Deep Reinforcement Learning

Ivaylo Popov
Research Data Scientist
Ocado Technology

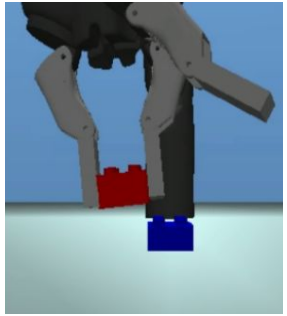
Motivation

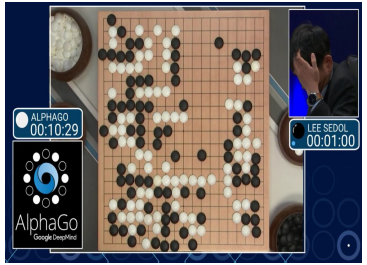
Research in Artificial intelligence

- Emergent behavior
- Multi-agent behavior
- Vision and control architectures
- Planning

Learning environments

- Atari
- Board games: Go, etc.
- Physics simulators: MuJoCo, Bullet
- OpenAI Gym, Universe
- DeepMind Lab
- Starcraft II





ocado TECHNOLOGY

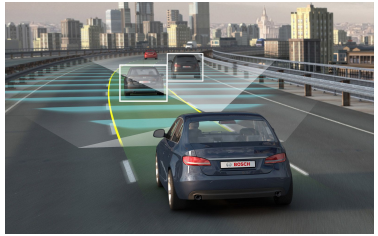
Motivation–continued

Robotics

- Manipulation
- Locomotion

Autonomous vehicles

- Aerial (e.g. drones, helicopters)
- Ground (e.g. cars, industrial robots)



