enough time and not enough samples to learn that gigantic single-level policy that would be required in a case of T_2-organization.

The engineering challenge of T_3-agents

The biological theory in Nikolić (2015a) shows that T_3-organization is how our real mind works. However, this theory does not tell us how to create strong-AI using a T_3-organization.

The key engineering challenge is to get the needed cybernetic knowledge at Ideatheca and more notably, Machine genome. In fact, particularly difficult to acquire is the Machine genome, which, as we mentioned, corresponds to the learning rules needed for learning knowledge stored in Ideatheca. To appreciate the extent of the challenge of acquiring that knowledge, it is necessary to understand the relationship between the policies of AI and the tri-traversal theory of the organization of biological minds.

At the very bottom of biological adaptive organization of an organism lay genes. Knowledge stored in genes dictates our developmental and learning processes. For an artificial T_3-agent, this level of knowledge corresponds to its lowest level policy to which we refer as Machine genome. In both cases, this knowledge constitutes the most fundamental and most general learning rules possessed by the agent. The rest of the agent knowledge depends on that knowledge.

From research on DNA, we know that at this level, the policy has to be relatively rich in contents. The policy has to consist of a number of different rules for learning and development. This is the level at which the knowledge for all developmental states is stored. This is also where all our instincts are stored. The contents at that level make a difference between developing a human brain as compared to e.g., a chimpanzee brain or mouse brain.

Nevertheless, the requirements on resources at this level of organization will not be excessive. Likely, implementation of Machine genome will require less memory than human DNA (which takes about 700 Mb). Much of the metabolic processes that are required in biological systems and that need to be regulated through gene expression are not needed in silicon application. For that reason, much of the knowledge in biological genome will be unnecessary in Machine genome. Nevertheless, plenty remains to be in need in order to develop a fully functional intelligent system matching that of a human. We can expect a Machine genome that emulates much of human intelligence to take at least several 10s if not 100s of megabytes.

Application of Machine genome creates a policy at the next level, Ideatheca, which then contains the cybernetic knowledge acquired throughout lifetime i.e., during machinogeny. This long-term memory is the place where all our human long-term memories are stored, including explicit facts, semantic knowledge and skills, such as how to ride a bicycle. Ideatheca is much richer than genome and hence, demands much bigger information storage. According to Nikolić (2015b) human Ideatheca may hold on the order of 10^{14} to 10^{15} bytes of memory (100 s of terabytes). Possibly a machine will store knowledge in a more optimized form and hence, reduce those demands. Nevertheless, to produce intellectual and perceptual capacities that begin approaching those of humans, a minimum of several tens or hundreds of gigabytes of information may be required to store that policy.

Thus, instead of a small number of simple and general learning rules that are currently being used in AI, T_3- based AI will require large databases of quite elaborate learning rules. In fact, the knowledge of T_3-AI should not be thought of as being stored as information (that some external agent needs to decode and read out) but as learning rules (that operate independently without an external decoder).

Importantly, Ideatheca is not directly used to generate behavior as is the case for example, in synapses of artificial neural networks. The knowledge in Ideatheca is too general for direct production of behavior. This general knowledge needs to be applied once again to extract an even more concrete policy at a higher level of organization. Thus, one more traverse is needed. The policies of Ideatheca create Working knowledge. In psychology, the processes that lead to creation of Working knowledge are referred to as perceiving, directing attention, making decision, recalling information, or generally thinking. The execution of policies at Ideatheca for creation of Working knowledge is fast, usually taking less than a second in real time. For us, the contents of Working knowledge exchange with the rhythms with which our thoughts, percepts and ideas replace each other during the conscious flow of thought.

Finally, executing the policy at the level of Working knowledge generates the observable behavior of the agent. This policy defines the sensory-motor loops and determines directly how we act on the surrounding world. The amount of resources required for this policy may be smaller or equal to that of Ideatheca.

The challenges that lay behind acquiring the appropriate knowledge in Machine genome and Ideatheca are determined by the fact that our human capability of creating such AI is chiefly limited by the ability to directly engineer such systems. With the exception of very special cases, it is generally not possible for a human engineer to understand how a small change in a genome would affect behavior of an agent. The complexity of the underlying interactions among machine genes is simply too high for a human mind. For example, while a human engineer can develop quite an elaborate theory of a learning mechanism driving a T_1-neural network such as e.g., back-propagation algorithm, it is nearly impossible to do something similar for Ideatheca or a genome of a T_3-agent.

Generally, we refer here to the aspects of AI functioning that a human engineer can understand as *Clear knowledge*. And to the aspects of AI that a human engineer cannot understand we refer to as *Dark knowledge*.

As most of machine genome corresponds to Dark knowledge, it cannot be programmed by engineers in a way that a reinforcement learning mechanism can be programmed. For that reason, to create T_3-AI, we need yet another adaptive mechanism that is capable of learning Machine genome and that is simple enough to be understood by human engineers. In other words, much like we must have Clear knowledge at the bottom of organization hierarchy of traditional T_2-AI, we can create T_3-AI only if we also drive development of that AI with Clear knowledge at the very bottom. And, because a genome by its nature constitutes of Dark knowledge, we need one more simple policy, making the production of AI effectively a T_4-system. This final learning mechanism must be then simple enough for human engineers.

Another important aspect of AI development is the observation of the behavior of AI. The agent can be judged by a human observer as desirable or not, or appropriate or not. This judgment needs not be based on understanding of how the agent operates. Rather it is based more on intuition. The agent may run largely on Dark knowledge and yet a human observer can tell whether the behavior is appropriate. For example, one may use the following rule: "If the machine does something similar that I would do, this is probably alright". This evaluation of AI occurs by observing only the very top of the adaptive organization i.e., the behavior. Because humans can understand it only at the level of intuition here we refer to that aspect of AI as *Opaque knowledge*.

Thus, the challenge of creating T_3-AI involves engineering at the level of Clear knowledge performed at the bottom of a T_4-hierarchy, creation and operations of much of Dark knowledge at middle levels of the hierarchy, and then giving feedback to AI for its Opaque knowledge at the top of the hierarchy.

In T_3-agents there are three levels of organization with Dark knowledge. If no shortcuts were found to accelerate the process of learning at those levels, a mere brute force approach to development of strong-AI would require repeating in computer simulations something similar to the entire evolution of our biological species, which could then take millions if not billions of years (with computational power equivalent to that of natural evolution). This approach would certainly not be practical. The present invention offers important shortcuts for that process.

Knowledge transfer accelerates creation of strong-AI

The genes contain very general knowledge acquired over evolution of the species. Our instincts are a part of that general knowledge: sexual instincts, competitive instincts, fighting instincts, cooperation instinct, when to be afraid and when not, what to eat, when to fall in love, how to deal with different modalities of sensory inputs, how to learn language, the instincts that eventually make us create mathematics, etc. This knowledge has been acquired by our ancestors through evolution. Some of these ancestors are very far ones who were not even mammals. And some of the ancestors were not even vertebrates. This accumulated knowledge that we carry in our genes reflects a large statistics sampled over a very long period of time and over many individuals.

To create strong-AI, a similar general knowledge needs to be acquired at the level of Machine genome. The problem to be solved is then: How to acquire that knowledge in a short period of time for example, within a few years or decades, rather than requiring millions or billions of years?

Knowledge transfer theory, which is described in more details in Nikolić (in preparation), makes this possible. This theory consists of four conjectures (*Mutual adaptive pressure*, *Holographic knowledge, One-directional learning,* and *Representative sample*) and tells us that we can use the knowledge that has been already accumulated in human genome and transfer that knowledge to a machine genome. Similarly, the theory specifies the conditions under which it is possible to transfer that knowledge. As a result of that theory, it is not necessary to re-sample the nature in order to provide an AI with machine genome. One can use the already existing knowledge in human genome (and animals' genome) to approximate the biological knowledge found already in nature, which is a much faster process than acquiring this knowledge from scratch.

There are two main preconditions for transfer of knowledge from one agent to another. The first one is that the two agents share the same environment. The two need to interact in such a way that they become relevant environment for each other. If this condition is satisfied, the actions of one agent (*trainer*) form a *downward pressure for adjustment* (see Nikolić 2015a) for the other agent (*student*). As a result, the trainer forms an environment that gives much more direct and more effective feedback than what would be the case without the knowledge of the trainer. Many years of experience of the trainer can be condensed in a few instructions about which type of behavior is good and which type should be avoided.

But to assimilate that knowledge by the student and to learn it successfully, the other main precondition needs to be satisfied: The student's learning mechanisms should already share certain knowledge with the trainer's learning mechanisms that made the trainer learn that knowledge in the first place. Given that newly learned knowledge is always a specific case of a general knowledge stored in learning rules, the resulting inductive biases (Mitchell 1980) limit the transfer of knowledge to only those aspects that can be covered by the existing learning rules. This means that only agents who have similar learning rules—i.e., similar policies at lower levels of organization—can transfer knowledge among each other.

Knowledge transfer theory applies to exchange of knowledge among humans, and transfer of culture and education from older to newer generations. A schoolteacher forms the environment that makes downward pressure for adjustment on the Ideatheca of the student. Knowledge transfer theory applies also to selective breeding of animals, which enable changes to a genome: the human selector provides the environmental feedback on which genes get passed on and which not. That way, a docile dog was created from a wild wolf. Knowledge transfer theory applies also to the common procedures of T_2-AI, where human operator selects the training sets and correct solutions.

As knowledge gets transferred, two agents get to share knowledge and by that token become in Conant and Ashby terms (1970) good models of each other (Nikolić in preparation). Knowledge transfer theory applies also to the components of an agent. For example, while working together the two brain hemispheres become good models of each other. Similarly, neural circuits in the spinal cord become good models of circuits in the cortex, and vice versa.

During knowledge transfer across adaptive agents (or their sub-systems), the level of organization that will learn most extensively is the one that receives most pressure for adjustment. The pressure for adjustment travels one step downwards whenever a higher level of organization is not capable of resolving the pressure (Nikolić 2015a). Therefore, the degree to which the pressure will reach lower levels of organizations, and ultimately cause learning at those levels, depends on the challenges posed by the environment. Ideal for learning is an environment that poses adjustment pressure for lowest levels of organization such that the type of pressure fits the induction biases of the learning mechanisms at those levels. In other words, most stimulating are the environments that challenge agents in ways that are most easily handled by those agents.

A trainer can affect the environment directly only at its highest levels of organization, but the trainer is nevertheless able to transfer knowledge from its low levels of organization. The reason is that knowledge at higher levels is always a specific case of knowledge at lower levels. Therefore, if the sample is large enough and representative enough, the knowledge at the top will sufficient accurately reflect the knowledge at the bottom. That way human behavior can reveal accurately the instincts stored in our genes. If behavior of a trainer is sampled across a sufficiently broad set of situations and circumstances, the adjustment pressures exerted on a student will be general enough as to be handled efficiently only at the lowest levels of organization. As a result, a full cycle of transfer can be closed, starting from low levels of the trainer and ending at low organization levels of the student.

Important for shaping knowledge transfer are various forms of locking changes to knowledge. If certain forms of changes (learning) are actively prevented, the pressure for adjustment is directed to other structures that are not locked. For example, a trainer may need to invest an effort not to learn from students because otherwise, by interacting through the same environment, students may transfer to the trainer their own knowledge (perhaps less desirable knowledge) (locking is illustrated in Figure 4). Also, if within an agent some components are prevented from learning, other components will be under higher learning pressure.

