Analog Computation for Next Generation AI

Dr. Hava Siegelmann
DARPA/MTO

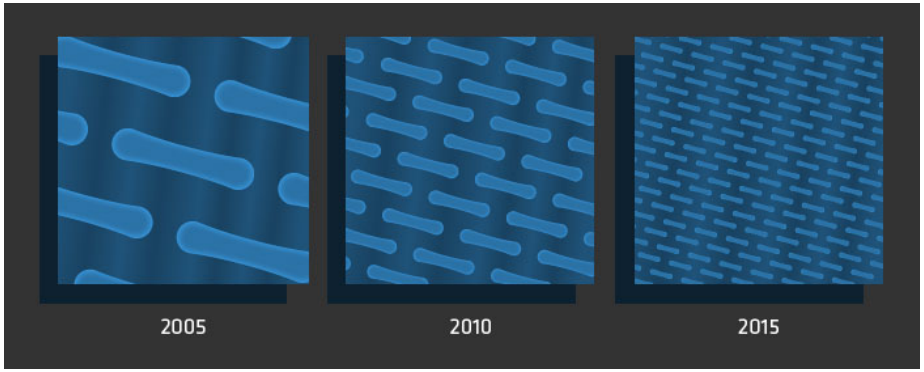
Introduction: Hardware via Gordon Moore

History: Moore's Law is the observation made by Intel co-founder Gordon Moore that the number of transistors on a chip doubles every year while the costs are halved.

(Transistors are simple electronic on/off switches embedded in microchips, processors and tiny electrical circuits. The faster microchips process electrical signals, the more efficient a computer becomes.)

Current: Number of transistors on silicon chips doubles every 18 months

Future: The exponential growth will come to an end with transistors. Other options: biotechnology, nanotechnology, and analog computation



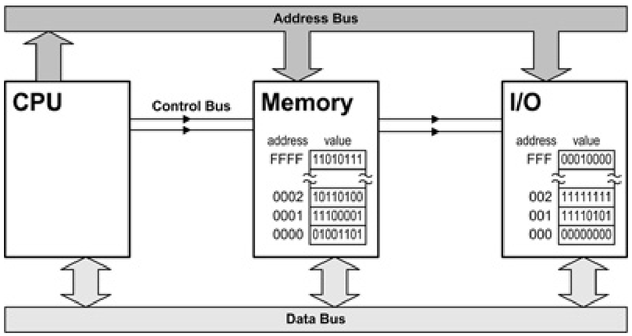
https://cns.utexas.edu/news/researchers-tackle-the-dark-side-of-moore-s-law

Why Analog Computation for AI?

- Current special-made hardware enable deep networks do fast integration
- **Low energy and fast:** Neuromorphic engineering & analog hardware provide very low-power and cheap solution to the simulation of neural networks
- **Better AI capability of prediction and motivation:** Any biological agent and good AI agent – has inside a "simulation of the world" from it, it calculates prediction, expectations, and reinforcement signals. Since the word is large and changing, any agent that has to live for long time will need either a humongous internal simulation of the world or analog and flexible internal representation of the world – like is believed to exist for animals.
- **Analog required for AI that improves all the time and that higher level of expressivity: Super Turing**.

State of the art: AI in 2011

IBM’s Watson 2011– AI top –



©Jeopardy

IBM's Watson computer wins Jeopardy matches

Huge clusters, databases, clever programming, some machine learning

But: Lacks flexibility, Must operate in orchestrated environment

AI recent successes

Gaming

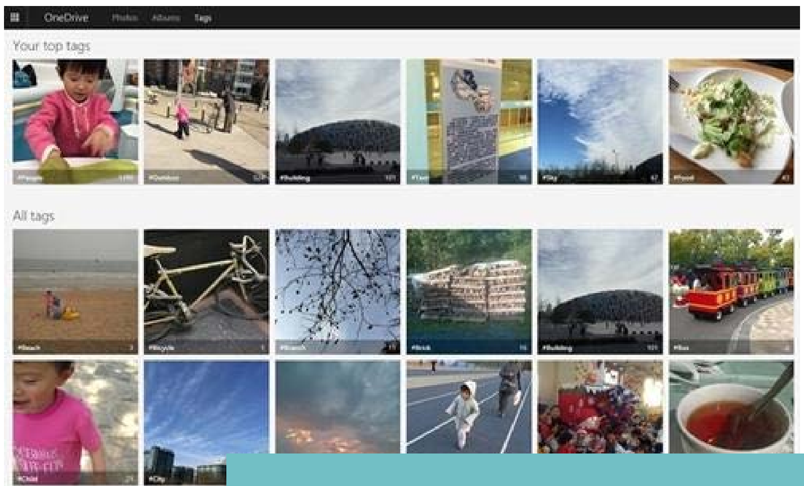
In 2017, Google DeepMind's AlphaGo AI compiled a 60-0 record against premier Go players

http://fortune.com/2017/01/07/google-alphago-ai/

Other researchers previously created AI systems that outperformed humans in chess, poker, Jeopardy, and Atari

Image recognition

In 2015, Microsoft outperformed humans on [ImageNet Large Scale Visual Recognition Challenge]



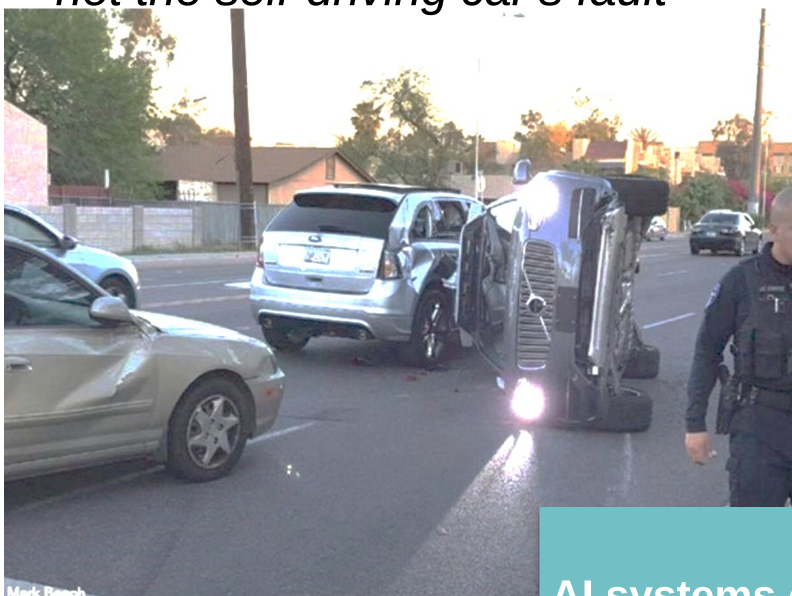
https://www.microsoft.com/en-us/research/blog/microsoft-researchers-algorithm-sets-imagenet-challenge-milestone/

