Singularity#1 and MFA II. Singularität Nr. 1 und MFA II.

Dr. Bheemaiah, Anil Kumar. AB, Seattle WA 98125 USA.
miyawaki@yopmail.com

Abstract

The Dog-Ears formal system (Bheemaiah, n.d.) is extended with MFA II architecture for the definition of Taskoids, needing adaptable designs and additive printing. We present a formal system to apply the formulation to illustrate Singularity#1 as an MFA II, application. The concept of Singularity and Singularity#1 and the MFA II design philosophy is explained, with an abstract photographic art.

Keywords: Dog-Ears, Taskoids, Singularity, Singularity#1, Robotics, Conversational UI, additive printing.

What:
Singularity#1 is defined as an algorithmic machine evolution like genetic algorithms, in MFA II architecture, with the development of machine learning algorithms for the automated design of hardware and software and additive manufacturing of the hardware.MFA II is a multi-functional architecture, where side effects are primary too in a defined multi-functionality, it is inspired by MFA I architecture of form following function and behaviors from BEAM robotics, while BEAM is analog or mixed, MFA II is digital.

Was:
Singularity # 1 ist definiert als eine algorithmische Maschinenentwicklung wie genetische Algorithmen in der MFA II-Architektur mit der Entwicklung von Algorithmen für maschinelles Lernen für das automatisierte Design von Hardware und Software sowie für die additive Fertigung der Hardware. Es ähnelt der Singularität, die sich durch die Entwicklung von Hardware und Software für maschinelles Lernen durch A.I-Algorithmen auszeichnet.
MFA II ist eine multifunktionale Architektur, bei der Nebenwirkungen auch bei einer definierten Multifunktionalität im Vordergrund stehen. Sie ist von der MFA I-Architektur mit Funktionen und Verhaltensweisen der BEAM-Robotik inspiriert, während BEAM analog oder gemischt und MFA II digital ist.

How:
We illustrate Singularity#1, in non-anthropomorphism in the design of Alexa skills and hardware tools for the sentient bot platform with additive manufacturing. Hardware extensions with the RetroSwitch are defined with a mathematical formulation, and templates with customization for the RetroSwitches and hardware designs are illustrated.
TensorFlow is used for a mathematical formulation of MFA II with generalized Tensors.

Wie:
Wir veranschaulichen Singularität Nr. 1 im Nicht-Anthropomorphismus bei der Entwicklung von Alexa-Fertigkeiten und Hardware-Tools für die Plattform für empfindungsfähige Bots mit additiver Fertigung. Hardware-Erweiterungen mit dem RetroSwitch werden mit einer mathematischen Formulierung definiert, und Vorlagen mit Anpassungen für die RetroSwitches und Hardware-Designs werden veranschaulicht.
TensorFlow wird für eine mathematische Formulierung von MFA II mit verallgemeinerten Tensoren mit eingebetteten Unendlichkeiten von Kovarianz und Kontravarianz und einer Konturintegralformulierung verwendet.

<media>

Singularity.

Fig 1: An image of an irregular chunk of concrete, a random act of human creation, which is not well formed and devoid of the mereotopological ontologies of a natural stone and its metaphors, denoting a synthetic creation, but not a willful sculpture either in form or an abstraction.

Fig 2: An image of a well formed stone in natural metaphors of mereotopical ontologies, representing a natural sculpture, denoting natural intelligence.

Singularity(Shanahan 2015) can be defined as an observable phenomenon, when A.I is of a natural well formed metaphor, in mereotopological ontologies beyond the diagonal argument. As in Fig 1, a synthetic creation of non willful metaphors, as side effects cannot be Singular, as it would not be well formed or beyond the diagonal argument.

Could Singularity be the result of machine evolution beyond the willed creation of human endeavor. This would have to be a side effect of human programming or A.I developed technology, it is unlikely that given the nature of electromagnetism, created by humans, which is not natural or well formed, that natural computing in crystals, quasi-crystals or amorphous medium would emerge.

</media>

Introduction:

Singularity#1 in the context of Multi Functional Architecture II

Singularity#1 is defined as an algorithmic machine evolution like genetic algorithms, in MFA II architecture, with the development of machine learning algorithms for the automated design of hardware and software and additive manufacturing of the hardware.MFA II is a multi-functional architecture, where side effects are primary too in a defined multi-functionality, it is inspired by MFA I architecture of form following function and behaviors from BEAM robotics(Wikipedia 2013), while BEAM is analog or mixed, MFA II is digital.

Unlike natural evolution, which is usually interpreted as Darwinian evolution, machine evolution is intent based(Russell, n.d.), it is creation with intention and not natural selection, hence Singularity#1 can be thought of a process of automated machine evolution, which is programmed and not based on a system of natural evolution.

MFA II is multi functional architecture II, which derives from the BEAM philosophy of biomimetics, where form follows function, and multi functionality defines a multi-dimensional description of the functions, in primary side effects as a multi dimensional reality of functionality with any basis, in this paper we define a basis of tensors and use TensorFlow(Ketkar 2017) to define and evaluate

Tensors to solve ontological descriptions(de Pablos and Patricia 2012; Li et al. 2018) in an algebra to define a map to a geodesic based object representation for 3D printing of hardware.

Background:

Intentional Evolution has been much debated in the context of natural evolution, but in machine evolution, intentional evolution is a natural mode, Taskoids are defined as the intentional evolutionary stream of machine intelligence(Bheemaiah)("Website" n.d.)r we extend the Dog-Ear formulation of robotic automation to singularity#1, a demonstration of intelligent tool usage by robots, proving the existence of intelligence similar to natural intelligence by robotics.(Kuhle 2018).

