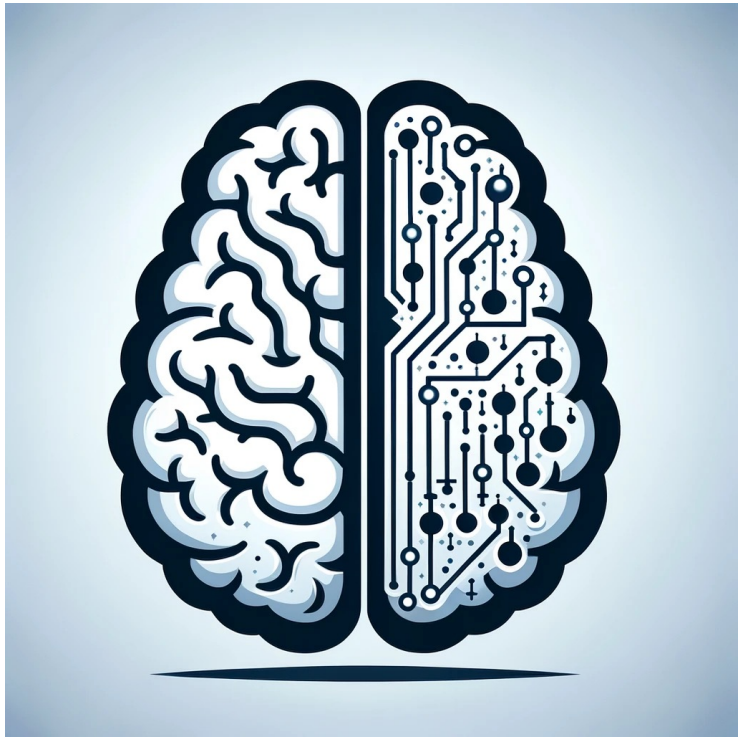
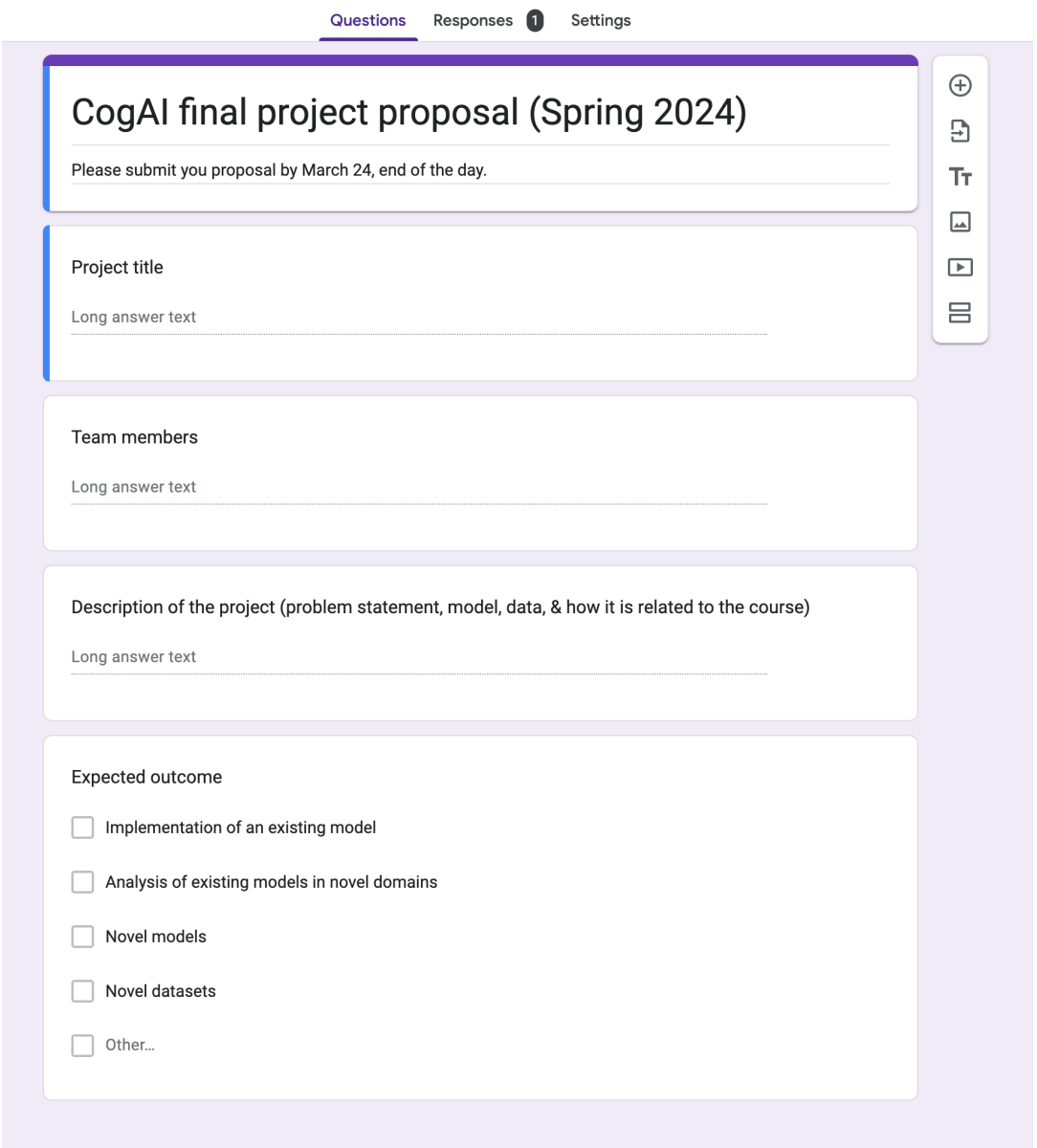
EN 601.473/601.673: Cognitive Artificial Intelligence (CogAI)



Lecture 17: Physical scene understanding

Tianmin Shu

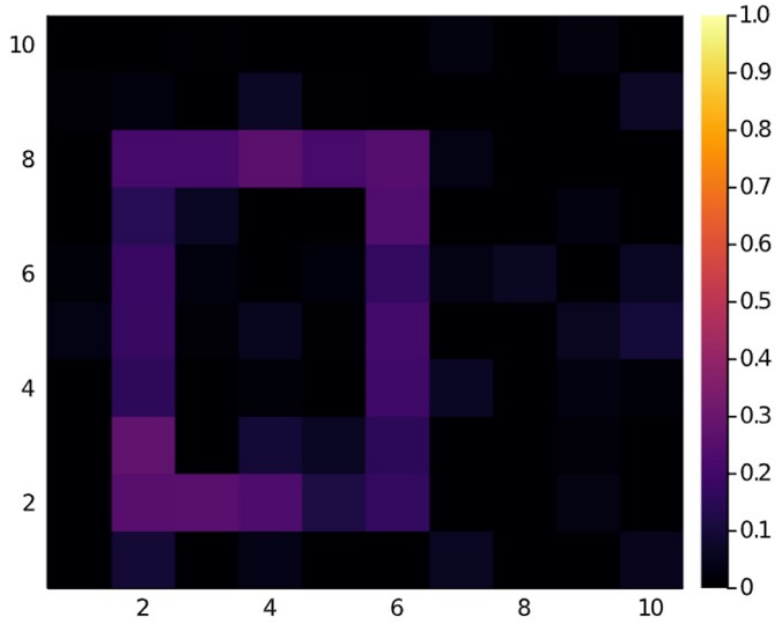
Course project – submit your proposal now!



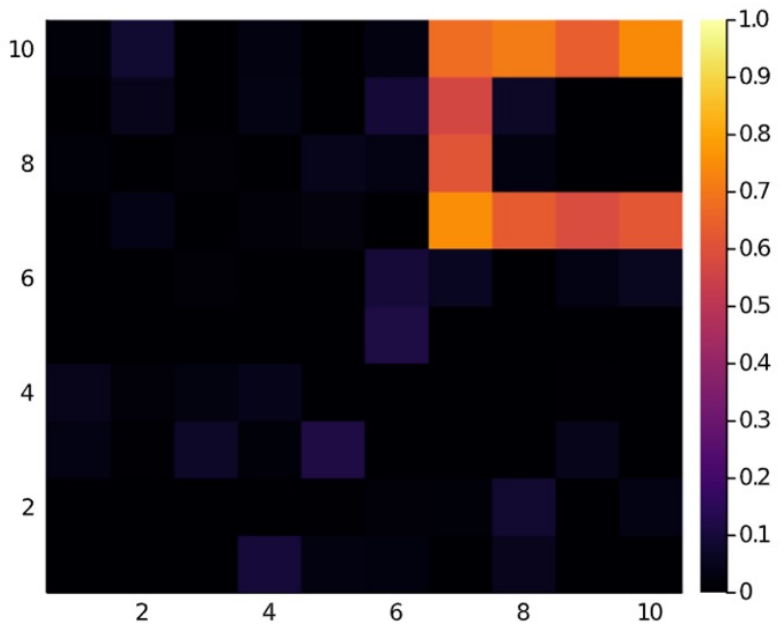
Problem set 3 – the last pset!

- Neural amortized inference for inverse graphics
- Train NN for variational approximation q(h | x); use it for proposals
- Due by April 14, end of the day

bottom-left corner at [2, 2]
width_val = 4
height_val = 6
brightness_val = .20



bottom-left corner at [7, 7]
width_val = 4
height_val = 3
brightness_val = .66



Problem set 3

- NN using differentiable probabilistic programming in Gen (you don't have to use PyTorch)
- See the example in Section 5 of Data-Driven Proposals in Gen.ipynb notebook
- Expected result: neural amortized inference achieves better results (higher average log probability) than importance sampling without data-driven proposals

All three problem sets

- Problem set 1: 10%
- Problem set 2: 20%
- Problem set 3: 10%

This week’s office hours

- 10 am – 11 am on Friday (30 min earlier than the usual time)

Physical and social reasoning

- Common sense scene understanding, intuitive theories

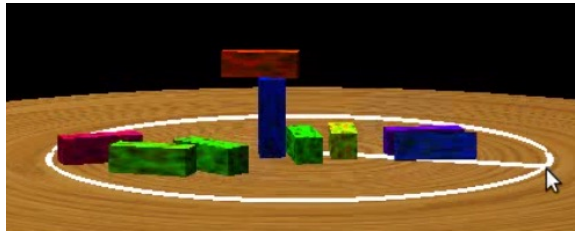
- Physical reasoning

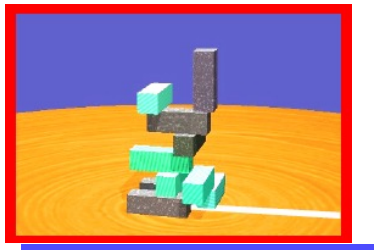
- Social reasoning

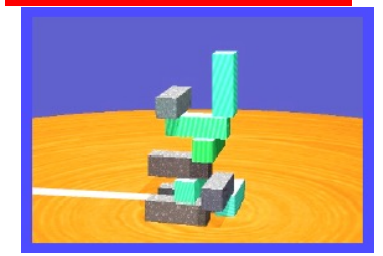
The intuitive physics engine

Will this stack of blocks fall?

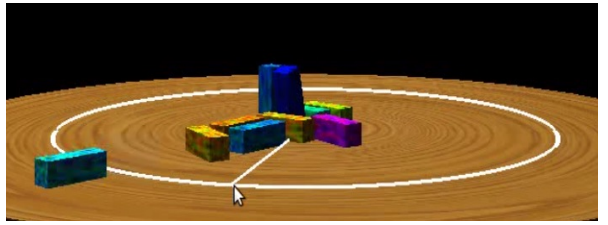
Which way will they fall?



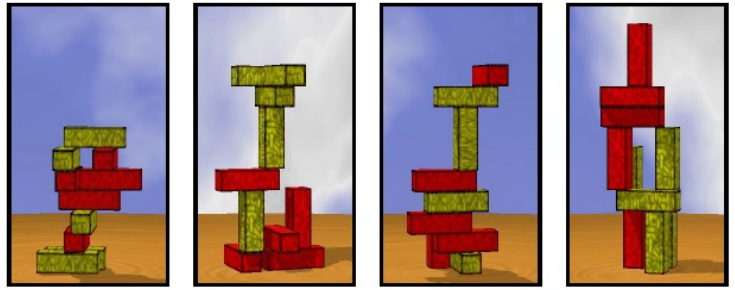




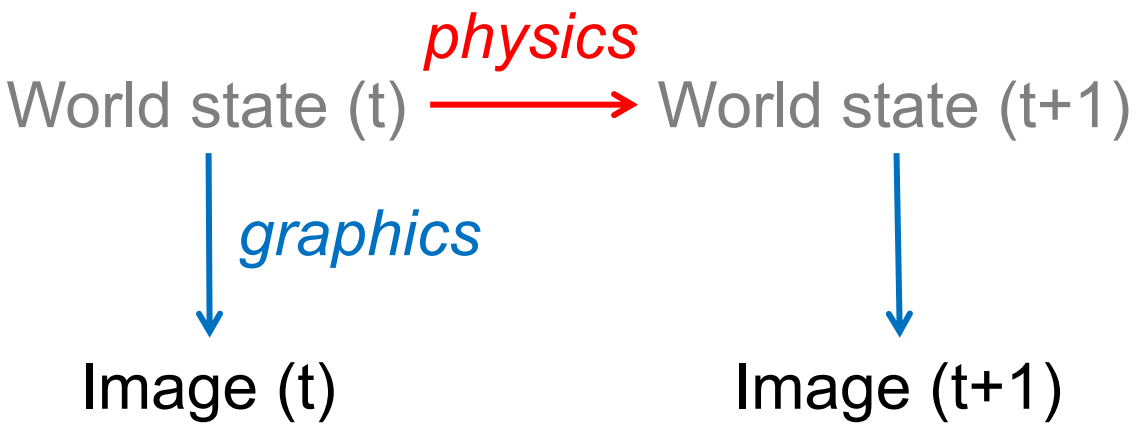
How far will they fall?



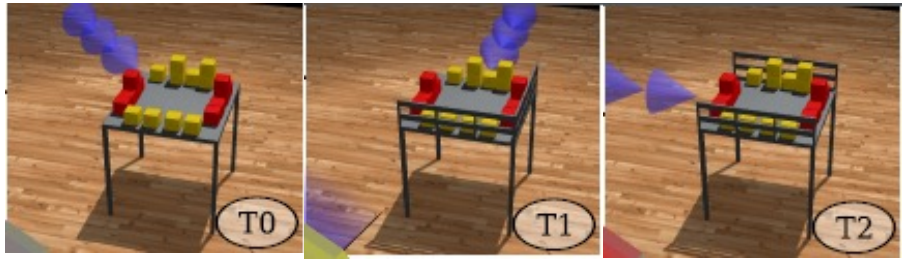
Is red or yellow heavier?



