RL&AI

Reinforcement Learning and Artificial Intelligence

rlai.net

The RL&AI group at the Univ. of Alberta in 2011

Principal investigators:

Rich Sutton

Michael Bowling

Csaba Szepesvari

Dale Schuurmans

Patrick Pilarski

et al.

The Future of Artificial Intelligence Belongs to Search and Learning

Rich Sutton

Reinforcement Learning and Artificial Intelligence Laboratory
Department of Computing Science
University of Alberta, Canada

Outline:
Understanding AI in the…

- Present
    - Success, excitement, and fear
    - Moore’s law (generalized) drives it all
- Past
    - The impact of Moore’s law can be seen throughout the history of AI
    - The longest trend: Scalable methods are initially disfavoured, but eventually win
- Future
    - A key remaining challenge: Knowledge (of the world’s state & dynamics)
    - How can we make knowledge scalable with Moore’s law?

Advances in AI abilities are coming faster; in the last 5 years:

- IBM’s Watson beats the best human players of *Jeopardy!* (2011)
- Deep neural networks greatly improve the state of the art in speech recognition and computer vision (2012–)
- Google’s self-driving car becomes a plausible reality (≈2013)
- Deepmind’s DQN learns to play Atari games at the human level, from pixels, with no game-specific knowledge (≈2014, *Nature*)
- Univ of Alberta’s Cepheus solves Poker (2015, *Science*)
- Deepmind’s AlphaGo defeats the European Go champion 5-0, vastly improving over all previous programs (2016, *Nature*)

Corporate investment in AI is way up

- Google's prescient AI buying spree: Boston Dynamics, Nest, Deepmind Technologies, …

- New AI research labs at Facebook (Yann LeCun), Baidu (Andrew Ng), Allen Institute (Oren Etzioni), Maluuba…

- Also enlarged corporate AI labs: Microsoft, Amazon, Adobe…

- Yahoo makes major investment in CMU machine learning department

- Many new AI startups getting venture capital

Why are these things happening now?

Is it because of big progress in AI algorithms?