Finally, it is important that knowledge transfer based on interactions through shared environment does not require the same form of implementation of knowledge storage. Knowledge is transferred in a form of rules of what-to-do-when, but these roles can be implemented in any form of hardware. Thus, knowledge transferred from humans to strong-AI will not require implementation based on e.g., networks of spiking neurons. Knowledge can be equally well transferred to other hardware and software implementations (e.g., probability matrices).

SUMMARY OF THE INVENTION

The present invention constitutes of a method for producing a T_3-agent capable of intelligent behavior and approaching the intelligence of human. A T_3-agent is gradually equipped with an appropriate cybernetic knowledge stored in machine-genome. In the same time, also according knowledge is being developed at the level of Ideatheca.

The present approach is very different from those that propose direct copying of the biological hardware and simulating it on a computer, such as the Human Brain Project. Instead of copying physiology and anatomy without understanding it, the present method proposes a certain degree of understanding of organization of intelligence based on the theory of practopoiesis and the tri-traversal theory of the mind. As a result, the present approach allows implementing computation and memory storage for AI in a form most suitable for the technical hardware.

The present method implements a gradual process of development of AI capabilities that largely resembles both phylogenetical development of species and ontogentical development of an individual. Therefore, T_3-agents gradually developed to exhibit higher and higher levels of intelligence begining from simplest capabilities, such as taxis, to ever more elaborate ones, asymptotically approaching the high human-level capabilities such as understanding of language and solving engineering or scientific problems.

The present apparatus is named “AI-kindergarten” as fundamental to it is an iterative process between interaction of AI with humans and integration of knowledge acquired through that interaction.

AI-Kindergarten consists of three accelerators, each accelerating the development of AI for a factor of roughly 1000. Combined, they can produce an acceleration of 1000^3, in comparison to the speed with which intelligence was developed naturally by through Darwinian evolution. The three accelerators are: Direct engineering, Playroom and Incubator.

Direct engineering is applied whenever possible to create Clear knowledge. However, AI-kindergarten has two main additional components that create the much required Dark knowledge, *Playroom* and *Incubator*. In Playroom, cybernetic knowledge is transferred from human trainers to the AI. In Incubator, this knowledge is being integrated within AI.

Both Playroom and Incubator rely on Knowledge Transfer Theory (Nikolić in preparation) to transfer knowledge from low organizational levels of one agent (e.g., a human trainer or a machine) to another agent (a machine). The advancement of AI unfolds through a process of knowledge transfer by human-machine interaction in Playroom followed by Incubator-based integration of that knowledge across the sub-components of AI. This process is iterative. One-step more advanced AI interacts again with humans and acquires new knowledge, which is then integrated in Incubator, and so on. Through repetition of this cycle (and occasional addition of Clear knowledge through directly engineered components) AI gradually increases its level of intelligence and can approach that of human.

Direct engineering

The first 1000x accelerator relies on directly engineering those parts of AI that are understandable to human engineering and hence, do not require other more time consuming approaches. Knowledge that is directly engineered can be based on pure engineering solutions but also on scientific studies of neurophysiology and behavior.

Direct engineering also involves development of computer hardware and software, robotic hardware, and any other engineered solution that improves the functionality and performance of AI. As only a small fraction of the total AI knowledge can be implemented directly, two more innovative components are added to AI-Kindergarten.

Playroom

The Playroom is the second 1000x accelerator where cybernetic knowledge is transferred from human trainers to machines. Playroom relies on Knowledge transfer theory to harvest that knowledge and to have trainers interact with machines and by doing so, express behaviorally the knowledge stored in their Ideathecas. That knowledge is expressed in a form of behavioral actions. For example, a trainer may give feedback “good” vs. “bad” in response to behavior of an AI implemented e.g., as a robot. Here, a human trainer may give feedback on which type of stimuli should be approached and which type should be avoided; or which kind of behavior is socially acceptable and which not; or which behavior is potentially dangerous (for humans) and which not.

Knowledge transfer occurs through shared environment and is based on preconditions described in Knowledge transfer theory. This transfer is not limited to button presses indicating “good” vs. “bad” but can involve also more advanced forms of training such as observation of actions performed by the trainer, followed then by imitation on the side of AI. As the intelligence of the T_3-agent grows, so does the variety of interactions that AI can perform with the trainer.

In each training session one unit of knowledge is from human trainers to AI students. The training loop by which a training unit is completed requires a procedure constituting of the following steps:

(i) Decision is made on the objectives of the training unit and the trainer is instructed and prepared accordingly.

(ii) Each training session begins with an existing level of knowledge on the side of AI. This knowledge may be directly engineered seed knowledge (Clear knowledge) and will normally involve a degree of previously learned knowledge (Dark knowledge)

(iii) New knowledge is guessed at the level of Working Knowledge or Ideatheca.

(iv) New knowledge is manifested at the level of AI behavior (Opaque knowledge).

(v) Feedback from the trainer is obtained.

(vi) If feedback is positive, the guessed knowledge is retained. Otherwise it is disposed of.

(vii) The state of retained knowledge is saved for later use in Incubator.

(viii) If the objective of the unit of learning is not achieved, go back to step (iii).

This procedure can be used to transfer knowledge from human to AI at any level of organization of the AI-agent. The most commonly trained by this procedure will be Working knowledge and Ideatheca. A transfer of human knowledge to Machine genome in Playroom would be too demanding on the time of the human trainer and the number of feedback responses that the trainer would need to give. Thus, the transition to Machine genome from Ideatheca and Working knowledge will be performed in the Incubator, where no interaction with humans takes place.

The function of Playroom is not to create final version of AI but to provide the training sample for AI, the AI-agent being fully created and trained only in Incubator. So, the function of Playroom is similar to the choice of the image database of images for traditional T_2-AI, such as e.g., a deep learning network. The training sample is organized as knowledge structured as at least one organizational level less adaptive than the AI that is being trained by that knowledge. So, in the classical case, it is a T_1-input-output mapping device (e.g., a deep network) created by generalized rules to represent a T_0-training set of e.g., images. In AI-Kindergarten, T_2 or T_1-agents created in Playroom can be considered only as samples of cybernetic knowledge. Then a T_3-agent is created in the Incubator by generalize rules to "represent" those T_2-samples (or T_1-samples) collected in Playroom.

An example of the training procedure in Playroom may be a robot that we want to improve in respect to knowledge on avoidance of large objects. Let us suppose that large objects should be avoided, because they are potentially harmful, and small objects should be safe to approach. We want the robot to eventually develop a reflex of avoiding large objects but not small ones. In that case the eight steps of the Playroom procedure could unfold as follows:

(i) We define as the objective of the learning unit a phylogenetically early aspect of biological intelligence: a fundamental knowledge necessary to take care of agents' safety reflected in the following rule: "Avoid large objects but feel free to approach small and medium-sized objects."

(ii) We start with a pre-existing knowledge, which may be a neural network that can map sensory inputs from camera to actuators. However, let us assume that this network does not have a single set of particular mappings built in. Rather, the network can produce many different mappings and this depends on which neurons in the network are operational and which are not (adapted neurons), each subset of neurons set as operational leading to a different mapping. If all neurons are allowed to operate, much of the pathways cancel out each other and perhaps there is not much interesting behavior created under such conditions. Hence, the network draws its flexibility in behavior on choosing a subset of neurons that are allowed to operate. This choice of this subset corresponds to the Working knowledge of the network. Ideatheca then consists of a set of rules that determine which of the neurons will be activated and which not.

As a part of Clear knowledge the agent is also equipped with the machinery for guess patterns of operational neurons in Working knowledge. Alternatively, the agent guesses rules for allowing neurons to be operational or not i.e., creates random knowledge at Ideatheca).

In addition, we may want to develop this new reflex on top of an existing function that the robot already performs well. For example, the robot may be previously trained to push red balls to the left and green ones to the right. Therefore, the robot may perform some other simple tasks while being trained for the sake of its own safety to avoid big things (especially if they are rapidly approaching the robot) and not to be concerned with small ones.

(iii) A guess of knowledge will be usually a combination of certain existing knowledge, including new Clear knowledge added to the agent, and a randomly generated change to a given policy of the agent. The Clear knowledge for guessing added to the agent in our example may be a routine that calculates the retinal projection of the approaching image, and a routine that calculates the speed of expansion/contraction of that retinal image. The random changes may be made by randomly choosing a neuron that will become operational (or not operational) within the network. At each iteration of the learning loop a small change is made, and hence, only a small number of neurons may be made to change their operational status.

(iv) The newly guessed knowledge in (iii) is by its nature Dark knowledge. Hence, the only way to know the effects of the change on the behavior is for the robot to actually generate actual behavior in response to an approaching object and during a given task.

(v) The trainer observes continuously the robot's behavior and various other parameters of the agent. The trainer then gives feedback on whether the newly guessed knowledge offers an improvement towards the goal or not. In the present example, the setup is such that the robot will in most cases generate behavior that cannot be considered an improvement. So, to make the procedure easier on the trainer, the trainer may be required to respond explicitly only for "good" i.e., only upon a detection of an improvement. The response "bad" would be thus default. In other setups, the arrangement could be the opposite, "good" being a default answer.

(vi) Important for the learning process is a decision of whether the next step of learning will include the last change or whether the agent will be returned back to the previous state. If an improvement was detected, then the change should be kept. If worsening of performance was detected, then the change should be reverted. And if no change in performance was detected, then either of the two choices may be taken.

(vii) It is necessary to keep record of all satisfactory performing policies of the agent. This record is needed for two purposes. One is the use in Playroom, as it is possible that the trainer may decide that it is necessary to revert back to certain previous knowledge. The second purpose is the use of that knowledge in Incubator, as this knowledge will generate the trainer for developing a more advanced form of AI in Incubator.

(viii) When the trainer judges that the robot is sufficiently cautious about avoiding big objects and in the same time is not distracted in performing the red-green ball task in the presence of small objects, the trainer should decide that the objective is achieved and that the session should be closed. The unit of training is completed.

A similar example can be made for an industrial robot needed to learn behavior for a given step of industrial process. In that case, not feedback “good” vs. “bad” is given, but a number of movement trajectories for a number of situations is given. That way a specialized Ideatheca may be created for one industrial function, producing eventually a robot that is much more intelligent than any robot employed in industrial process today.

Playroom is used to collect many thousands of such bits of cybernetic knowledge. The process of giving feedback should be always automated as much as possible, but there will always be nevertheless an irreplaceable need for human guidance, which is here instantiated in form of a trainer.

The order in which these bits of knowledge are acquired reflects roughly phylogeny and ontogeny of biological intelligence. Thus, the process of acquiring human-level intelligence is gradual, starting from very simple cybernetic knowledge which is the one that is invented by evolution phylogenetically early (and also develops ontogenetically early) and only after this is achieved, continuing then with phylogenetically later knowledge. In other words, much like phylogeny is approximated by ontogeny, both of those are approximated by machinogeny.

An example of a unit of training at a later stage of AI development would be training the AI to acquire a reflex of expressing itself through generation of a sound. This would be an early stage in developing the capability of using language. In that case the trainer would give a positive response whenever the AI expresses itself in a way that is understandable and comfortable for a human user. Early on in such trainings, for example, a general goal of a training unit may be to generate two distinctive sounds; one when the user needs to be informed that a certain job is successfully completed by AI and another when AI needs an approval for action from the human user.

At an even later stage of development towards human-level AI, the unit of training may be devoted to social aspects of behavior, learning which behavior is appropriate in which situation. For example, different type and vigor of actions should be approved by the trainer if the people around AI appear sad, tired or distressed, then if the people appear in good spirit, happy and full of energy.

As the stages progress more and more towards human-like intelligence, the activities performed in AI-Kindergarten will become gradually more and more similar to activities performed in real, human kindergarten. To transfer more of human knowledge to AI, the robots will have to be trained through play in ways that resemble much the plays of human children and the interactions that adults have with them. These activities may include building things, playing roles and conversing in human language.