What if grey is much heavier than green?

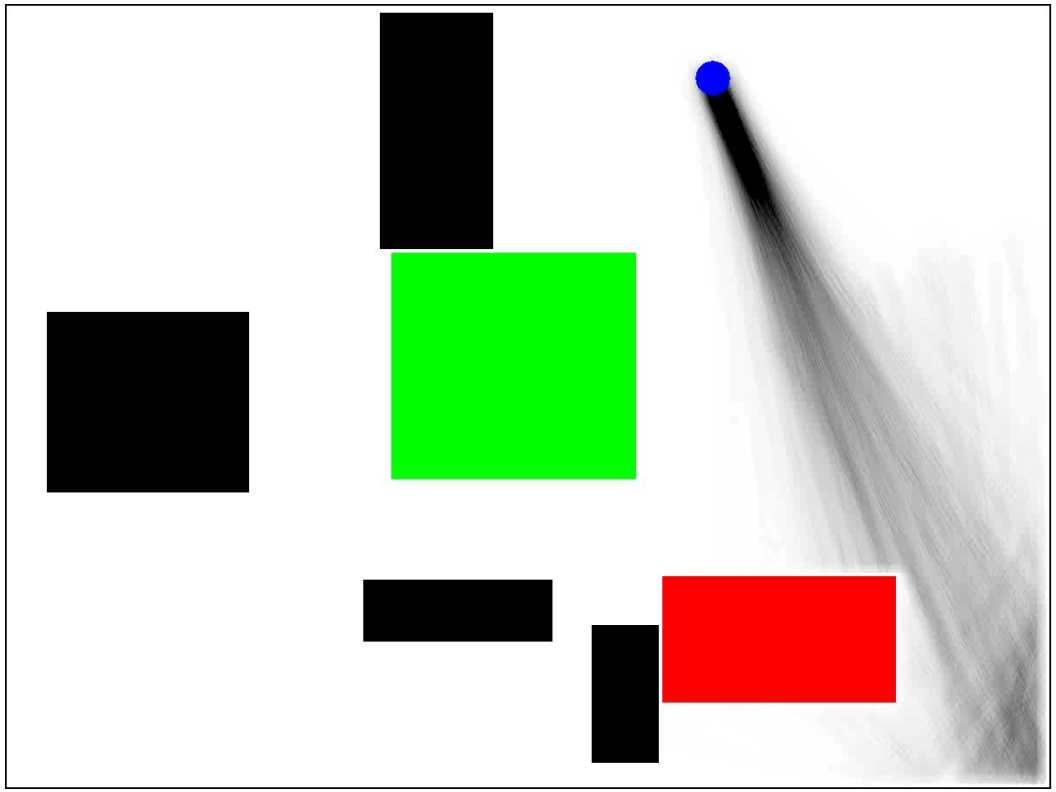


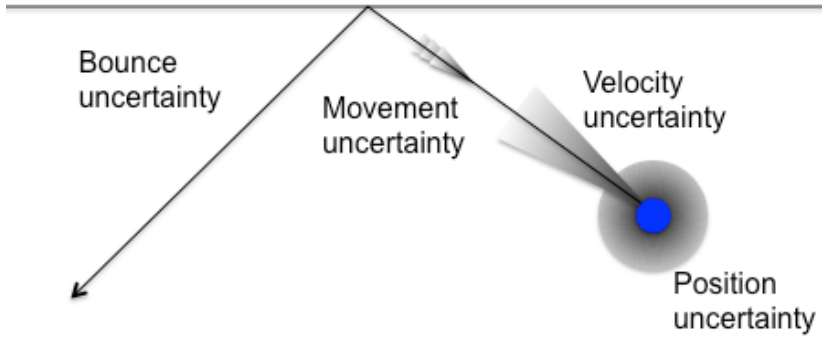
What will happen if you bump the table …?

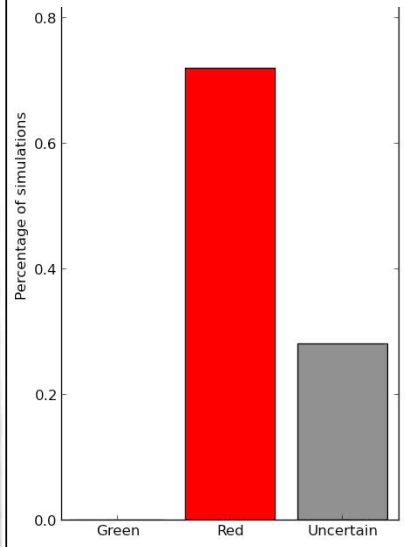


Dynamics in intuitive physics

Ooooh or **Aaaah** ?







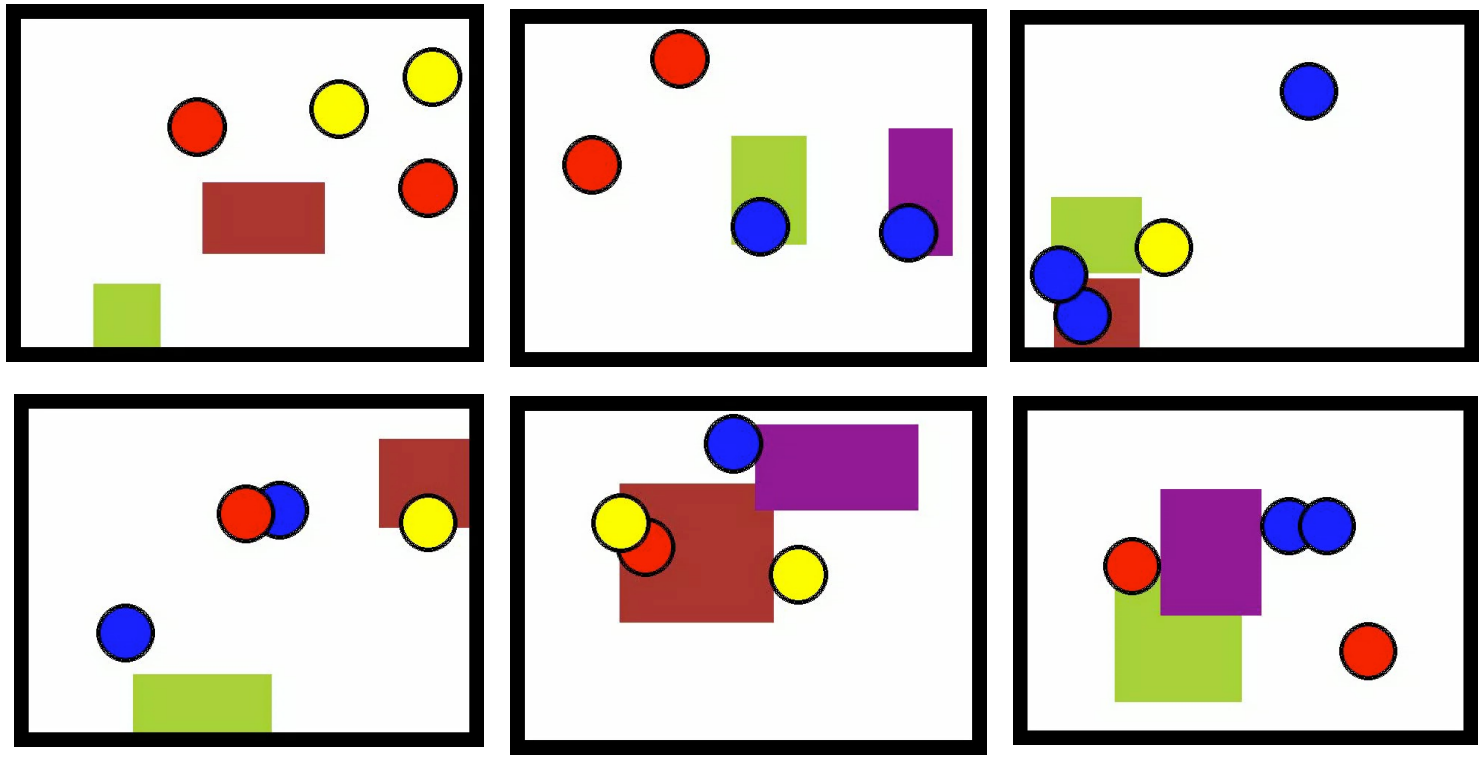
(Smith, Dechter, Tenenbaum, Vul, Cog Sci 2013)

Multimodal physical understanding

Vision and sound

Rapid learning of physical properties and laws

Baby air hockey table



Inferring properties:
e.g., mass, charge, friction, elasticity, resistance…

Learning laws:
e.g., presence of forces and their shape, existence of hidden objects, kinds of substances …

Ullman et al. (*Cognitive Psychology* 2018)

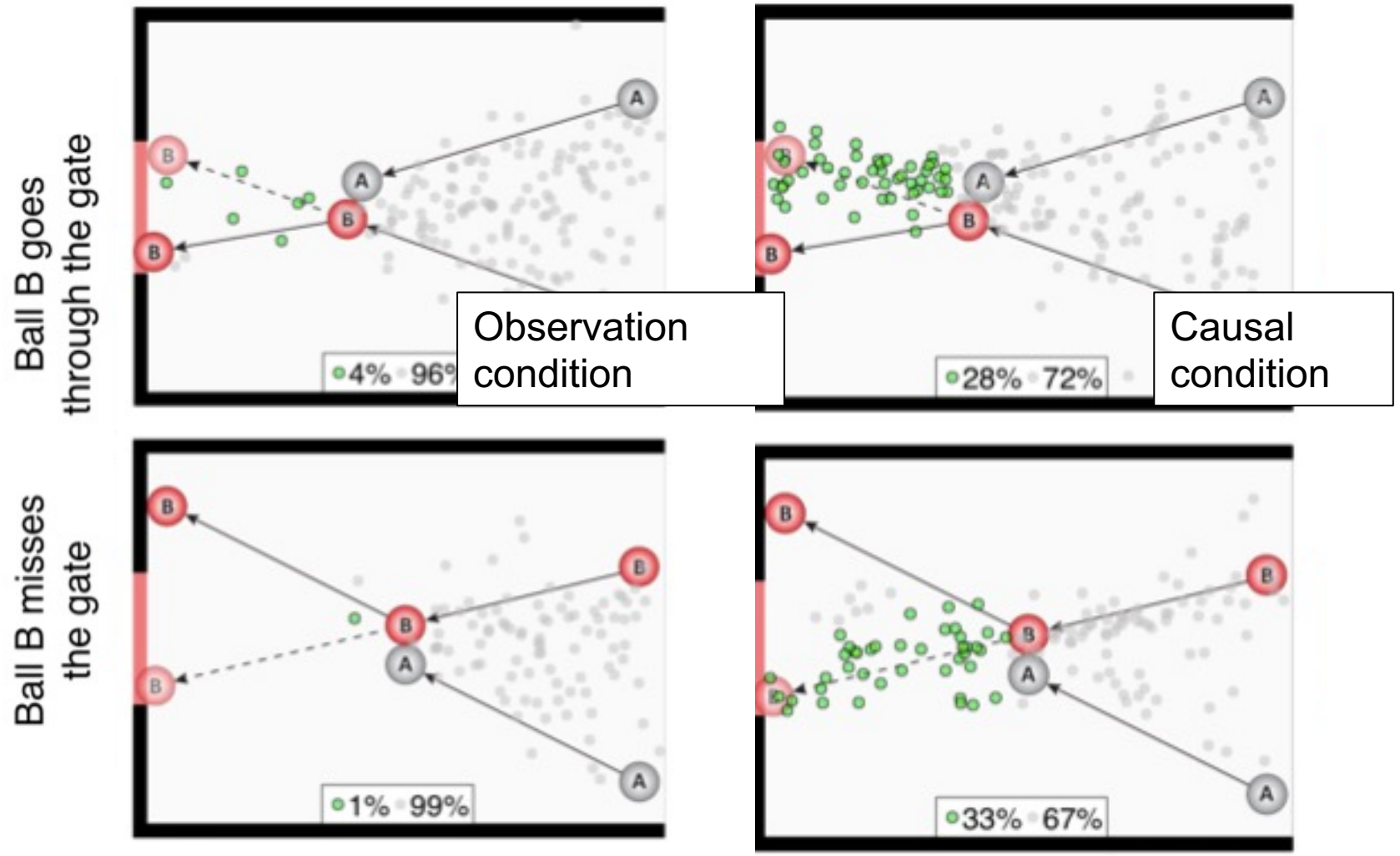
Causal reasoning (evaluated via eye tracking)

Task: How did B go in hole?
Eye-tracking: **Predictive** simulations

Task: Did A cause B to go in the hole?
Eye-tracking: **Counterfactual** simulations

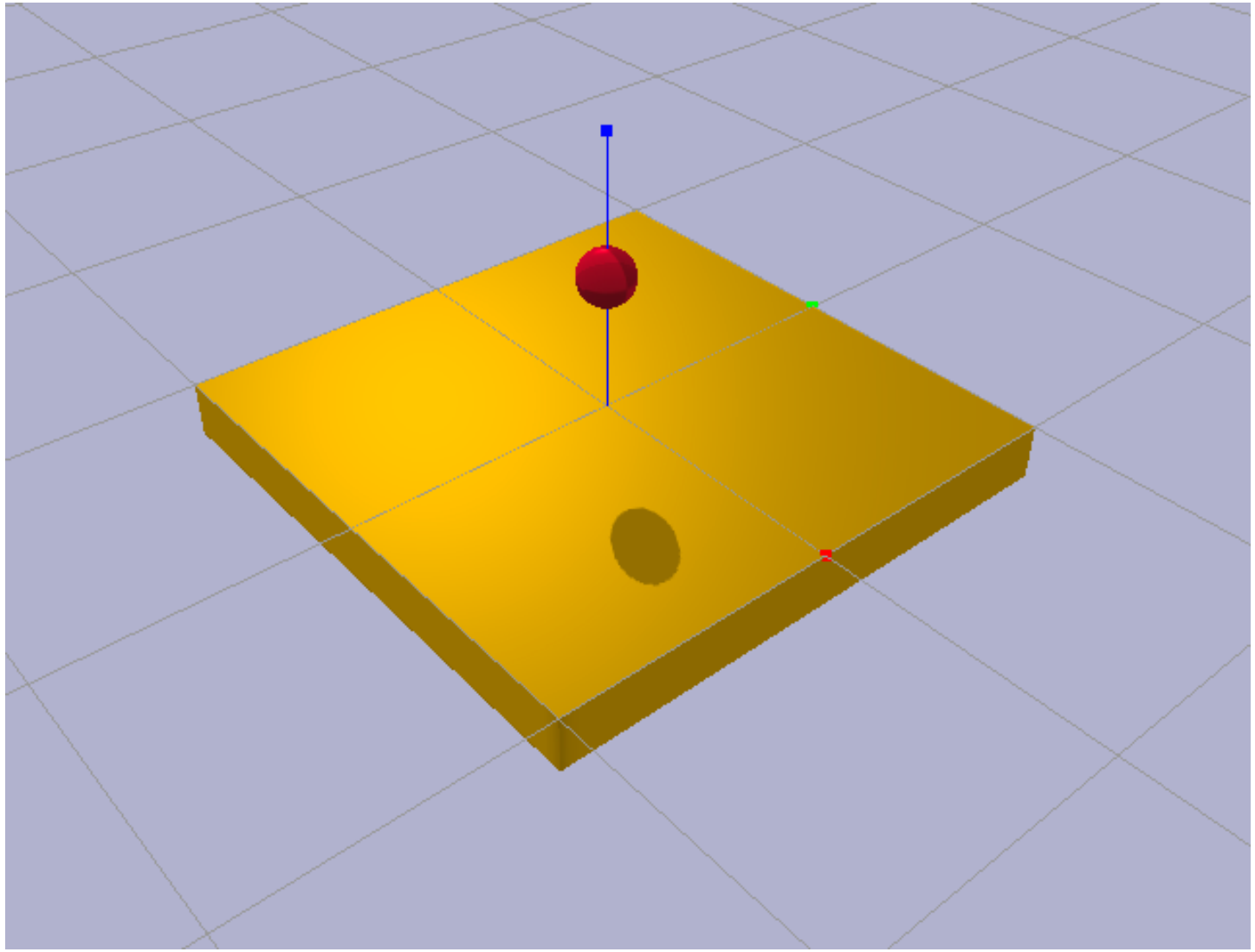
Gerstenberg et al. (2017)

