

POSITION PAPER

Open Access

# AI-perspectives: the Turing option

Frank Kirchner<sup>1,2</sup> 



## Abstract

This paper presents a perspective on AI that starts with going back to early work on this topic originating in theoretical work of Alan Turing. The argument is made that the core idea - that leads to the title of this paper - of these early thoughts are still relevant today and may actually provide a starting point to make the transition from today functional AI solutions towards integrative or general AI.

**Keywords:** Artificial intelligence, AI technologies, Artificial neural networks, Machine learning, Intelligent robot interaction, Quantum computing

## Introduction

When Alan Turing approached the topic of artificial intelligence<sup>1</sup> (AI) in the early first half of the last century, he did so on the basis of his work on the universal Turing machine which gave mankind a tool to calculate everything that can effectively be calculated.

To take the next step and to think about AI seems almost imperative in retrospect: if there are computational phenomena on the hand then there must be the ‘non’ computational phenomena on the other and to deal with the question of the structure of this class of phenomena is only consistent and leads straight to the only existing evidence that it is at least possible to deal<sup>2</sup> with this class of problems and that is human or natural intelligence.

In light of the increasing number and areas of applications of AI research, it is interesting to follow Alan Turing to structure himself in this ‘Landmark Paper [1]’ which approaches the phenomenon of AI. After he introduced and discussed a number of machines of different but limited computational power, he went on to

introduce a special class which he called ‘unorganised Machines [1]’ and which has already anticipated many features of the later developed artificial neural networks. E.g. many very simple processing units which, as a central property, draw their computational power from the complexity of their connections and the resulting interactions.

For as far sighted and visionary his ideas have been it was clear – also to himself – that the resources to effectively built such machines were not available in the early half of the twentieth century and he actually argues to focus instead on less resource demanding tasks like cryptography, game playing (chess), speech recognition and understanding and of course logic.

So early AI could be coined the age of great visionary insights and theoretical developments in the face of drastically limited resources in terms of computational power, storage capacity, communication band-width and of course data availability. Consequently, a general debate on anything concerned with the implications of AI research did not take place as it appeared to be more science fiction rather than something that could be a reality any time soon.

This changed for the first time with the upcoming age of inference engines when computational logic was able to unfold some of its capacity being able to rely on already increasing computational power which allowed implementing some of the early insights in logic, programming and formal languages.

<sup>1</sup>He does not call it AI. This term was only coined at the famous 1956 Dartmouth conference

<sup>2</sup>This is not to say Deep-Methods per se. It rather shall refer to the fact that the systems learns from sampling the Environment in whatever way this will be realized.

Correspondence: [Frank.Kirchner@dfki.de](mailto:Frank.Kirchner@dfki.de)

<sup>1</sup>Faculty of Mathematics and Computer Science, University of Bremen, Robotics Group, Bremen, Germany

<sup>2</sup>DFKI GmbH (German Research Center for Artificial Intelligence), Robotics Innovation Center, Bremen, Germany



© The Author(s). 2020 **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

As the availability of computational resources increased it became clear that applying some of the early approaches - especially those relying on statistical methods - was no longer a vision but could become a reality. Today the computational power is at a level that allows AI methods to be implemented in online learning systems such as streaming data learning in robotic systems that act as tour guides in museums or real-time face recognition systems at airport security check points and medical diagnosis systems that can identify cancer tissue in MRT scans with a higher precision than even an experienced medical doctor. At the same time, the internet and associated media applications increased the amount of available data of all sorts to an amount that allows for training of algorithms and applications like deep neural networks which require these extensive amounts of data. It seems like only yesterday when we had to ask questions like: is a computer chip powerful enough to run a special machine learning algorithm in a reasonable time or is there enough storage capacity available to store the data that would be needed to feed the algorithm for training.

#### Early AI: THE TURING OPTION

*'You cannot make a machine to think for you'. ... It will be the purpose of this paper to question it. ' [2].*

Developments happened with such speed that it seems as if we were simply not capable of keeping up with the pace of technological development until we realized only a moment ago that we have created a tool which actually begins to challenge us on grounds that we thought were reserved for human intellect and cognitive capability.

While this discovery is driving a chill down the spines of one half of humanity, it is a thrilling possibility for the other half and a stimulus to apply these technologies to applications that challenge human cognitive capabilities on an even deeper level. A prominent example is of course the fact that a 'machine' was able to beat the world champion by 3:0 in the game of Go.

However, frightening this may has been to the first half of humanity even the second has started to ask questions when an improved version of the 'machine' was able to beat the earlier version (the one that beat the human champion) in all of 100 games played against its predecessor [3].

The reason for the second half to begin to ask themselves some questions was a result of the fact that the improved version actually learned to play the game not by analyzing thousands and thousands of games played by human champions but just by been given the rules of the game and then teaching itself to play and finally win the game. Which means nothing less but that the

machine learned strategies that are so far superior to human strategies that even the world champion would look like a novice to the game. So, do we see an example of AI-Superiority here, does this result mean that some of the darkest dystopias are becoming a reality, a world where machines actually take over all levels of human cognitive abilities and are superior to the human intellect?

#### Early AI: the Turing option

A reasonable reply could be: 'Maybe, but not yet!'. The reason I would share this reply is the fact that what we are seeing in these cases is something that we call an 'Island Talent' or an example of what I would like to term Functional-AI, because these machines can do this one thing and they can do it with enormous precision and speed but they cannot do anything else. Present the Go playing machine with a game of chess or a simple poker game and it would be unable to perform even at beginner's level.<sup>3</sup>

Of course, it must be questioned whether game playing is actually something that indicates AI-superiority to be around the corner. As any game has a fixed Domain and framework it is therefore very different from the real world which is the reason I called this phenomenon 'island talents'. However, the point one should carefully note is that the 'isolated' problems that can be solved by these examples of Functional-AI are becoming more and more complex or difficult.

But does it mean that we will never get there? Absolutely not, the scientific community has already begun to take up the challenge and is now turning towards what is called the integration problem of AI, referring to approaches to develop architectures and frameworks that allow for the integration of different AI technologies in one system to work in parallel or to complement each other and that should consequently be called Integrative-AI.

