Imagination-Augmented Agents(I2A) for Deep Reinforcement Learning

Google DeepMind

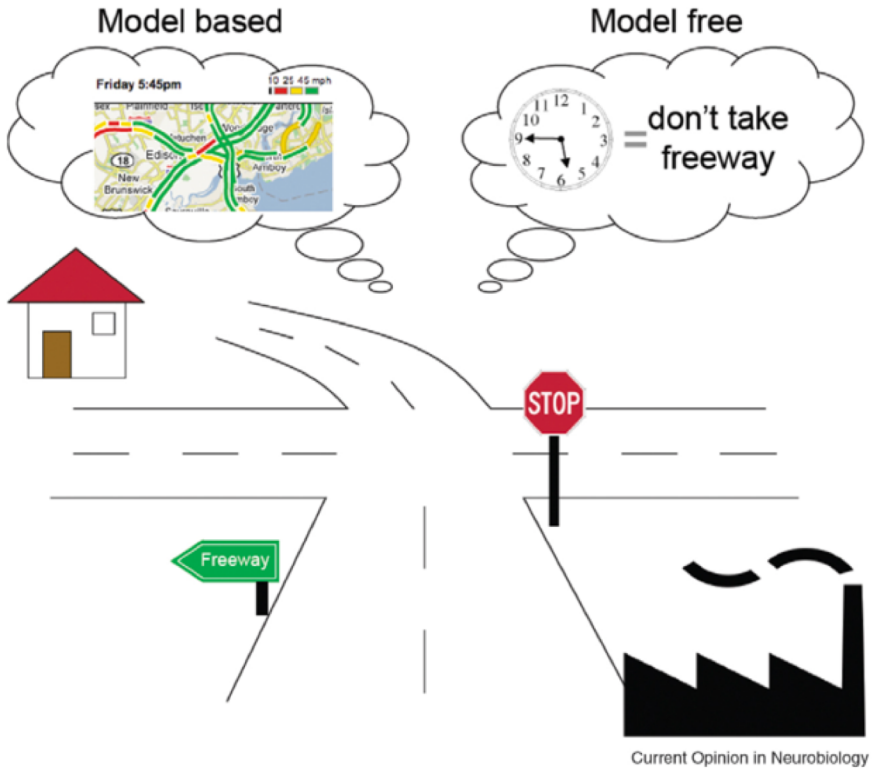
NIPS 2017

Model-free Reinforcement Learning

- Mapping raw observations directly to values and actions
  - many advances in this field
  - e.g., DQN, A3C(Asynchronous Advantage Actor-Critic)
- However:
  - data inefficiency
  - lack of generalization