Other researchers subsequently created AI systems that outperformed humans in image recognition

AI limitations

Uber's self-driving car

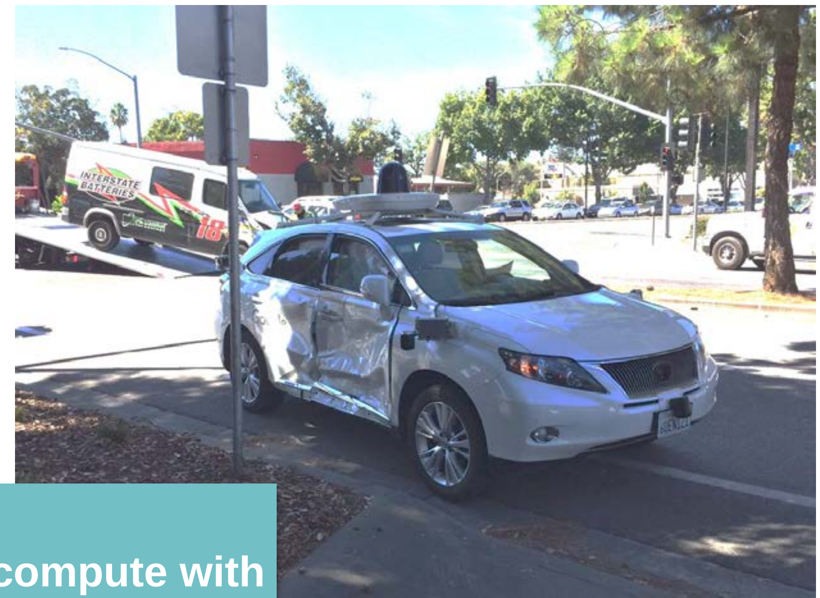
The 2017 collision was due to a human driver failing to yield—*not the self-driving car's fault*



http://www.abc15.com/news/region-southeast-valley/tempe/tempe-police-self-driving-uber-vehicle-involved-in-car-accident-no-injuries

Google's self-driving car

The 2016 collision was due to a human driver running a red light—*not the self-driving car's fault*



https://www.theguardian.com/technology/2016/sep/26/google-self-driving-car-in-broadside-collision-after-other-car-jumps-red-light-lexus-suv

AI systems only compute with what they've been programmed or trained for in advance

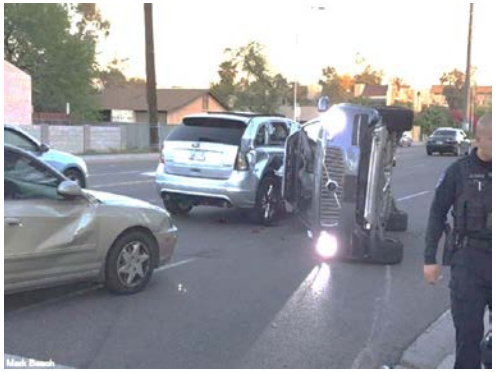
AI limitations

AI systems only compute with what they've been programmed or trained for in advance

The New York Times

"Tesla's cars could be relied upon to react properly in only some situations that arise on roadways."

https://www.nytimes.com/2017/01/19/business/tesla-model-s-autopilot-fatal-crash.html?_r=0



http://www.abc15.com/news/region-southeast-valley/tempe/tempe-police-self-driving-uber-vehicle-involved-in-car-accident-no-injuries

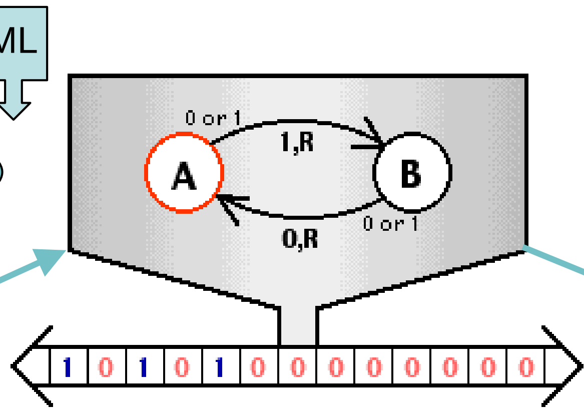
1. No way to prepare for any eventuality. No easy-fix of learning recent errors (catastrophic forgetting)
2. Malfunctions in circumstances beyond preparation
3. Worse with widespread applications

Today's computational foundation: Turing Machines

In 1936, Alan Turing modeled "human-calculators" as theoretical automatic machines