However, it must be mentioned that the idea of overcoming some of the obvious limits of AI approaches has been addressed already with the first AI-Summer when approaches like 'deep-reasoning' [4] have been discussed. Actually, series of workshops have been organized to discuss the 2nd generation expert systems towards the end of the 1980's that also discussed issues closely related to the idea of integrative AI, a good overview is given in [5].

In his early work Alan Turing has offered an approach to the integration problem that I would like to call the Turing Option and that has established itself today in the form of AI-enabled robotics as a tool for studying AI by stating: 'A great positive reason for believing in the

<sup>3</sup>At least until it has learned the new game, which can be pretty fast with today's computational Resources...

possibility of making thinking machinery is the fact that it is possible to make machinery to imitate any small part of a man.<sup>4</sup>

Even if this sounds to today's ears as if he would propose to create cyborgs instead of solving the problem at hand - How to create intelligence in an artificial machine - it becomes clear very quickly that Turing has in fact understood very well that in order to achieve artificial intelligence it is not enough to have an efficient calculation tool e.g., the above-mentioned 'unorganized machines' [2], but that it is mandatory to embed this tool in a complex physical shell - or body. This thought is stressed to a level that we only begin to understand fully today, when he talks about 'education of machinery' instead of 'programming of machinery', as we are approaching robots of a level of complexity and integrated computational power that approaches like 'learning from demonstration' can be effectively implemented in real systems.

How much did Turing actually foresee the need for integrative AI. Obviously, he did not coin the term itself in his papers but he does argue for all the ingredients of an integrated theory of AI when he refers to the concept of building robots that should cite: "roam the English country site..." in order to learn for themselves from the interaction with this real-world environment. How else could a machine be able to 'roam the English country side' if it would not integrate methods like perception, planning, reasoning and action execution? On top of that he argues that the machine should just do this 'to roam around' in order to learn! Yet he also asks the machine to include learning capabilities, which in his view is the only way to achieve - on a step by step basis and most presumably over an extended period of time - intelligent functionality inside a machine.<sup>5</sup> Actually, at this point Turing also makes a clear distinction between the real world and game playing when he later acknowledges that to build these types of robots would be impossible, as they would be simply too heavy and impractical - literally collapsing under their own weight - due to the limited technology available at that time and instead to

<sup>4</sup>Intelligent Machinery, report written by Turing for the National Physical Laboratory, 1948. The paper was first published in 1968, within the book *Cybernetics: Key Papers*, eds. C. R. Evans and A. D. J. Robertson, University Park Press, Baltimore Md. and Manchester (1968). It was also published in 1969 in *Machine Intelligence 5*, pp 3-23, Edinburgh University Press (1969), with an introduction by Donald Michie.

<sup>5</sup>It is worth a thought as to whether or not Turing was striving for something that was later termed strong AI or if he recognized the fact that this would be yet another ball game altogether and in fact he meant to describe a machine showing intelligent functionality without being mistaken for an intelligent being. The later work on the Turing test would suggest that however it is unlikely he would have thought of a Turing test without the wall (no sight) between the human and the Maschine.

study game playing (chess) among other (symbolic) subjects instead. For that reason, I would personally grant him the right to have been the first one to point out the concept of integrated AI, yet not explicitly inventing the terminology but clearly drawing a line between a kind of Functional-AI as we see it today - with the implementation of 'island talents' like face recognizers, logistics optimizers and a like - and systems that integrate many of these functions and that represent Integrative-AI systems.

It is this thought that confronts us with a hard reminder to our future challenge as AI researchers by pointing out that these machines must learn their ability to deal with<sup>6</sup> the phenomena of 'non-computability'. They should do this step by step and through complex interaction with a complex environment which is why the physical body is indispensable - and that any attempt to make these machines factory - new with all capabilities built-in is basically impossible but must be achieved in a 'data-driven'<sup>7</sup> process of learning and becoming better...

### Functional AI: the era of island talents

But can we still rely on Turing's thoughts today? Is it still a blueprint for achieving artificial intelligence in machines? Does it mean that we just need to implement some deep learning into a robot eh voila... AI will emerge? If there is a lesson that we can learn from Turing's analysis of artificial intelligence I believe it is the fact that he concludes that AI is not some ordinary function that we can simply implement in a machine. Instead he lists a set of explicit requirements:

### Learning

Turing clearly concludes that it is an iterative process of improvement, yet learning is involved. This is not to say that we just need to implement Deep-Learning methods and we are done. Learning will definitely be data driven, as the system needs to sample the environment and will have to learn from this data. However, in an integrated AI-approach it must make more use of this information. Learned results must be stored and organized in a way that lets them be reused in later events - remember the process is iterative, so there is potentially the full life span of the system available for learning - and the learned results must be integrated to create Meta-Knowledge that will allow the systems require much less samples from the environment to come to conclusions in later stages of the process. In the best-case future learning will not be bound to look at millions of data

<sup>6</sup>To deal with is different from to solve.

<sup>7</sup>This is not to say Deep-Methods per se. It rather shall refer to the fact that the systems learns from sampling the Environment in whatever way this will be realized.

points (which is an extremely expensive process) to learn the statistical correlations and hidden dynamics for each new case, instead a combination with more classical reasoning approaches will result in single shot (or with only a few samples) learning.

### Physical interaction

He draws the conclusion that it needs to be a process of improvement from interaction with the real world. A simulation apparently is not enough. He specifically points out the need for interaction and refers to the very physical part of it. Not information exchange or some passive form of interplay of the system and the world rather he explicitly speaks of the physical interaction manifested in the ability to move around and to manipulate the environment.

From these explicit requirements we can or must derive some implicit requirements:

### Complex body

Therefore a physical body is mandatory and this physical body needs to be of a minimum of structural complexity. As a robot will hardly be able to navigate a real-world environment or manipulate objects in the environment if it does not have legs or arms/hands to accomplish this.