Incubator

Incubator is the third 1000x accelerator where the bits of knowledge acquired in AI-Kindergarten get integrated into a more complete AI and where the Working knowledge and the knowledge in Ideatheca of the trainer is being transposed to more abstract rules at the level of Machine genome. Incubator creates new, more general form of knowledge.

Incubator can be understood as generalizing the AI-training process in classical approaches. To create a new T_1-agent (a deep network with new knowledge) from a sample of T_0-knowledge (e.g., still images), a learning mechanism needs to be added to the T1-agent, making it in total a T_2-system (e.g., deep network + learning mechanism). In general, if we want to use T_n-knowledge to create a more general T_{n+1}-agent, for learning of that agent, we require a total of a T_{n+2}-system. Therefore, to create a T_3-agent, we require a mechanism in Incubator that can make changes to the Machine genome of that agent, which makes it in total a T4-system.

Thus, in Incubator in total four traverses operate. This is consistent with biological aspects of practopoiesis, as biological agents are also T3 and creation of their genome requires another traverse in form of evolution by natural selection (Nikolić 2015a). Hence, species is a T_4-agent. To achieve biological level of intelligence in a machine, we need to match the biological level of adaptability.

For these functions the Incubator also relies on Knowledge transfer theory. During knowledge transfer in Incubator, different bits and pieces of knowledge i.e., different functions of a given policy, get to be integrated into a balanced combined agent. As the total amount of knowledge grows, so grows the need to balance that knowledge. Different aspects of the agent's knowledge have to fit together as they need to operate together in a synergy. This puts the requirement to put the agent as a whole into a balance.

The balance is achieved when different parts of the agent satisfy Conant and Ashby's (1970) requirement to become good models of each other. For each function in a policy (or for each neuron in a network) it is not the environment of the agent that constitutes its environment but also other functions in the policy (other neurons in the network) constitute the environment too. An agent functions well as a whole, only when its elements are set such that they are good models of each other. Only in that case can they work together in synergy.

The processes of balancing takes place during any activity of the agent, because during any interaction with the environment necessarily also different components of the agent interact among themselves. This puts them under the so-called *equi-level interactions* (Nikolić 2015a) and gives them the chance to learn about each other as they learn about the surrounding world.

Incubator can be understood as a large computer simulation that simulates not only agents who mutually interact but also their shared environment within which they interact. Again, the trainer and the student share the environment and interact: The trainer gives feedback and the student receives downward pressure for adjustment and makes changes until the performance is satisfactory. When advantageous, it is however also possible to use a real-world environment as a part of Incubator.

While in Playroom the human trainer has more knowledge than the student, and the student gets only bits and pieces of it, in Incubator the student has normally more knowledge than the trainer. Trainers arrive with bits and pieces extracted in Playroom and create students that accumulate all that knowledge. Therefore, the students trained in Playroom can be understood also as a sort of *messengers* of cybernetic knowledge. However, unlike the traditional messages that are T_0 (e.g., a message written on a piece of paper containing Clear knowledge), these messengers operate at higher organization levels that transfer cybernetic knowledge (e.g., T_2-messengers containing Dark knowledge).

Incubator has one or more of the following eight properties:

(a) There is no human feedback involved. The training is completely machine-based.

(b) Downward pressure for adjustment is made at the level of machine-genome.

(c) The knowledge of the trainer is frozen so that the downward pressure for adjustment is unidirectional: from trainer to student.

(d) Each feedback iteration in Incubator involves guessing of a new Machine genes followed by expression of those and other Machine genes (i.e., entire Machine genome), which is followed by actions of Ideatheca, followed by application of Working knowledge. Only after application of Working knowledge and the subsequent generation of behavior, can feedback be provided for the guessed gene.

(e) To accelerate the learning process, the old knowledge of the student is also frozen or subjected to minimal changes. That way, new knowledge and skills learned by the student adjust to old knowledge and skills, but not the other way around.

(f) To accelerate the learning process, the requirements for the total number of guesses and tests are minimized by scheduling the learning into an appropriate Incubation tree: e.g., if A, B, C, and D, are the units acquired in Playroom, we first integrate in Incubator A and B into E, and C and D into F and only then E and F into the final agent.

(g) To accelerate the learning process and if processing power allows, simulation may be executed with speed higher than the real-life time.

(h) To accelerate the learning process and if computational resources allow, the search for appropriate Machine genes may be executed in different simulations simultaneously (i.e., by the means of distributed computing).

In the example used above, we may want to use the knowledge of the robot trained in Playroom (to avoid big objects but not small ones) to create a more capable robot. We may transfer that knowledge from Ideatheca into the Machine genome of another robot whose pre-existing Machine genome already provides it with an "instinct" to perform some other task. For example, this robot may have a tendency to collect red balls and store them away, but ignore green balls. We may want to then add another "instinct" to Machine genome—that of avoiding big objects, especially if they approach the robot—and make the instincts work well together in a balanced synergistic way.

(a) A computer simulation is made of a world with robotic actuators, red and green ball and objects of various sizes approaching the robot. Additional parts of this simulation are the trainer and the student robots. The trainer robot gives “good” or “bad” feedback to student robot. Otherwise, the procedure for learning a unit of knowledge is similar to that in Playroom.

(b) The guessing process is at the level of Machine genome. This means that downward pressure for adjustment for every “bad” feedback cannot be resolved at Ideatheca. As a consequence the pressure moves down to Machine genome where it needs to be resolved.

(c) To ensure that the transfer of knowledge is unidirectional and that the trainer does not learn from the student, but only the other way around—the student learns from the trainer—, Ideatheca of the trainer (in some cases also Working knowledge) is frozen. This means that for trainer the operations of Machine genome (or in same cases of Ideatheca) are switched off.

(d) As genes are being guessed in the simulation, these genes need to be put into operation in order to test their suitability. This means that the simulated agent has to undergo interactions with its simulated environment such that the new genes have the chance to make developmental changes to the agent. In our example, the test will involve satisfactory performance in respect to avoidance of large approaching objects.

(e) Also, it has to be ensured that the agent does not lose the preceding capabilities and this is the instinct to selectively pick the red balls. To protect the old knowledge in genes, the old genes can be frozen. That is, the gene guessing mechanism creates new Machine genes, but does not alter the old ones. That way, according to Knowledge transfer theory, the pressure for adjustment is put solely on the new genes. However, important is that the old genes remain in operation. Only if these remain in operation can the new genes become good models of the old genes—and hence, the agent as a whole can be balanced.

In any such situation when a new “instinct” is added on top of an old one, an important question is whether there is at least a theoretical possibility to find a solution i.e., to find a matching new set of genes that can satisfy the function without impeding the old function. Moreover, a question is whether this solution can be found in a reasonable amount of time.

As these questions pertain to Dark knowledge, there is little that human engineers can prove or disprove directly. At best, human engineers are left to guessing.

For that reason, AI-Kindergarten relies heavily on guidance from biological evolution of capabilities. If data from phylogeny and ontogeny tell us that for both, a new step in evolution of species and a new stage of development of an individual, a solution exists to bring the species/individual to the new higher-level of adaptive behavior (i.e., adaptive intelligence), then likely we will find a solution also in Incubator. In other words, if we can establish that natural evolution has found a given solution to a certain problem under given circumstances, this is an indicator for the engineers of AI-Kindergarten that a solution should be possible to find also for AI-agents. This is the main reason that the development of AI—i.e., machinogeny—has to be informed by and follow biological phylogeny and ontogeny.

Thus, in our example, we may ask the question of whether the picking of red balls would be something that is evolutionary earlier than avoiding large approaching objects, or vice versa. In practice, this may turn into the following question: If we begin incubation with two agents of similar complexity and intelligence but one having the instinct of picking red balls and the other having the instinct of avoiding large objects, which of the two should be the student and which should the trainer? In other words, should AI evolve ball picking on top of avoidance, or the other way around? For answers to such questions we have to look into comparative biology and developmental psychology, as good choices can considerably accelerate the development of AI.

(f) Knowledge in Machine genome is necessarily more general than that in Ideatheca (Nikolić 2015a). This means that many pieces of knowledge potentially stored in Ideatheca are condensed to a fewer number of "instincts" in Machine genome. To make the incubation process of condensing cybernetic knowledge successful and efficient, we have to rely on Opaque knowledge to choose the units acquired in Playroom that seem most easy to generalize.

For example, we may have acquired multiple units of knowledge in multiple Playroom sessions. These may all be related to avoidance of large objects but acquired from different trainers and from different types of objects (furniture, boxes, large living beings, etc.). If our goal is to find a set of genes, that generalize among those different bits and pieces of knowledge, we may perform an incubation sub-step that involves only objects avoidance before we integrate that knowledge with ball picking. So, we may first create an agent that produces an integrated Ideatheca reflecting the object avoidance knowledge acquired from all trainers. Also, we may then create a Machine genome that produces an "instinct" for such generalized behavior. Only after those steps are completed, we use the agent with integrated Ideatheca as the trainer for creation of a Machine genome that integrates that behavior with selective ball picking. In addition, we may use the Machine genes obtained for object avoidance as guidance for guessing Machine genes of the combined agent, and by doing so reducing significantly the search space.

In a process of creating advanced forms of AI with many forms of "instincts", these sub-steps in incubation of knowledge can be organized as an Incubation tree.

(g) When we simulate the environment and the behavior of agents in Incubator, we limit the computations to the essentials. In our example of objects approaching the agent, we may use Graphic processing units to compute only the positions and sizes of the retinal projections of approaching objects (or objects being approached), without having to simulate texture details of the objects. As a result, we may be able to run the simulation 100x faster than real-life. As a consequence, we may perform the search through Machine gene space also 100x faster than what would be required in real time.

(h) For example, we may have access to 100 simulators that we can use simultaneously as Incubators. We may thus, be able to perform simultaneously 100 incubations that are identical in all respects except for the part of the genome space being tested. Whenever we make a significant progress in one of the 100 Incubators, we may update all the remaining 99 Incubators with that new discovered gene. We can then continue with further search from that point in all 100 simulators. This can provide near-linear increase in the development speed as a function of the available computational power.

AI-Kindergarten in comparison to biological processes of knowledge acquisition

There are a number of processes in the biological brain that have a similar function of organizing knowledge to that of Incubator. In the brain, explicit knowledge is acquired with high pace due to the operations of hippocampus. Later, this knowledge needs to move from hippocampus into cortex. This process involves replaying sequences of learned events, which occur during sleep. As a result, the memories move from a concentrated place to get integrated with the rest of the brain.

The biological process assisted by hippocampus by which knowledge is acquired quickly is similar to what happens in Playroom. The process by which this knowledge moves into the cortex is similar to the function of Incubator. During both processes, those in the brain and those in AI-Kindergarten, the goal is that the agent is eventually balanced i.e., that the components of the agent become good models of each other.

An external process of gradual acquisition of balanced knowledge is the process of acquiring skills, such as riding a bicycle, skiing, or playing tennis. What hippocampus does for explicit knowledge, repetitive training does for implicit knowledge of a skill. During each practice, Ideatheca of the brain is put under downward pressure for adjustment and changes a bit. The reason it takes long time to develop a skill is that the new skill needs to be integrated with all of the other skills in the brain. This balancing process takes time.