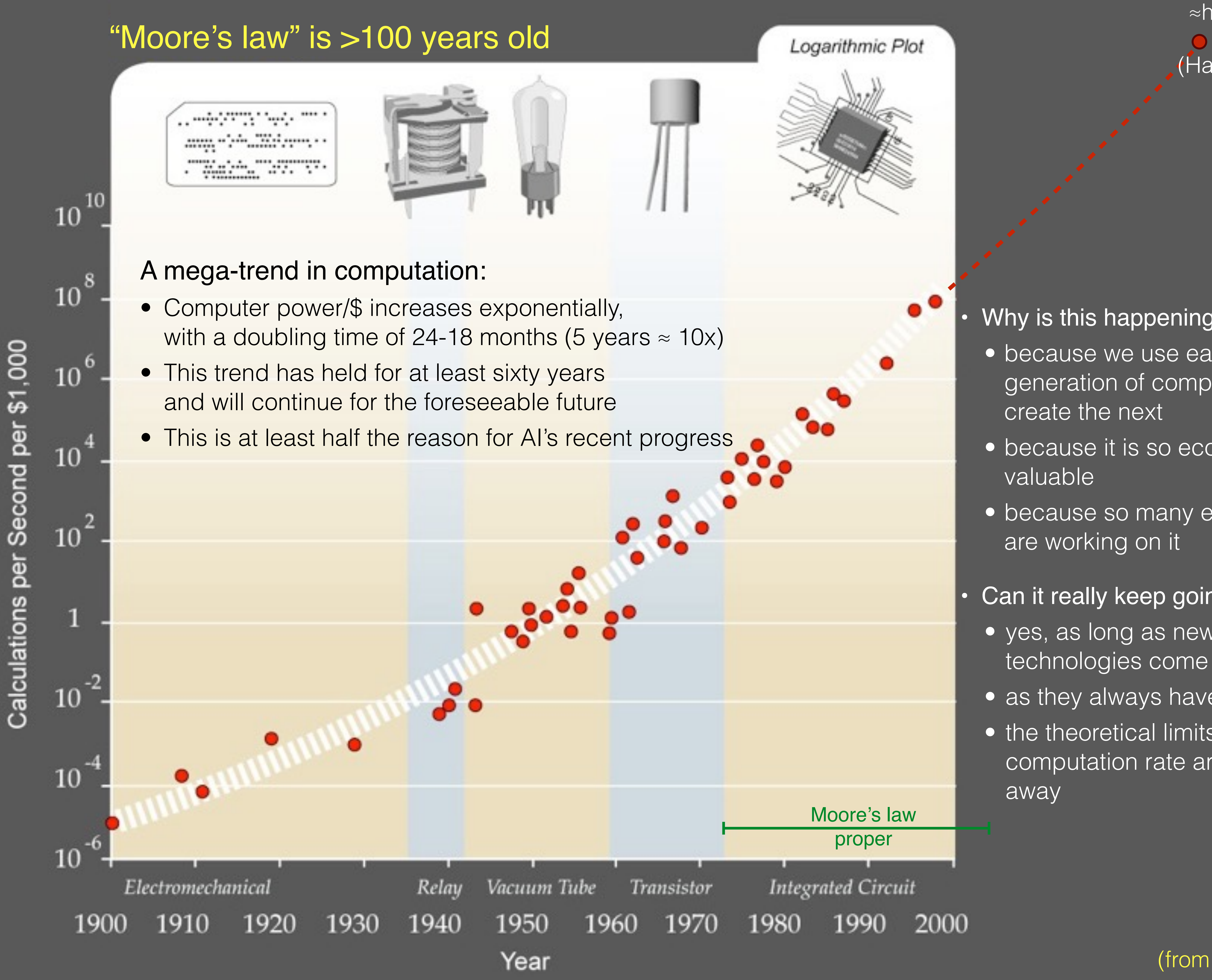
Or…

Moore’s Law

The long-term exponential improvement in computer hardware

Moore’s law proper: The number of transistors that can be placed inexpensively on an integrated circuit doubles approximately every two years

“Moore’s law” is >100 years old



≈human brain in ≈2030 (Hans Moravec)

Moore’s law proper

A mega-trend in computation:

- Computer power/$ increases exponentially, with a doubling time of 24-18 months (5 years ≈ 10x)
- This trend has held for at least sixty years and will continue for the foreseeable future
- This is at least half the reason for AI’s recent progress

- Why is this happening?
  - because we use each generation of computers to create the next
  - because it is so economically valuable
  - because so many engineers are working on it
- Can it really keep going?
  - yes, as long as new technologies come along
  - as they always have in the past
  - the theoretical limits to computation rate are still far away

(from Kurzweil AI)

Fear of AI is also up

- Many people fear the success of AI, that it may be unsafe and threaten humanity
- One fear is that AIs will be much smarter than us
    - Nick Bostrom, author of “Superintelligence: Paths, Dangers, Strategies,” worries that the first strong AI might take over and cause an “existential catastrophe”
    - Elon Musk — “[Strong AI would be] releasing the demon” “our greatest existential threat” “there should be some regulatory oversight” “I think there is potentially a dangerous outcome there”
    - Stephen Hawking — “The development of full artificial intelligence could spell the end of the human race” “It would take off on its own, and re-design itself at an ever increasing rate” “world militaries are considering autonomous-weapon systems that can choose and eliminate targets” “humans, limited by slow biological evolution, couldn’t compete and would be superseded by AI”
- AI researchers are sometimes too dismissive of these fears
    - Andrew Ng compares worrying about strong AI to worrying about over-population on Mars
    - Geoff Hinton says that if strong AI does ever happen it won’t be for a long while

My view

- Understanding human-level AI will be a profound scientific achievement (and economic boon) which may well happen by 2030 (25% chance) or by 2040 (50% chance) — or never (10% chance)

- It will bring great changes! We should certainly prepare ourselves

- But the fear is overblown, unhelpful, misplaced, and poorly expressed

  - One big fear is that strong AIs will escape our control;
    this is likely, but not to be feared

  - AI will arrive much slower than feared, at the rate of Moore's law

  - The greatest risks come not from AI as much as from the people who would misuse it; this is a pre-existing, ongoing problem with our societies

    - The problems that need solving are not primarily technical or mathematical, but societal in conclusion, about the present:

AI is not like other sciences

- AI has Moore’s law, an enabling technology racing alongside it, making the present special

- Moore’s law is a slow fuse, leading to the greatest scientific prize of all time

- So slow, so inevitable, yet so uncertain in timing

- The present is a special time for humanity, as we prepare for, wait for, and strive to create strong AI

Outline:
Understanding AI in the…

- Present
  - Success, excitement, and fear
  - Moore’s law (generalized) drives it all
- Past
  - The impact of Moore’s law can be seen throughout the history of AI
  - The longest trend: Scalable methods are initially disfavoured, but eventually win
- Future
  - A key remaining challenge: Knowledge (of the world’s state & dynamics)
  - How can we make knowledge scalable with Moore’s law?