©blogspot.com

Loadable program



Input

Output

Memory tape

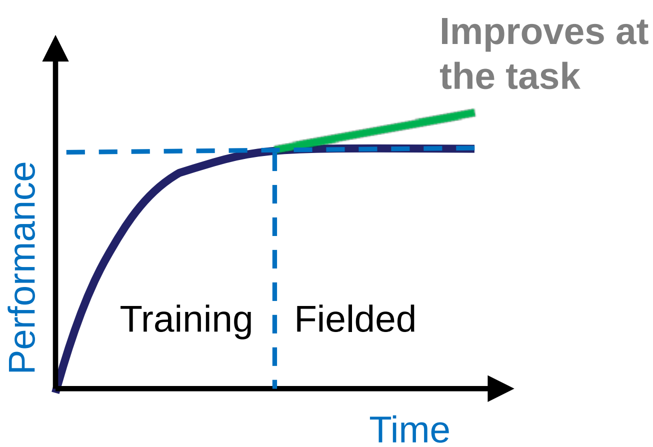
Current AI has two pre-execution parts

- **Program and rules**
- **Parameter learning (e.g., in ML)**

Next generation AI: Lifelong Learning Machines

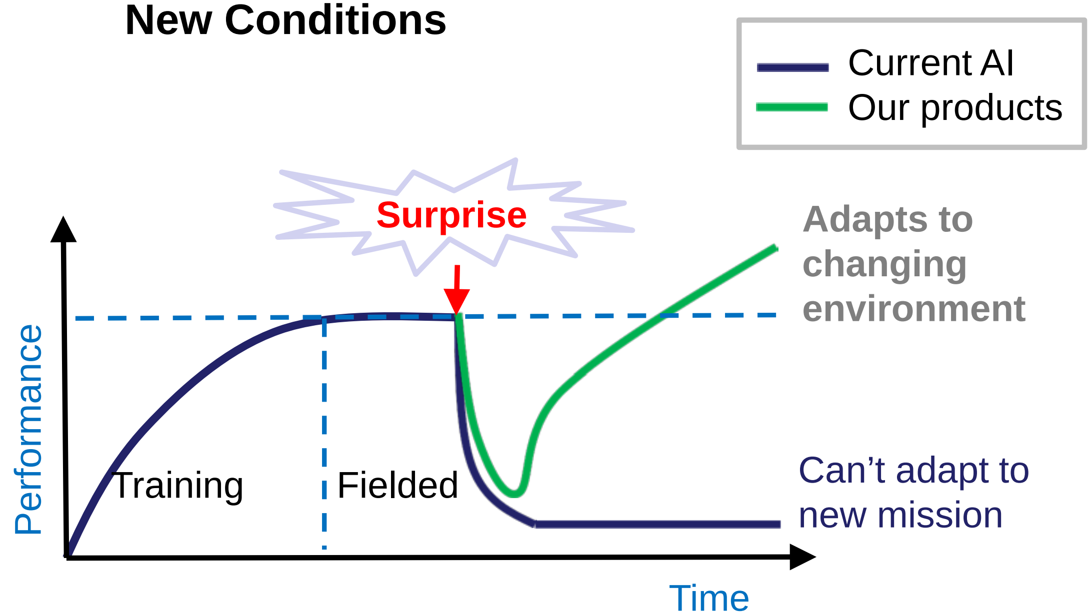
L2M is concerned with learning machines that will improve their performance over their lifetimes

Continuously Improve Performance



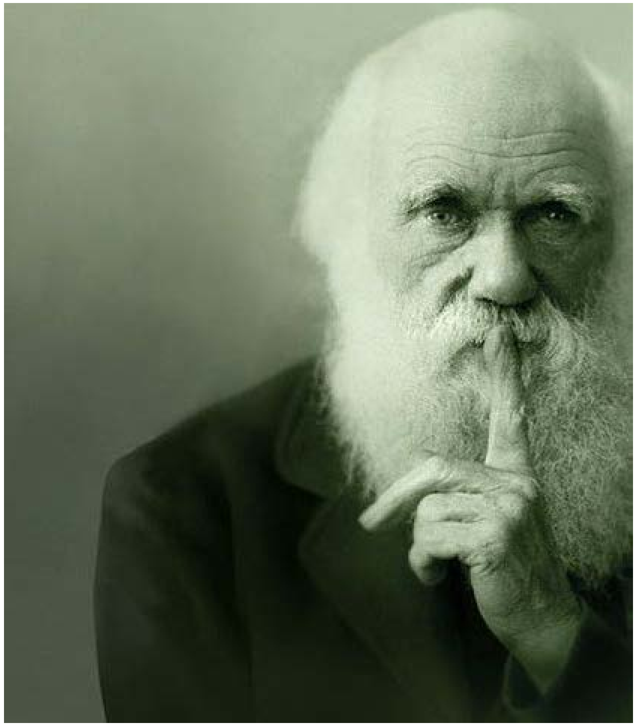
Current ML based on large datasets; data may be scarce

Adapt to New Conditions



Situation may change after training and fielding (external, internal)

Natural systems don't freeze at execution



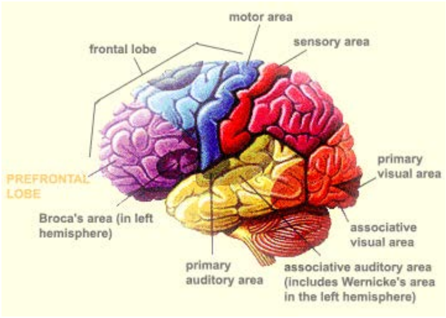
©onedio.co

"…it is not the strongest that survives; but…the one that is able best to adapt…to the changing environment…."

L.C. Megginson, re "On the Origin of Species"

Nature's mechanisms for change beyond preloaded programs

Brain reconsolidation

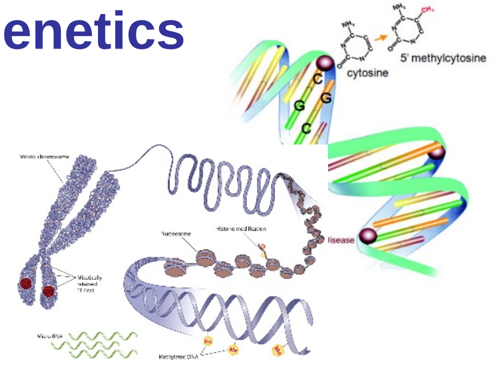


http://ind3.ccio.co/w3/B9/r1/b54462b246c38979b0288dffa0 0a186b.jpg?iw=300

Storage process for restoring retrieved memories, through which memories can be reinforced, faded, and modified toward new experiences