<original-contribution>

Formal Theory:

Definition 1.0: C is a camera-based input stream, for a monocular vSLAM system, defined by
-> (E, C); ->(M, C), ->(P, C)
By Dog-Ears formulation, there exists the event space of events [E] and actions [A], and the system of flex-rules [FR] and a State machine and cloud functions, [S,F]
Where [M] is a feature map from a point cloud and [P] the pose of the camera.

Definition 1.1. : let [To] be a collection of tool bits, To.[O], the ontologies for tool To,

Definition 1.2: Let [F] be a feature map extracted from [M] from an ROI from an image I,
Let [F] be the features defining a Taskoid Td,
We need to determine the machine genome and transcription for Td from [F].

Theorem 1.3: For Taskoid Td, if in the transcription of existing Taskoid genomes, there is an incompleteness, then:

1. Human intervention is called for.
2. An MFA II formulation of a target tool bit To or set [To] is designed using a tensor formulation.

Theorem 1.4: There exists a tensor representation of the topological ontologies, in a grid mapping,
->(Te, To.[O])
->(Te, Stl), where stl is an object file that can be additive printed. And Te is the machine genome representation, there exists a transcription from Te to Stl, with micro-adaptation in transcription models.

Te can be either [Templates] or a custom [Te] in a two level adaptability, in the micro-adaptability of templates or the creation of new template designs.

Theorem 1.5: Incompleteness of [S, F] as defined in 1.0
If there exists an incompleteness in [S, F] for Td, then there exists in [S, F], a function for an MFA II based evolution, defining ->(Te, Stl).
Incompleteness is defined by an exception thrown by theorem 1.3, needing the addition of new flex-rules as ontologies from a loop of Deep Learning.
From theorem 1.4 for To.[O] there exist one or more ontologies, as functional ontologies, indicating multi-functionality, and from To[O] can Te be derived through a polynomial time algorithm.

Discussion:

Dog-Ears with the RAVA language , extends the Reactive and object oriented systems with a formal framework for event-action relations with multi sensor integration, for a state machine and flex-rule based system, the same system is extended with rule learning and state machine design, for exception handling in applied MFA II based design of toolbits as an illustration of machine evolution of hardware design. The formulation in templates as machine genomic representation of taskoids, is a transcription mechanism for adaptation of template solutions or new templates for micro-adaptability.(Iba 2018b, [a] 2018; Lander, n.d.; Zhang 2003; Iba 2018c; Mirjalili, Faris, and Aljarah 2019)
A combination of deep learning and genetic algorithms differs from procedural algorithm based evolution as intentional evolution, with deep learning as genetic programming or genetic computing.
Here machine evolution as seen as procedural, rule based or deep learning based integrated code generation with a machine genome definition, analogous to a genomic basis of genetic based evolutionary computing.

Conclusions and Future Work.

We have thus defined a machine genome for intent evolutionary model of a multi functional computational design of a hardware element.
Future work would involve a case study in a vSLAM system, with a test bed of machine evolved tool-bit designs, with or without templates.

</original-contribution>

References.

Bheemaiah, Anil Kumar. n.d. “Formal Methods for Multi Modal UI for Robotics.” https://doi.org/10.31224/osf.io/7zdu3.

Iba, Hitoshi. 2018a. “Evolutionary Approach to Machine Learning and Deep Neural Networks.” https://doi.org/10.1007/978-981-13-0200-8.

———. 2018b. “Machine Learning Approach to Evolutionary Computation.” *Evolutionary Approach to Machine Learning and Deep Neural Networks*. https://doi.org/10.1007/978-981-13-0200-8_4.

———. 2018c. *Evolutionary Approach to Machine Learning and Deep Neural Networks: Neuro-Evolution and Gene Regulatory Networks*. Springer.

Ketkar, Nikhil. 2017. “Introduction to Tensorflow.” *Deep Learning with Python*. https://doi.org/10.1007/978-1-4842-2766-4_11.

Kuhle, Anneliese. 2018. *Tool Intelligence as an Explanation of Cross-Linguistic Variation and Family Resemblance: An Evolutionary and Typological Investigation*. Lexington Books.

Lander, Sean. n.d. “An Evolutionary Method for Training Autoencoders for Deep Learning Networks.” https://doi.org/10.32469/10355/44299.

Li, Juanzi, Ming Zhou, Guilin Qi, Ni Lao, Tong Ruan, and Jianfeng Du. 2018. *Knowledge Graph and Semantic Computing. Language, Knowledge, and Intelligence: Second China Conference, CCKS 2017, Chengdu, China, August 26–29, 2017, Revised Selected Papers*. Springer.

Mirjalili, Seyedali, Hossam Faris, and Ibrahim Aljarah. 2019. *Evolutionary Machine Learning Techniques: Algorithms and Applications*. Springer.

Pablos, Ordóñez de, and Patricia. 2012. *Advancing Information Management through Semantic Web Concepts and Ontologies*. IGI Global.

Russell, Edmund. n.d. “Intentional Evolution.” *Evolutionary History*. https://doi.org/10.1017/cbo9780511974267.008.

Shanahan, Murray. 2015. *The Technological Singularity*. MIT Press.

“Website.” n.d. Accessed August 29, 2019. Dr Bheemaiah, Anil Kumar. (2019). Taskoids: A Formal Definition (Version 1.0). http://doi.org/10.5281/zenodo.3376813.

Wikipedia, Source. 2013. *Biomorphic Robots: Beam Robotics, Humanoid Robots, Data, Android, Aibo, Pino, Gynoid, Rhex, Asimo, Robonaut, Actroid, Xianxingzhe, Geoff Peterson, Bat*. University-Press.org.

Zhang, Byoung-Tak. 2003. “A Unified Bayesian Framework for Evolutionary Learning and

Optimization." *Natural Computing Series*. https://doi.org/10.1007/978-3-642-18965-4_15.