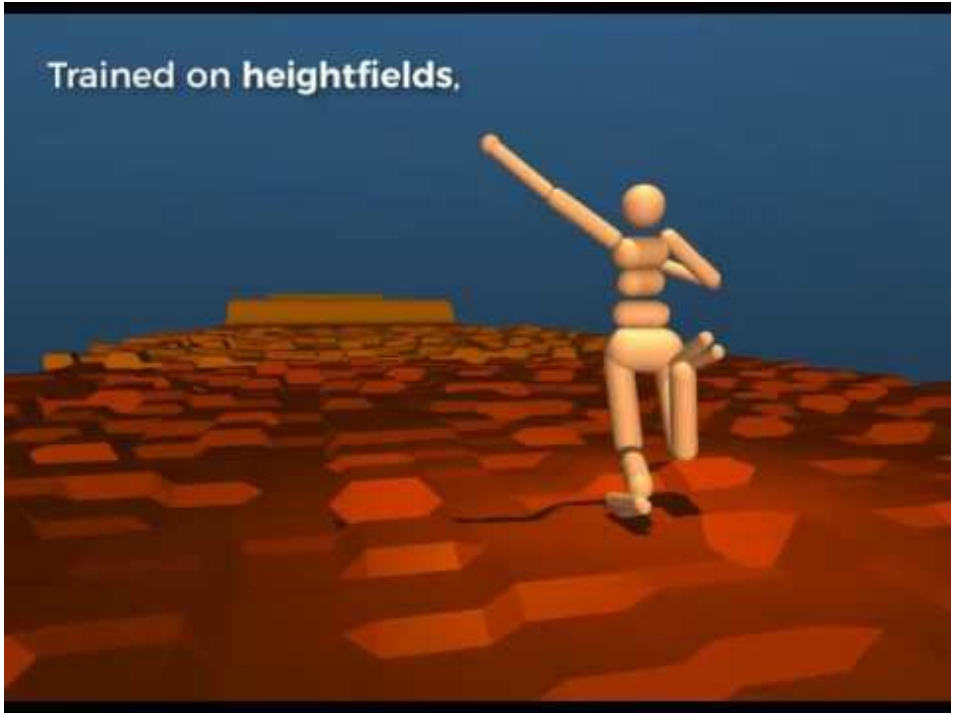
Factory and warehouse control

Business applications

- Marketing / sales automation
- Support

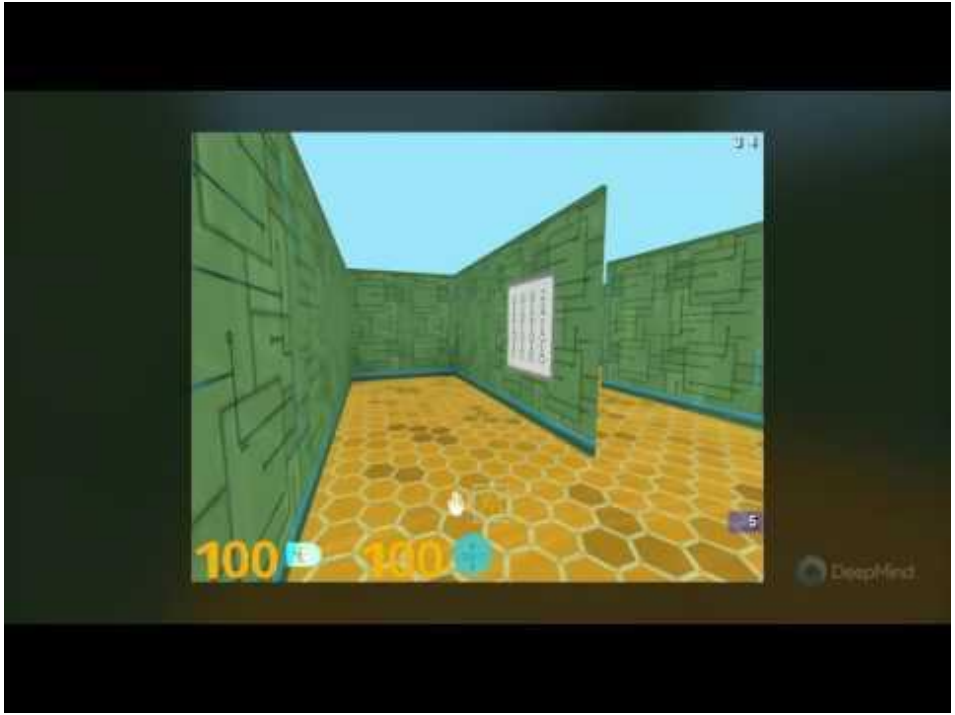
ocado TECHNOLOGY

Complex locomotion behaviors (DeepMind)



ocado TECHNOLOGY

3D maze navigation (DeepMind)



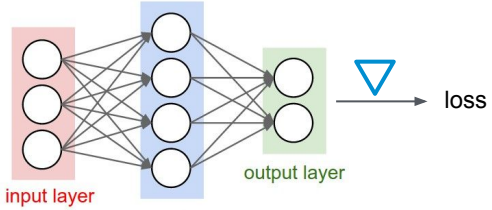
ocado TECHNOLOGY

Robotic picking of objects (Google Brain)

ocado TECHNOLOGY

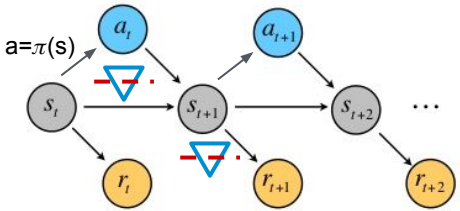
How is RL different to deep learning

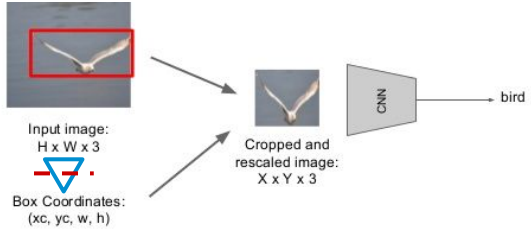
Deep learning: differentiable loss function and model



RL: no differentiable loss function given

- Sequential decision processes
- Non-differentiable parts of a model (e.g. “hard” attention)