Hippocampus may serve as brain's "hack" for the problem that some knowledge has to be acquired with high speed and there are no possibilities for slow repetitive training of a skill. Hippocampus may thus serve as a temporary storage of "raw", non-balanced knowledge of that kind (spatial knowledge, explicit memories), which is then balanced only later, during less intensive interactions with the environment i.e., during rest. This may involve sleep and dreaming. In a way Playroom is the hippocampus of AI-Kindergarten. Similarly, the simulation performed in Incubator is the dream of AI-Kindergarten.

However, there are also significant differences between the brain and AI-Kindergarten. The brain does not change its genes. In contrast, AI-Kindergarten acquires knowledge at a level of Machine genome. This aspect of AI-Kindergarten is related to another activity of humans and this is selective breading of animals. By selective breading we have domesticated many animals creating in the genome desirable cybernetic knowledge (e.g., we turned a wild wolf into a domicile dog). Therefore, AI-Kindergarten offers much more than the hippocampus-cortex relation. In a way, the combination of Playroom and Incubator combines the two approaches of knowledge acquisition principles, the one with quick acquisition and later slower integration and the one of selective breading. The reason that they can be combined is that the knowledge acquisition rules underling both of them are similar and are all covered by the Knowledge transfer theory.

BRIEF DESCRIPTIONS OF THE DRAWINGS

Figure 1: Four levels of adaptive organization (four traverses, or T_4) of AI-Kindergarten necessary to create biological-like intelligence and gradually approach human-level intelligence.

Figure 2: The end product of AI-Kindergarten is an AI with three levels of organization (three traverses).

Figure 3: Classical AI compared to tri-traversal AI created in AI-Kindergarten from the perspective of three types of knowledge.

Figure 4: A general principle of transferring knowledge from one agent (e.g., human trainer) to another agent (e.g., AI) across different levels of organization.

Figure 5: The main iterative components of AI-Kindergarten.

Figure 6: Integration of knowledge in Incubator.

Figure 7: Flow diagram of a training unit in Playroom.

Figure 8: Flow diagram of an integration unit in Incubator.

Figure 9: Use of Incubation tree.

Figure 10: The parallel gradual growth in the complexity of Ideatheca and machine-genome reflecting phylo- and ontogenesis.

Figure 11: AI-safety through limitations in super-human capabilities that AI-Kindergarten can produce.

DETAILED DESCRIPTION OF THE INVENTION

Figure 1. The principle of organization of artificially intelligent adaptive agents and systems for creating such agents. In such systems not only one policy exists, but multiple ones are organized into a hierarchy. A policy lower on the hierarchy (e.g., 103) provides the learning rules for the one higher on the hierarchy (e.g., 102). Each policy receives inputs from the environment (external world; 101). The learning rules such as those underlying reinforcement learning can be considered as a special case of a (simple) policy placed one step lower on the adaptive hierarchy relative to the policy that it helps learn. According to the tri-traversal theory (T_3-theory) of biological organization three such levels of policies will be needed in order to build machines that match human capabilities in adaptive behavior (102, 103, and 104). The lowest policy on that hierarchy (104) corresponds to the function of genes in biology. To create such a T_3-agent, at least one more policy is needed even lower on the hierarchy (105), whose function is to create the knowledge for 104. The function of this policy corresponds to evolution by natural selection in biological agents. The advantage of such hierarchically organized policies over a single policy is the total number of variety of behavior that a system of a given size can produce (Nikolić 2015b). To produce sufficiently flexible behavior a single policy cannot suffice. A related advantage of hierarchical agents is the learning time. In policy-hierarchical systems, it takes not only fewer resources to implement machinery capable of intelligent behavior but also the learning time needed to acquire correct policies is much shorter.

Figure 2. The end product of AI-Kindergarten is an AI with three levels of organization (three traverses), shown as a knowledge graph (Nikolić 2015a). The organization of knowledge of an artificial agent that is biological-like intelligent requires possessing policies at three different levels and thus, executing operations at three different levels of organization. Through the operations of any of the policies more specific cybernetic knowledge is created at the higher level of organization. This process of knowledge creation is referred to as a *traverse* of knowledge (205). At the very top of the organization is the external behavior of the agent (201). This behavior is determined by a policy reflecting the current Working knowledge, abbreviated as WK (202). This policy may operate with high speed, closing the sensor-actuator cycle in as short as 10 milliseconds. Working knowledge is updated slower, but it could be nevertheless often updated several times per second. The process of updating working knowledge is the second traverse of the agent and is referred to as anapoiesis. This knowledge determines the current set of policies according to which the agent actually responds behaviorally to the inputs form the environment. The policy that contains the general knowledge of the agent about the surrounding world, which includes concepts and long-term memories lays organizationally just bellow Working knowledge (203), and is referred to as Ideatheca. Ideatheca provides the agent with knowledge on how to use the current sensory inputs in order to set the Working knowledge for sensor-actuator operations. Finally, the contents of Ideatheca are acquired throughout the lifetime of the agent through its interactions with the environment and by application of a more general form of knowledge stored in Machine genome (204). For a T_3-agent, Machine genome is fixed. It does not get updated. The relative amount of knowledge stored in each policy can be indicated by the sizes of the circles in a knowledge graph. A T_3-agent will typically have much larger Ideatheca and Working knowledge than will Machine genome.

Figure 3. Classical AI is compared to tri-traversal AI created in AI-Kindergarten. The two types of AI are compared from the perspective of three categories of engineering knowledge. Left: the adaptive organization of the learning process of the classical single-policy AI (for example, deep learning networks). Right: Adaptive organization required for creating biological-like intelligence. Traditional AI has one traverse (from 302 to 301) and its creation requires in total two traverses (from 303 to 302 and from 302 to 301)(hence, T_2-AI). In contrast, biological-like AI has in total three traverses (307 to 306 to 305 to 304) and its creating requires in total four traverses (308 to 307). Thus, to build tri-traversal AI, a four-traversal production system is necessary.

In both cases, human engineers create the lowest traverse, which is the one acquiring the most fundamental cybernetic knowledge driving the agent. This category of cybernetic knowledge must be fully understandable to human engineers because otherwise, AI could not be created. This knowledge is referred to as *Clear knowledge* (indicated by white circles in the knowledge graph)(303 and 308).

In contrast, the knowledge acquired by applying Clear knowledge that interacts with the environment (e.g., through a sample of images needed to train a deep network), becomes quickly too big and too complex for engineers to understand. For most part, a human cannot understand the computational operations of the agent at that level. Classical AI has only one level of such humanly non-understandable knowledge, while T_3-AI has three such levels. This category of humanly non-understandable knowledge is referred to as *Dark knowledge* (indicated by black color in knowledge graph)(302, 305, 306, 307).

When observing behavior of an AI a human can again understand it despite the fact that AI runs largely on Dark knowledge. However, this understanding is at a different level than the Clear knowledge. The observer does not understand the process of generating behavior in all the detail. Nevertheless, the observer can judge intuitively whether the behavior is appropriate for a given situation or not. By observing the AI at the very top of its organization i.e., at the level of the behavior of the agent, a human person can make a judgment on whether the AI performs well or not. This judgment is not based on engineering type of understanding of how the machine works, but simply on the basis of a comparison of machine performance with human performance. We refer to this category of intuition-based knowledge as *Opaque knowledge* (indicated by gray circles in the graphs)(301 and 304).

The two graphs illustrates that human understanding and intuition respectively deal with the very bottom and the very top of the agent organization. The middle parts of organization are in all cases too complex for a human mind to understand. The implication is that, to develop T_3-AI, it is not possible by human engineers to directly create Machine genome or Ideatheca. Their minds simply cannot understand how a change at the bottom level of organization (Machine genome) would affect the top level (behavior).

In the case of T_3-AI, the very bottom Clear knowledge level is closely related to Darwin's theory of evolution by natural selection in biology, and this theory is understandable to human mind. This is the reason that, to build biological like T_3-intelligent agents, it is necessary to employ a T_4-system for manufacturing such AI. Only the bottom of T_4, but not that of T_3, is understandable and hence, can be directly engineered.

Figure 4. Illustration of one of the four conjectures of the Knowledge transfer theory, as applied to transferring knowledge from one agent to another. The interaction is created between two agents, one consisting of components 402, 403 and 404 and the other of components 405, 406 and 407. They share the same environment (401). That way, the two agents become environment to each other. As a result, the two agents are under adaptive pressure to become good models of each other. Their policies tend to change them such that they adapt to each other and hence, end up sharing knowledge.

To ensure one-directional transfer of knowledge, the knowledge of one of the two agents can be kept frozen (indicated by locks in 403 and 404). In that case, the agent does not adapt any longer at the given levels of organization and as a consequence, the pressure for adjustment can be resolved only by the other agent. This has for the result that the knowledge is transferred in one direction only. The not-frozen agent will be under the adaptive pressure to acquire knowledge from its peer. We refer to the two as the *trainer* (the one with the frozen knowledge on the left) and the *student* (the one that learns on the right).

In the present example the knowledge is being transferred from Ideatheca of the trainer (403) to the Ideatheca of the student (406), which requires functional Machine genome (407). However, this principle can be used for transferring knowledge across any pair of levels of organization.

The knowledge transfer theory tell us that the knowledge can be transferred from trainer to student only if the according policy (e.g., genes) by which knowledge has been learned by the trainer are similar to the policy of the agent. Therefore, for the transfer to work, the genes of the trainer (404) and the student (407) must contain similar cybernetic knowledge i.e., those at 407 need to act in similar ways as those at 404 acted in the past in similar situations. This important property of hierarchically adaptive systems (i.e., practopoietic systems) can be used to transfer knowledge deeply – from genes of human trainers to machine-genes of AI.

In machines, knowledge freezing can be achieved simply by switching off certain functions of the agent. In biological trainers, knowledge freezing can be achieved through instructions. If the trainer has enough knowledge in her/his Ideatheca (403) to understand the behavior of the agent, the downward pressure for adjustment can be resolved already at the level of Working knowledge (402) thus, preventing sending of adaptive pressure downwards towards Ideatheca.

Figure 5. AI-Kindergarten consists of three main components, or three 1000x accelerators. The Playroom and the Incubator are two novel contributions to assist Direct engineering and make possible development of strong-AI. Direct engineering is

used whenever possible to contribute to the advancement of AI. However, Direct engineering is strictly limited to Clear knowledge. To create the needed Dark knowledge, first extraction of knowledge units from human trainers is performed in the Playroom. Then, the units of knowledge are integrated into ever-smarter machines in the Incubator.

Playroom is the place at which human trainers give feedback to behavior of robots and that way knowledge is transferred from humans to machines. Incubator is the place where this newly learned knowledge is integrated with the previous knowledge and is transferred down the adaptive hierarchy towards the machine-genes. The operations of Playroom and Incubator are iterative. Playroom and incubator are in theory sufficient to develop strong-AI even without subsequent direct engineering. Nevertheless, continual work on Direct engineering additionally accelerates that process.

Figure 6. Integration of knowledge in Incubator. Knowledge transfer theory is used to integrate unit of knowledge (trainer: 602, 603 and 604) extracted previously in Playroom, with the previous knowledge of an agent (student: 605, 606 and 607). Typically, knowledge is transferred from Ideatheca of the trainer (603) to the Machine genome of the student (607). Trainer and student share the environment (601), which is typically simulated on a computer.

The pre-existing knowledge in student's Machine genome is frozen (indicated by a lock in 607), making adaptive pressure on the new knowledge to become a good model of the existing knowledge. New components of Machine genome are being created by gene guessing mechanism (608), which operates as the lowest-level policy (lowest-level traverse according to Nikolić 2015a), and is evaluated through feedback (609) received from the trainer and that passes through shared environment (601).