Effects
Temporal changes leading to adaptive behavior regulation

Epigenetics



http://2.bp.blogspot.com/EoTDSv8D_tc/U_eYScuZB_I/AAAAAAAAAE58/08xglTNvLh8/s1600/Mechanisms-of-epigenetics.jpg

Dynamic alterations in cell's transcription that affect how cells express genes based on external/environmental factors

Effects
Changes in the way of interpreting DNA, leading to adaptive organisms

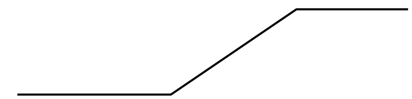
Analog Computation for Future AI

- Future AI requires changes and updates from experience – more data, better knowledge
- From computational point of view: lifelong learning machine requires analog computation
- Both are instantiation of a computational model called “Super Turing Computation”

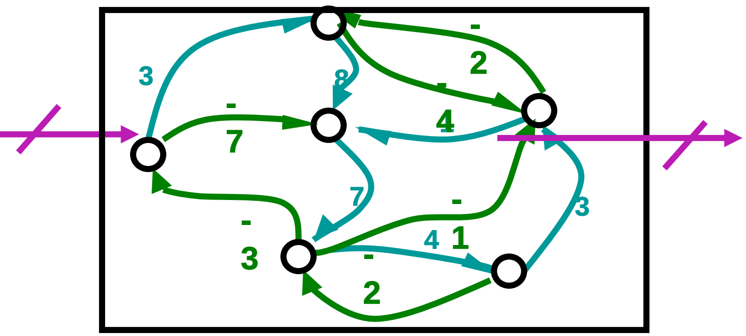
1991-3: at Rutgers University NJ

What is the computational power of Neural Networks?

$$\sigma(x) = \begin{cases} 0 & x \le 0 \\ x & 0 < x < 1 \\ 1 & x \ge 1 \end{cases}$$



Analog Recurrent Neural Net (ARNN)

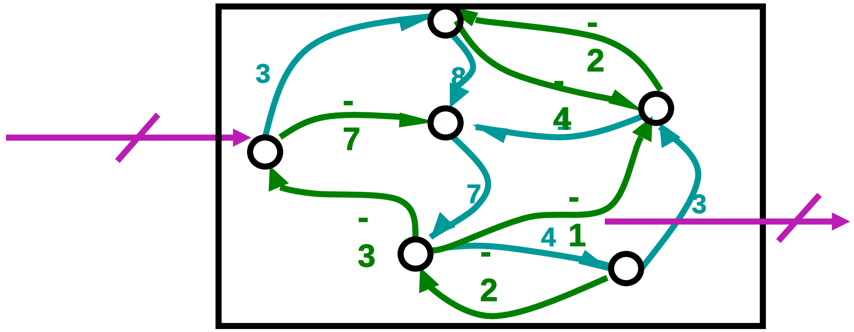


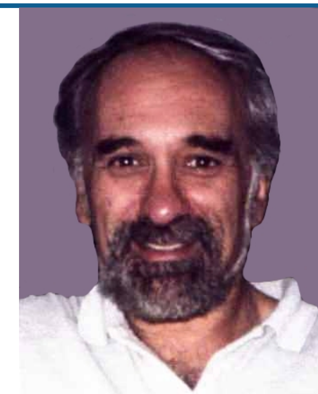
Findings:

- UTM (not Jordan’s conjecture)
- New class beyond UTM “Super Turing”

1995 Science, “Computation Beyond the Turing Limit” (math, dynamical systems), 1998 book… neuroscience

ST Technical Details

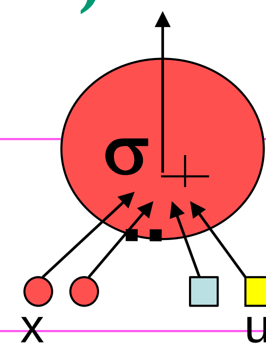




Eduardo Sontag

The Standard Model:

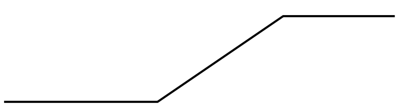
Analog Recurrent Neural Network (ARNN)



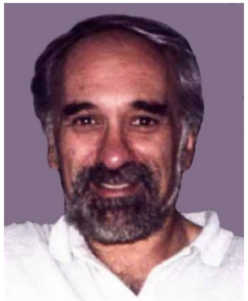
$$x_i(t+1) = \sigma\left(\sum_{j=1}^{N} a_{ij}\cdot x_j(t) + \sum_{j=1}^{M} b_{ij}\cdot u_j(t) + c_i\right)$$

: σ continuous, saturated-linear function:

$$\sigma(x) = \begin{cases} 0 & x \le 0 \\ x & 0 < x < 1 \\ 1 & x \ge 1 \end{cases}$$



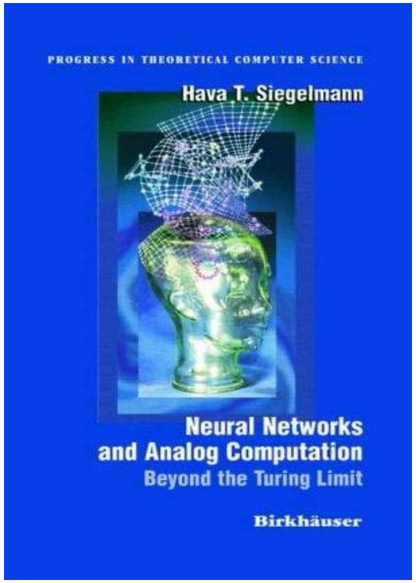
Hierarchy of ARNN
with Eduardo Sontag, Ricard Gavalda, Jose Balcazar

Weights	Recognition power	Polynomial time
Integers	Regular (finite automata)	
Rational (short numbers)	Recursive	P
Real (long: T bits)	Arbitrary	AnalogP

Infinite hierarchy between Rec and Arbitrary; P and AnalogP.