Sequential decision processes

s_t - observation state

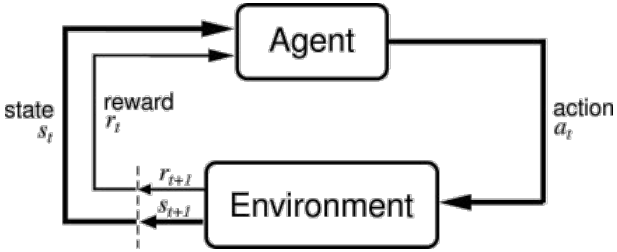
a_t - action

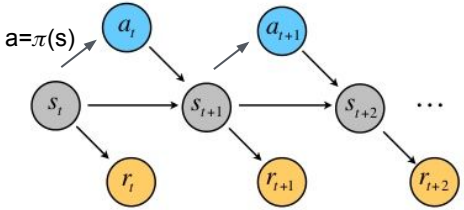
$P(r_{t+1}, s_{t+1} | s_t, a_t)$ - transition probability

r_t - reward

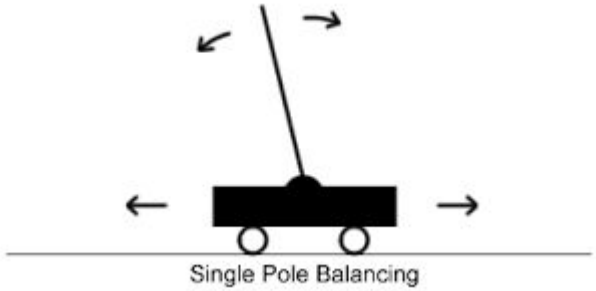
Goal: maximize cumulative reward

$\max_a R_t = r_t + r_{t+1} + \cdots + r_T$



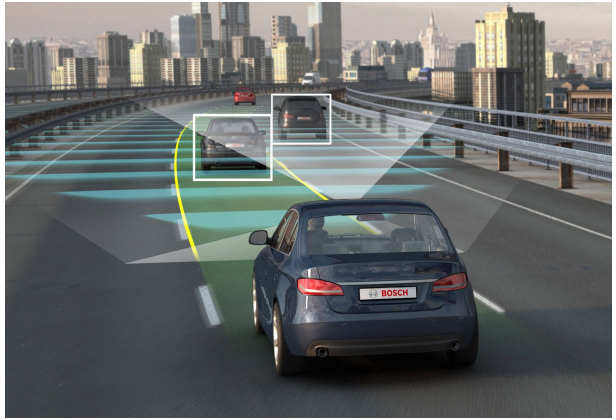


Example–cartpole



Goal	Keep pole upright
State (s)	Pole position and angular velocity Cart position and horizontal velocity
Actions (a)	Push cart left / right
Reward (r)	+1 x each step before failure
Episode	Until failure or 50 steps reached

Example–autonomous driving



Goal	Move car to destination adhering to safety constraints
State (s)	Camera, lidar, GPS Wheel velocity and position Accelerometer
Actions (a)	Steering wheel position Acceleration pedal position Breaking pedal position
Reward (r)	-1 x GPS distance to destination (shaping) -Fi if failure type i triggered (e.g. speeding, crash)

ocado TECHNOLOGY

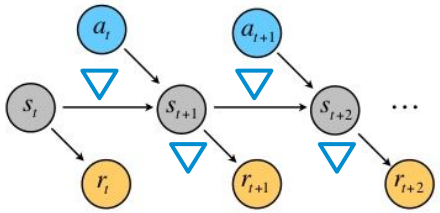
Model-based or planning methods

Model types

- Model known (e.g. board games)
- Hand-engineered (e.g. physics models)
- Learnt (e.g. neural networks on collected data)