3 waves of neural networks

- First explored in the 1950-60s: Perceptron, Adaline…
  - only one learnable layer
- Revived in the 1980-90s as Connectionism, Neural Networks
  - exciting multi-layer learning using backpropagation (SGD); many successful applications; remained popular in engineering
- Revived again in ~2010 as Deep Learning
  - dramatically improved over state-of-the-art in speech recognition and visual object recognition, *transforming these fields*
  - the best algorithms were essentially the same as in the 1980s, except with faster computers and larger training sets

i.e., NNs won (eventually) because their performance scaled with Moore’s law, whereas competing methods did not

We have seen this story before

- In chess

  we thought human ideas were key, but it turned out (deep Blue 1997)
  that big, efficient, heuristic search was key

- In computer Go

  we thought human ideas were key, but it turned out (MCTS 2006–)
  that big, sample-based search was key

- In natural language processing

  we thought that human-written rules were key, but it turned out (~1988)
  that statistical machine learning and big data were key

- In visual object recognition

  we thought human ideas were key, but it turned out (deep learning 2012–)
  that big data sets, many parameters, and long training was key

Scalable methods

- A method is *scalable with the computational mega-trend (Moore's law)* to the extent that its performance improves roughly in proportion to the quantity of computation it is given

- Scalable means you can take advantage of (use effectively) an arbitrarily large amount of computation (e.g., learning, search)

- A method is not scalable if the improvement it gives is not much affected by the computation available (e.g., the opening book in chess)

- *Search* and *learning* are scalable; prior knowledge, human assistance, and taking advantage of special-case structure are not

- By definition then, scalable methods improve automatically with time; they tend to be disfavoured initially but perform better in the end

- This is a pattern that can be seen over and over in the history of AI

Scalability is the key, but it tends to be correlated with other issues

- Symbolic vs statistical,
  hand-crafted vs learned,
  domain-specific vs general-purpose
  - Symbolic, hand-crafted, and domain-specific methods all rely more on human understanding and participation in their design; they begin non-scalable and tend to stay that way
  - Over the history of AI, statistical, learned, and general-purpose methods have steadily increased in relative importance
- In the early days of AI (pre-1980), a similar distinction was made between "strong" methods (powered by human input) and "weak" methods (relying on general principles)
  - The terminology is telling; the founding fathers favorite methods failed to scale and have fallen from favor; the weak have inherited AI

The choice is always before the AI researcher: To work on what scales, or what does not?

- We almost always reach for what does not scale
- It is usually easier, less abstract, and quicker to payoff
  - Improving a scalable method may bring little payoff for years
- But Moore’s law is progressing; with each further doubling in computation the relative advantage of scalable methods increases and becomes more quickly visible
- If you want to have a long-term impact, you should work on methods that scale with computation; timing is important

in conclusion, about the past

The longest trend in AI, and the biggest lesson, is that in the long run scalable methods always win

- Scalable = they can use arbitrary quantities of computation,
  and their performance improves proportionately

- Learning and search are scalable
  to the extent that they reduce dependence on people

Outline: Understanding AI in the…

- Present
  - Success, excitement, and fear
  - Moore’s law (generalized) drives it all
- Past
  - The impact of Moore’s law can be seen throughout the history of AI
  - The longest trend: Scalable methods are initially disfavoured, but eventually win
- Future
  - A key remaining challenge: Knowledge (of the world’s state & dynamics)
  - How can we make knowledge scalable with Moore’s law?

How scalable is supervised learning?

- Classically, supervised learning involves a training set of examples of desired behavior
- The learning and guessing processes has been greatly scaled with neural networks
- But scalability is limited by the training set
  - Training sets grow somewhat with Moore's law
  - But typically must be provided by people

Not so much

How scalable is reinforcement learning?

- In classic, model-free RL, we learn a policy (a mapping from states to actions) and a value function (a mapping from states to future reward)
  - these can be learned by trial and error, by trying actions and seeing what rewards follow
  - no labels are required (good for scaling)
    and it is computationally cheap (good? or bad?)
- If experience is plentiful (e.g., self-play) then RL scales beautifully
- But in the classic, model-free case, you do just a small computation per time step, and then there is nothing much else to do; there is little scaling (the policy and value mappings can be made more complex)

Not so much

The most important advance in machine learning over the next 12 months will be…

- The ability to **learn at scale from ordinary experience**
  - from interaction with the world
  - without the need for a training set of labeled data
  - in a more naturalistic way, like how a child or animal learns
  - about how the world works, about cause and effect
- Enabling machine learning to scale to the next level
- Using *deep reinforcement learning* for *long-term prediction* (probably) and/or *unsupervised learning*