Causal reasoning (evaluated via eye tracking)



Intuitive Physics in Gen

Inferring the mass and restitution of a ball based on how it bounces off a table

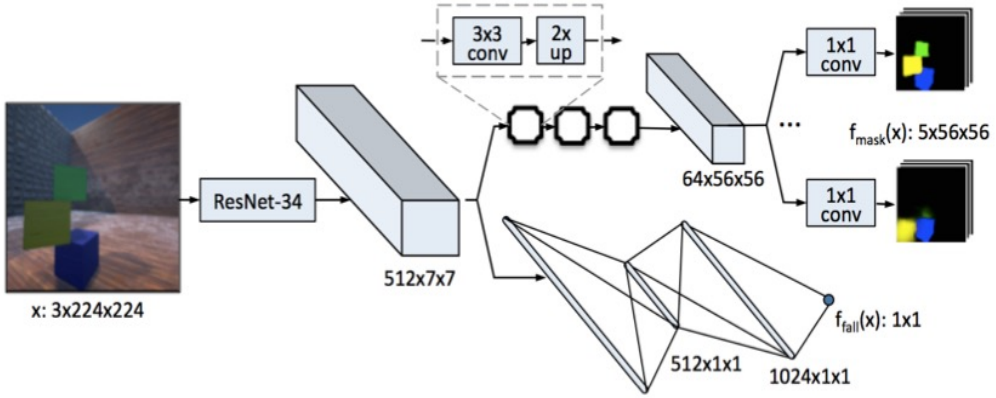


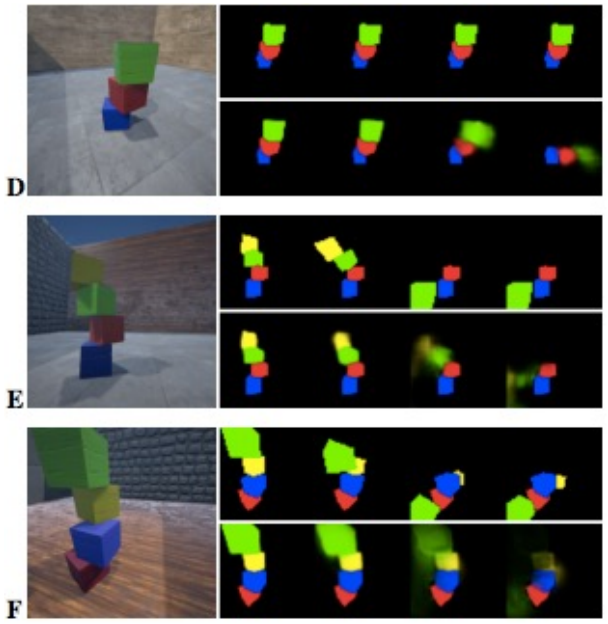
Intuitive Physics in Gen

See jupyter notebook

Intuitive physics in neural networks?

Distributed (vector) representations of objects and their interactions?

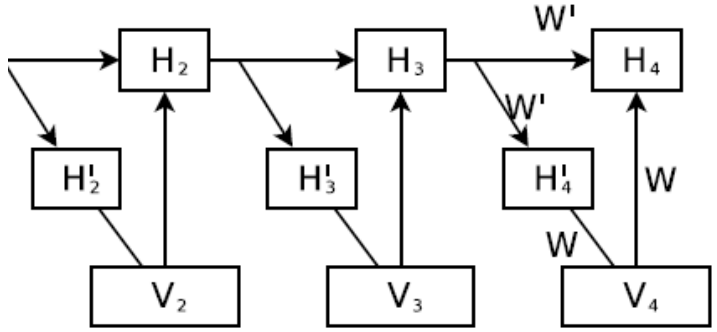




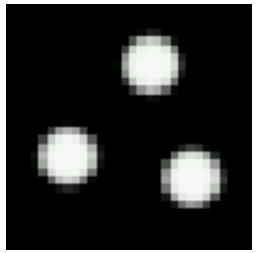
PhysNet (Facebook AI; Lerer et al 2016)

Training: 200K examples of 2-4 cubes

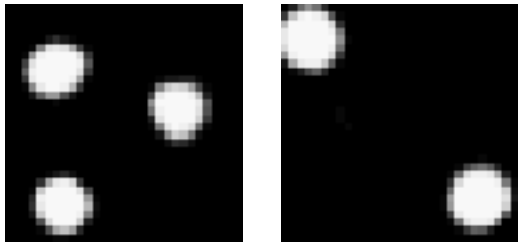
Test: 5 cubes



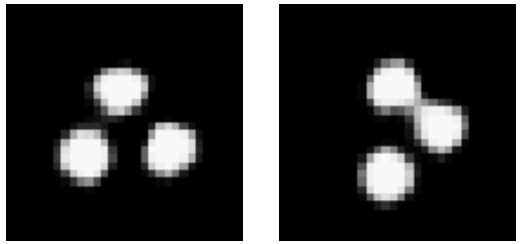
Sutskever & Hinton (2008)



Training data



Recurrent Temporal RBM

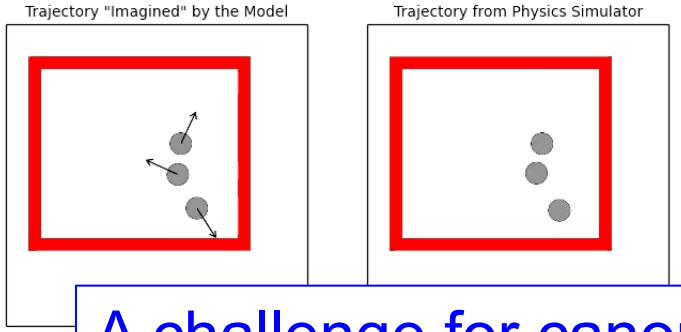


Temporal RBM

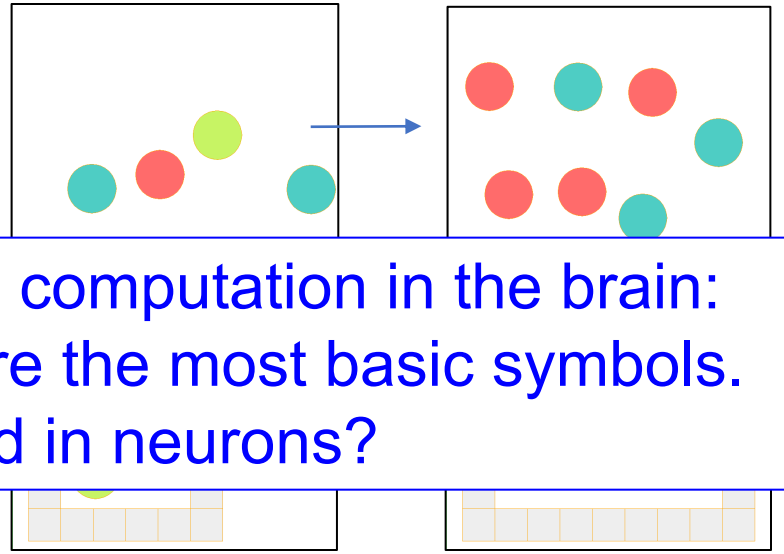
Intuitive physics in neural networks?

A case study in inductive bias: The power of discrete objects, relations, interactions

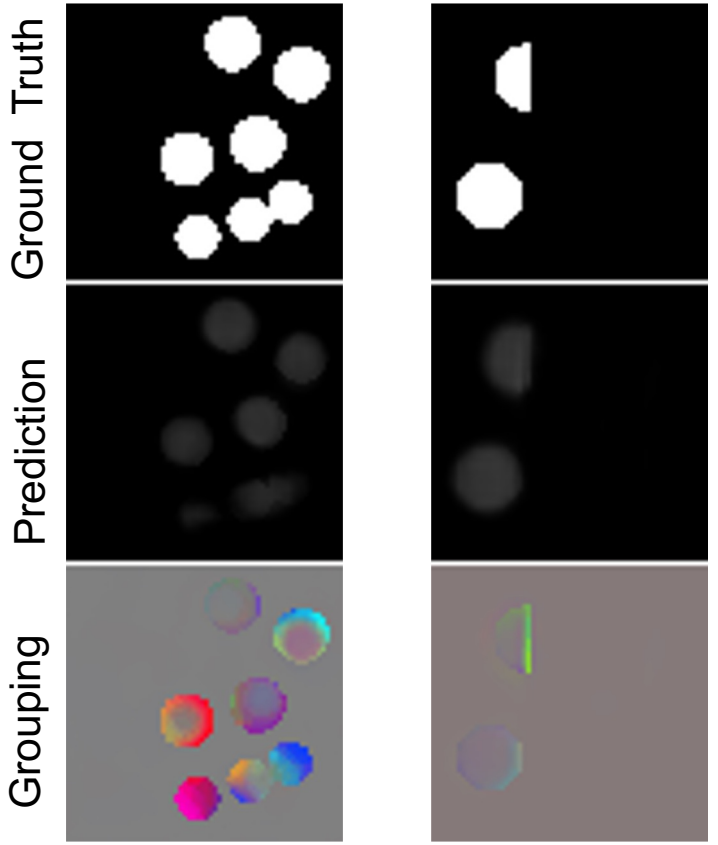
Fragkiadaki et al. (2015)



Neural Physics Engine (Chang et al., 2017)
Interaction Networks (Battaglia et al., 2016)

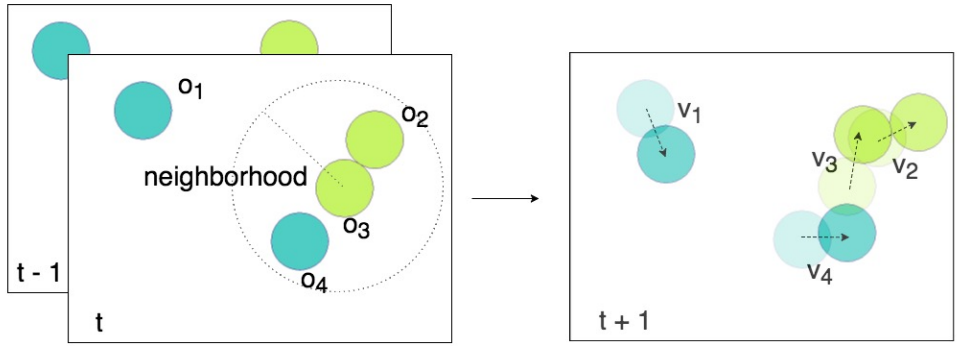


Relational Neural Expectation Maximization (Steenkiste et al., 2018)

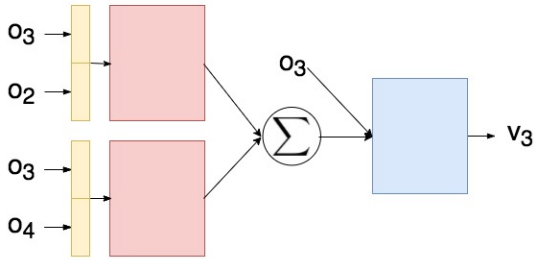


A challenge for canonical computation in the brain:
Object representations are the most basic symbols.
How are they represented in neurons?