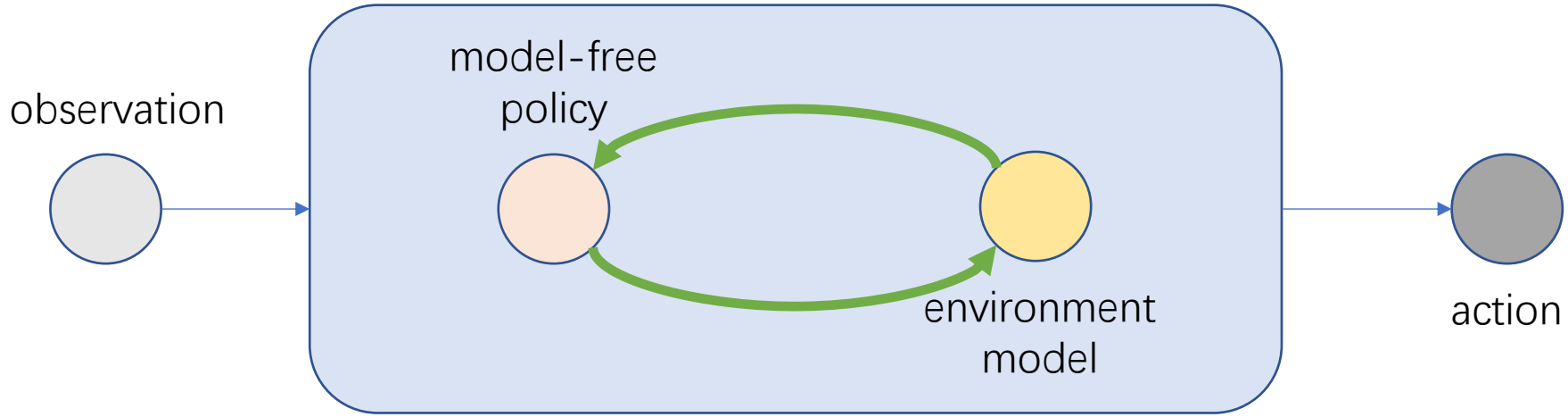
Model-based Reinforcement Learning

- Build the agent with model of the environment
  - better generalization
  - imagination: reasoning about the future using the model
- High dimensional environments:
  - costly model
  - inaccurate model leads to catastrophe



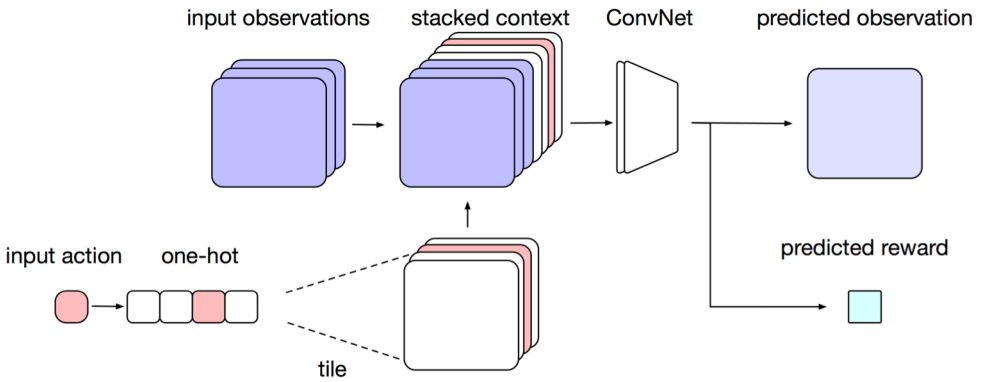
Motivation

- Combining both model-based and model-free aspect
  - Learn a model of the environment
  - Query the model for predictions
  - Learn to interpret the predictions in order to perform actions