The grand challenge of knowledge

- By knowledge I mean empirical knowledge of the world
  - Analogous to the laws of physics,
    to knowing how the pieces move in chess,
    to knowing what causes what,
    to being able to predict what will happen next for various actions
- The knowledge must be
  - Expressive: able to represent all the important things, including abstractions like objects, space, people, and extended actions
  - Learnable: from data without labels or supervision (for scalability)
  - Suitable for supporting planning/reasoning
- There is a substantial body of technical machinery for this in RL (options, PSRs, TD nets); but I will just sketch the challenge and its scalability

Examples of stuff to know

- Twitching this muscle lifts that finger
- There is a wall behind me
- The toilet is down the hall on the left
- The shape of a teacup
- Knowing how to ride a bike
- Knowing how to call a taxi

- My keys are in my pocket
- There is an apple in the box
- There is a book on the table
- My car is red
- People usually have two feet
- The Eiffel tower is in Paris
- John has the flu

The Sensorimotor View

- In which an agent’s knowledge is viewed as facts about the statistics of its sensorimotor data stream
- This point of view is interesting because
  - it is reductionist and demystifies world knowledge
  - it provides a clear way of thinking about semantics
  - it implies that knowledge can be verified and learned from data – “the knowledge is in the data”

It's hard to implement the Sensorimotor View well

- Where "well" means such that it is
  - sound, stable, and efficient with function approximation
  - scalable to large numbers of predictions learned in parallel from the same experience
  - real time (online with many updates/second)
  - captures multi-step facts
- Achieving these modest goals is highly constraining

Strategy

- To understand the world is to have *many predictions* about your sensorimotor data stream
- The predictions must be multi-step and policy contingent
  - because almost all interesting predictions are more-than-one-step and policy-contingent
- You must be able to learn from partial executions
  - because then you can learn about *many policies in parallel*
  - this will require *TD* and *off-policy* learning, and *FA*

Skilled perception and action…learned without labels

The knowledge is about the data; The knowledge is in the data

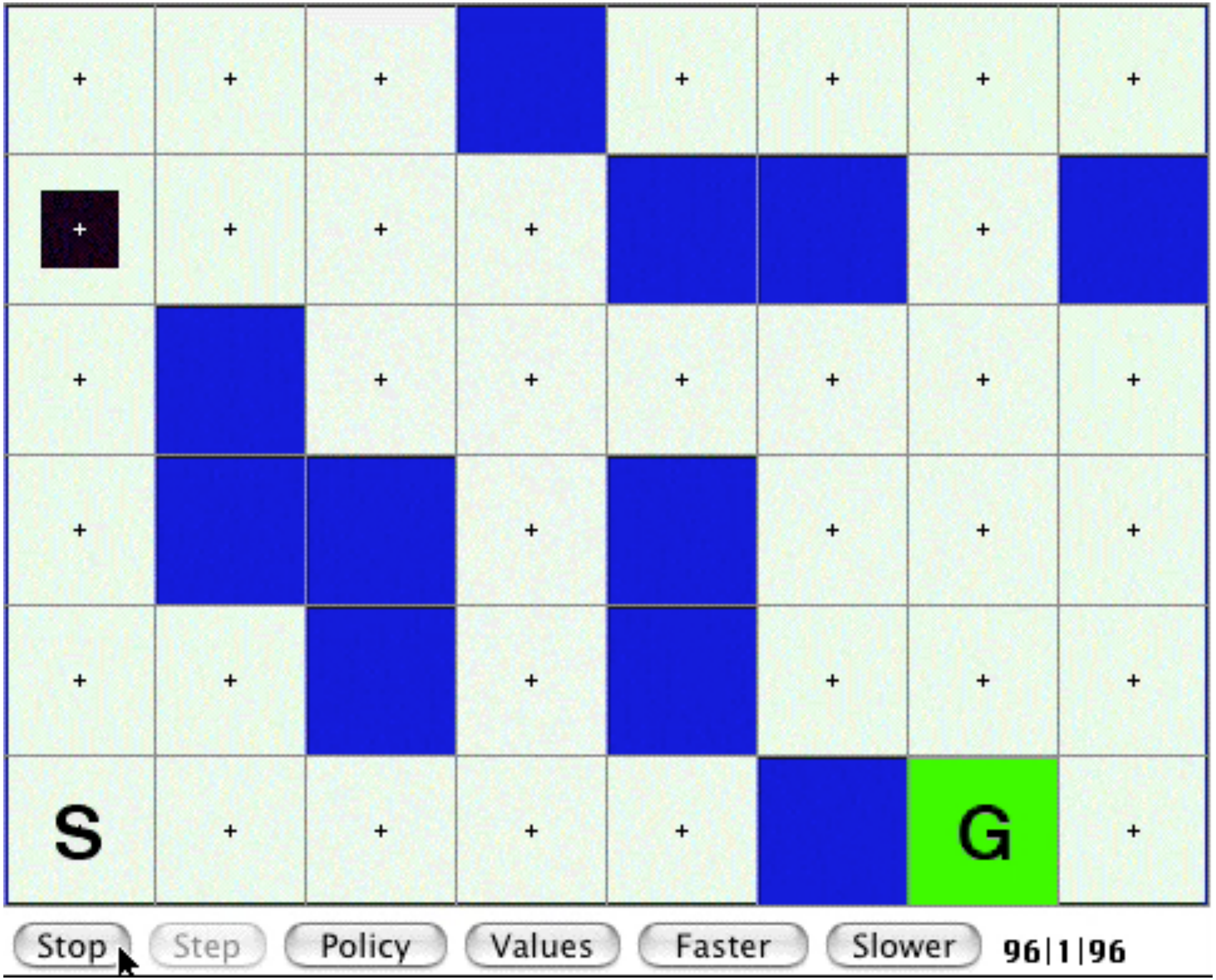
- It is there
    - that’s why people can know it, and provide labels
- We just have to find a way to learn about it
    - even though it is all very abstract