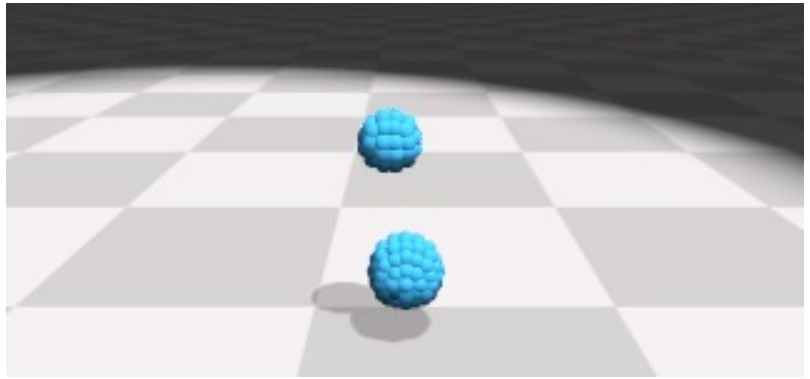
Scenario

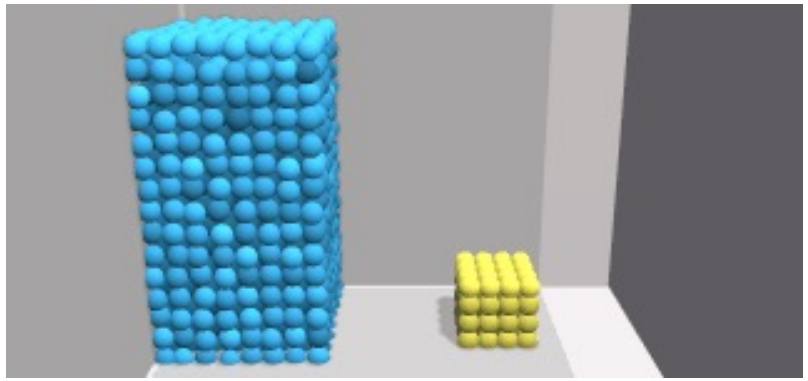


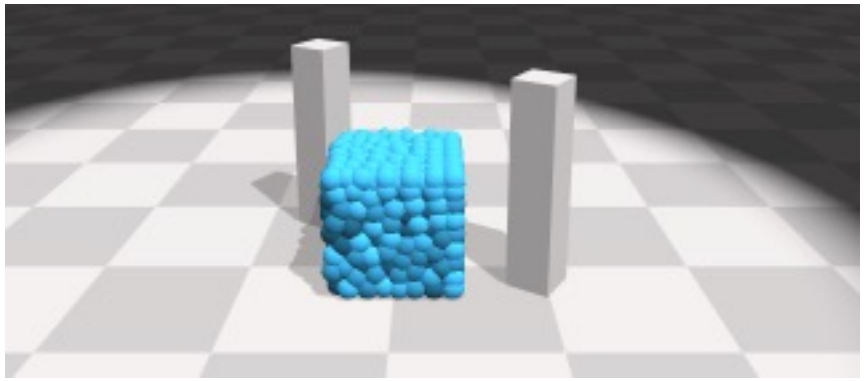
NPE applied on object 3

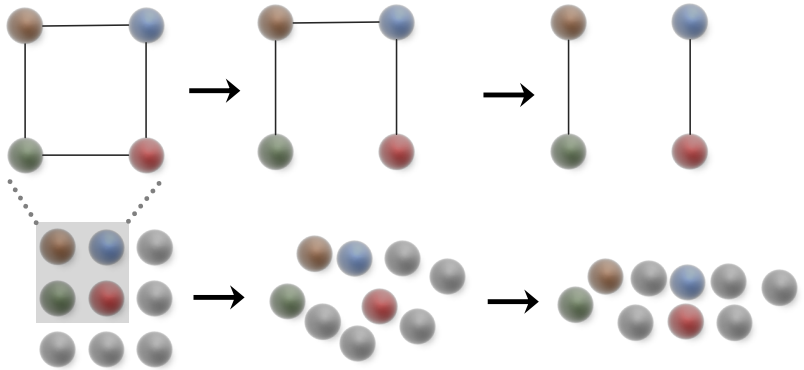


Dynamic Particle Interaction (DPI) networks: Learning to simulate different forms of matter for prediction and control

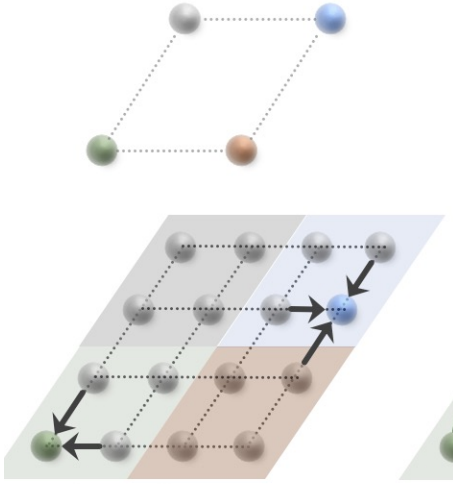




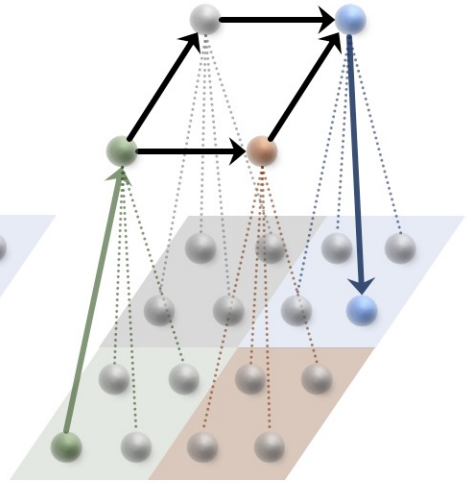




Dynamic Interaction Graph



Local Propagation



Hierarchical Propagation

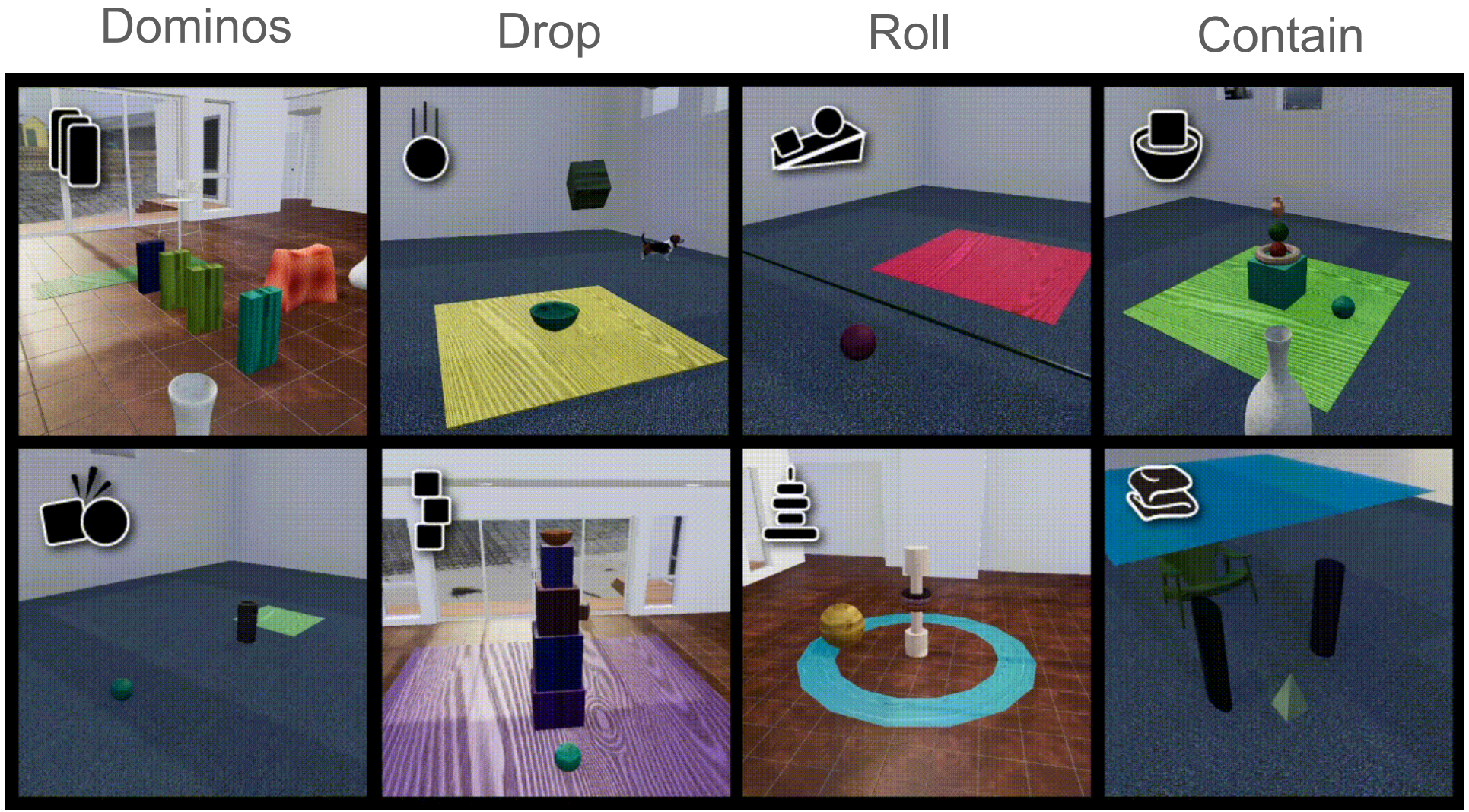
Rigid body:
Hierarchical graph, global motion

Deformable object:
Hierarchical graph, local motion

Fluid:
no hierarchy, local motion

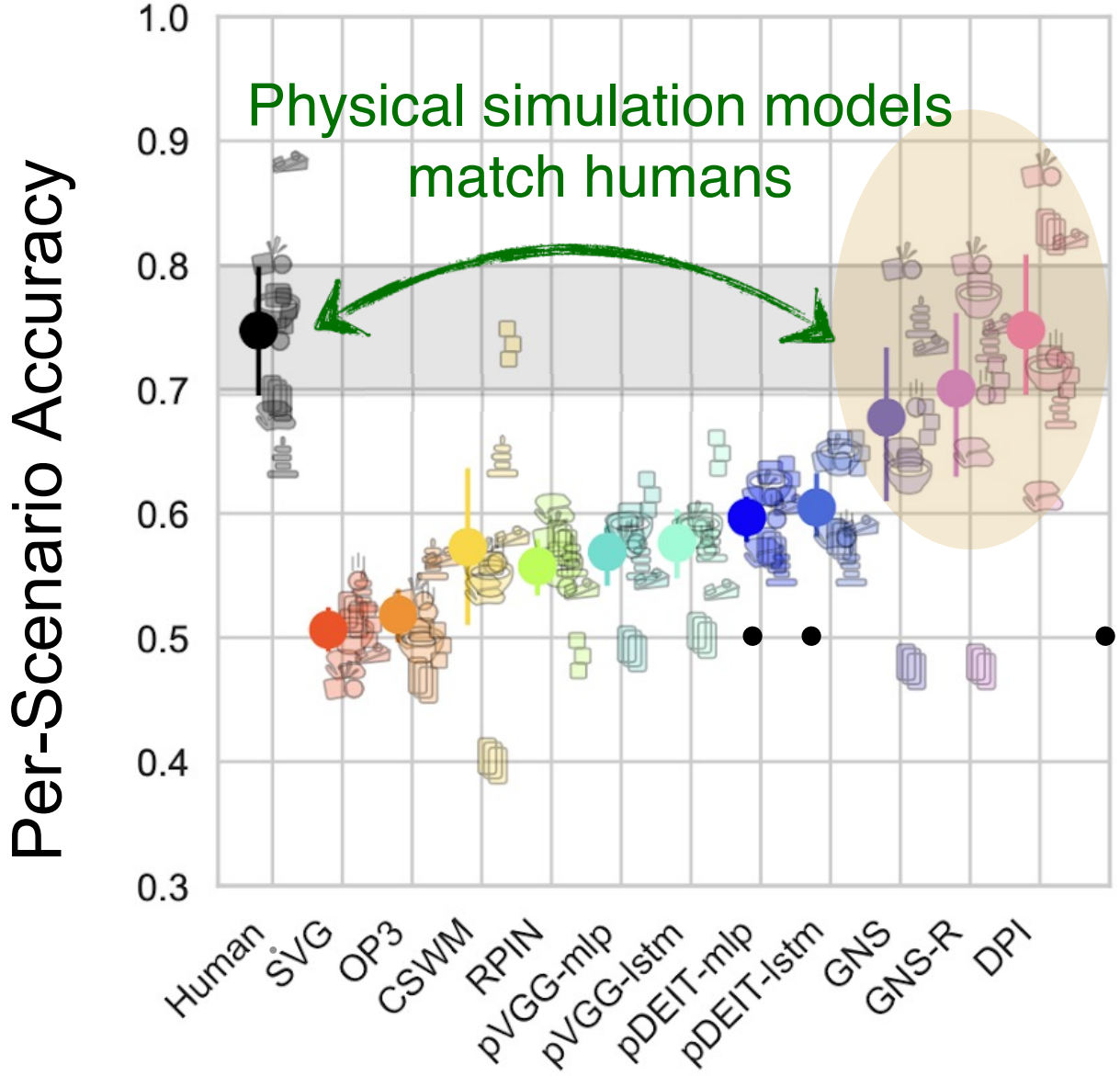
Li et al., (ICLR'19, ICRA'19)

Physion: Physical prediction from vision in humans and machines



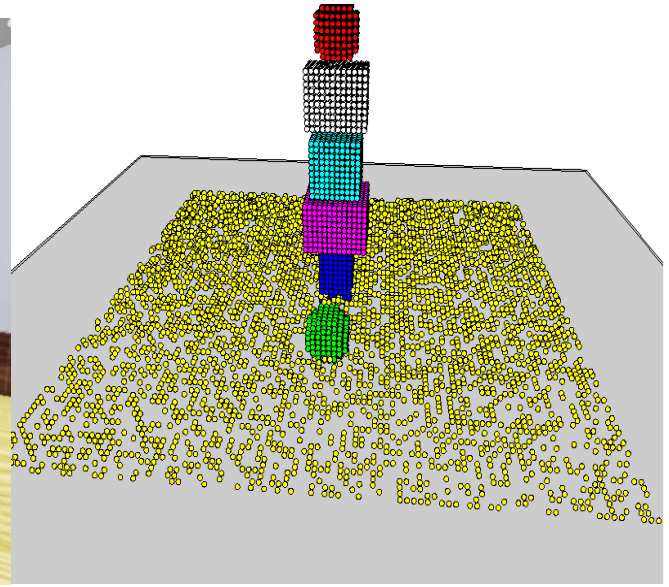
Dan Bear et al. (NeurIPS Benchmark 2021)