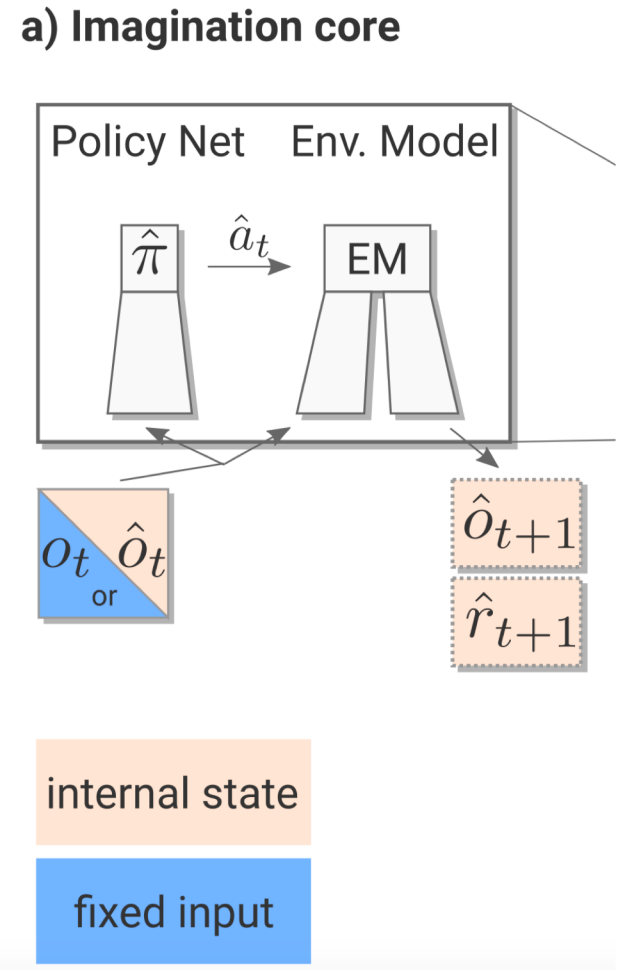
I2A architecture – Environment model

- Environment model:
  - recurrent model
  - makes environment prediction at each time step
- Training of the environment model:
  - using cross entropy loss l_{model}
  - either trained before DRL, or add l_{model} as auxiliary loss to the total loss



I2A architecture – Rollout policy

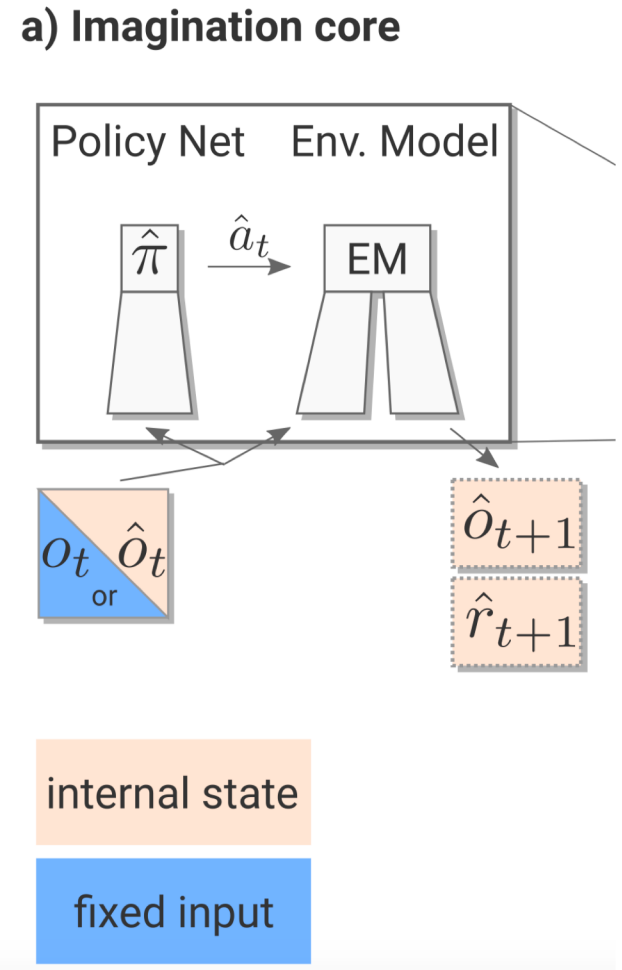
- Rollout policy net:
  - imagine future actions
  - trained by imitating output of the overall I2A using cross entropy loss



I2A architecture – Imagination core

- Imagination core:
  - Environment model + Rollout policy
  - predicts next time step using current observation and action