Continuous systems

- Backpropagate through system
- Linear / nonlinear dynamics optimization



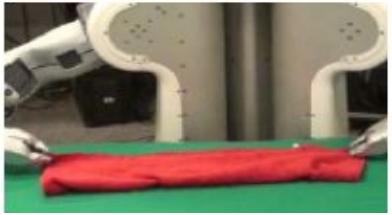
Discrete systems

- Monte Carlo tree search (MCST)

ocado TECHNOLOGY

Challenges with dynamics models

- Model engineering very hard
  - Ambiguous state
  - Unstructured environments
  - Deformable objects
  - Changing environments
- Optimal policy often much simpler
- Long control sequences



ocado TECHNOLOGY

Model-free reinforcement learning

Policy-based (Actor)

- Back-box optimization
- Policy gradient

Value-based algorithms (Critic)

- Monte Carlo learning
- Temporal difference learning

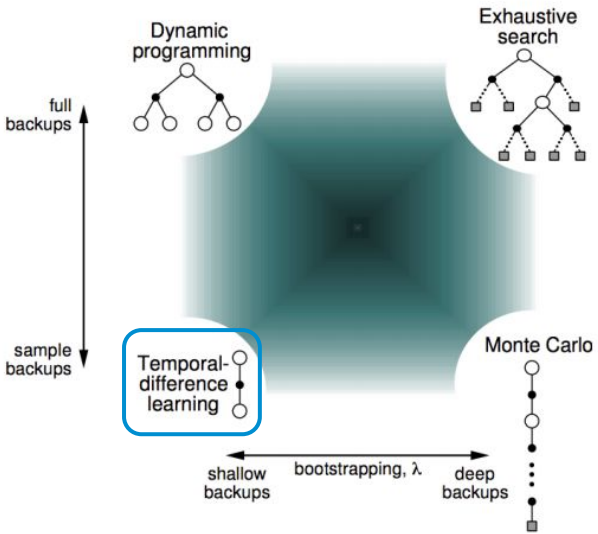
ocado TECHNOLOGY

Value-based methods

Value function $V^{\pi}(s) = E[r_0 + r_1 + r_2 + \ldots \mid s_0 = s]$

Action-value function $Q^{\pi}(s, a) = E[r_0 + r_1 + r_2 + \ldots \mid s_0 = s, a_0 = a]$

Advantage function $A^{\pi}(s, a) = Q^{\pi}(s, a) - V^{\pi}(s)$



- Monte Carlo
  Sampling instead of full summation
- Bootstrapping
  Estimates of the value in state s' instead of full trajectories

ocado TECHNOLOGY

Temporal difference learning

Temporal difference learning - estimating value function of a policy

$$V^{\pi}_{i+1}(s) = E_{\pi}[r + \gamma V^{\pi}_{i}(s') \mid s]$$

Q-learning - estimating the optimal action-value function

$$Q^{*}_{i+1}(s,a) = E_{s',s,a}[r + \gamma \max_{a'} Q^{*}_{i}(s',a') \mid s,a]$$

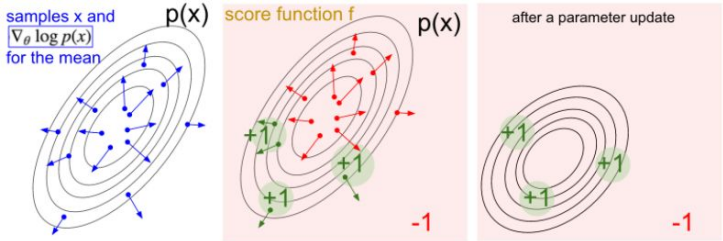
ocado TECHNOLOGY

Policy gradient methods

Optimize policy by deriving gradient of R w.r.t. policy parameters $\boldsymbol{\theta}$

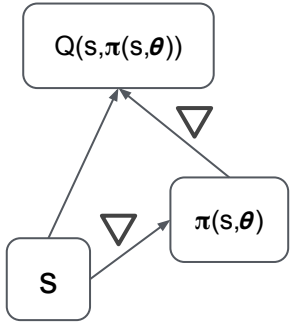
Policy gradient (stochastic policies)

$$\nabla_\theta \mathbb{E}_\tau [R] = \mathbb{E}_\tau \left[R \nabla_\theta \sum_{t=0}^{T-1} \log \pi(a_t \mid s_t, \theta) \right]$$