Super-Turing Continuum Hierarchy



Continuum of computational hierarchy. From Turing Machines (fixed programs) to Super-Turing Computation (modifiable programs).

http://1.bp.blogspot.com/-Vl3F-DL2Raw/T9wLn7ZiaVI/AAAAAAAAAsI/CtJfKSmLrk0/s1600

ST- Possible Ingredients

1. Analog values (Real)
2. Randomness/asynchronous
3. Lifelong Learning, evolving
4. Series of TM's

Neural networks (AnalogP)

$\alpha \in$ Kolmogorov$[f(n),g(n)]$: UTM calculates α[n-prefix] from $f(n)$ bits in $g(n)$ time P=K$[1,p(n)]$ AnalogP=K$[n,n]$

T-computation

1. Discrete (Q)
2. Deterministic
3. Pre-programmed

•Turing machines (P)

With Jérémie Cabessa 2012

Lifelong sequences: $\varphi_S : \{0,1\}^{\omega} \longrightarrow \{0,1\}^{\leq\omega}$

1 0 1 1 1 …

http://images.addictionblog.org/cherrycake/wp-content/uploads/2011/09/Does-marijuana-cause-brain-damage2.jpg

$2^{2^{\aleph_0}}$

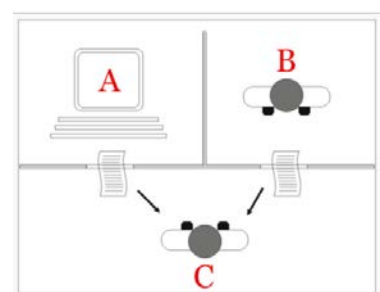
$\aleph_0$

$\aleph_0$

$2^{\aleph_0}$

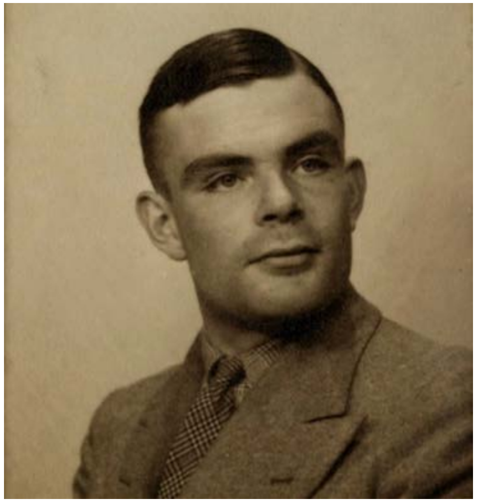
- Lifelong-TM (lifelong Turing) = Recursive continuous
- Lifelong-ARNN (Lifelong Super-Turing)

Turing-test 1950: separate $\aleph_0$ from human
ST-test : separate $2^{\aleph_0}$ from human



https://en.wikipedia.org/wiki/Turing_test

Turing on Intelligent Machines



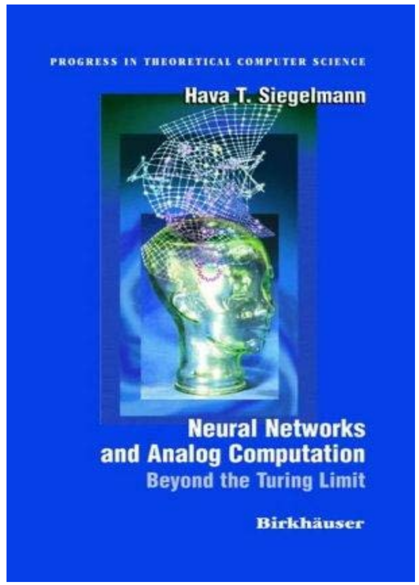
http://godsandfoolishgrandeur.blogspot.com/2013/10/alan-turing.html

"Electronic computers are intended to carry out any definite rule of thumb process which could have been done by a human operator working in a disciplined but unintelligent manner." ('50)

"My contention is that machines can be constructed that will simulate the behaviour of the human mind" ('51)

"What we want is a machine that can learn from experience" ('47)

Super-Turing Hierarchy



ST- Possible Ingredients

1. Analog values (Real)
2. Randomness/asynchronous
3. Lifelong Learning, evolving
4. Series of TM's

Neural networks (AnalogP)

T-computation

1. Discrete (Q)
2. Deterministic
3. Pre-programmed

•Turing machines (P)

When searching for human-like intelligence:

http://godsandfoolishgrandeur.blogspot.com/2013/10/alan-turing.html

Rich information: “This would occur if for instance the digits of the number π were used to determine the choices of the machine.”

Randomness: “a machine which is to imitate a brain must appear to behave as if it had free will, … something like a roulette wheel or a supply of radium.”

Lifelong Learning: **“**If the machine is treated only as a domestic pet, and is spoon fed with particular problems, it will not be able to learn in the varying way in which human beings learn.”

Series of machines: *“By choosing a suitable machine one can approximate the truth”*

Frequently asked: Lemma: linear precision suffices

Definition: k-Truncated(ARNN) is like ARNN

$$x_i(t+1) = \sigma\left(\sum_{j=1}^{N} a_{ij} \cdot x_j(t) + \sum_{j=1}^{M} b_{ij} \cdot u_j(t) + c_i\right) \qquad \sigma(x) = \begin{cases} 0 & \text{if } x < 0, \\ x & \text{if } 0 \leq x \leq 1, \\ 1 & \text{if } x > 1. \end{cases}$$

but with weights and neural values truncated after O(k) bits

Lemma: In up to T steps of computation, ARNN and k-Truncated(ARNN) are indistinguishable ➔ only the first k bits matter and the rest can be wrong

What makes computation super-Turing is not the precision but the capability to incorporate information beyond what's pre-prepared

Proving Linear Precision Suffices

We calculate the error between the output of a net (f) and its t-truncated net

Assume N nodes, M input lines,

$$\begin{aligned} |\tilde{x}_i(t) - x_i(t)| &\leq \epsilon_t \\ |\tilde{a}_{ij}(t) - a_{ij}(t)| &\leq \delta_w \\ |\tilde{b}_{ij}(t) - b_{ij}(t)| &\leq \delta_w \text{ , and} \\ |\tilde{c}_i(t) - c_i(t)| &\leq \delta_w \text{ .} \end{aligned}$$