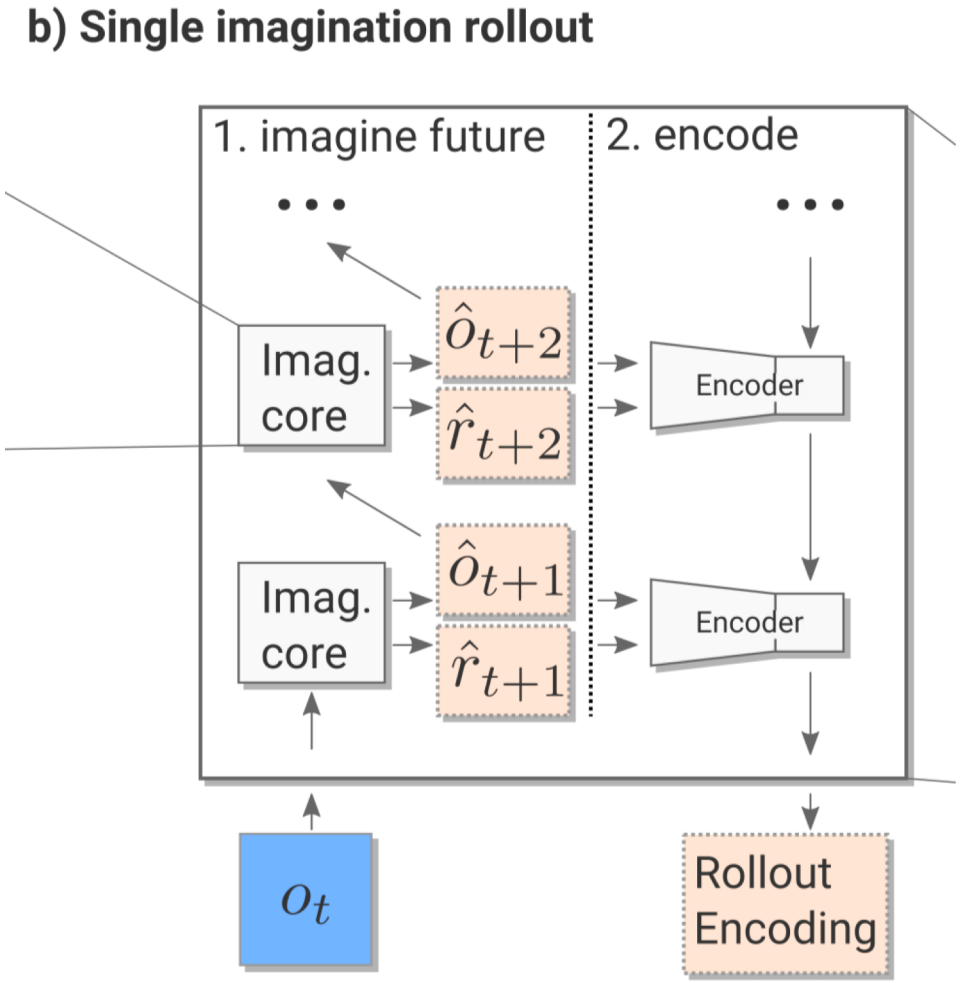
By calling **Imagination core** over multiple time steps, we make the agent imagine the future