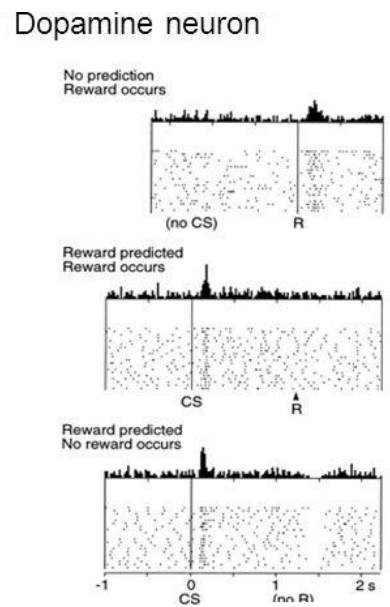


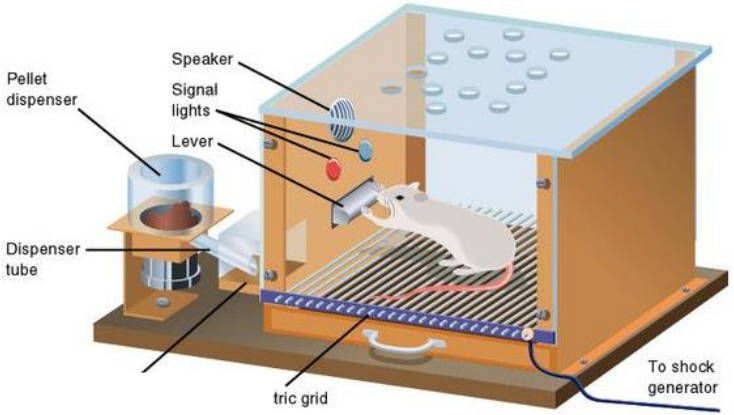
Deterministic policy gradient

$$\nabla_\theta \mathbb{E}_\tau [R] = \mathbb{E}_\tau \left[\sum_{t=1}^{T} \nabla_\theta Q(s_t, \pi(s_t, z_t; \theta)) \right]$$



Related mechanisms in animal brains





Dopamine neurons encode TD error (Schultz, 1997)

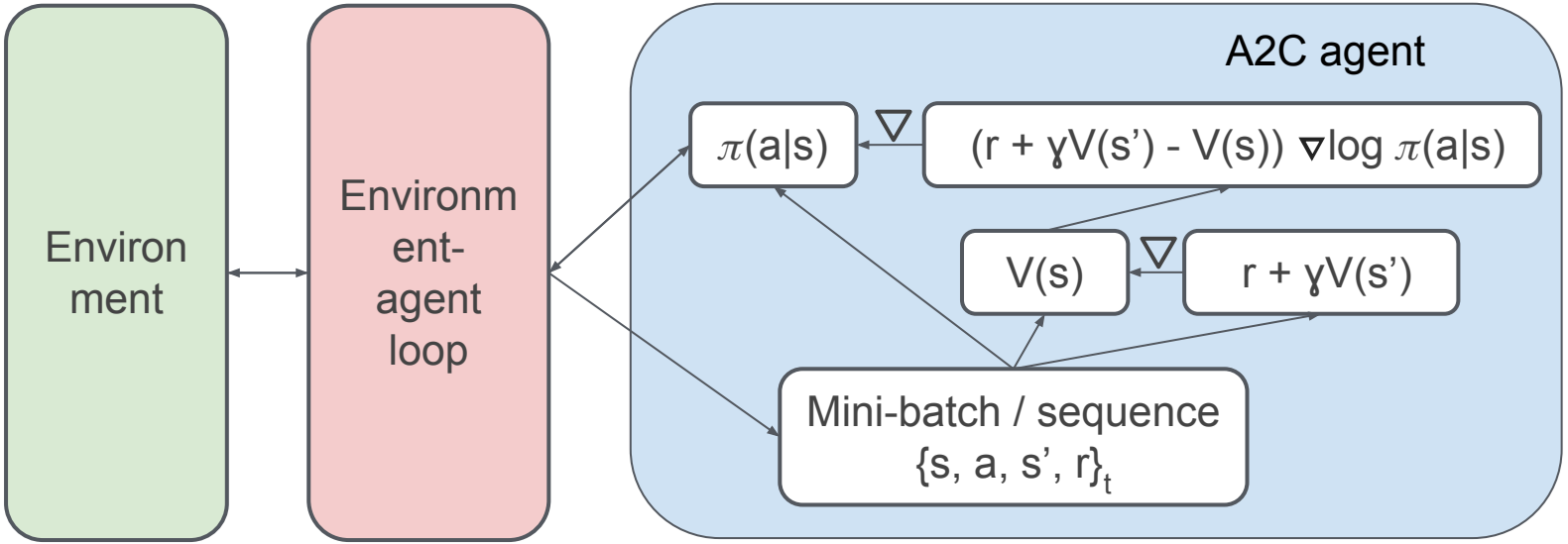
Operant conditioning (Skinner, 1948)