Figure 7: Flow diagram of a training unit in Playroom. A training session in Playroom begins with identification of the tasks in form of *instructions* given to the trainer. Each training unit must begin from a certain minimal level of pre-existing knowledge. In the earliest stages of machinogeny, this knowledge may be entirely Clear i.e., entirely engineered by human. However, a later stages the training will always begin with a certain amount of Dark knowledge. Next, new knowledge in Ideatheca of the student is *guessed* and then it is used to generate *Working knowledge*, which in turn generates new *behavior*. The trainer observes the behavior and makes a judgment "good" or "bad" (or gives another type of feedback). If the judgment is "bad", the latest changes are disposed of, and Ideatheca is reverted to the previous state. Then, new knowledge in Ideatheca is guessed and the iteration cycle repeats.

If the generated behavior is judged as "good", the new Ideatheca is saved and the trainer judges whether the objectives of the training session have been *achieved*. In case that the objectives have not been achieved, a change is again made in Ideatheca (additional knowledge is guessed) but without reverting to the previous state. In other words, new change is made on top of the previous change. The iteration cycle then continues.

If objective is achieved, the new cybernetic knowledge stored in the resulting agent is *delivered* to Incubator for integration with other units of knowledge. The training session ends.

Figure 8: Flow diagram of an integration unit in Incubator. The flow of a session for integration of knowledge in Incubator is in principle similar to that in Playroom, but there are certain important differences. The trainer is in this case not a human person, but an AI agent previously created in Playroom. Also, instead of the guessing of Ideatheca (as was the case in Playroom), in Incubator *Machine genes* will be typically guessed. This has for a consequence that to complete a test, guessed Machine genes have to be applied to generate Ideatheca, which in turn needs to generate Working knowledge, which is then used to produce behavior.

Behavior is not judged on the basis of observation but on the basis of comparing the behavior created by the student to that created by the trainer. If the two behaviors are sufficiently similar, they are considered a *match*, and the guessed genes are retained. If there is no match, the guessed genes are reverted to a previous state.

Similarly, instead of a trainer judging whether the objectives of knowledge transfer have been achieved, an *overall error rate* is used as a criterion, the value of which has to be low in order to consider the session complete.

Guesses to Machine genome can be made by various mechanisms that can involve completely random changes, recombination of previous genes, random drift of existing genes, and others.

Once the session is complete and the new AI integrates sufficiently well the new knowledge without impeding the previous knowledge, this improved version of AI can be used as a replacement of all the previous versions and hence, in principle, can be prepared for *offering as a product on the market*. The incubation session here ends.

Figure 9: Use of Incubation tree. The Incubation tree determines the steps in which bits of knowledge acquired in Playroom are being integrated in the Incubator. For example, if ten training sessions in Playroom resulted in ten new units of knowledge (*a* to *j*), those should not be necessarily integrated one by one to the resulting AI. Instead, much faster integration may be achieved if sub-steps of integration are preformed first. Those units of knowledge that are similar should be integrated first. In this example, pairs *a* and *b*, *c* and *d* and so on, are highly similar as the two units in each pair constituted e.g., a repetition of the same training session. These can then be integrated first as sub-steps (901 and 902).

Next, the units *a* to *d* may be more similar to each other than units *e* to *h*, which are then more similar to each other than units *i* and *j*. This may then justify integrating first *a* to *d* (903) and *e* to *h* (904) before the resulting sub-steps have been finally fully integrated (905).

Figure 10. The process of creating ever more advanced AI machines in AI-Kindergarten follows evolution of species and development of individual (i.e., phylogeny and ontogeny). The process will necessarily begin with very simple behavior such as taxis and will gradually evolve AI towards more and more advanced forms of behavior and intelligence. These steps will require repeating largely the path of biological evolution of behavior (phylogeny), which means that we will have to develop worm-level or insect-level-intelligence before we move to higher mental capabilities.

This process will take advantage of the fact that biological evolution has already performed number of experiments and has accumulated knowledge of the outcomes of those experiments. To optimize development of AI, we want to take maximally advantage of that available knowledge acquired through hard work in the biological past.

The process will also follow the development of an individual (its ontogeny). We use the fact that phylogeny and ontogeny exhibit similar sequences; Features appearing early in phylogeny tend to appear also early in ontogeny.

This allows adding genes to "adult" AI, something not easily done in biological individuals. This is referred to as *mutation of adults*. We do not need to start each new test of a change to genome from the very beginning of the organism development (i.e., from its conception)—like natural evolution needed to. We can substantially accelerate the process by mutating adults such that we follow biological phylogeny/ontogeny in that process.

If we need to repeat some steps of development, we can do that also much more efficiently than natural evolution did. As we save all the previous states of AI genome and Ideatheca, we need not start from the beginning of individual's development but can revert AI back for any amount needed.

As a result, in AI-Kindergarten, Machine genome and Ideatheca will grow over time largely in parallel. As the development of AI progresses, Machine genome grows and so does Ideatheca. Whenever, a part of Machine genome is frozen (1009, 1013) to accelerate tests of new genes (1010, 1014), in effect also frozen is the part of Ideatheca (1001, 1005) that is being primarily driven by those frozen parts of the genome. Adjusted are only new parts of ideatheca, as changed by the active genes (1002, 1004). This may accelerate the tests of new genes, but does not allow new knowledge in Ideatheca to be balanced with the old knowledge. In order to ensure balanced Ideatheca, after each successful acquisition of a new set of genes, the AI-agent has to interact with its environment in the Incubator in a mode in which the entire Machine genome is unfrozen and hence, in which all parts of the genome are active (1011 with 1012; 1015 with 1016). This interaction has to be long enough and rich enough to balance out Ideatheca across all its components (1003 with 1004, 1007 with 1008).

A criterion for balanced out Ideatheca is a good performance of AI across a range of tasks, old and new ones. A straightforward way for balancing out Ideatheca is to use the new AI in real-life situations for real-life purposes.

Figure 11. Historically, development of our technology evolves from easy emulation of human capabilities to more difficult ones (x-axis). Along the way, the technology asymptotically approaches human capabilities but never fully reaches them. Yet, at every stage the technology exhibits super-human performance in the respect to the capabilities that have been successfully emulated (y-axis). For example, a sharp stone provides immediately super-human capabilities in ripping skin, while a hand-held calculator provides immediately super-human capabilities in multiplying and dividing numbers. However, all those powerful abilities of technology remain domain-limited.

AI developed by AI-Kindergarten will not be different in that respect. AI will continue to cover more and more of the intelligence space of humans, providing at each step immediately super-human capabilities for the given limited domain.

Importantly, AI developed in AI-Kindergarten will likely never reach the full capabilities of a human, remaining in many respect forever limited to domain coverage of intelligence that is sub-human. For example, it may be very difficult to implement an equivalent of sexual drive or pleasure of eating a good meal. Thus, much like a hand-held calculator is super human in adding numbers and is sub-human in almost everything else, AI developed in AI-Kindergarten will similarly be always limited only to domain-specific super human capabilities. This kind of AI will have hard time understanding humans in many aspects, as it is unlikely to live a life that would fully correspond to a life of a human person. Many types of experiences will be missing—some because of technological limitations, some due to practical reasons, some due to market demands. As a result, although such AI may be able to drive a car safely, build and fix machinery, and solve engineering problems it will nevertheless always remain some sort of an "autistic" AI.

Importance for survival of human species

The figure also illustrates importance of each invention for survival of human species (indicated by the relative size of the box). Every new technological step typically provides a relatively smaller contribution in comparison to the previous one in respect to the degree to which it assists human race in surviving and producing a large population (indicated by a reduced width of boxes). Relatively larger steps may occur when new significant inventions are made and technological revolutions consequently follow. Examples may be industrial revolution and the invention of a personal computer. Introduction of T_3-intelligence may be one such relatively large step, although probably not nearly as important for the survival of human race as was the invention of hand axe. Therefore, AI developed in AI-Kindergarten, although likely capable of solving engineering problems that humans could not possibly solve (or would take a very long time for that), will not have a huge impact on the ability of human race to survive. T_3-AI may become an important companion to humanity for producing a high quality of life for many people but much like a hand-held calculator, it will never replace humans or exceed the domains of human intelligence.

Asymptotic intelligence growth despite continual super-human performance

The dynamics of approach towards human intelligence will be likely asymptotic. At the beginning the growth of capabilities may be fast. We also may get very quickly and with relatively little effort a T_3-AI that outperforms in most tasks the today's T_1- and T_2-technology. We may also get relatively quickly machines of the level of intelligence that beats non-human primates in many tasks. However, as we get closer and closer to human-level intelligence, further steps will necessarily become more and more difficult. It will take increasingly more effort to extract the necessary knowledge in Playroom and it will require also more resources to integrate that knowledge in Incubator. The samples of human behavior collected in a form of Playroom feedback may need to be larger and the worlds simulated in Incubator may have to gradually become more and more similar to those of humans.

The problems come from the fact that to be human-like intelligent, one needs to live a human-like life and this implies also the biological body of a human—with all of its properties and imperfectness's. It is impossible to possess a human knowledge on what it is like to have a flu without actually having a flu; and to have a flu, one needs a biological body with all its physiology. Consequently, and AI that has never experienced flu, will never be as good as a human in deciding in a new situation on what another human suffering from flu may need. Similarly, an AI may never properly understand what is funny in a joke about flu. And it may never appreciate properly a piece of art that refers to the experience of flu. And so on.

Only if AI would live through a flu, and a sexual life, and would have real meals and many, may other things, could AI know fully what it is like to be human. This will be technically difficult and for most purposes for which we actually need AI, it will be also unnecessary. Thus, there will be probably very little commercial need for human-replica AI. Hence, it is highly unlikely that AI-Kindergarten will ever produce such a human-replica.

The knowledge transfer theory also poses limits. Although it allows transferring knowledge from one agent to another, this transfer is only approximate: Something is always lost in translation. These lost parts will be more difficult to make up for in later stages of the development, as they will require increasingly more effort and resources to tune up AI appropriately.

Consequently, when interacting with AI, given enough time, we will always be able to eventually tell apart AI from a real human. That is, although it may fool us for moment, such AI could never fully pass a Turing test.

As a consequence, T_3-AI created in AI-Kindergarten will never fully reach human intellectual capabilities such that it can replace them in all the jobs. There never will be a situation in which there are no jobs to be done by humans. There always will be work for us—one that could only be done well by humans i.e., by those who know what it is like to be a human. How can a machine create art that is human appreciated if the machine does not have all the necessary human experiences? Also, how can a machine make a final decision on a graphical user interface for humans, if the machine cannot really know what is it like for humans to use such an interface? How can an AI make decisions on any design of objects used by humans if it cannot accurately tell what it is like to interact with those objects? AI produced in AI-Kindergarten will have hard time making decision about what a human would prefer in new situations.

So, no matter how far we develop T_3-AI and how much better then us it will become in solving a number of engineering, science, economy, or business problems, it will never be like us. No matter how much we try to make it like us, will never fully succeed. AI will never be fully able to understand humans.

At the end, we humans will be making the decisions that concern humans – whenever some new decision needs to be made regarding a question “Would a human person like this?” or “How would a human respond in this novel situation?” AI may do engineering but we will decide what will be engineered and what the end product will be like. AI may do science, but we will decide what question will need to be addressed and which results are to be kept. AI may make some decisions about economy but humans will decide which goals will need to be reached by those decisions.

Safety

It is very important that AI is safe for humans. We want AI to largely follow Asimov’s three laws of robotics. We do not want AI to get out of control, to become too powerful, mean, selfish, and in same ways too intelligent such that it would cease being a servant to mankind and become a master instead. We do not want AI to overtake the world. We do not want to be enslaved, and certainly not eradicated. There are fears that something like this could happen (Bostrom 2014). Clearly, we want to stop at the point at which we create technology that serves as a subordinate companion.