I2A architecture – Rollout

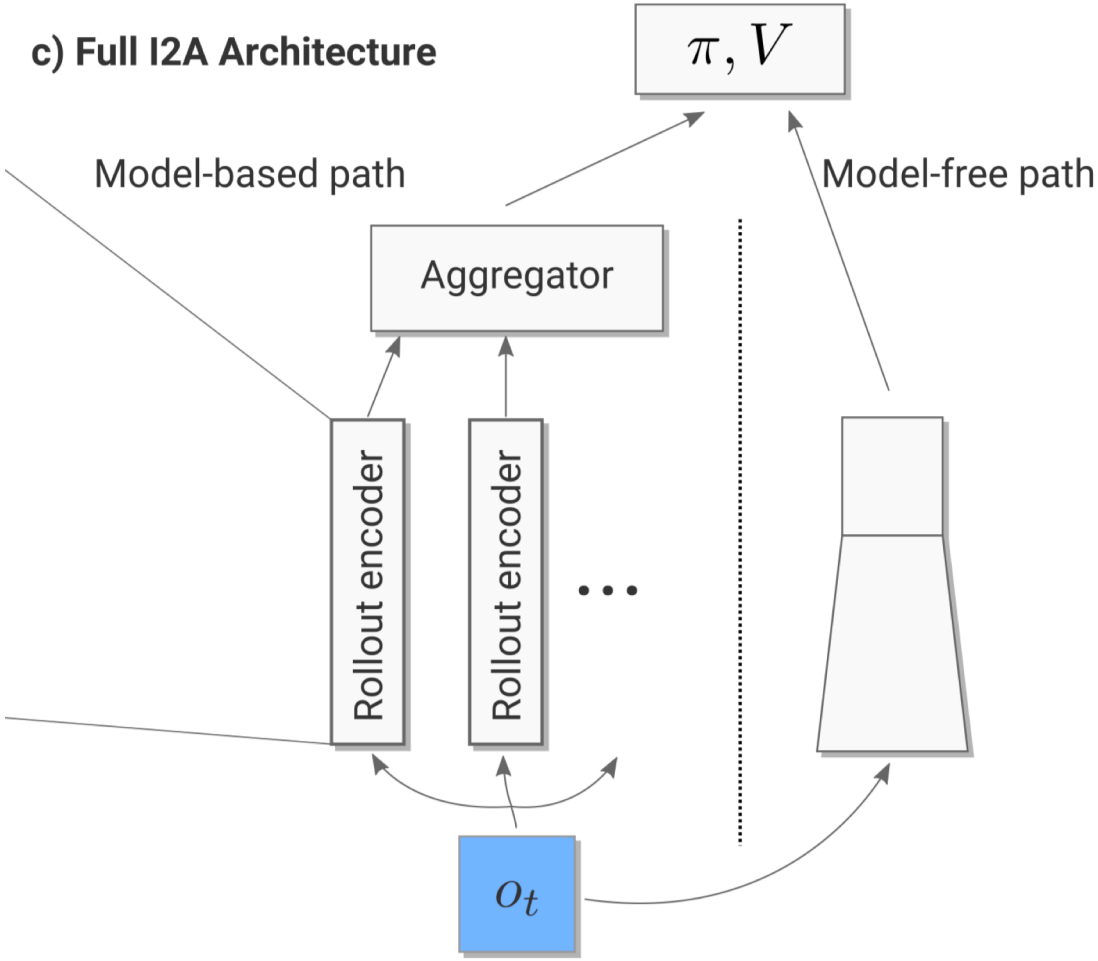
Imagination about the future is inaccurate and with redundant information

Using rollout encoder to process all the observations along the rollout and learn to interpret it



I2A architecture

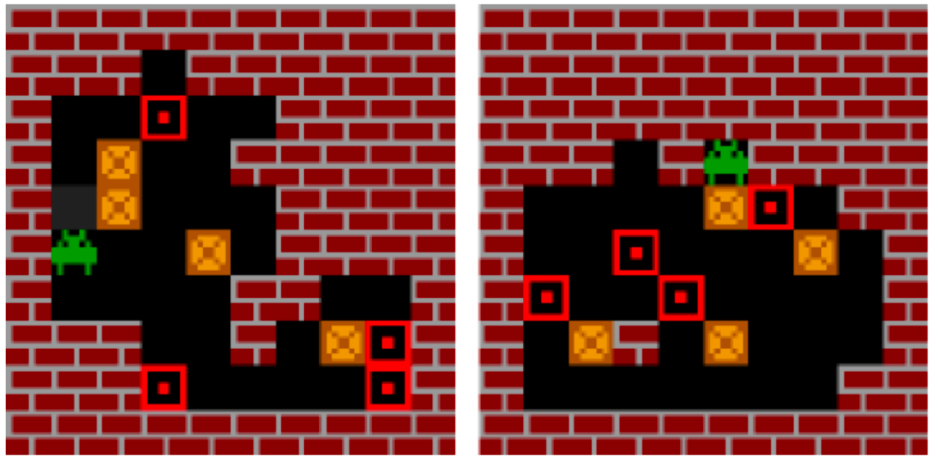
- With aggregator, we obtain an **imagination-augmented code**
- With classic model-free method, we compute the **model-free code**
- Combine both code to compute the **imagination-augmented policy** and **value** with one fully connected layer