### Island skills

And finally, this requirement also asks for the ability to be able to master some of the 'island talents' that were already discussed. E.g. such a system must be able to perceive the environment and extract features of the environment with high precision and speed, to name just two of those island talents that computers are extremely good at today.

### Reasoning and planning

Moreover, it must be able to make sense of these features and objects in the sense that it must have a model of the environment and the objects within it that relates the features and objects to each other in order to be able to reason and plan.

### Control

And finally, it needs to possess of a control regime that allows it to move and manipulate in a meaningful and goal directed way in order to use the movements and manipulations for learning.

One can argue if or if not, this is a recipe to achieve AI in machines and one can also argue if the physical part (the body as well as the real world) is really so important. My personal opinion here is an absolute yes! Physics is important simply because the fact that it withdraws itself from perfect modelling and surprises us with

effects for which a priori solutions cannot be pre-compiled instead it requires to develop mechanisms and concepts to handle these not to be modelled effects in an efficient way. Once again to handle a problem is not the same as to solve it and I believe that here lies the clue for future research on AI and robotics and why integrative AI is an important next step and is in contrast to the contemporary application-oriented AI or Functional-AI. The 'Turing Option' will open up a new dimension to these machines. The physical world...

### Functional-AI: the era of island talents

#### *Finding the right path in an era of increasing resources*

Today we can still relax and watch with amusement the helpless steps of some of the robotic systems at the DARPA challenge (<https://www.darpa.mil/program/darpa-robotics-challenge>) to coordinate their own two feet just to walk up a stairway and we can smile at the helpless looking efforts trying to use a key to open a door... However, we should not be laughing too loud.

What are the requirements for modern robots that would increase their performance to a level where they would actually be of any usefulness to humanity; reliability; resilience; traceability of actions; fault tolerance; learning from a few examples instead of millions of data points; collaborating with people in a team and proactively solving problems themselves are just a few and they seem far away.

However, today, we can already produce structurally (kinematically) very complex robots (e.g. humanoids), which can be produced by lightweight construction methods, new and intelligent materials and especially by generative production techniques. These robots have a strong disposition for interaction with the real world (which is the prerequisite not only for learning motoric skills), which in many areas comes very close to human abilities, and can be operated effectively at the same time, i.e. they are efficient in terms of their energy requirements and the ratio of size/mass and the number of active/passive degrees of freedom.

#### *The importance of a physical body*

But one should ask the question of what are the fundamental conceptual aspects of why a body is needed. Why is embodiment so important and how does this relate to the concept of integrated AI. The idea of embodiment is actually around for a long time, see [6] as a landmark paper featuring this concept. The original idea of the concept of embodiment was to provide a new approach to robot control that diverged significantly from the so far dominant Sense-Plan-Act (SPA) concept. In order to get around the classical SPA flow it was mandatory to consider the structural and morphological features of the system in question. One has to admit that

this approach yielded some impressive results given the limited computing power and software (control) concepts involved. E.g. Wall-following suddenly became a piece of a few dozen lines of software running on an 8-Bit Micro-Controller. The reason this was possible was that instead of the classical SPA approach no modeling at all was involved and hence no sophisticated algorithms were needed. However, generality of the approach of course was lost as it was a piece of software that would implement Wall-Following on this one particular machine<sup>8</sup> and no other. I know because I spent a good time of my career building such systems thinking these would conquer new worlds... [7]. Instead of complicated mathematical models, e.g. mathematical representations of the environment and the robot, this approach used the given morphology of the robot as the model of the environment itself. This was done with respect to nature that was cited as an architect that designed systems (in this case biological ones) according to the needs of a given environmental niche and yet the system itself (given all its kinematic structures and possibilities) was the best available model of the environment.

It became obvious very quickly that strictly following the embodied approach would not push us beyond the border and yet hybrid architectures have come up that tried to combine the best of both worlds, fast none model based reactive layers with rather slow but model based higher level planning layers (see [8] for a summary) and in fact today most robots doing useful things in real world environments would employ a hybrid architecture in one form or the other.

What we can learn from some 30+ years of research on embodiment in Robotics and AI is twofold: On the one hand we understood that exploiting the features of the physical structure of the system we are trying to control makes a lot of sense and helps to achieve more robust performance on the other hand without the higher-level planning and reasoning layers these systems do not cross the threshold of significance for any useful application.

However, I think that we have not exploited the idea of embodiment deeply enough<sup>9</sup> before it became unpopular, or to put it in different words, before other developments became more promising and therefor more

popular. This is to say that the increase in computing power was very fast over the last 30+ years actually so fast that you just had to wait a little while until a very complex algorithm would become possible to be executed on a computer chip on your robot. As a consequence of this development it simply did not make a lot of sense to dig deeper into embodiment and to come up with systems that would employ what I would call 'kinematic intelligence', referring to features built into the mechanical structure of the system that enable, facilitate or just simplify certain 'intelligent' function (a good example are passive walkers [9]). Instead the more powerful computer chips allowed to use very powerful algorithms that accounted for the very low kinematic intelligence of the systems by 'modelling the pitfalls of the hardware away', in other words very complicated control laws could be used that were able to deal with low intelligent hardware concepts instead of putting more effort into the design of the systems hardware or body (and using the extra algorithmic power for other things...). I was again among those who took the bait when at the end of the 1990's a colleague and I were trying to make a robot autonomously navigate in sewage pipes.<sup>10</sup> It turned out to be a real challenge to design a system that could just physically travel down a concrete pipe [10]. We had many concepts in mind that would be able to deal with the challenging environment, however in the end we decided to screw a laptop and a few sensors to a modified radio-controlled toy truck and instead used the power of the laptop to implement a neural network that learned [11] to classify the structure of the pipes and types of junctions that the system encountered. Because it was extremely difficult to physically back track to last known positions in the pipe network – it would have required to actually use the much more sophisticated system designs that we had already thrown overboard – we developed a navigation system based on a distribution of so-called belief states an approach that was later developed by others [12] to be called probabilistic navigation. I cannot say I regret the path I followed at that time but I sometimes ask myself what if we would have used the much more complex designs we had figured out and used the extra computing power that we got for free to use it for higher aspects of cognition? Already at that time we dreamed of systems that would just stay down in the pipe systems forever (their live time) and continuously learn about the dynamics and changes in the environment and the system itself to become what we would call today a long-term autonomous system or a live long learning system or as Nils Nilsson termed it in the 1970's a 'never to turn off system'.