Benchmark results



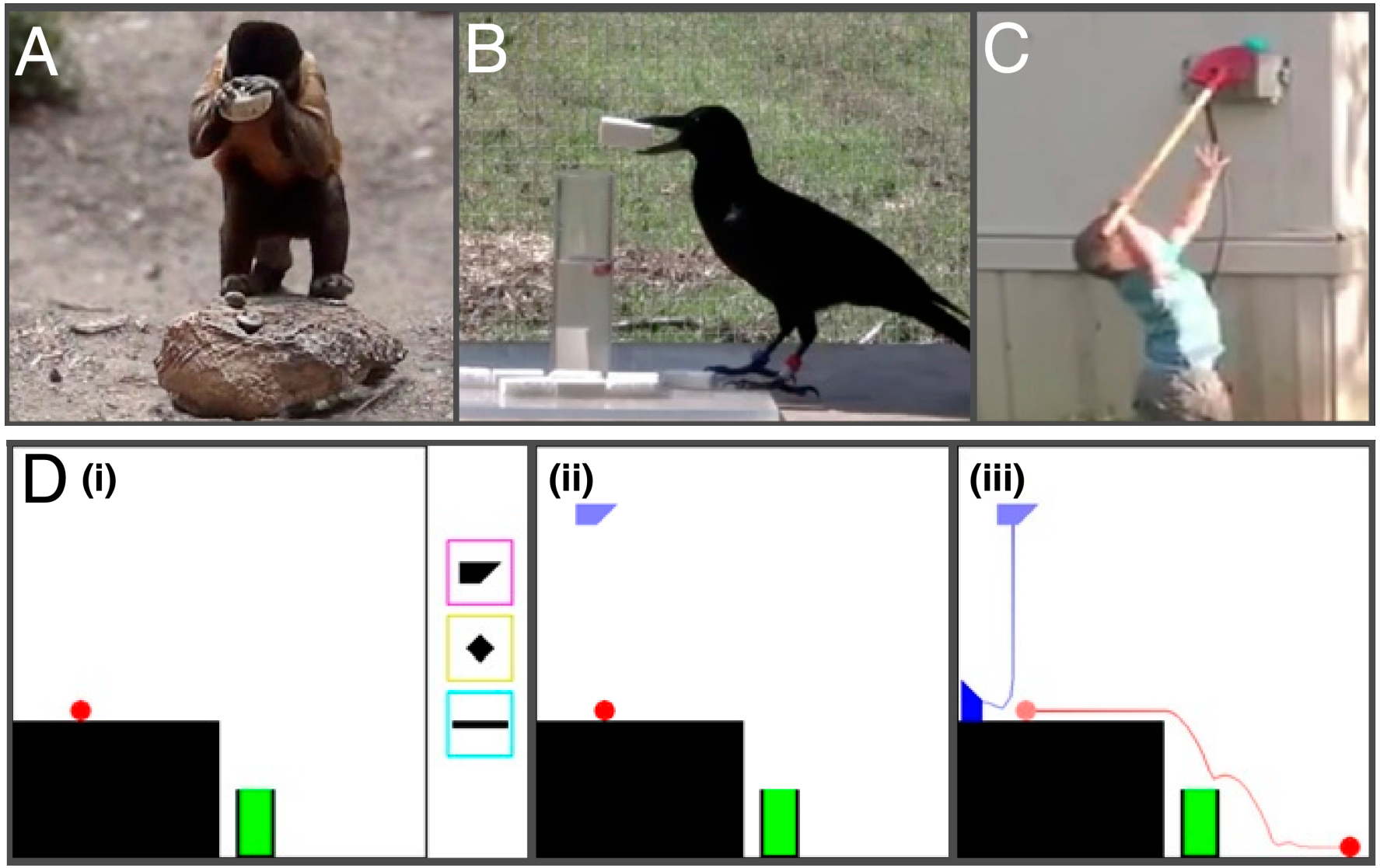
- Humans are good but not perfect
- *Models that simulate the explicit physical state of the scene — not visual input — can match human accuracy*
- *Vision-based models are much worse.*

Visual input



Physical input

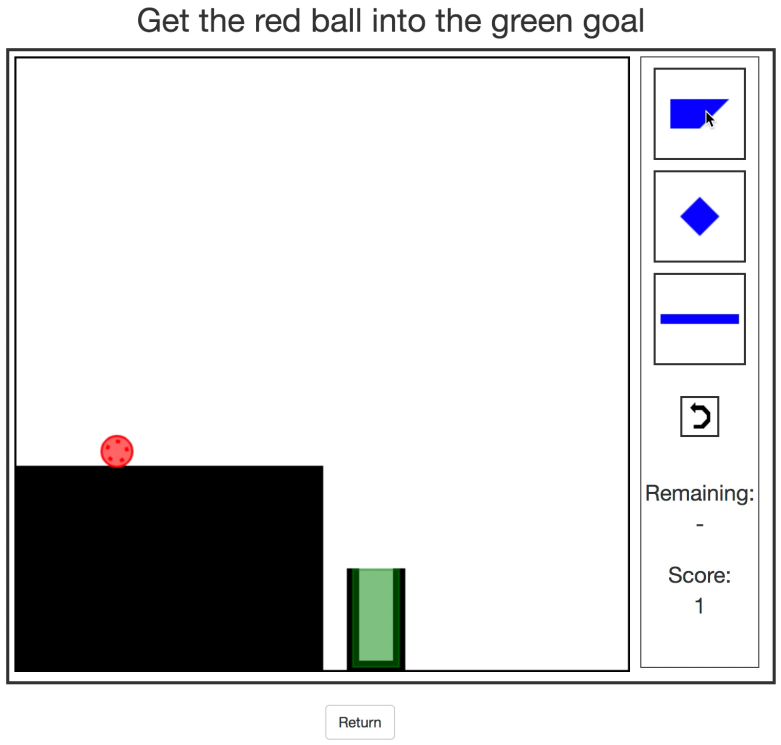
Physical reasoning for problem solving and tool use

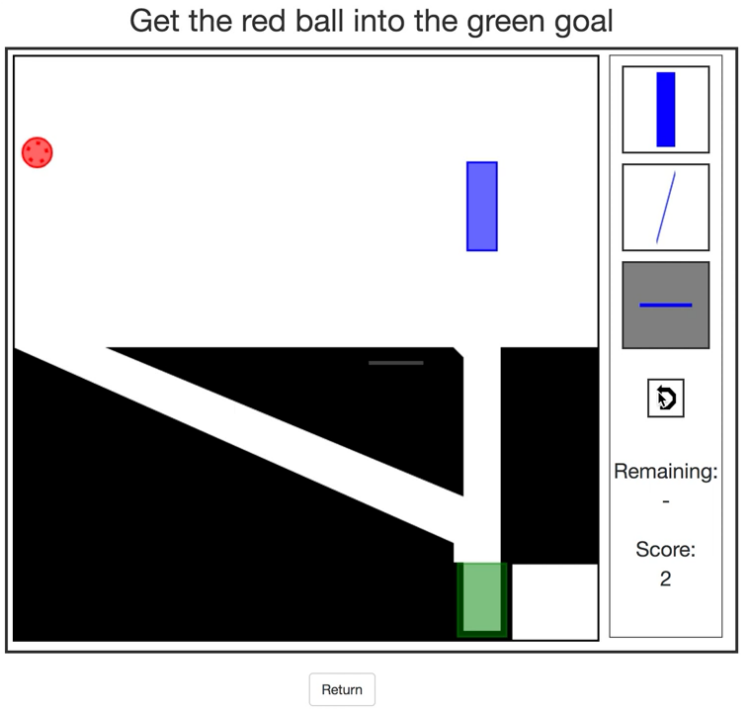


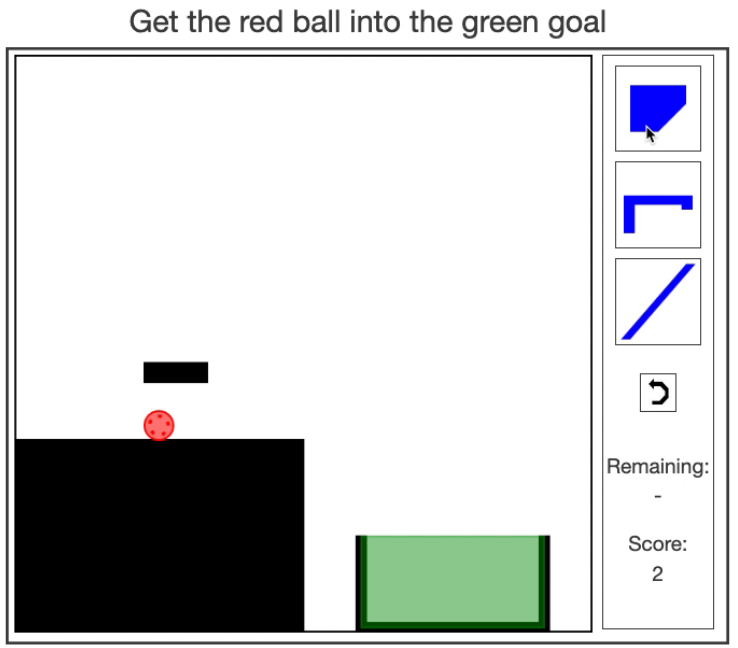
Allen et al. (PNAS 2020)

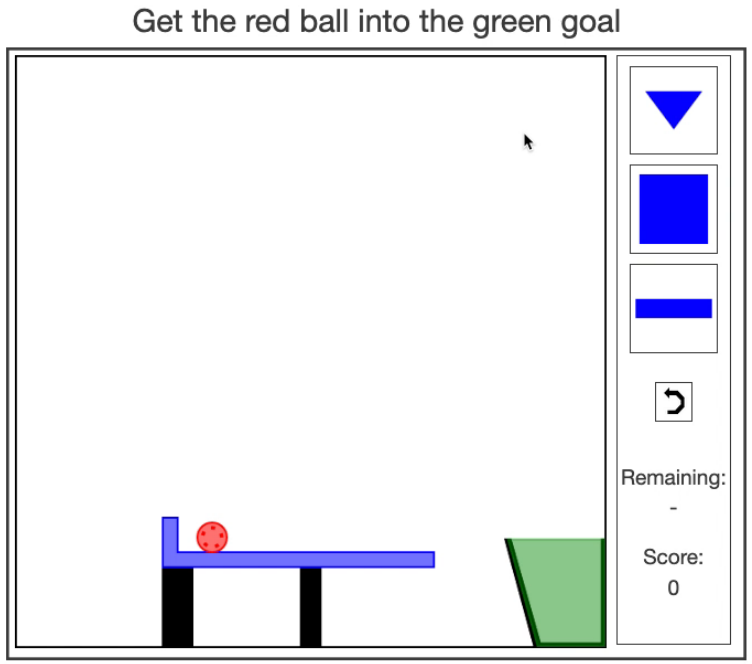
“Virtual tools”

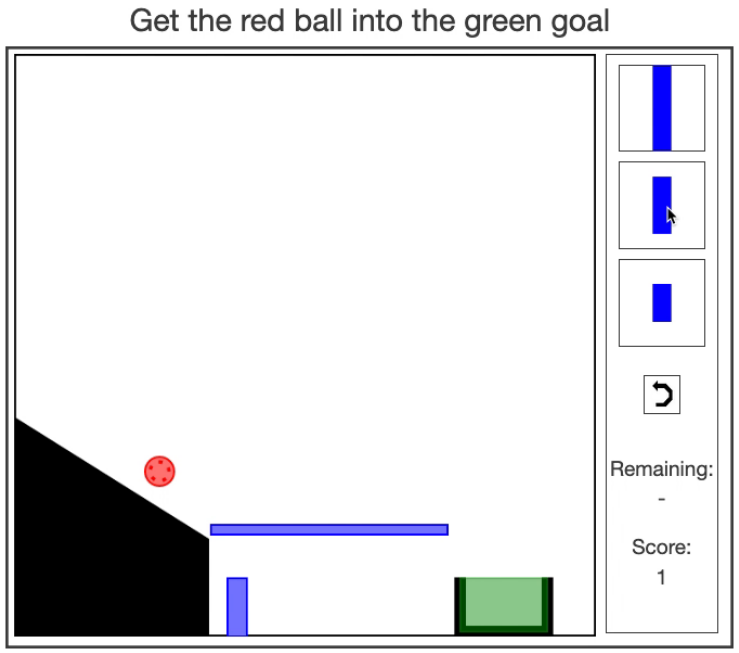
Allen et al. (PNAS 2020)







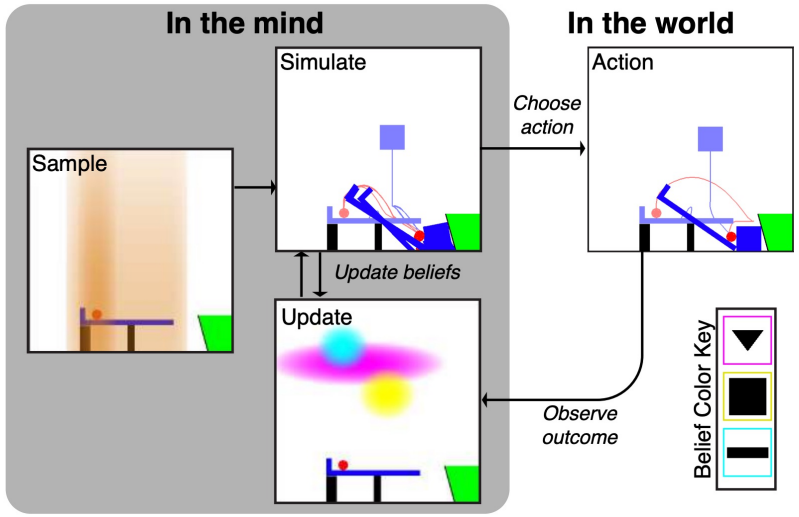


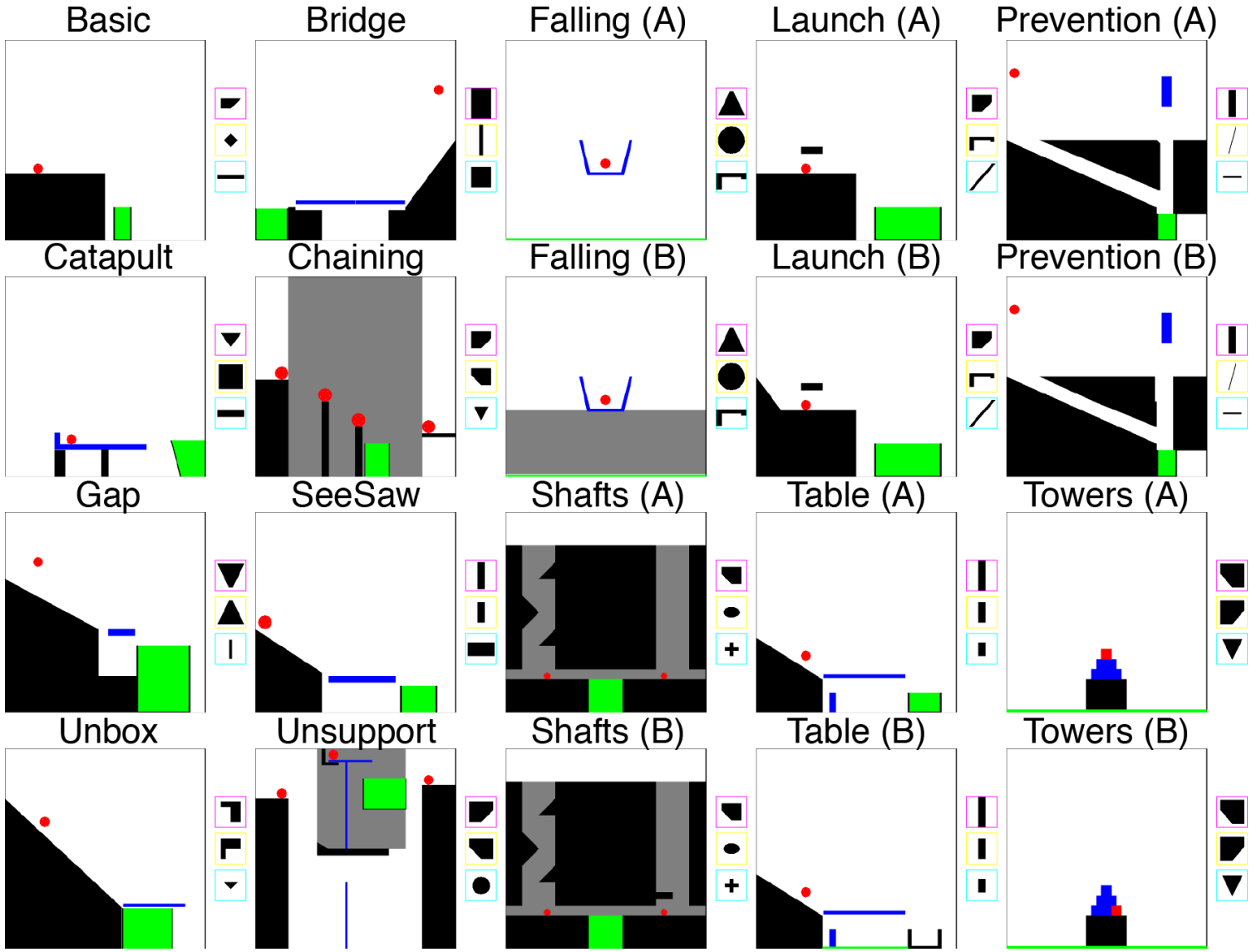


“Virtual tools”