A demo code without RL training of the agent can be found at
https://github.com/vasiloglou/mltrain-nips-2017/blob/master/sebastien_racaniere/I2A%20-%20NIPS%20workshop.ipynb

Experiment: Sokoban

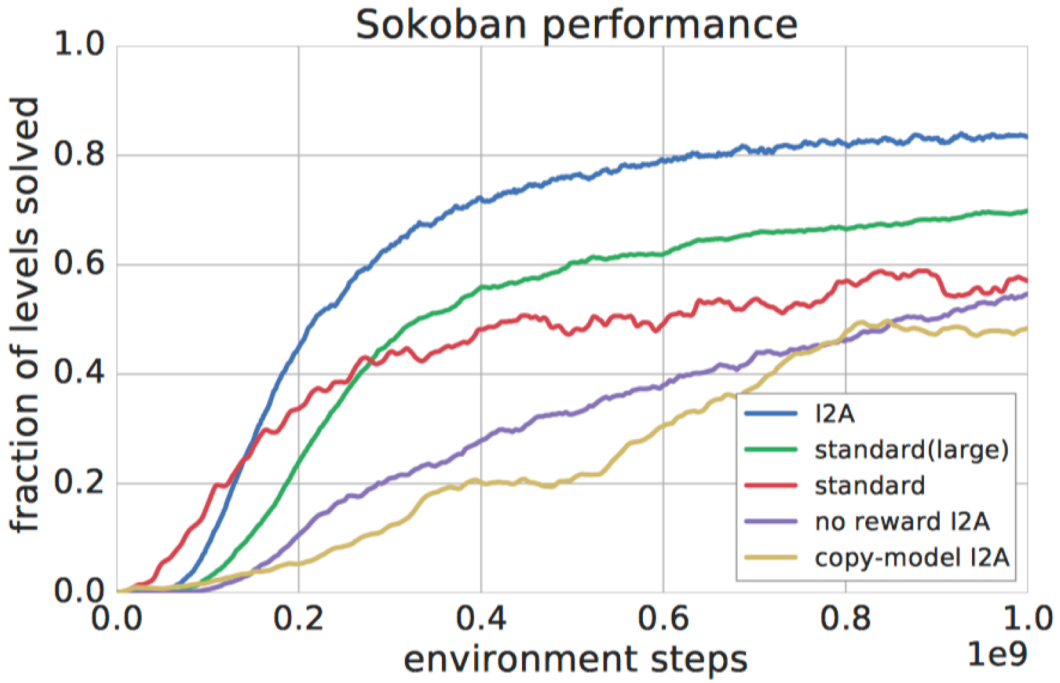
- Player (green sprite) needs to push all 4 boxes onto the red target squares to solve a level
- Difficulty:
  - no immediate reward: the red dot next to the box may not be the feasible solution
  - irreversible move: may cause game insolvable