<sup>8</sup>Interestingly one can see the same hardware dependency in modern quantum computers, where a program would run solely on the machine it was written for.

<sup>9</sup>Please note that I do not question the advances that have been made in complex robot design. These are impressive especially in the area of humanoid systems and could profit enormously from various technologies in Chip-technologies, 3D-Printing, Sensor-Systems technologies etc... I would rather question if we studied and understood enough how kinematic-intelligence can take away some of the burden to achieve intelligent function in a machine from the algorithmic parts of that machine.

<sup>10</sup>Don't ask... We were young scientists and we needed the money...

My corollary on the importance of a body for integrative AI would be the following:

Complexity of the systems has to grow beyond a certain threshold in order for Integrative AI approaches to be reasonable or in other words for the ‘Turing Option’ to become available. Once the complexity of our systems does cross this threshold we will be able to observe these developments:

- a) Methods of integrative AI will be developed on a conceptual and framework level
- b) As a result, the level of intelligence in these systems will grow fast and
- c) complexity will come down again.

In fact, with the increasing complexity of kinematic chains (e.g. in manipulators or the legs and arms of humanoid robots) a solution using classical differential equations is no longer efficient or even becomes impossible when it comes to parallel kinematics or closed kinematic chains [13]. Only recently (deep) learning methods have been used to derive efficient models for control [14] and it seems to be a very reasonable assumption that these methods will be the tool of choice to cope with the dynamics of complex kinematic systems interacting with unpredictable environments,<sup>11</sup> especially if model-based approaches are combined with data driven learning methods, yet the need for integration is already visible even if today only in partial areas and not so much yet on a system-environment long term interaction level.

As a side note we should recognize that improvements in natural language processing nearly stayed a flat line in the chart of historical developments up until Neural Network based approaches and especially Deep-Learning

methods entered the scene, when the performance curve sky rocketed [15].

Therefore, the day we will be able to see a humanoid robot that integrates several AI-Technologies to run thru the forest, open a door with a key, or stitching a wound of a soldier while talking to him or her in a calm decent voice using the right words to psychologically calm the person down, while in the background it is planning the fastest path to the hospital given the current weather forecast and available transportation options, is most likely not too far away.

#### *Towards integrative-AI*

The greatest needs for research are effective approaches to the organization of the different processes that must be used to effectively operate e.g. robots as described above (<http://www.willowgarage.com/blog/2012/04/16/open-source-robotics-foundation>). If one looks at the list of required characteristics of these systems - in particular to be able to cooperate with humans in a team - a system is indeed described or required that can be described as AI-complete, in the sense that it actually requires and has to integrate all aspects of AI and that cannot be reduced to a simpler (less complex) solution. The methods range from the use of machine learning methods to control highly complex kinematics, the use of deep neural networks in sensor-based perception, the planning of complex action sequences, the reasoning from facts (those given to the system and those generated by the system itself) and finally the recognition of intentions of the human partner and the action of the robot adapted to a complex context.

Learning from millions of data points cannot be the right way to learn. A retired colleague of mine from the University of Bremen [16], who studied the frogs brain for decades, keeps nagging me by saying: “How is it possible that my frog can solve the problem to catch a fly with a brain of 7 grams of weight requiring a few Watt of power and your robot needs to look at millions of flies just to learn what a fly looks like – left alone to manage to catch it – and requires kilowatts of power...”.

Apart from being embarrassed I am trying to tell him that we have missed out to study how to organize and structure the things that we have once learned. Instead we focused a lot in the past decades to the process of learning itself and we apparently made some very good progress but we made less progress on studies on how to structure, organize and eventually network the things we have learned. Biological systems must have found ways to learn things much quicker and with less effort from what we are currently doing. There are many ways of learning and one aspect of learning is what could be called learning over generations. This concept refers to the fact that by generations of evolution some of the

<sup>11</sup>As an interesting side note: I have many discussions with my control students these days about replacing all of control theory with machine learning approaches. Instead of approximating the kinematic solutions by solving a set of differential equations, which rely on so many error-prone and changing parameters, we could just as well use a (deep) neural network to learn the kinematic dependencies and even the dynamic dependencies of these parameters. What I observe in my students and of course in the eyes of my colleagues from the control domain, is great staring eyes, disbelief or just a smile of sorrow. In these situations, I remind my discussion partners to remember the Alpha Go Example. Similar to my colleagues none of the Go champions would have ever believed that he or she could be beaten by a machine. It was just unthinkable; Go works on a level that is reserved for human cognitive capabilities no machine can ever even get close. Well, it did and just as well may the human cognitive abilities and experiences to formulate the set of differential equations in a way that solves the kinematics problem to be subject to an update by a machine that derives from many examples the inherent parameters and control laws that make a complex kinematic chain act in the desired way, maybe even in ways that go far beyond our imagination of how dynamically stable a given kinematic chain can be controlled.

things that have been learned by earlier generations of learners gets built into the hardware of the next generation of learners. This occurs as a co-development process in biological systems: on the structural (mechanical) level the frog evolved a longer tongue but at the same time also evolved a brain region (algorithmic level) to control the tongue. It can also be observed on the neuro functional level where e.g. the part of the brain that was developed to control the tongue was linked to the input from the visual part of the brain of the frog to form a more complex ensemble that solves the fly catching problem in coming generations even without any thinking. So, what was once a very costly process for many generations of frogs has been preserved and transformed into a system of lesser complexity.<sup>12</sup> One could say that the investment (to spent so much effort to learn fly catching) finally paid off for the species. We have not come to this level of Design principals in AI-Research or in AI-enabled-Robotics yet. But I think this is where we should be heading for and I think this is what the Turing Option meant at its core.