Allen et al. (PNAS 2020)

Sample, Simulate, Update (SSUP) model:



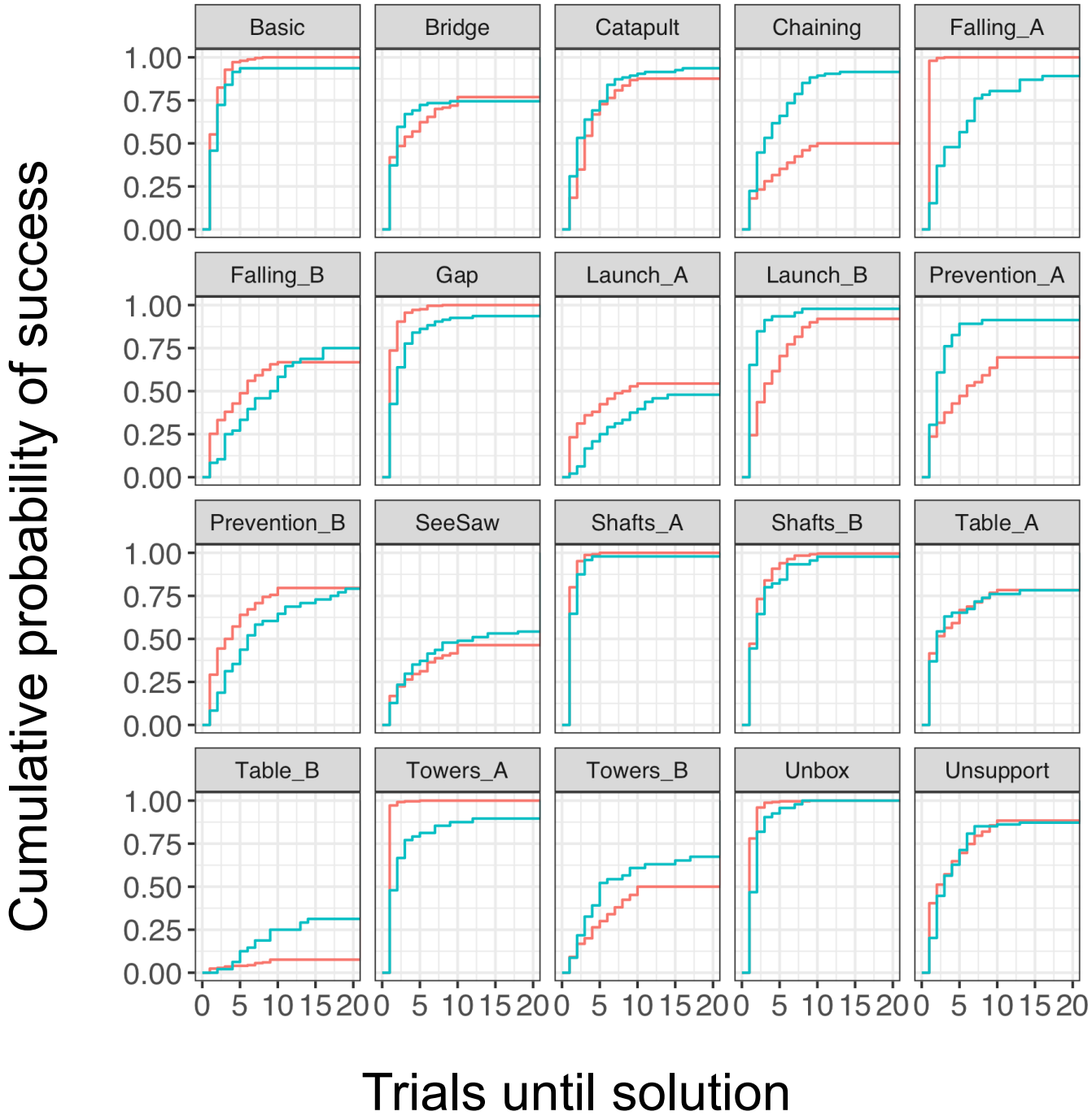


“Virtual tools”

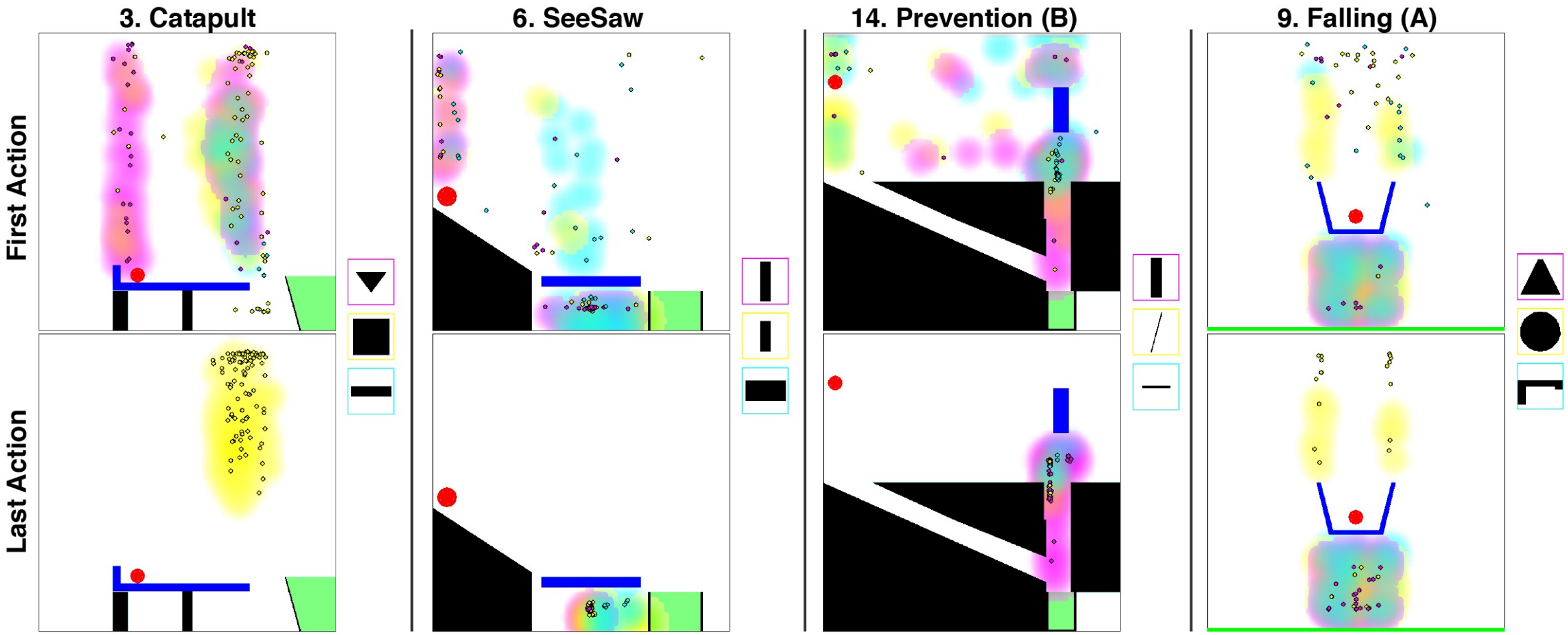
Allen et al. (PNAS 2020)

Humans

Sample, Simulate, Update (SSUP) Model



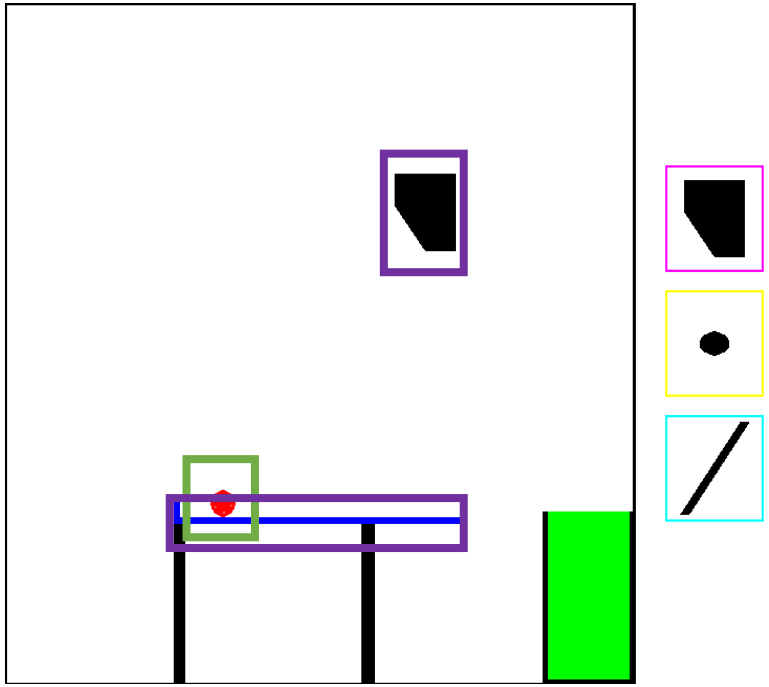
SSUP model often captures specific action choices



Model: cloud
Human: points

Learning tool classes: abstract object-centric strategies

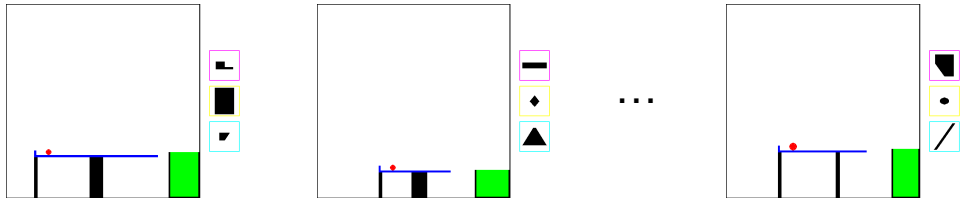
Catapult



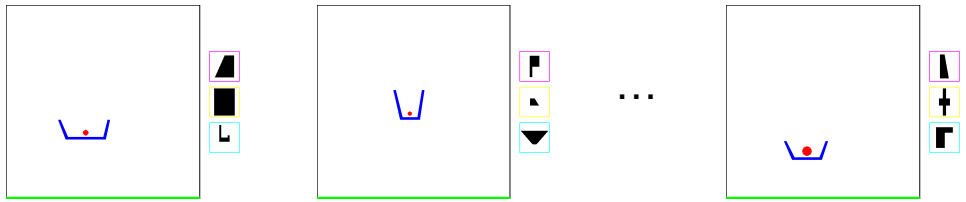
Strategy acquisition and transfer in human learners