a view of (empirical) knowledge:

Knowledge is about the world's state and dynamics

- State is a summary of the agent's past that it uses to predict its future
- To have state knowledge is to have a *good summary*, one that enables the predictions to be accurate
- The predictions themselves are the dynamics knowledge
- The most important things to predict are *states* and *rewards*, which of course depend on what the agent does
  - if these are predicted in the right way, then the predictions can be used as a model of the world to support planning (the analog of self-play and reasoning)
- How can such knowledge be learned, represented, and used in a scalable way?

Model-based RL: GridWorld Example



The one-step trap:

Thinking that one-step predictions are sufficient

- That is, at each step predict the state and observation *one step later*
- Any long-term prediction can then be made by simulation
- In theory this works, but not in practice
  - Making long-term predictions by simulation is exponentially complex
  - and amplifies even small errors in the one-step predictions
- Falling into this trap is very common: POMDPs, Bayesians, control theory, compression enthusiasts

Can’t we just use our familiar one-step learning methods?

- Can’t we just wait until the target is known, then use a one-step method? (reduce to input-output pairs)
    - E.g., wait until the end of the game, then regress to the outcome
- No, not really; there are significant computational costs to this
    - memory is O(span)
    - computation is poorly distributed over time
- These can be avoided with learning methods specialized for multi-step
- Also, sometimes the target is never known (off-policy)
- We should not ignore these things; they are not nuisances, they are <u>clues</u>, hints from nature

AI is not an information problem; it's a computation problem

- Both data and computer power tend to increase exponentially over time

- But the computer power needed to process the data fully (to produce an exact solution) is exponential in the data

- Thus, AI is not about making the best use of limited *information* (e.g., Bayesian)

- But about making the best use of limited *computation*

    - about making the best use of *massive but insufficient* computational resources to find *approximate* solutions