While in the last decades we have made considerable progress in the area of the different sub-disciplines of AI, the Turing Option (robotics) forces us to study the integration of these sub-disciplines into one system. On the background of the story of the frog it is important to note that this cannot and should not be considered a 'trivial' (software) engineering problem. Instead it is a problem that challenges us to act more economically on our resources and to find ways to melt down things that have once been learned with great effort into simpler, less complex structural elements of our systems.<sup>13</sup>

Here is also a reason why the body is so indispensable for generating multi-purpose-AI systems. The physical body quasi serves as a long-time storage medium for things that have once been acquired (learned) by the system on a purely algorithmic level. While the algorithmic level is where we are very flexible and fast we can evolve new concepts but when it comes for these concepts to be efficient tools they need to be implemented in a less computation demanding way. Somehow, we are re-approaching the original idea of embodiment by seeing the body as the best model of the environment. But while the last time we stopped at building single examples or proof of concept that in fact the body can be a good environmental model we should this time go for a deeper approach and study ways how we can systematically take advantage of this concept by building systems that improve over generations and with every generation they outsource some of the costly acquired knowledge

into the structural or functional design of the next generation of systems. Of course, this requires first that we have a notion of generations of systems (robots) we should in fact develop 'Generation Thinking' when it comes to AI-system design. Interestingly enough we do have notions of generations of smart phones or cars, but we do not have a notion of generations of AI systems at least not in a systematic way.

The challenge and scientific question are how to efficiently integrate the different complex levels; from control to reasoning, planning, and interaction. The term efficient here does not mean deterministic but refers to the above-mentioned ability to handle such complex systems. The important difference is that we need very complex machines (robots) to study or create artificial intelligent systems, which can only develop intelligence step by step from the learning interaction with a natural environment, and which, however, due to their structural complexity and the inherent complexity of the natural environment, force us to use non-deterministic methods to control these complexities.

Thus, we are required to develop organizational principles or integration structures that make these systems immanent non-determinism manageable to the extent that the resulting systems remain efficient machines, i.e. that they accomplish their tasks in reasonable time and with reasonable resources.

It should be pointed out that integrative AI is not Strong-AI as one may speculate. In fact, it is not even something that will take us beyond of the set  $P$  of problems efficiently solvable in polynomial time.

I would like to support my argument by a simple set theoretic argumentation. If we construct a set of problems that collects all the AI algorithms, then we can describe the set of Problems solvable by AI-algorithms combined by the functional composition of at least one element of this set. Functional AI could then be described as the set of problems solvable by one specific AI algorithm or a single element of the above collection of all AI-Algorithms, e.g. NLP or Human Face recognition. We can safely conclude that:  $\text{Functional}^{\text{AI}} \subseteq P$ .

In contrast to this class of problems we can quantify Integrative AI as the set of problems that requires to apply at least two AI-algorithms. So, we can describe the set of problems that are solved thru methods of Integrative AI as the cross product of the set of all AI-Algorithms. However,  $\text{Integrative}^{\text{AI}} \subseteq P$  still holds and therefore we must assume that we will have to be able to solve at least one problem outside of  $P$  to achieve Strong AI. Because we must assume that  $\text{Strong}^{\text{AI}} \subset \text{NP}$ , we can conclude that  $\text{Integrative}^{\text{AI}} \neq \text{Strong}^{\text{AI}}$ .

This argument implies that integrative AI can be defined as the set of combinations of one or more AI algorithms, note that this definition does not say anything

<sup>12</sup>This is an example why in my corollary on embodiment and AI I predict that complexity will come down again.

<sup>13</sup>Instead of being learned over and over again...

about how these algorithms are to be functionally composed. However, this definition also leads to the result that integrative AI is not something that will solve problems beyond P and it is no way to achieve strong AI or AI superiority.

### Perspectives on AI: the quantum option

Integrative AI is not a mystical step to create super intelligent superior systems but just one step further to create multi-purpose or general AI systems with a broader spectrum of possible applications. However, it is important to stress the need for an increase in complexity of the systems especially when talking about robotic approaches to achieve AI. Because only after a certain threshold will be met the option proposed by Turing will become available and the fundamental paradigm shift that is required when moving from pure function-oriented AI systems to multi-purpose AI systems will be addressed. The paradigm shift can best be described as a way to move from systems which performances can be accurately measured, predicted and maybe formally verified to systems which performances can only be described qualitatively and failure is part of the equation. The big challenge in new architectural concepts and programming or design frameworks will be to come up with methods that minimize the possibility of failures while exploiting the advantages of AI,<sup>14</sup> like generalization, robustness and fault tolerance in the presence of noise etc. Ways to minimize failures have been described earlier when the concept of learning over generations and the stepwise externalization of knowledge into the hardware of the system was discussed. Note that this does not only refer to mechanical parts and structure but can also be exploited when it comes to the codesign of Hard and Software. E.g. when a new piece of hardware or a sensor is added to a system it would be advantageous if that piece of Hardware would come with a piece of software that allows to use it.<sup>15</sup> This will initially raise the complexity of the systems, because we will first have to learn how to effectively build such architectures. But then it will drive the complexity curve downwards just as we saw it happen in other technologies e.g. automotive. In this domain the platform concepts actually first had to be understood and industry consortia had to form to better exploit the advantages etc. until one could observe that the complexity of the cars actually slowed down and more and more focus was

put on IT Technology inside the car rather than in the gear train, motor, clutch, breaks etc... Finally, this pathed the way for a complete change in car technology and with E-Mobility we now see a drastic reduction in the overall complexity of cars. What if we can observe the same to happen in AI-Technology.

### Perspectives on AI: the quantum option

*The era of computability and humanities responsibility in the light of unlimited resources.*

One consequence of new paradigms in programming or designing AI-systems lies in the possibility that these systems will make mistakes and that these errors will be minimized in the course of their 'education' - to put it in Turing's words - but will never be fully eliminated.