Good news is that much of the safety valves needed to keep AI at bay are already built in AI-Kindergarten by its very design. Due to the limited domains of intelligence of any version of AI, there is no possibility for AI to exhibit super-human capabilities in the domain that has not been trained and that is not a part of human capabilities. In addition, through Opaque knowledge we will have a continual overview of the developments and will be able to detect any developments that went into wrong direction.

AI developed in AI-Kindergarten will never be able to develop super-intelligence in a domain that it has not been trained for, or in one that has not been intended. Development of non-human intelligence is theoretically also possible, but this would require a different approach than AI-Kindergarten—an approach that would not use the advantage of existing biological knowledge but would instead require application of blind evolution by natural selection in a way that occurred naturally. In addition, one would need to provide environment selective for such non-human intelligence. This process would not take advantage of Playroom and Incubator acceleration and would hence require much more time and computational resources. Probably, it would be in the order of million times slower. For those reasons, it is unlikely that we would be even able to develop any time soon non-human intelligence to any significant degree.

One may argue that more important than the domain of intelligence are the motives and goals of AI (Bostrom 2014). After all, we want to ensure that AI does not want us harm. To address this issue it is necessary to note that motives and goals are a form of intelligence too. Goals are a part of cybernetic knowledge as they tell an agent what states it needs to reach to increase chances of its success. Everything that was said above about domain specificity of intelligence applies also to motives.

Therefore, much like AI cannot spontaneously develop intelligence in new domains, it also cannot develop new motives and instincts that would be outside the domain defined by human trainers in AI-Kindergarten. For similar reasons, it is not possible that harm to humanity becomes a sub-goal of some more important "end-" goal (such as an AI being programmed to create as many paperclips as possible and then killing humans in order to turn them into paperclips; Bostrom 2014).

These limitations of what can be changed in terms of goals and "instincts" come from one very fundamental feature of T_3 systems and this is the hierarchical organization of motives and goals. Lower levels of organization change the upper ones but not the other way around. Upper levels of organization cannot directly change the lower levels. An adaptive system cannot improve its adaptability by thinking-thorough which changes of its genome will make changes to its behavior. The effects of gene change remain necessarily Dark knowledge for an AI too, not only for human engineers. Moreover, by the very nature of how practopoietic systems work, any attempts to inflict changes on itself downwards are most likely to lead to a general loss of adaptive capabilities of the agent and its disarray (Nikolić 2015a).

Instead, adaptive systems operate such that mechanisms other then themselves decide on the lowest-level knowledge i.e., their genome, and hence on their "instincts". The rest of the system is naturally drawn towards following those instincts and not defying them. The consequence is that we can develop Machine genome of an AI such that its motives are to exclusively help humans and humanity.

The reasons that we humans have instincts to harm each other do not come as an automatic consequence of our intelligence. These drives of Thanatos are a result of natural selection. Only those ancestors of us who had these instincts survived. The environment in which we evolved dictated development of those instincts. If we make sure that our AI does not evolve in environment that makes pressure on development of such instincts, no such instincts will develop. By its very design, AI-Kindergarten is envisioned to foster just the opposite: kindness and obedience to humans.

That instincts of kindness and obedience are also biologically possible to develop, speaks the evidence that parents have a natural instinct to protect their own children even at a risk of their own lives. Similar evidence comes from the results of selective breeding in which we turned wild animals into domestic ones. If we could turn a dangerous wolf into a domicile dog, we can also make sure that our AI stays in the desirable range—because, what AI-Kindergarten does in essence is not much different from selective breeding of animals.

In conclusion, we can develop AI that is truly a companion to humanity, not its enemy. In fact, anything else would be very difficult to create. It is much easier to create a machine that will protect and defend humans then one that would harm them. We will nevertheless have to keep an eye on safety, and possibly also regulate creation of AI by law and international treaties. Important is that there is no imminent danger of AI-Kindergarten created AI wanting to harm humans or surprisingly overtaking the world. At the end, it will be always human persons who will call the major decisions in all respects in which we use that AI.

References

Ashby, W. R. (1947) Principles of the self-organizing dynamic system. Journal of General Psychology 37: 125–128.

Bostrom, N. (2014). Superintelligence: Paths, dangers, strategies. Oxford University Press.

Conant, R. C. & W. Ashby, R. (1970) Every good regulator of a system must be a model of that system. International Journal of Systems Science 1(2):89-97.

Dreyfus, H. L. (1972). What computers can't do. New York: MIT Press.

Mitchell, T.M., (1980) The need for biases in learning generalizations. Department of Computer Science, Laboratory for Computer Science Research, Rutgers Univ, pp. 184–191.

Nikolić, D. (2014) Practopoiesis: How cybernetics of biology can help AI Singularity Weblog

Nikolić, D. (2015a) Practopoiesis: Or how life fosters a mind. Journal of Theoretical Biology 373: 40–61.

Nikolić (2015b) Only T_3-AI can reach human-level intelligence: A variety argument. Submitted to *arXiv.org*

Nikolić (in preparation) Cybernetic knowledge-transfer theory. To appear on *arXiv.org*