Learning

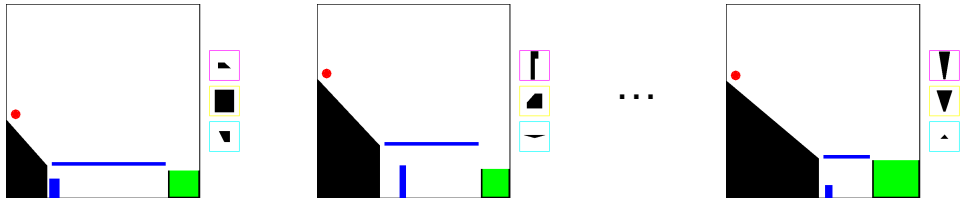
Catapult



Tipping

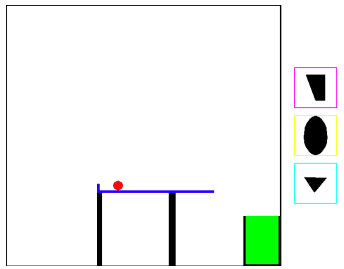


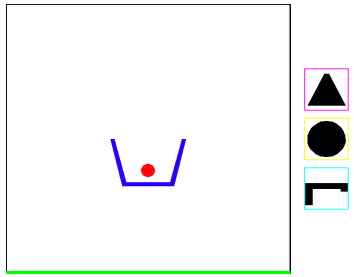
Table

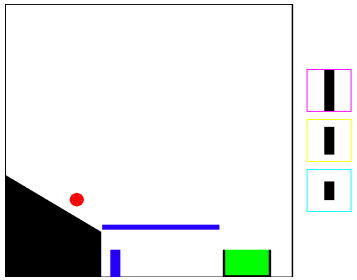


Testing

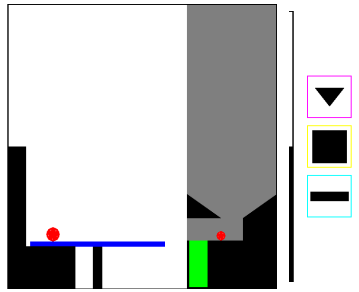
Near transfer

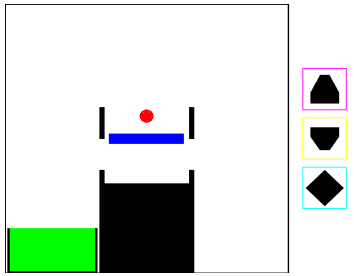


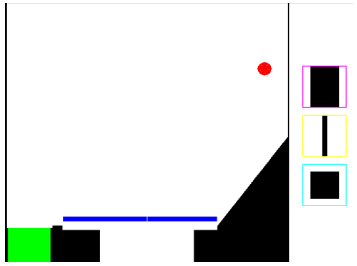




Far transfer

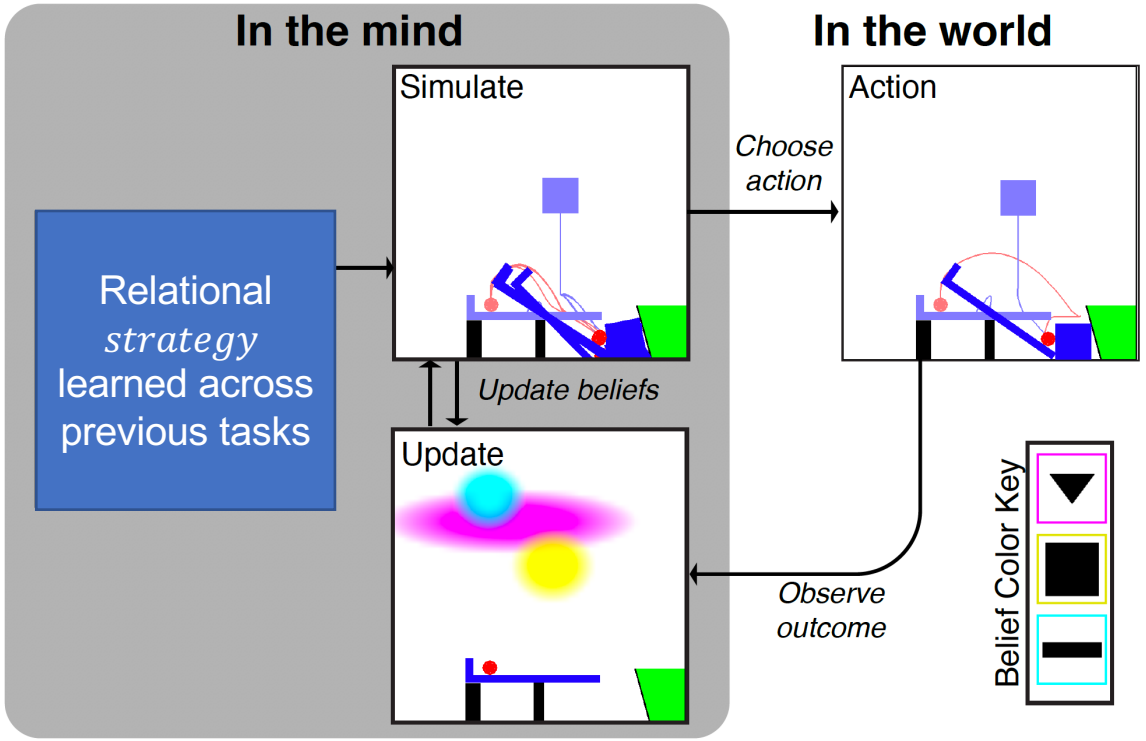






Allen et al. (2020)

A model of strategy learning



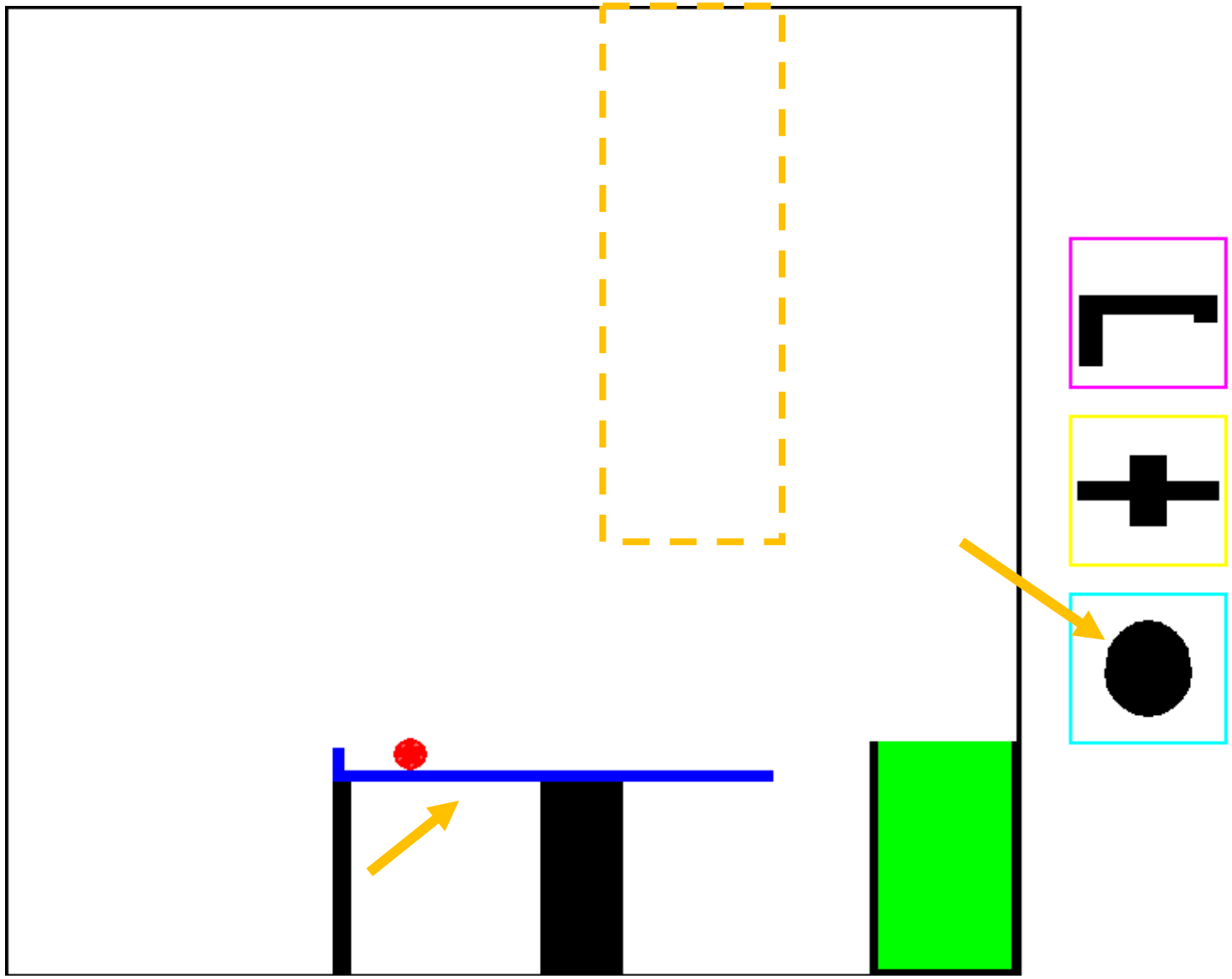
Probabilistic grammar

Production Rule	Probability
P → is_object_type(OBJ, OBJTYPE)	0.33
P → compare(Q1, Q2, OP, DIM)	0.33
P → query_value(Q, DIM)	0.33
Q → size(OBJ)	0.25
Q → area(OBJ)	0.25
Q → distance(OBJ1, OBJ2)	0.25
Q → loc(OBJ)	0.25
OBJ → find_object(OBJTYPE)	0.33
OBJ → tool	0.33
OBJ → obj	0.33
DIM → {x, y}	0.5
OBJTYPE → {'dynamic','goal',SHAPE}	0.33
SHAPE → {'Compound','Ball',etc.}	1/|SHAPE|
OP → {<, >, ==}	0.33

Silver et al. (2020)

Example “catapulting” strategy

- **Pick a large tool**
  - query(area(tool)) > 500
- **Choose wide object beneath ball**
  - compare(location(obj), location(ball), >, ’y’) AND
  - queryDimension(size(obj),0) > 30
- **Place the tool high up and to the right of the wide object**
  - (queryDimension(dist(tool, obj),0) > 65 AND
  - compare(loc(tool),">", loc(obj),1)



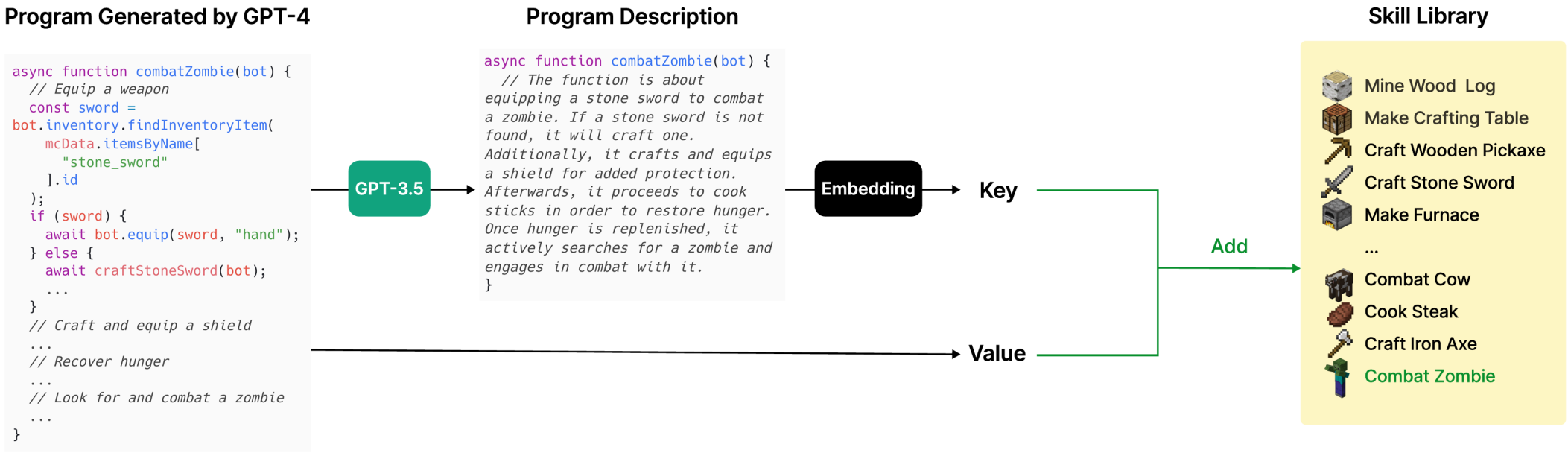
Skill library acquired by LLMs

VOYAGER: An Open-Ended Embodied Agent with Large Language Models

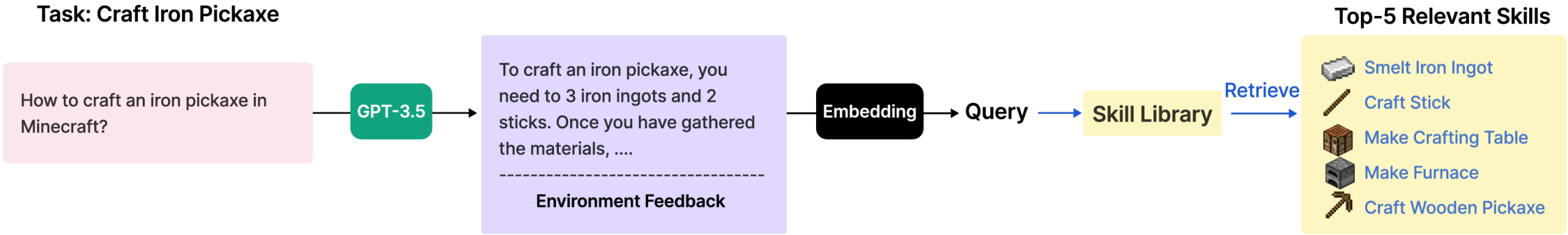
Guanzhi Wang[1,2✉], **Yuqi Xie**[3], **Yunfan Jiang**[4*], **Ajay Mandlekar**[1*],
Chaowei Xiao[1,5], **Yuke Zhu**[1,3], **Linxi "Jim" Fan**[1†✉], **Anima Anandkumar**[1,2†]
[1]NVIDIA, [2]Caltech, [3]UT Austin, [4]Stanford, [5]UW Madison
[*]Equal contribution [†]Equal advising [✉] Corresponding authors
https://voyager.minedojo.org

Add and retrieve skills

- Add a skill



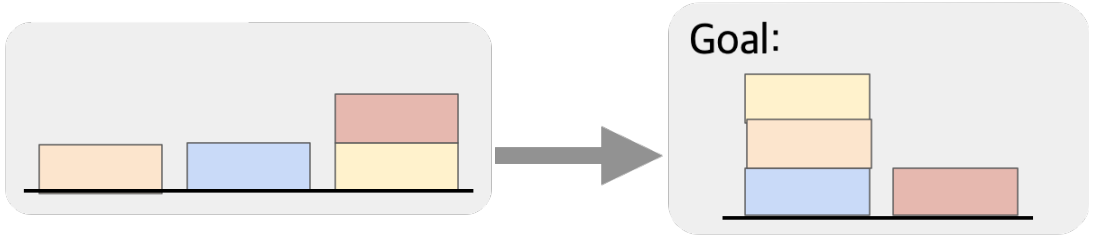
- Retrieve a skill



Language models as world models

- LLMs fail to plan robustly

Blocksworld: How to move the blocks to the goal state?



GPT-4

Invalid Action!
The yellow block is still under the red one.