Predicting right and left bumps conditional on going forward

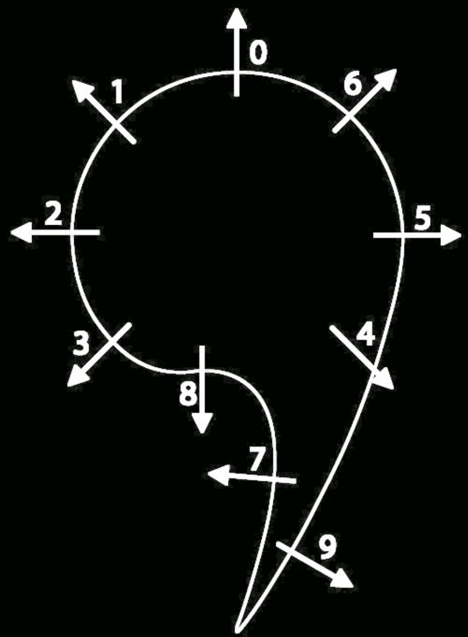
pred data

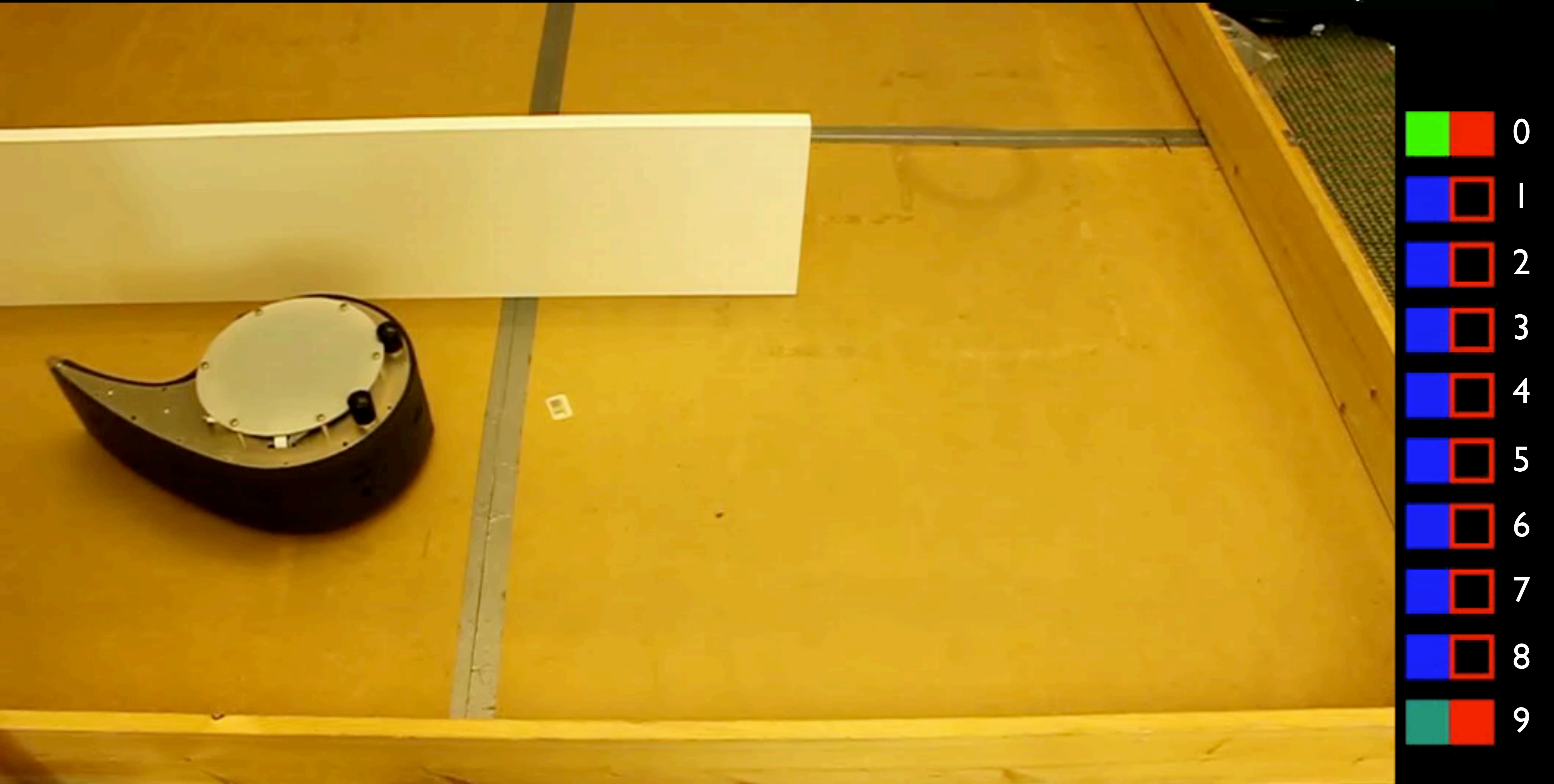
🟩 □
left bump

🟩 □
both bump

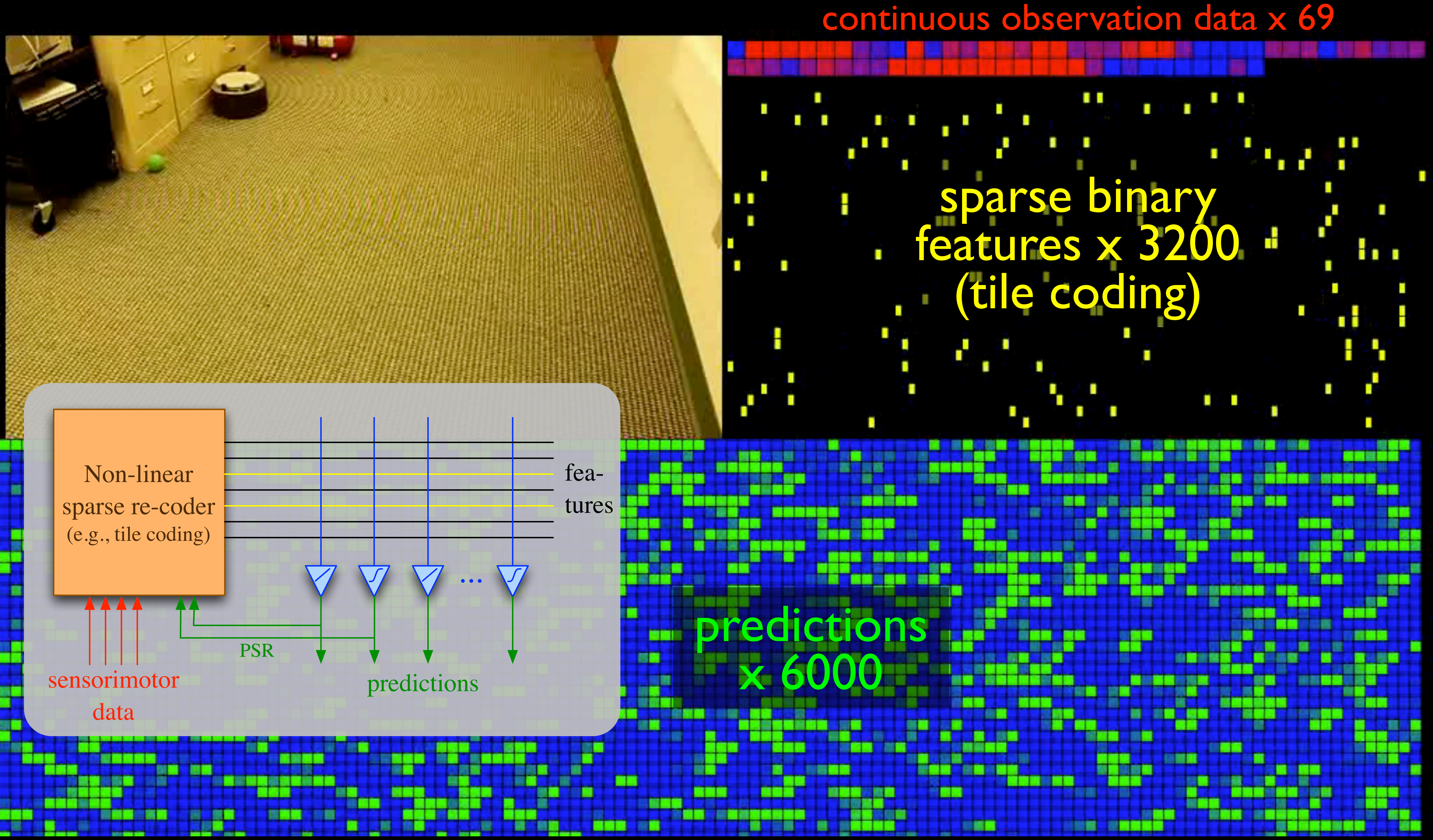
🟩 □
right bump

Predicting 10 infrared proximity signals





Massive real-time prediction learning
Up to one billion weight updates/second



continuous observation data x 69

sparse binary features x 3200 (tile coding)

predictions x 6000

Non-linear sparse re-coder (e.g., tile coding)