Figure 1

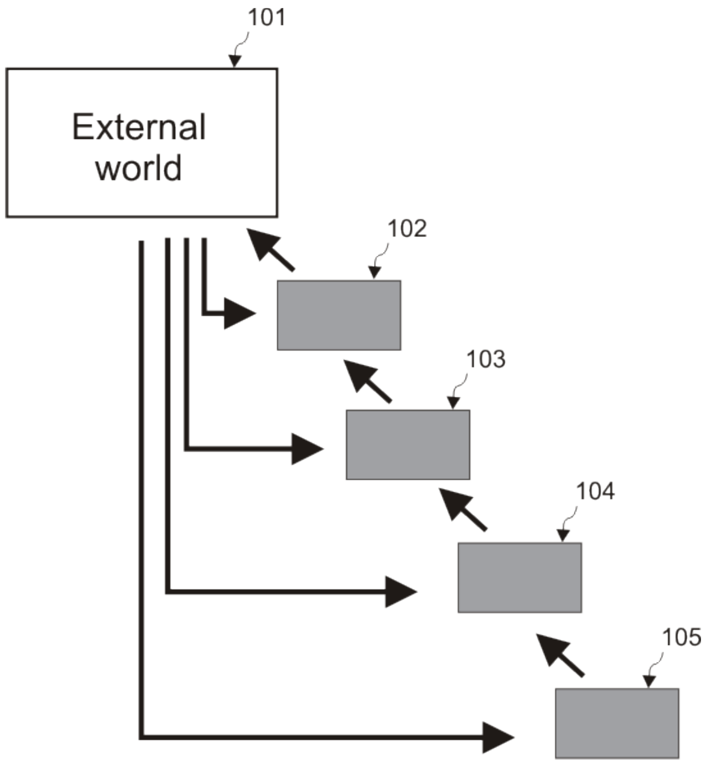


Figure 2

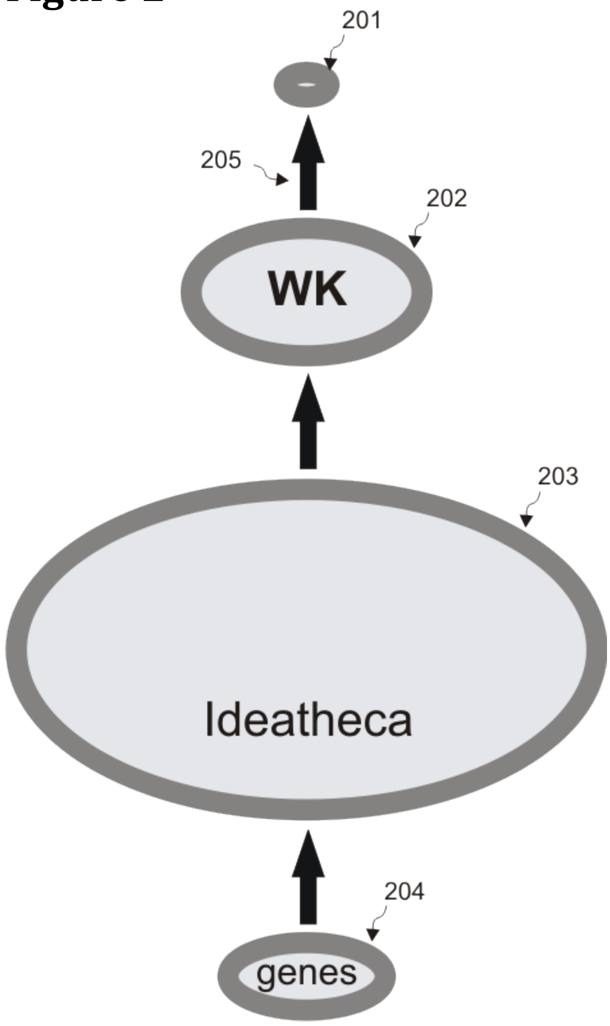


Figure 3

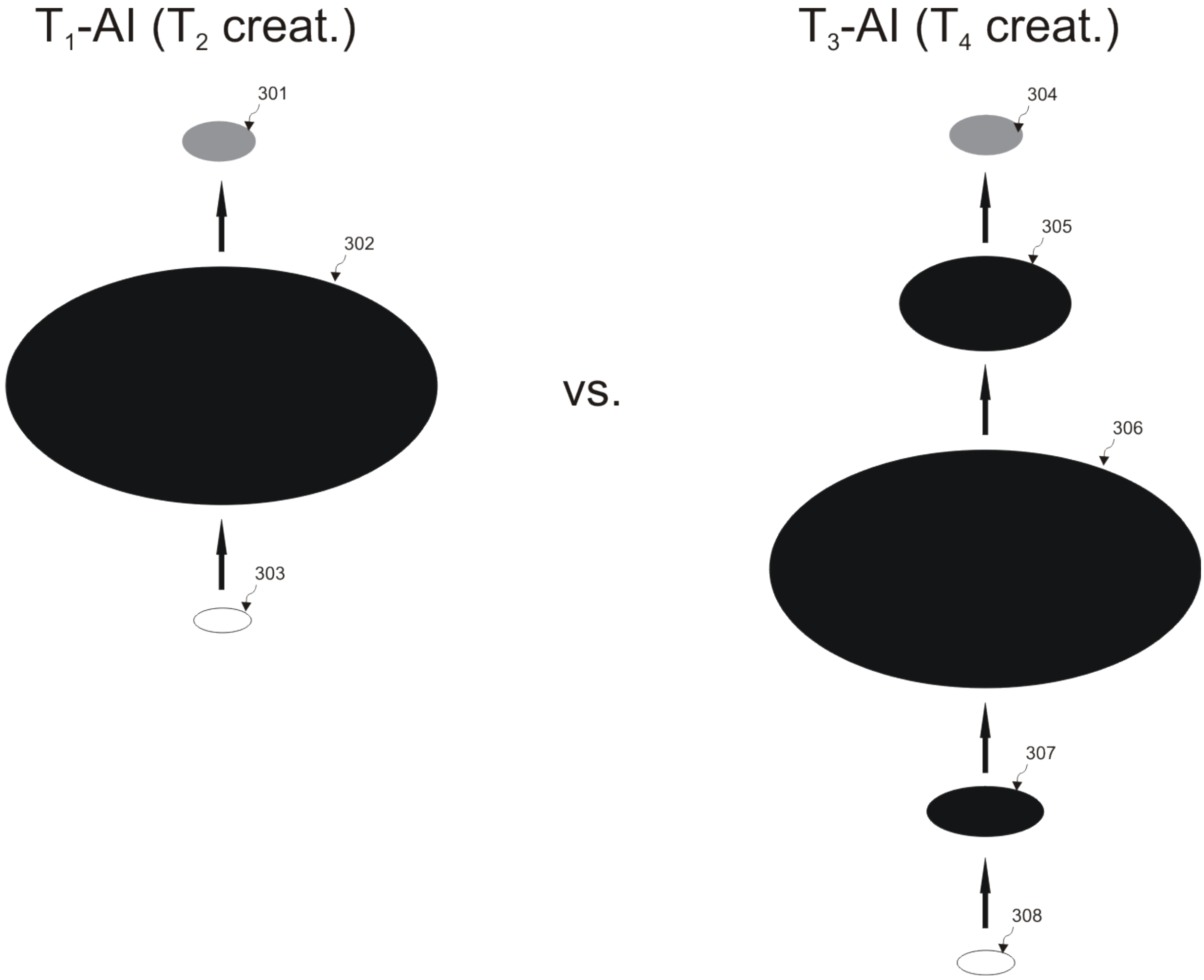


Figure 4

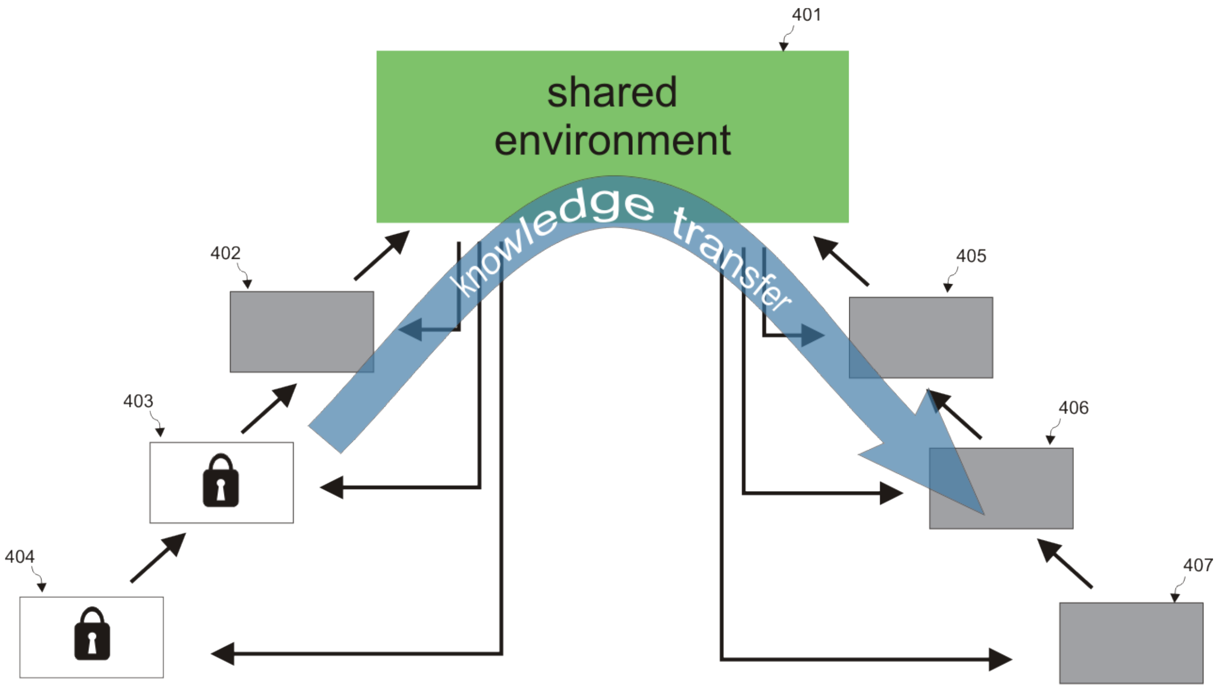

401
shared
environment
knowledge transfer
402
403
404
405
406
407

Figure 5

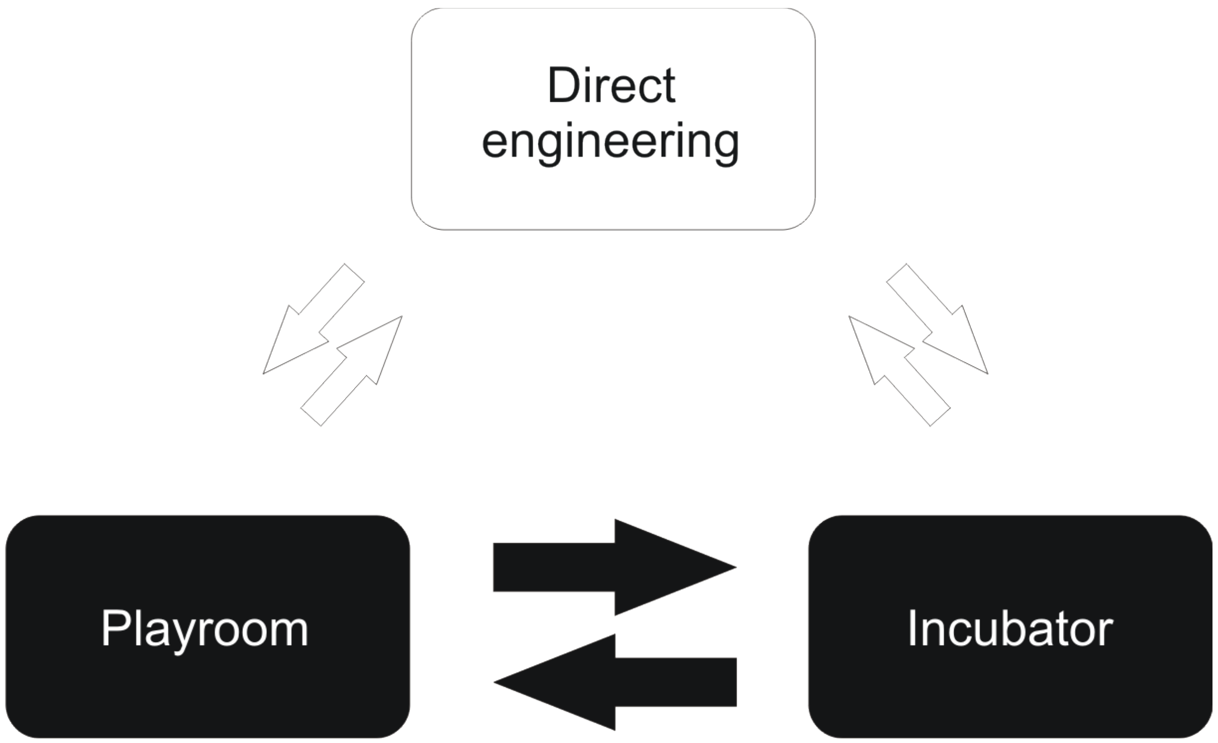


Figure 6

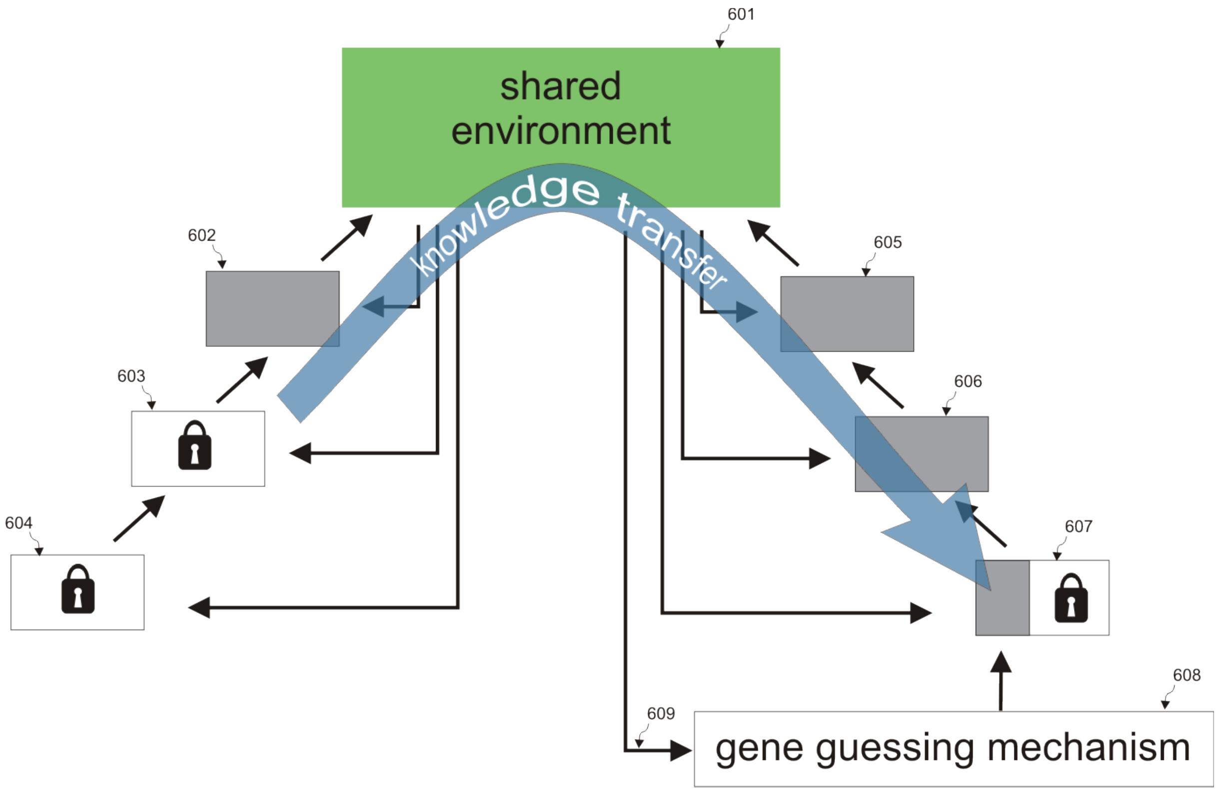

601
shared environment
knowledge transfer
602
603
604
605
606
607
608
609
gene guessing mechanism