Deep reinforcement learning algorithms

- Advantage actor-critic (A2C)
  - Stochastic policy gradient
  - TD learning for V^{π}
- Deep deterministic policy gradient (DDPG)
  - Deterministic policy gradient
  - TD learning for Q^{*}

ocado TECHNOLOGY

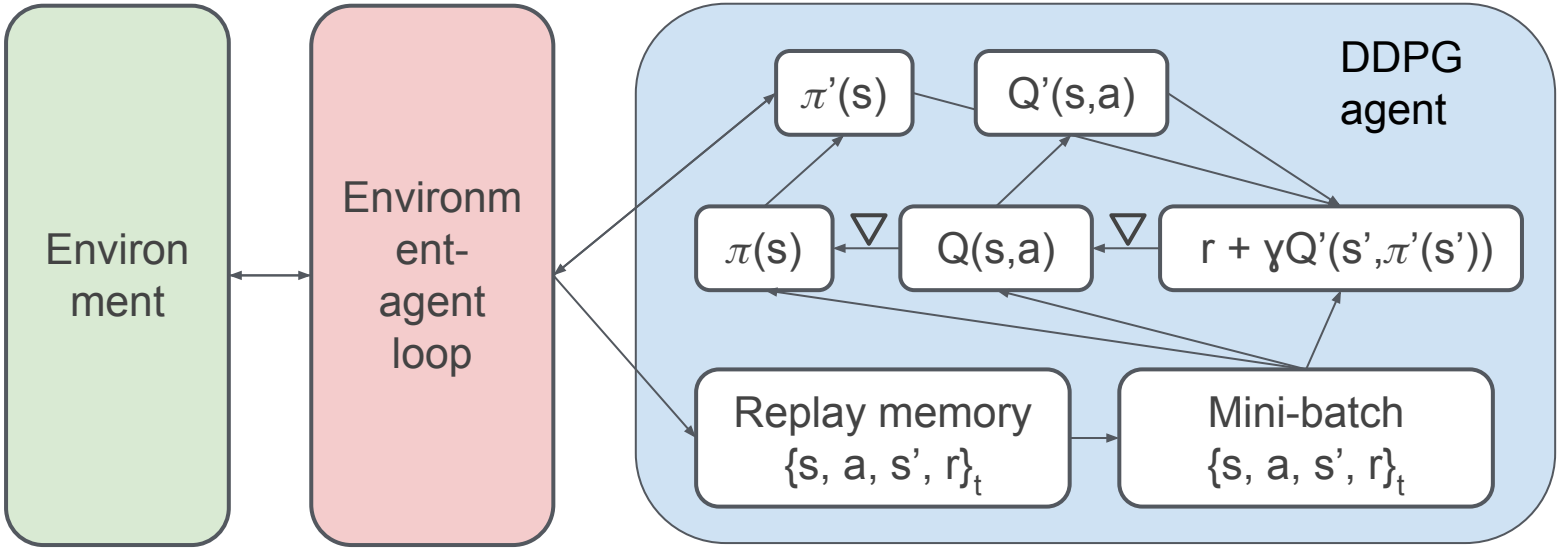
Advantage actor-critic (A2C)