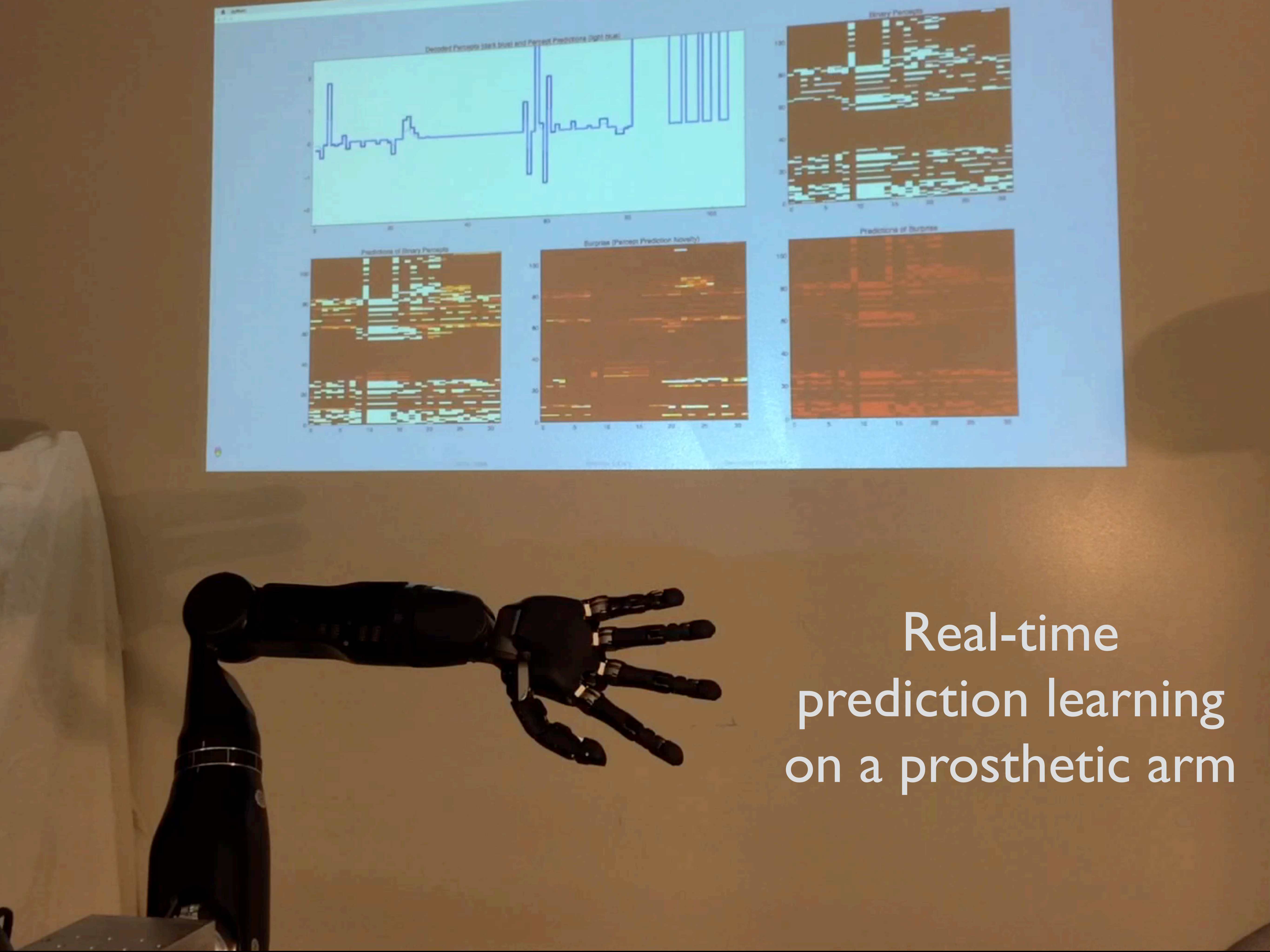
features

…

PSR

sensorimotor data

predictions


Decoded Percepts (dark blue) and Percept Predictions (light blue)
Binary Percepts
Predictions of Binary Percepts
Surprise (Percept Prediction Novelty)
Predictions of Surprise

Real-time prediction learning on a prosthetic arm

An old, ambitious goal:

To understand the world in terms of sensorimotor data

- Making predictions at multiple levels of abstraction
- Finding the abstractions that carve the world at its joints
- Expressing cause and effect compactly, supporting planning and decision making
- This goal is well suited to scaling;
  it can utilize arbitrarily large quantities of computation
  - in learning the predictions
  - and in searching for the best abstractions

New tools

- General value functions (GVFs) provide a uniform language for efficiently learnable predictive knowledge
- Options and option models (temporal abstraction)
- Predictive state representations
- New off-policy learning algorithms (gradient-TD, emphatic-TD)
- Temporal-difference networks
- Deep learning, representation search
- Moore’s law!

Conclusions

- Moore's law strongly impacts AI
  - It makes the present special, as hardware races alongside ideas
  - It causes scalable methods to have the *greatest long-term impact*
- The future belongs to scalable methods: search and learning
- Scalable learning from ordinary experience is the next prize
- Learning knowledge from ordinary experience is the big prize
  - it is possible (the knowledge is in the data!) and very scalable
- Our plans should be ambitious, scalable, and patient/stubborn

Thank you for your attention

and thanks to

Rupam Mahmood, Adam White, Joseph Modayil,
Harm van Seijen, Doina Precup, Hado van Hasselt,
Thomas Degris