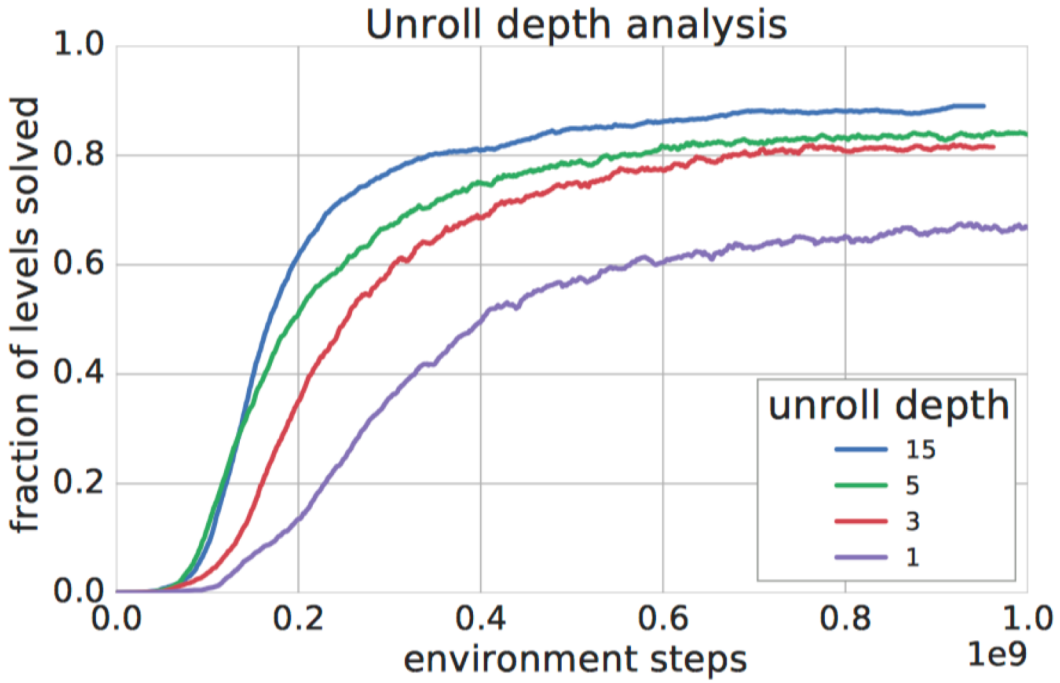
Experiment: Setting

- Using pre-trained environment model
- Using A3C as the model-free method
- Trained over 32 to 64 workers