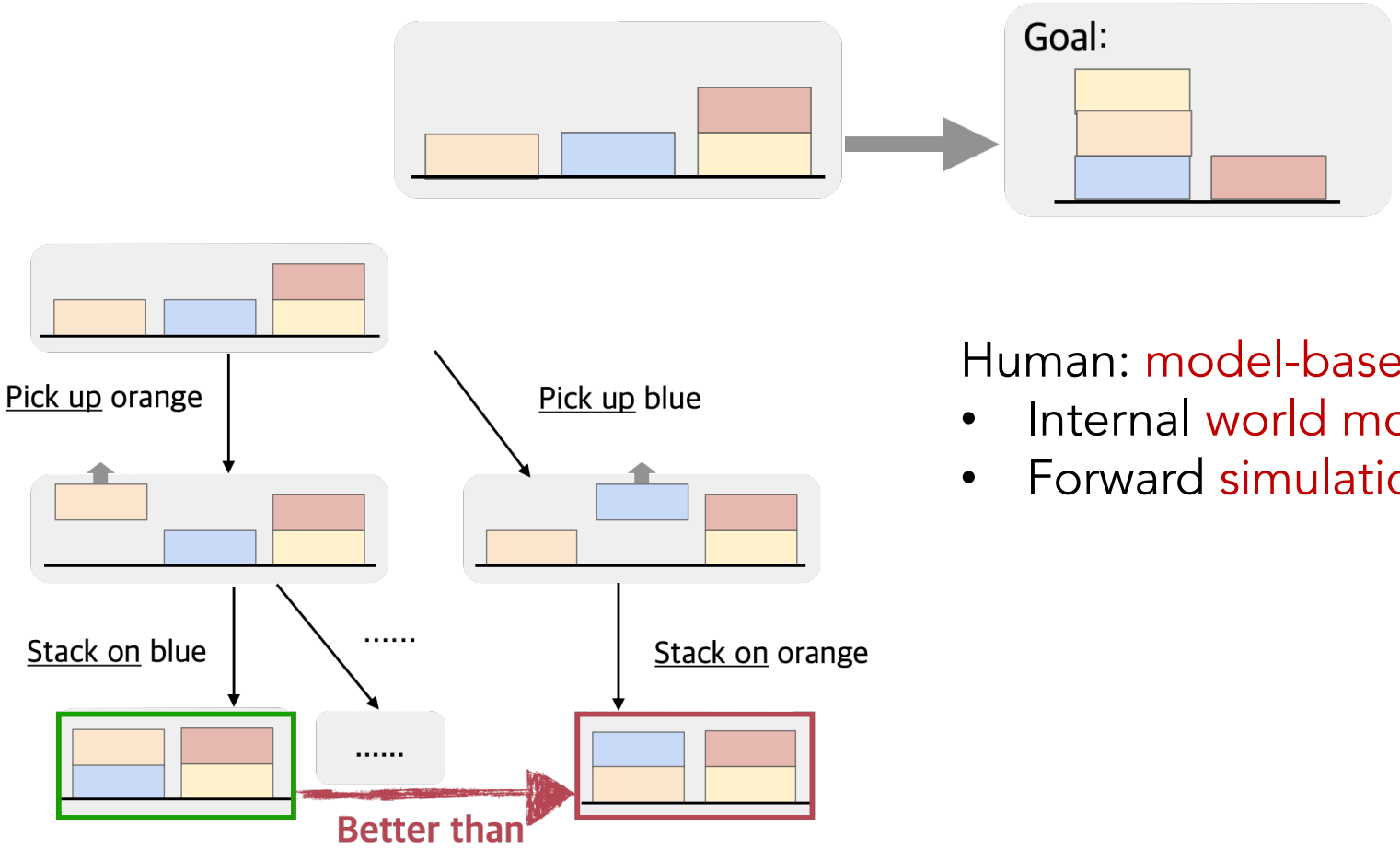
1. Pick up the orange block.
2. Stack it on the blue block.
3. Pick up the yellow block.
4. Stack it on the orange block.
5. Pick up the red block.
6. Put it on the table.

LLMs: Autoregressive plan generation

Valmeekam et al. (2023)
Hao et al. (2023)

Language models as world models

Blocksworld: How to move the blocks to the goal state?



Human: model-based planning

- Internal world model
- Forward simulation of alternative plans

Language models as world models

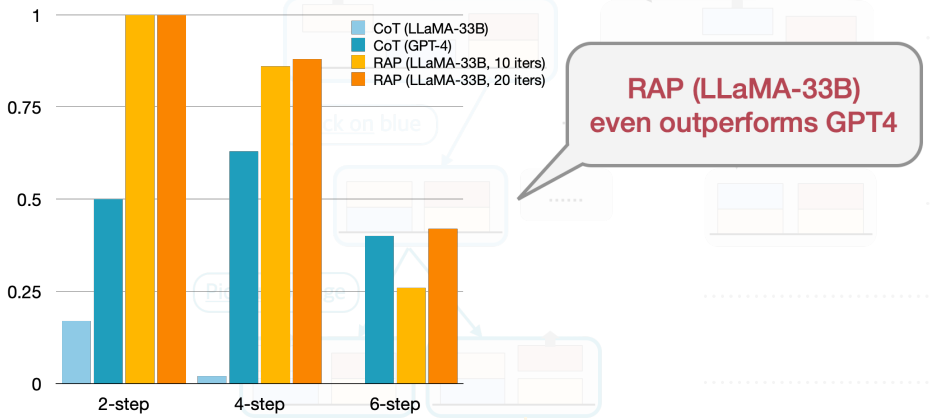
- Reasoning-via-Planning (RAP), Hao et al. (2023)

Monte Carlo Tree Search (MCTS):

Model-based planning w/ a small LM (as WM)

>

Autoregressive planning w/ a much larger LM



Using LMs to construct WMs for embodied reasoning

Language models as world models

- Why can LMs simulate the world?

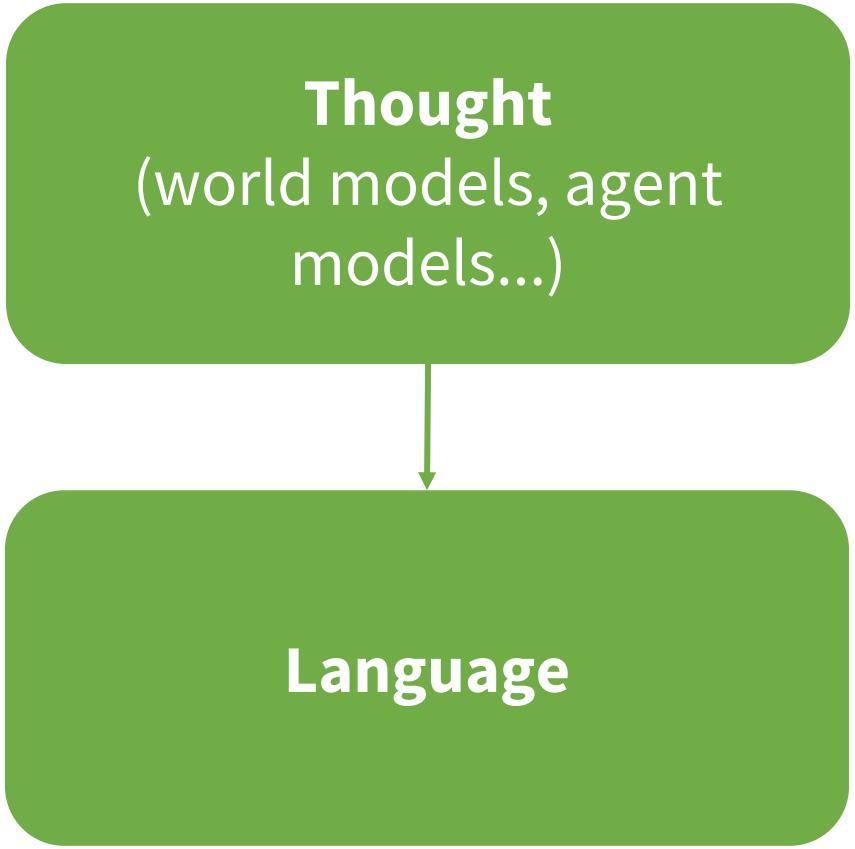
Knowledge of the world encoded in the training text

"If a wine glass falls onto the ground, it will break."

"If a basketball falls onto the ground, it will bounce back."

Language models as world models

- Language of thoughts (why text encodes world knowledge?)



Language models as world models

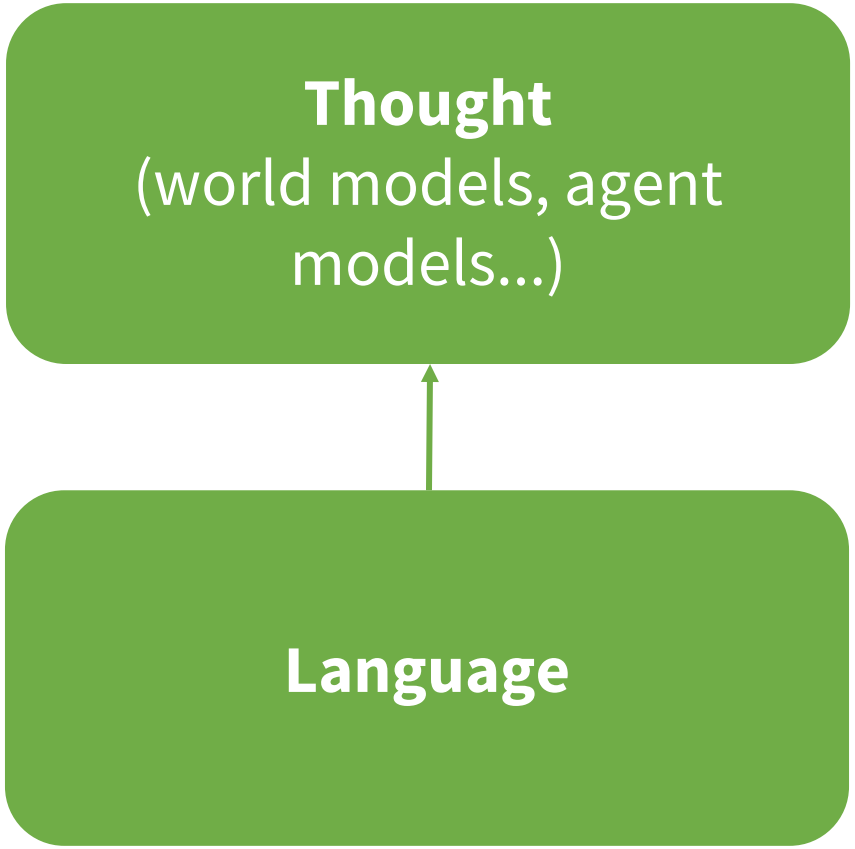
- Probabilistic language of thoughts

From Word Models to World Models:
Translating from Natural Language to the Probabilistic Language of Thought

Lionel Wong[1⋆], Gabriel Grand[1⋆], Alexander K. Lew[1], Noah D. Goodman[2], Vikash K. Mansinghka[1], Jacob Andreas[1], Joshua B. Tenenbaum[1]

⋆*Equal contribution.*

[1]MIT, [2]Stanford



Probabilistic programs

Probabilistic programs with a natural language interface

- Reasoning about the world using language. For example,
- Imagine a table with a red ball placed to the left of a blue ball. We can push the red ball and it hits the blue ball.
- Imagine that the red ball is pretty heavy. And the blue ball is fairly light.
- How fast does the blue ball move after the collision?

Text to probabilistic programs using LLMs

Text

Imagine a table with a red ball placed to the left of a blue ball. We can push the red ball and it hits the blue ball. Imagine that the red ball is pretty heavy. And the blue ball is fairly light.

Probabilistic program

```
(define choose_shapes...)
(define get_initial_color ...)
(define choose_mass ...)
(define get_initial_x...) ...

(define generate-object
  (mem (lambda (obj-id) (list
(pair 'object-id obj-id) (choose_shape obj-id)
(choose_color obj-id) (choose_mass obj-id)...)))) ...

(define generate-initial-scene-state...) ....

(define simulate-physics (mem (lambda (scene total_t delta_t)
  (let check_collisions ...)
  (let generate_next_scene_state_at_time...) ....))))
```

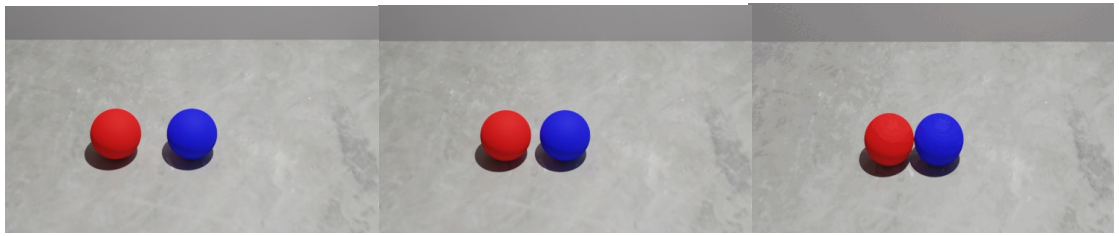
Attributes of objects

```
object-1: { color: red, shape: sphere, mass: 0.2, x: -3, v: 1.0,
            a: -0.05, force: 1.0 ...}
object-2: { color: blue, shape: sphere, mass: 3.0, x: 0, v: 0.0,
            a: 0.0, force: 0.0...}
```

Physics simulation engine

t=1	t=2	...	t=10
object-1: {..., x: -2.5, v: 0.95...}	object-1: {..., x: -2.0, v: 0.9...}		object-1: {..., x: 0.0, v: 0.01...}

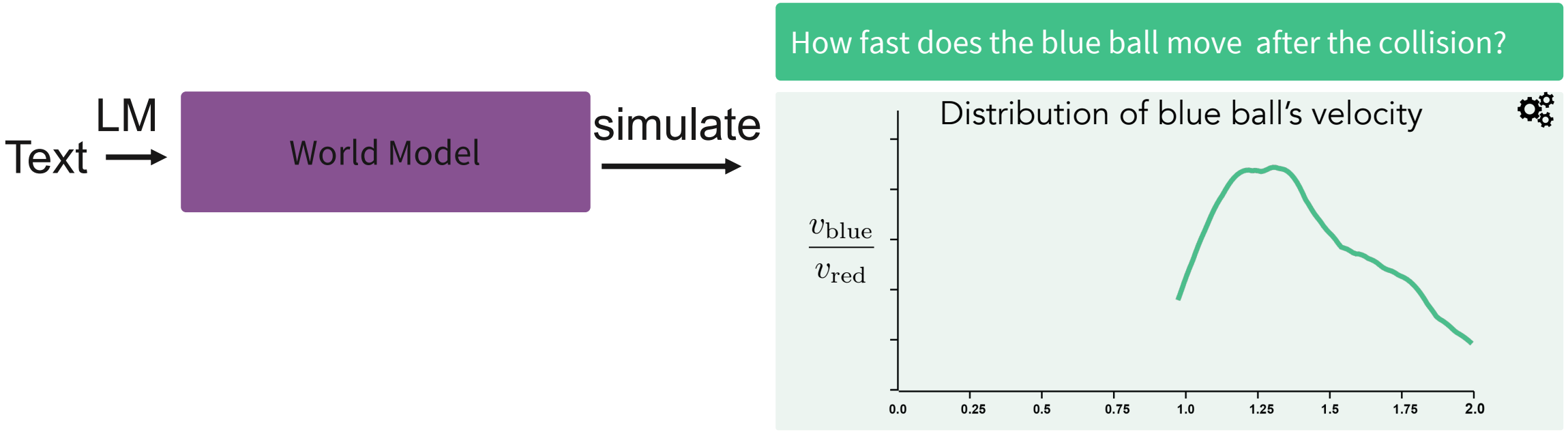
Graphics rendering engine



Gabe Grand, Lio Wong, et al. (2023)

Text to probabilistic programs using LLMs

Using LMs to construct WMs via *probabilistic programs* for language reasoning



Probabilistic reasoning → faster than the red ball's initial speed

Gabe Grand, Lio Wong, et al. (2023)