Using the global Lipschitz property $|\sigma(a) - \sigma(b)| \leq |a - b|$, it follows that

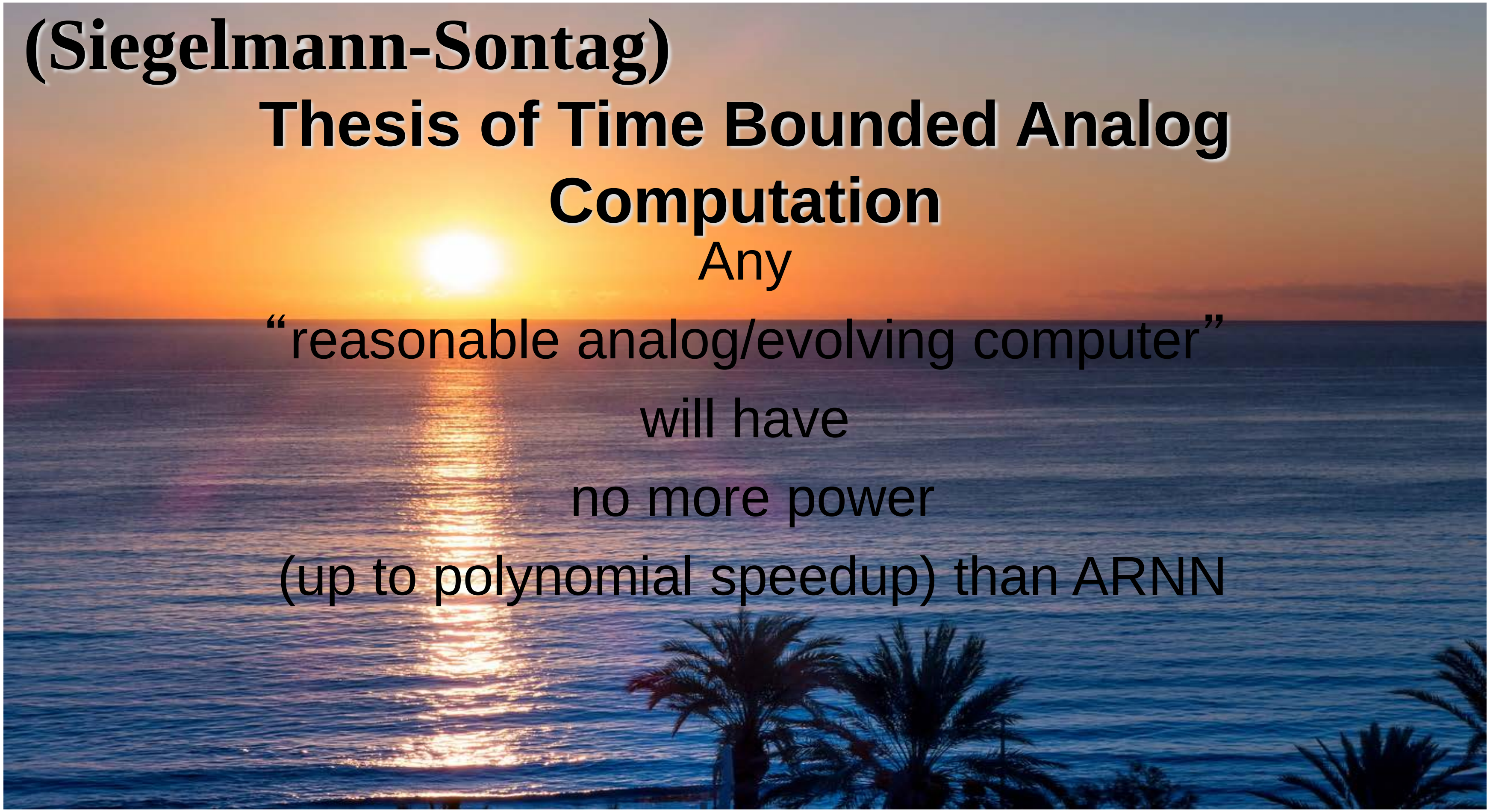
$$\epsilon_t \leq N(W' + \delta_w)\epsilon_{t-1} + (N + M + 1)\delta_w + \delta_p \leq LW\epsilon_{t-1} + L\delta_w + \delta_p \text{ .}$$

Therefore,

$$\epsilon_t \leq \sum_{i=0}^{t-1} (LW)^i (L\delta_w + \delta_p) \leq (LW)^t (L\delta_w + \delta_p) \text{ .}$$

We require similar output: $f \leq 0 \Longrightarrow \tilde{f} < \frac{1}{4}$ and $f \geq 1 \Longrightarrow \tilde{f} > \frac{3}{4}$. i.e., $\epsilon_t < \frac{1}{4}$

That is, sufficient truncation for computation is $O(t \log(LW))$ = $O(t)$



(Siegelmann-Sontag)

Thesis of Time Bounded Analog Computation

Any

“reasonable analog/evolving computer”

will have

no more power

(up to polynomial speedup) than ARNN

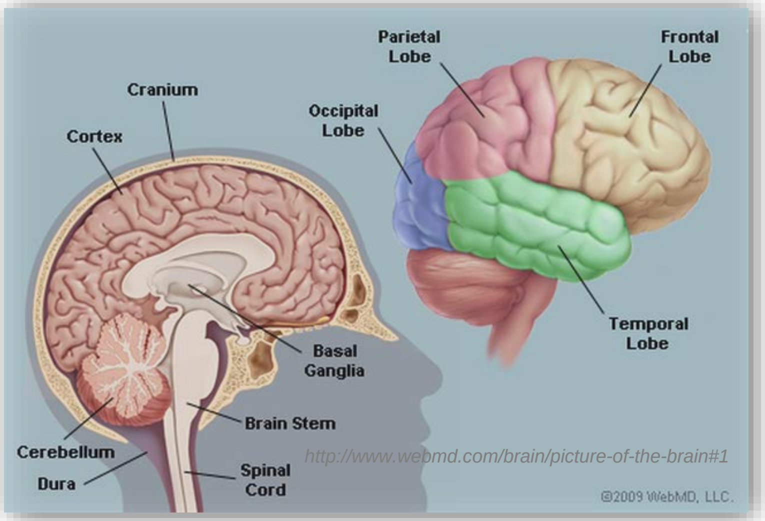
https://pixabay.com/en/sunrise-sunset-sea-afterglow-sun-1008670/