On the one hand this fact is exactly what in later decades may be a decisive criterion for the differentiation between artificial intelligence and simple automatons, on the other hand it is something that we – as designers – are deeply reluctant to accept; that our machines must make mistakes in order to qualify as intelligent machines...

To be able to accept mistakes might sound like an unthinkable suggestion to an engineer's ears and actually we should be very thankful for this mindset, just think about aircraft engineers would not try to eliminate even the smallest errors in their designs... However, there is a border line in the natural – or physical – world that appears to set the limits for precision and ultimately for engineering approaches, which can be described by the term complexity. E.g. to predict the turbulent air flow around an ice-covered aircraft wing is impossible as the interaction dynamics of air molecules with the rocky surface of the ice crust are too complex to be modeled and yet control strategies for such systems cannot be derived.

The solution to this class of problems is simply to avoid them<sup>16</sup> from happening in the first place. This is why we need to wait for take-off in winter flights for the de-icing service to finish...

However, we would not be humans if we would simply accept this border line, in fact we would have never been able to develop planes in the first place if we would have a mindset that accepts such borders. Computer science has always looked across the fence and was able to integrate solutions and theoretical results from other disciplines into their own field and to develop it further. We would not have formal languages to program computers if it would not have been for the linguists to lay down the formal foundations of Grammar and Language just to name one example. Consequently, computer science today is looking with more than just one eye at the

<sup>14</sup>Resembles the error correction efforts in Quantum Computing. Can we find a way to deal with the error because we cannot avoid them?...

<sup>15</sup>In a project called X-Rock, funded by the German Ministry of Research (FKZ 01IW18003) we are aiming to development of such an approach by defining a bottom up approach to co-development of robotic Hard- and Software. A publication on this approach is in press at the time of writing this report.

<sup>16</sup>Another way of dealing with a problem instead of solving it...

developments in a field of theoretical physics that actually deals with the ultimate border in the natural world. Quantum physics is trying to understand the world at the Planck scale, which is  $1.6 \times 10^{-35}$  m (<https://www.symmetrymagazine.org/article/the-planck-scale>). This is a scale that is so small that to be able for us ordinary people to grasp it we must resort to comparisons like: the size of the Planck length compared to the size of an atom is similar to the size of an atom compared to the size of our sun...

But what does this have to do with the perspectives of Artificial Intelligence. The thing is that on this scale things get a bit weird and actually the laws of physics as we know them and as we are using them to e.g. built aircraft do no longer apply. Matter is not observable, the concept of location is not defined, measurements do not work in the way we know it and all we are left with are probabilities... Probabilities of a particle being here or probably there in other words not a good ground to build reliable systems on. But is it really not - after all probabilities are the representation we are used to in machine learning. Actually, probabilities are the foundation of the success of the data driven machine learning techniques that are so powerful in nearly all the applications of AI and they are the foundations for stock market values of some of the biggest companies in this field to exceed the 1-billion-dollar threshold.

Quantum computing actually exploits these uncertainties. The fact that a particle on the Planck scale is in a superposition of states sounds like a nightmare to classical engineering but in quantum computing this phenomenon means that a qubit (quantum and bit) does not only represent 2 states but in fact infinitely many states and with every qubit that is added to a quantum computing system its computational power is doubled. This is very powerful and it is said that it outperforms the biggest supercomputer that we have today as soon as we reach about 50 Qubits [17]. Currently we stand at some 20 Qubits as IBM announced in 2017 already and they also announced to tackle the 50-qubit barrier<sup>17</sup> (<https://www.technologyreview.com/s/609451/ibm-raises-the-bar-with-a-50-qubit-quantum-computer/>). Even more interesting is the fact that IBM provides Qiskit an open-source quantum computing framework which is accessible for the public (<https://github.com/QISKit/>).

Some of the initial applications that especially companies are looking at today range from modelling materials on the molecular level, aeronautical simulation, cryptography and artificial intelligence. Quantum

supremacy, the point when a quantum computer will actually outperform all classical computers, is subject to a controversial debate because it is a moving target as classical computers are still getting better and faster and error correction in quantum computers proves to be a hard goal to reach. As described above, the theoretical computational power of a quantum computer rises exponentially with every qubit added unfortunately at the same time the errors in this machine will rise (eventually) exponentially too [18].<sup>18</sup>This is a result of a phenomenon called decoherence which in principle means that the environment can kill your computation before you had a chance to read the results and it refers to the fact that quantum states are very fragile and can be destroyed by even the faintest environmental interaction, which is why the generation of quantum bits is a mayor technical challenge and requires the systems to be held at close to 0 degrees Kelvin in order to be stable.

The actual shift however, comes from the need for programmers to change their view on programming with quantum computers. If it was said earlier that the quantum world is a strange place for engineers as it contradicts some of the most fundamental aspects of solid engineering the same is true for programming. Rather thinking in automation theory with all the properties that we have come to value so much like discrete states, discrete time and a set of defined properties that can be attributed to states we now have to acquaint ourselves to say goodbye to these fundamentals. Instead we have to think about programming rather as a composition of overlaying and interacting wavefunctions (which do have some defined properties at least) with a possibility for the programmer to intervene by modulating the properties of the wavefunction in order to direct the interaction and superposition of states in the associated qubits towards making desired interactions more likely than undesired ones. Quantum 'compilers' or development frameworks are helping us to make this shift in thinking by providing what is called a quantum gate. A famous example is the Hadamard gate which is a symbol for a complex operation that actually changes the properties of a qubit or better the properties of the wavefunction that we call a qubit. Actually, we use the Hadamard Gate to initiate the superposition state of a particular qubit. We could think of this operation as an initialization procedure that puts the desired qubit into the supposition state or creates the superposition property that results in the fact that we will measure a 50/50% chance of reading

<sup>17</sup>In fact during the review process of this journal Google announced to have achieved this goal, see (<https://towardsdatascience.com/google-has-cracked-quantum-supremacy-cd70c79a774b>) as a reference.