Experiment: Sokoban

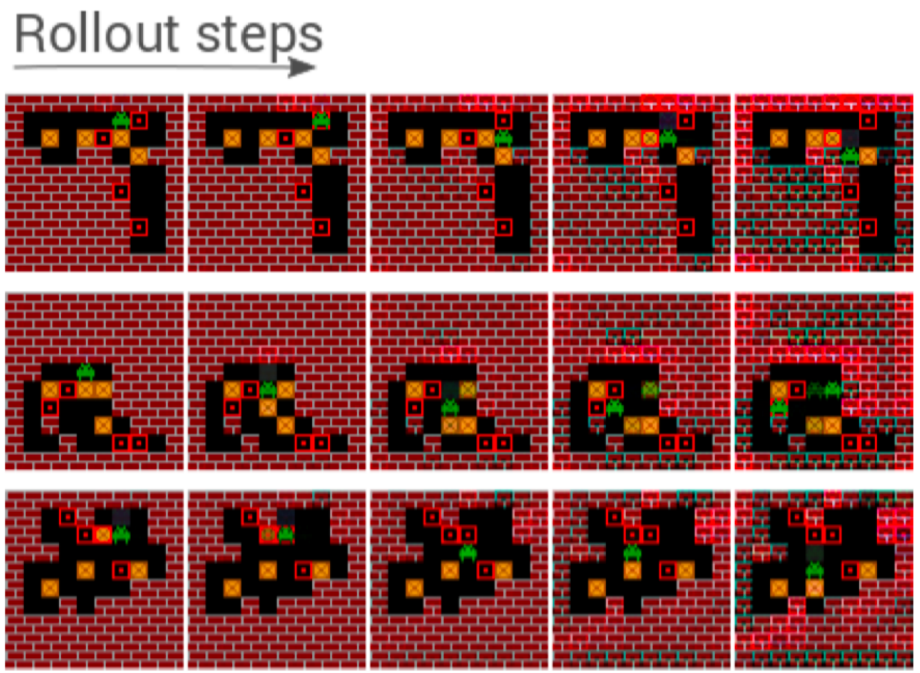


Outperform standard model-free method and I2A without model



Deeper rollout produces better results

Experiment: Sokoban



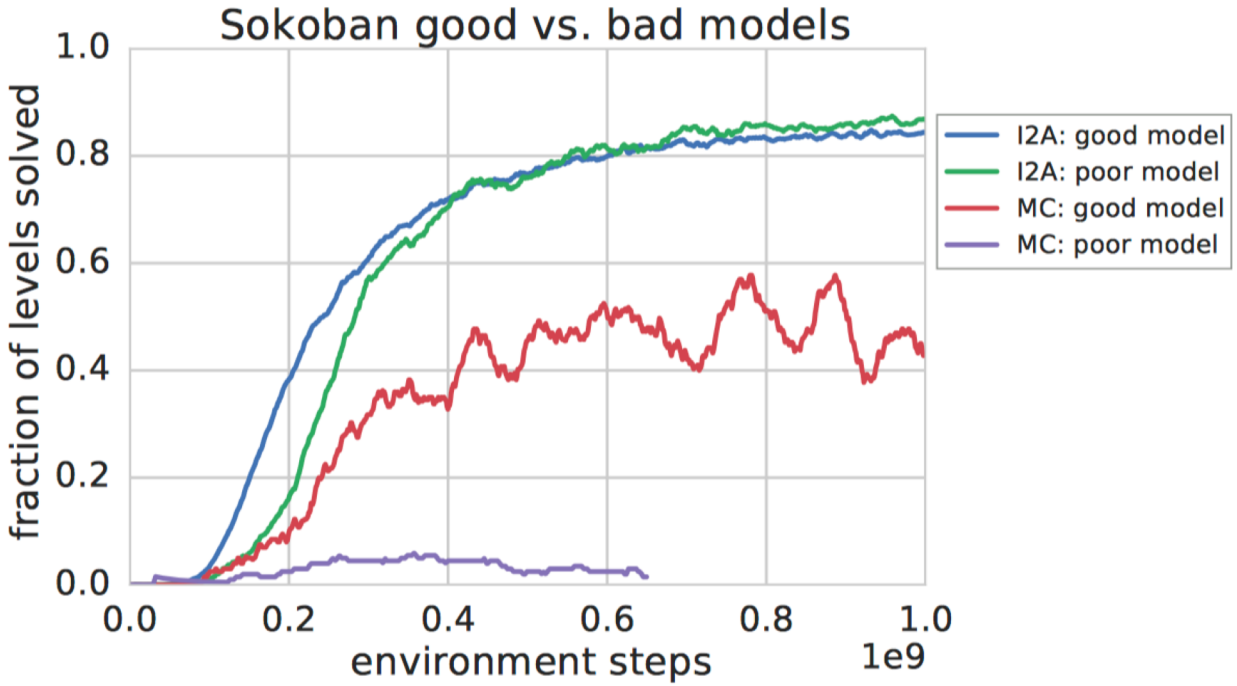


Figure 5: *Experiments with a noisy environment model. Left:* each row shows an example 5-step rollout after conditioning on an environment observation. Errors accumulate and lead to various artefacts, including missing or duplicate sprites. *Right:* comparison of Monte-Carlo (MC) search and I2A when using either the accurate or the noisy model for rollouts.

Experiment: Sokoban

I2A@87	$\sim$ 1400
I2A MC search @95	$\sim$ 4000
MCTS@87	$\sim$ 25000
MCTS@95	$\sim$ 100000
Random search	$\sim$ millions

Table 1: Imagination efficiency of various architectures.

More efficient than MCTS method with a value network (similar to AlphaGo)

Boxes	1	2	3	4	5	6	7
I2A (%)	99.5	97	92	87	77	66	53
Standard (%)	97	87	72	60	47	32	23

Table 2: Generalization of I2A to environments with different number of boxes.

Better generalization

Conclusion

- Summary
  - Imagination: I2A uses internal environment model to predict future
  - Agent learns to interpret plans from internal environment model and combines it with model-free result
- What I learn
  - An end-to-end agent with model-free RL method may have poor result due to inefficient amount of data
  - Combining the method with a simple model of the environment can improve the result