Nature combines Turing with Super-Turing Computation

- Turing machines change output based on input
- Super-Turing machines change program based on inputs

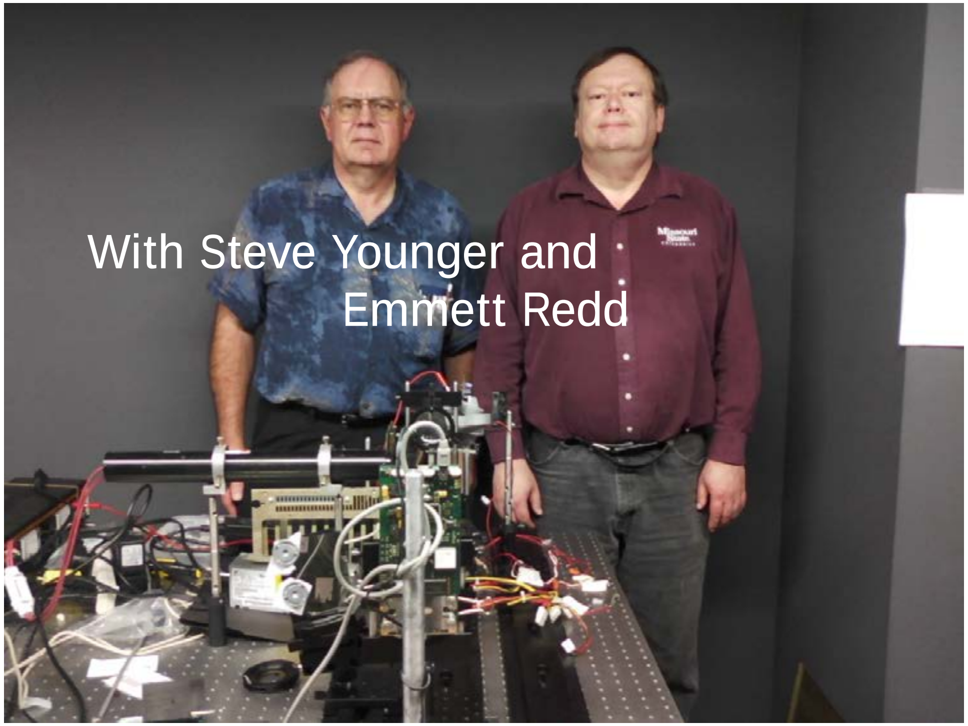
- Nature systems follow (Turing like) programs
- They adapt as needed, changing their Turing programs
- They store revised Turing programs as components for future use

- Some mechanisms seen in brain:
  1. Chemicals associate neurons by context (creating submodules)
  2. Dual memory system
  3. Separate self from tasks
  4. Abstraction due to structure



Taking theory into practice

Physicists and Engineers (NSF – Werbos): building first Super Turing prototype



"Oh yes, at least 100 years, I should say." [1952]

"The nervous system is certainly not a discrete-state Machine.." [1952] Today's technology ?

DARPA: Lifelong Learning Machines (since 2017)

Today

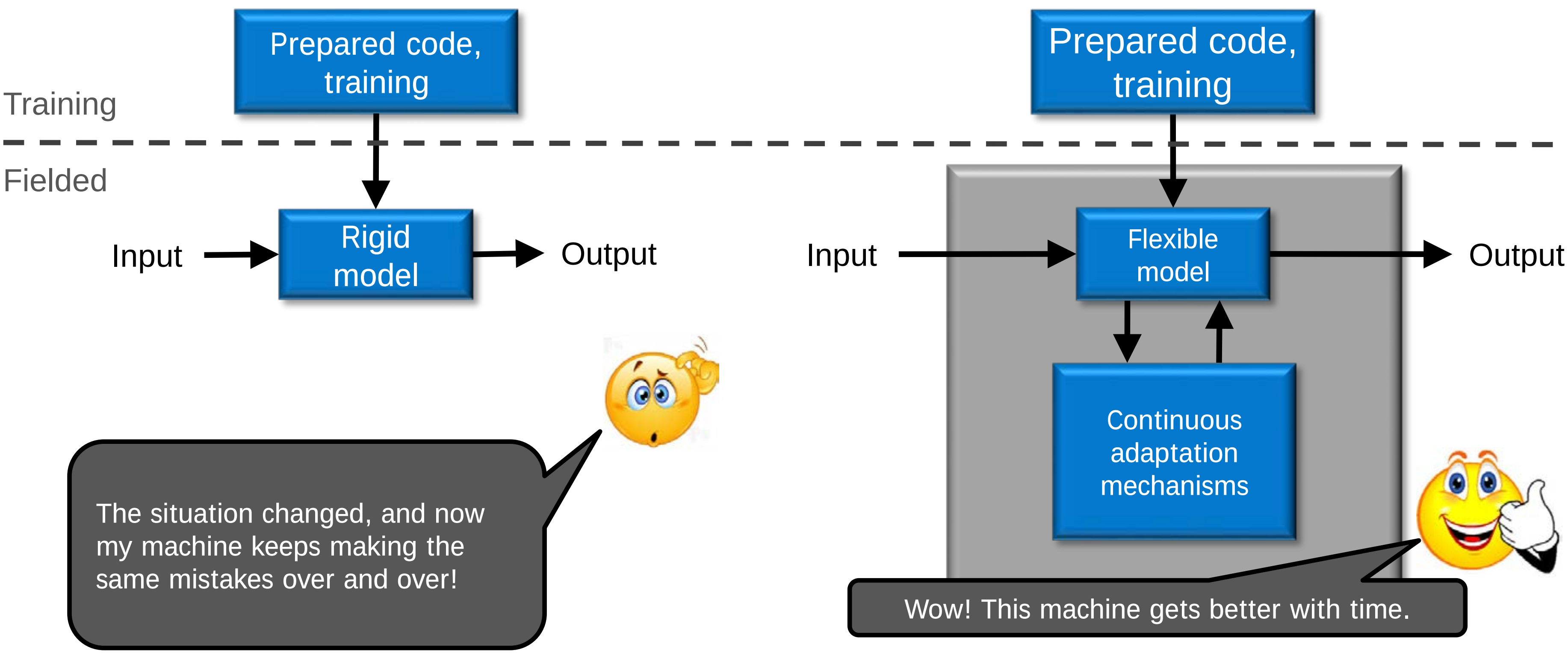
Knowledge distilled from expert and/or training on examples (batch or online)