<sup>18</sup>Just a minute ago errors have been discussed as a possible distinctive criterion for AI Systems. They will qualify as robust systems when we achieve methods for these systems to deal with their errors. Much effort in Quantum Computing is today actually directed towards research on how to deal with errors (Decoherence) in Quantum Computing.

a 1 or a 0 if we would actually perform the measurement. One fact that we have to get used to is that if one puts two qubits through a Hadamard gate, performing the identical operation on both these qubits, and then reads the values, there is a 50% chance of seeing different results. Here is one reason why I think this is a paradigm shift in programming as our programming today relies 100% on the reproducibility and determinism of a set of basic operations like constant function, successor function and projection function as well as operations like composition and recursion in other words the set of primitive recursive functions [19] as the fundamental basis of computational complexity theory.

To understand just how big the paradigm shift in Quantum Computing (QC) will be for programmers one has to consider the fact that in QC you actually do not have your ordinary instruction set that can be applied to a set of registers that will deterministically alter the state of these registers. Instead in QC if one applies an instruction to a quantum system what actually happens is that a series of infrared pulses<sup>19</sup> will be applied to the qubit that will change the state of the qubit into the desired direction. In fact, the instruction set of quantum computers is currently only very limited. There are four basic important instructions that I would like to mention here. The Z Operation is an operation that shifts the state of the qubit by 180 degrees it can be regarded as the equivalent of the NOT operation in ordinary computers. The Hadamard gate already mentioned is used to put a qubit into superposition and the so called CNOT operation is a two-qubit operation that is used to put the two bits in an entangled state. Finally, quantum compilers offer a measurement operation that is used to read out the value of a qubit. The instruction set actually varies from the developer of the actual quantum hardware, which is another drawback in QC because a program that is written for e.g. a GOOGLE machine will not work on a machine built by IBM. The reason for this lies in the fact that for any quantum computer the above notion of gates actually does not mean anything these are just symbols for us the programmers to design programs (not very complex ones at the time being). In fact, the notion captured by the above gates must be translated into what could be called the machine language of a quantum computer which is a series of (in most physical machines) infrared pulses that are applied to the qubits in order to change the state of those bits. There are basically three different ways one can modify these pulses: the frequency, the amplitude and the duration of the pulse. This way of programming reminds me of playing a musical instrument rather than programming a

machine. As you need to know what note to play (the frequency), how loud you need to play it (the amplitude) and how long the note should sound (the length of the pulse). But apart from this analogy there is a very fundamental difference between an ordinary computer and a quantum computer. In a classical computer one sends the data to the machine which is then modified by the instructions that the computers perform on the data. Quantum computing is the exact opposite of this approach. The data sits inside the machine (the array of quantum bits) and the programmer sends the instructions (pulses) to the machine to modify the data.

Looking at the difficulties and very limited programs that we see with today's quantum computers one could be tempted to believe that this will never become reality. Especially the enormous hardware designs with miles of cables dangling around and tons of equipment held at very low temperatures does not look much like there will be one of these machines in every household any time soon. But this is an engineering problem and if one takes a look back some 70 years and compares how huge, fragile and error prone classical computers have been at that time it is not too far-fetched to believe that these problems will be solved and quantum supremacy may be reached (<https://www.datasciencecentral.com/profiles/blogs/quantum-computing-deep-learning-and-artificial-intelligence>, <https://towardsdatascience.com/google-has-cracked-quantum-supremacy-cd70c79a774b>).

## Conclusions

This look backwards brings us right back to the time that was discussed in the beginning of this paper. The time of early AI and Allan Turing's pioneering work. This time was described as the time of limited computational resources when we had to resort to 'Gedanken-experiments' to derive our theories and the possibility of tackling non-computable or not efficiently computable problems was absolutely impossible. Today we are in the middle of the time of increasing resources, though still too limited to tackle problems of the class of not efficiently computable, but we can now efficiently solve problems that we did not think we would ever be able to solve, e.g. create systems that predict tumor cells in an MRT scan better than a trained medical doctor.

We are looking at a time when this border will eventually fall and we enter the time of nearly unlimited resources (at least computational wise) and we will be able to solve problems that have been out of reach so far<sup>20</sup> [20]. In fact, it would be possible to compute the turbulences around an ice-covered airplane wing and eventually provide efficient control strategies, it would be

<sup>19</sup>The technology is actually machine dependent and varies among the different Quantum Computers.

<sup>20</sup>Referring to the new complexity measure called BQP (bounded-error quantum polynomial time) which is a class of problems solvable by a quantum computer in polynomial time with less than 1/3 error cases.

possible to simulate materials on a molecular level or to design new and personalized drugs with higher efficiency and less side effects and of course it would make a huge difference to Artificial Intelligence as we would be able to solve problems that cannot be solved efficiently on classical computers, e.g. simulations of the chemical interactions of molecules could be achieved that will allow us to generate the huge amounts of data for training extremely deep neural networks (that we could not sample) to design the above mentioned personalized drugs.

To summarize the challenges for AI researchers in the near future will be manifold but it is obvious from my remarks here that I believe in the Turing Option, which involves the hardware of the systems and the interaction with the real world. As a summary of this paper I would like to extend the Turing Option by the notion of ‘Generation’ for technical systems especially for AI Technology. This is to say that based on the systems Turing proposes we should develop systematic ways to carry the achievement of one generation of such systems on to the next and implement some form of evolutionary robotics much as evolutionary programming already does it. The requirements of such an approach go beyond the current approach to design, build and program systems as they require much more standardization and a level of community platform thinking much as we see it for decades now in the Automotive industry.