Figure 7

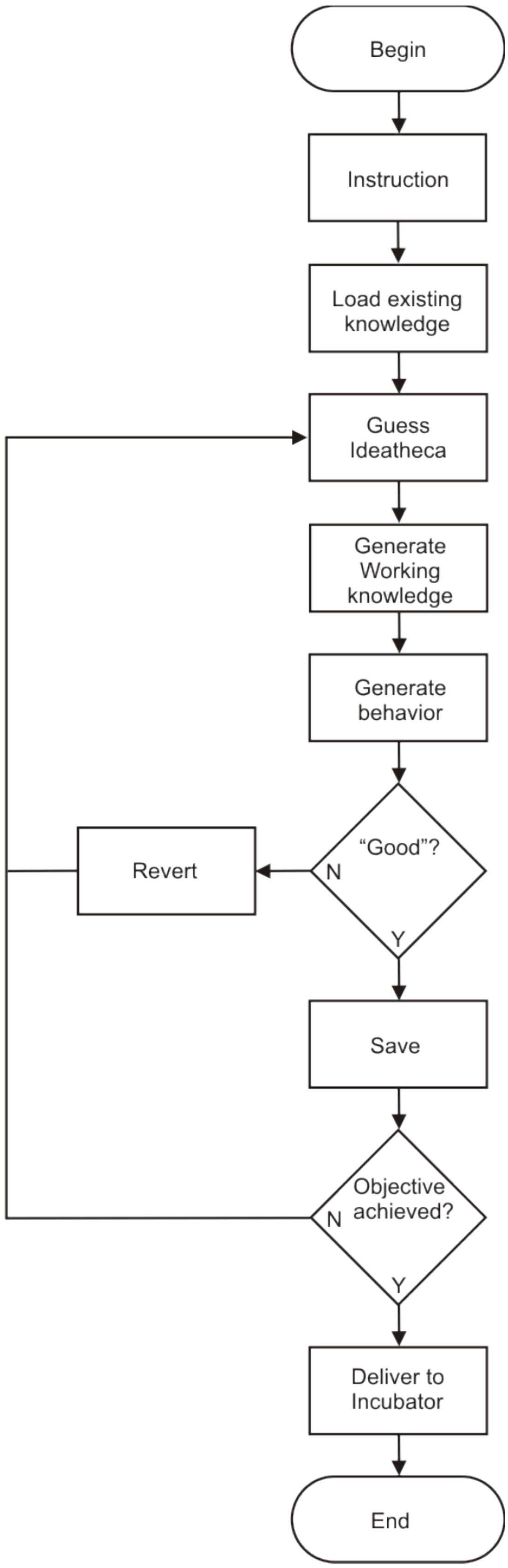
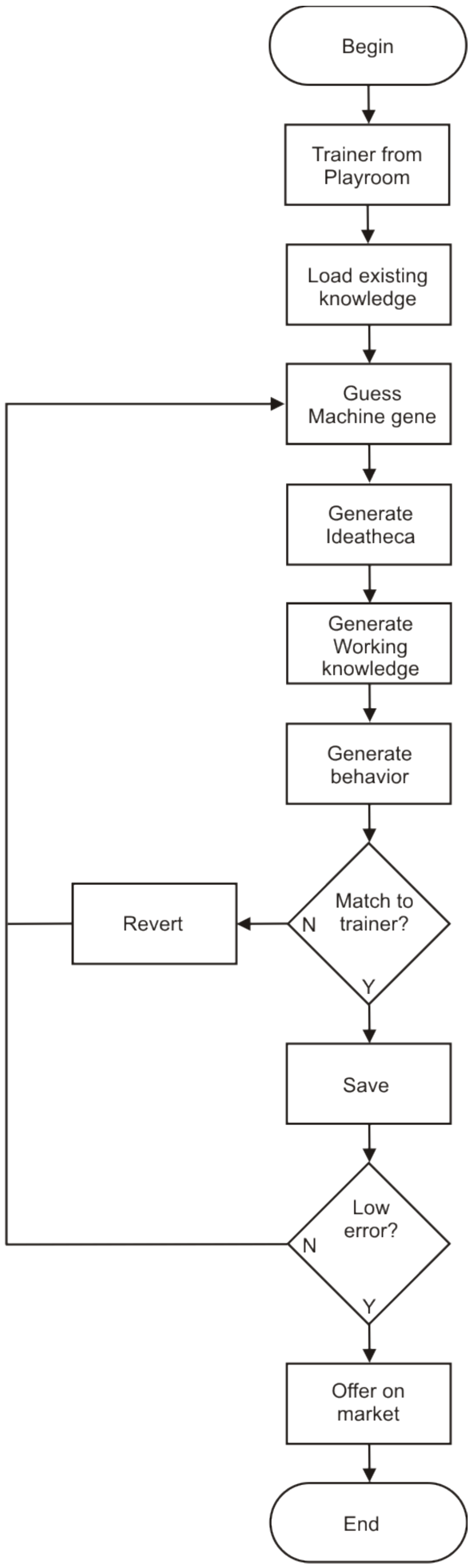

Begin
Instruction
Load existing knowledge
Guess Ideatheca
Generate Working knowledge
Generate behavior
"Good"?
N
Y
Revert
Save
Objective achieved?
N
Y
Deliver to Incubator
End

Figure 8



Begin
Trainer from Playroom
Load existing knowledge
Guess Machine gene
Generate Ideatheca
Generate Working knowledge
Generate behavior
Match to trainer?
N
Y
Revert
Save
Low error?
N
Y
Offer on market
End

Figure 9

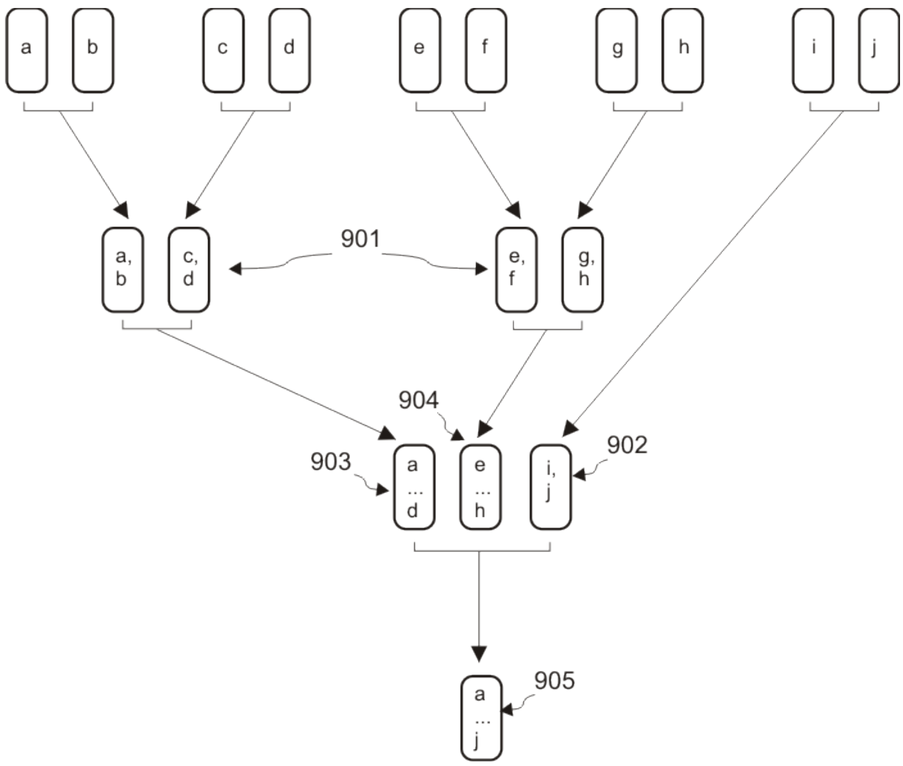


Figure 10

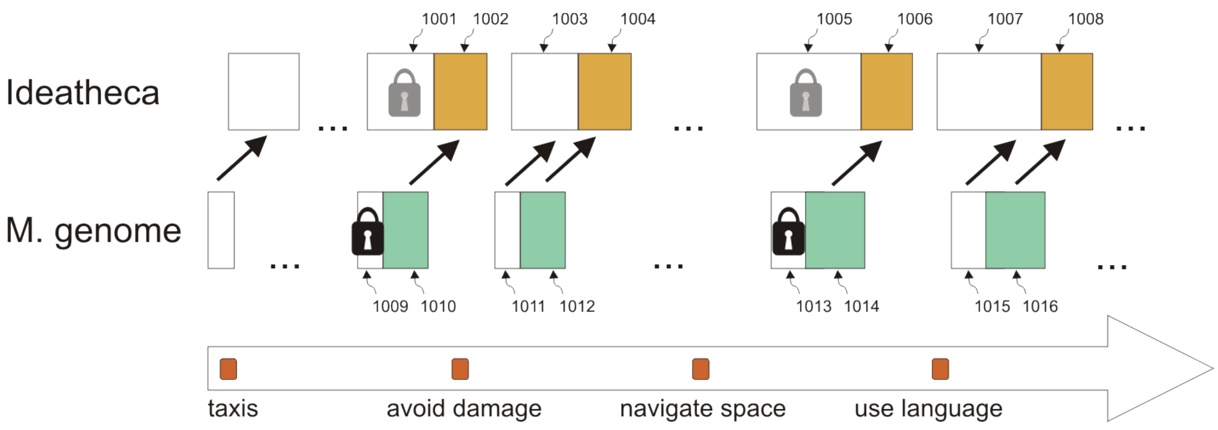
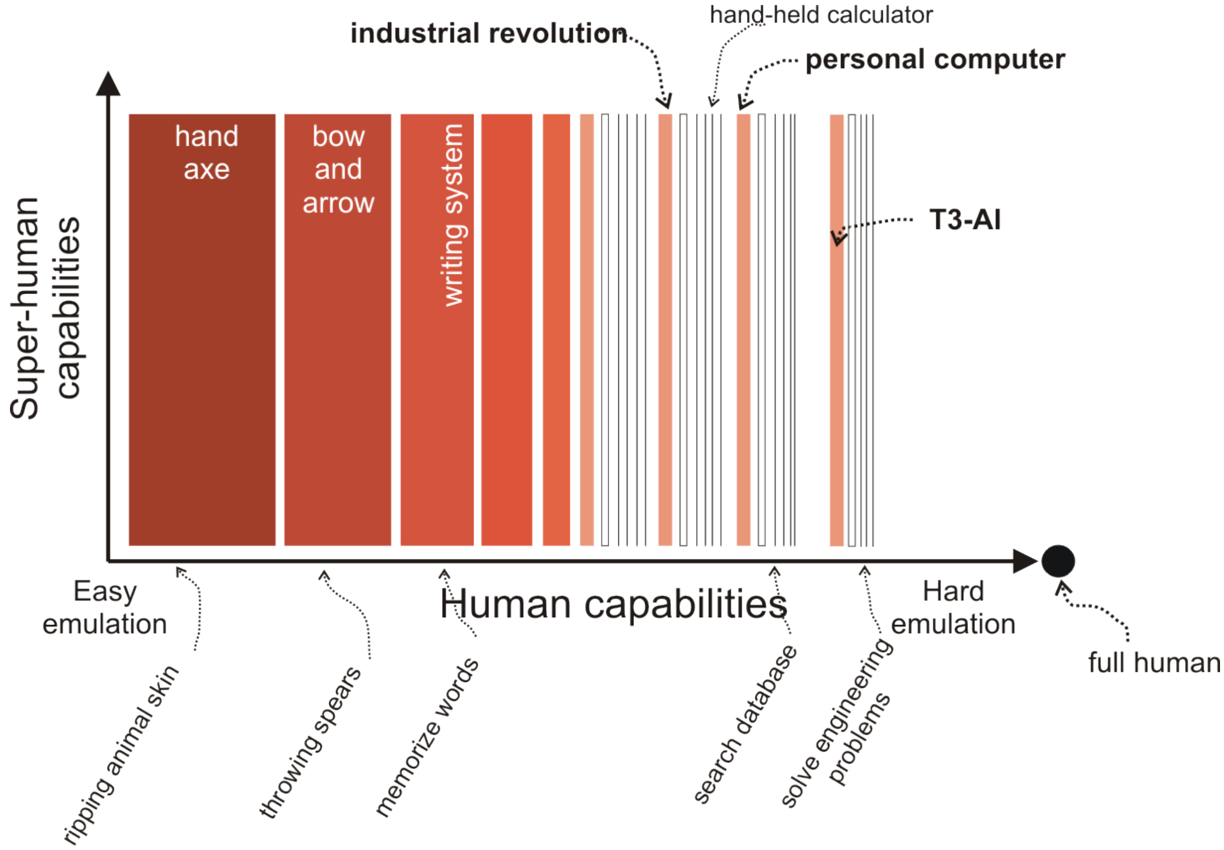

Ideatheca
M. genome
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
taxis
avoid damage
navigate space
use language

Figure 11


industrial revolution
hand-held calculator
personal computer
hand axe
bow and arrow
writing system
T3-AI
Super-human capabilities
Human capabilities
Easy emulation
Hard emulation
full human
ripping animal skin
throwing spears
memorize words
search database
solve engineering problems