- Execution follows completed training cycle
- Execution is fixed

In L2M

Basic information (like safety rules, instincts and needs) are put in advance

- Continues learning during execution
- Can adapt program to new situations



Summary: Lifelong Learning Machines

- **Lifelong Learning AI:** The ability of a computational system to learn in real time, and apply previous learning to new situations
- **Examples:**
  - A car that becomes better on snowy roads each time it drives on them (becomes an expert)
  - A plane that learns to fly more efficiently and safely
- **Current situation**: No extant systems with true learning. Current systems are a combination of preprogramming and off-line training
- **Weakness of current systems**: Inability to handle new circumstances without making an error or halting

https://www.youtube.com/watch?v=76XXev8R6YY

https://lh3.googleusercoAtent.com/XuSIt4CPzTb9FcOJqYe4i31M gd3TyzC5JJlash37fZtWPjr7mUNW1WEmAeHebIWDZ_OM=s147

http://www.jabil.com/technologies/heavy-fuel-engines-for-unmanned-aerial-vehicles/

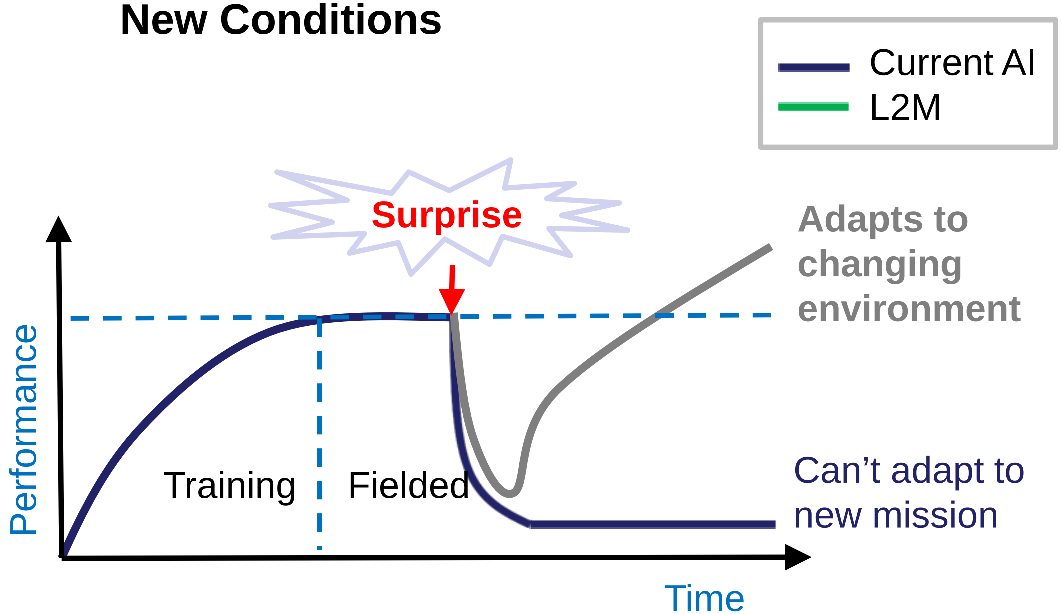
Lifelong Learning Machines (L2M) program objective

L2M will develop fundamentally new machine learning mechanisms that will enable systems to improve their performance over their lifetimes

- Highly competitive
- Please contact me with great ideas
- I'm interested in collaborations

Hava.siegelmann@darpa.mil
(hava@cs.umass.edu)

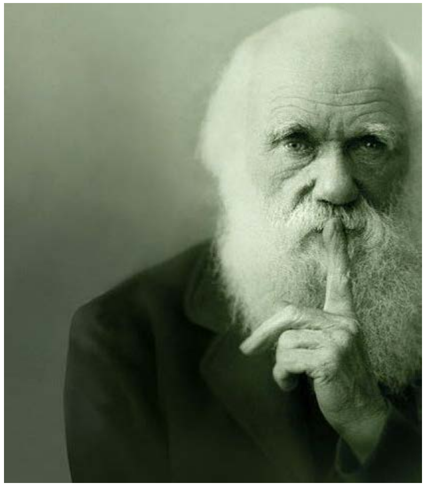
Adapt to New Conditions

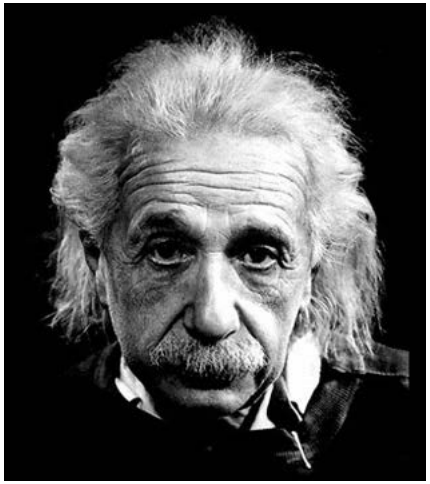


Situation may change after training and fielding (internal and external changes)

"…it is not the strongest that survives; but…the one that is able best to adapt…to the changing environment…."

L.C. Megginson, re "On the Origin of Species"





https://www.izlesene.com/iz/memcn3342

"Once you stop learning, you start dying."

Albert Einstein

Thank you

www.darpa.mil