Here is an attempt to create a list of the Topics and directions we should pursue more deeply:

- 1) Go to the real world and built systems of a complexity which is high enough to inter (act) with (in) the real world and to survive this over longer periods of time.
  - a. How can the system complexity threshold be quantified? How complex is enough?
  - b. How to use the Hardware of the system as storage medium for learned stuff.
- 2) ‘Generation Thinking’ can make systematic use of the body as a model of the environment how to implement the notion of generations of systems.
  - a. How to externalize learned knowledge into physical hardware... In order to be able to deal with the real world rather than trying to solve it?
  - b. Alternative massively parallel dataflow computer architectures should be implemented in our robots. What are the properties of those architectures?
  - c. How can the concept of ‘Generation Thinking’ be transferred to learning in AI-Systems?
  - d. How to use this concept to overcome learning from millions of data points?
  - e. How to systematically integrate Symbolic and sub symbolic Learning for recognition and perception as well as action and planning in these systems?
- 3) What are the architectural, programming and design methods that minimize the possibility of failures while exploiting the advantages of AI?
  - a. How much will integrative-AI bring the system complexity up?
  - b. If Integrative<sup>AI</sup> P is true can ‘Generation Thinking’ overcome this limit?
  - c. Can Quantum-Computing bring the raising system complexity down again.
  - d. What benefits can AI actually draw from QC? Is there more that standardization, optimization and simulation?
  - e. Can AI learn to program a quantum computer (do the paradigm shift better than us)?
- 4) Last but not least we must study how to better communicate the research we are pursuing in this field. This must be understood by the community to ensure the technology development is sufficiently well communicated and discussed with society and the results of this discussion must be reflected in the research that is pursued.

These developments point out that mankind must begin to learn to deal with its increasing power especially modern AI-research and the possibility of Quantum Computing makes a journal that focuses on the perspectives of AI extremely useful. It is the intention of this paper to stimulate the discussion about the future of our research area where are the frontiers now, where will they be in a decade? Are there limits at all and if so where are they and why are they real. If there are no limits,<sup>21</sup> what does this mean for our responsibility as researchers...

#### Abbreviations

AI: AI artificial intelligence

#### Acknowledgements

Not applicable.

#### Author's contributions

FK was the mayor contributor in writing the manuscript and read and approved the final manuscript.

#### Funding

Not applicable.

#### Availability of data and materials

Not applicable.

#### Competing interests

The author declares that he has no competing interests.

<sup>21</sup>As far as our current understanding can predict...

Received: 12 July 2019 Accepted: 16 July 2020

Published online: 04 September 2020

**References**

1. Intelligent Machinery, A Heretical Theory\* A. M. TURING *Philosophia Mathematica*, Volume 4, Issue 3, 01 September 1996, Pages 256–260, <https://doi.org/10.10193/philmat/43256> Published: 01 September 1996, <https://academic.oup.com/philmat/article-abstract/4/3/256/1416001?redirectedFrom=fulltext>.
2. Intelligent Machinery, report written by Turing for the National Physical Laboratory, 1948. The paper was first published in 1968, within the book *Cybernetics: Key Papers*, eds. C. R. Evans and A. D. J. Robertson, University Park Press, Baltimore Md. and Manchester (1968). It was also published in 1969 in *Machine Intelligence 5*, pp 3–23, Edinburgh University Press (1969), with an introduction by Donald Michie.
3. David Silver, Aja Huang u.a.(2016) Mastering the game of Go with deep neural networks and tree search. (Memento vom 28. Januar 2016 im Internet Archive). *Nature* 529: 484, doi:<https://doi.org/10.1038/nature16961>.
4. Randall Davis, (1982). *Expert Systems: Where Are We? And Where Do We Go From Here?* *AI Magazine* Volume 3 Number 2 (1982) (© AAAI).
5. David J-M, Krivine J-P, Simmons R. *Second generation expert systems*. Springer Science & Business: Media; 2012.
6. Brooks RA. A robust layer control system for a Mobile robot. *IEEE J Robot Autom*. 1986;RA-2:14–23.
7. Jessica Ebert, (2005) Scorpion robot could conquer worlds <https://www.nature.com/news/2005/050207/full/050207-14.html>.
8. Vukobratović M, Stojić R. Historical perspective of hybrid control in robotics: beginnings, evolution, criticism and trends. *Elsevier, Mechanism and Machine Theory*. May 1995;30(4):519–32.
9. McGeer T. Passive dynamic walking. *Int J Robot Res*. April 1990.
10. Kirchner F, Hertzberg J. Autonomous mobile robot system for sensor-based and map-based navigation in pipe networks. US Patent. 2000;6:108,597.
11. Hertzberg J, Kirchner F. Landmark-based autonomous navigation in sewerage pipes, proceedings of the first Euromicro workshop on advanced Mobile robots (EUROBOT'96); 1996.
12. Thrun S, Burgard W, Fox D. *Probabilistic robotics (Intelligent robotics and autonomous agents)*. Cambridge/London: MIT Press; 2005.
13. Siciliano B, Oussama Khatib (eds.) (2008). In: *springer handbook of robotics*, chapter 5.5., page 121 (ff). Springer Science & Business Media.
14. Aaron D'Souza, Sethu Vijayakumar and Stefan Schaal (2001) *Learning Inverse Kinematics*. Proceedings of the 2001 IEEE/RSJ, International Conference on Intelligent Robots and Systems, Maui, Hawaii, USA, 29 Oct – 03 Nov 2001.
15. Fisher IE, Garnsey M, Hughes ME. Natural language processing in accounting, auditing and finance: a synthesis of the literature with a roadmap for future research, *Intelligent Systems in Accounting Finance & management* 23(3):n/a-n/a; 2016. <https://doi.org/10.1002/isaf.1386>.
16. Roth G. *Das Gehirn und seine Wirklichkeit*. In: *Kognitive Neurobiologie und ihre philosophischen Konsequenzen*. Suhrkamp: Frankfurt am Main; 1994. isbn:3-518-58183-X.
17. John Preskill (2012-03-26). "Quantum computing and the entanglement frontier". arXiv:1203.5813 [quant-ph].
18. Peter W.Shor (1995). Scheme for reducing decoherence in quantum computer memory. AT&T Bell Laboratories.
19. Brainerd WS, Landweber LH. *Theory of computation*: Wiley; 1974.
20. Nielsen M, Chuang I. *Quantum computation and quantum information*. Cambridge: Cambridge University Press; 2000. isbn:0-521-63503-9.

**Publisher's Note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)