- Deep networks for V(s) and π(a|s)
- TD learning and policy gradient
- Advantage estimate to reduce variance of policy gradient

ocado TECHNOLOGY

Deep deterministic policy gradient (DDPG)



- Deep networks for Q(s,a) and π(s)
- Q-learning + Deterministic policy gradient
- Replay memory + Target networks Q' and π'(s)

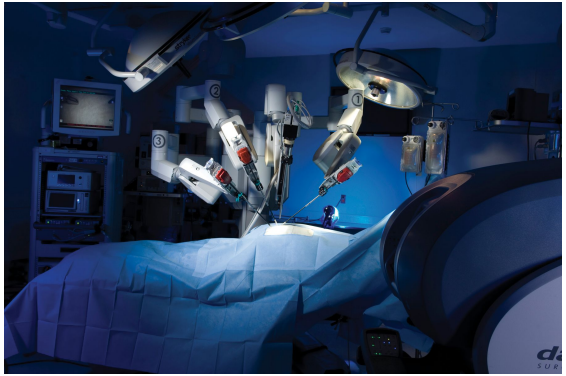
Advanced research topics

Efficient exploration

- Data-efficient algorithms
- Curriculum learning
- Auxiliary objectives
- Imitation learning
- Transfer learning

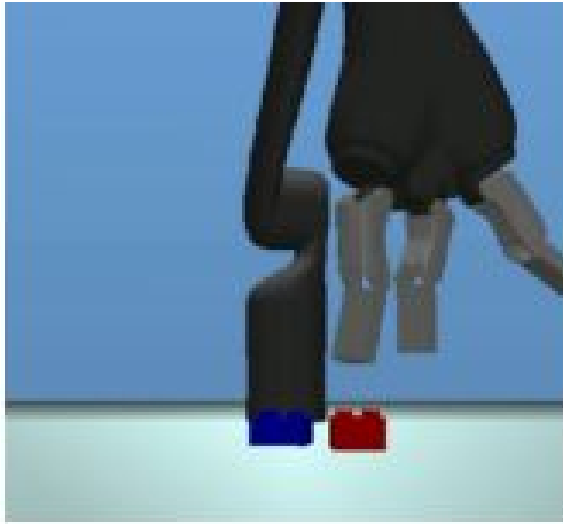
Safe exploration

- Hard control constraints
- Curriculum learning
- Transfer learning (e.g. from simulation)



ocado TECHNOLOGY

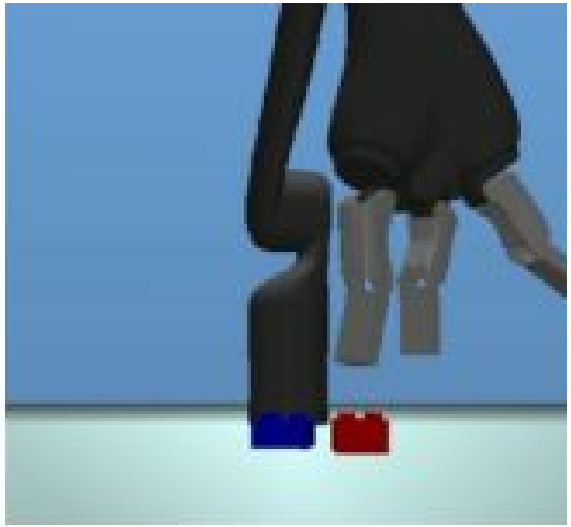
Exploration



Goal	Stack red brick on blue one
Reward	+1 if bricks stacked (red on blue)
Outcome	Initial random agent never sees the reward
Solutions	• Curriculum learning • Shaping rewards • Instructive starting states • Learning from human demonstrations

ocado TECHNOLOGY

Data-efficiency



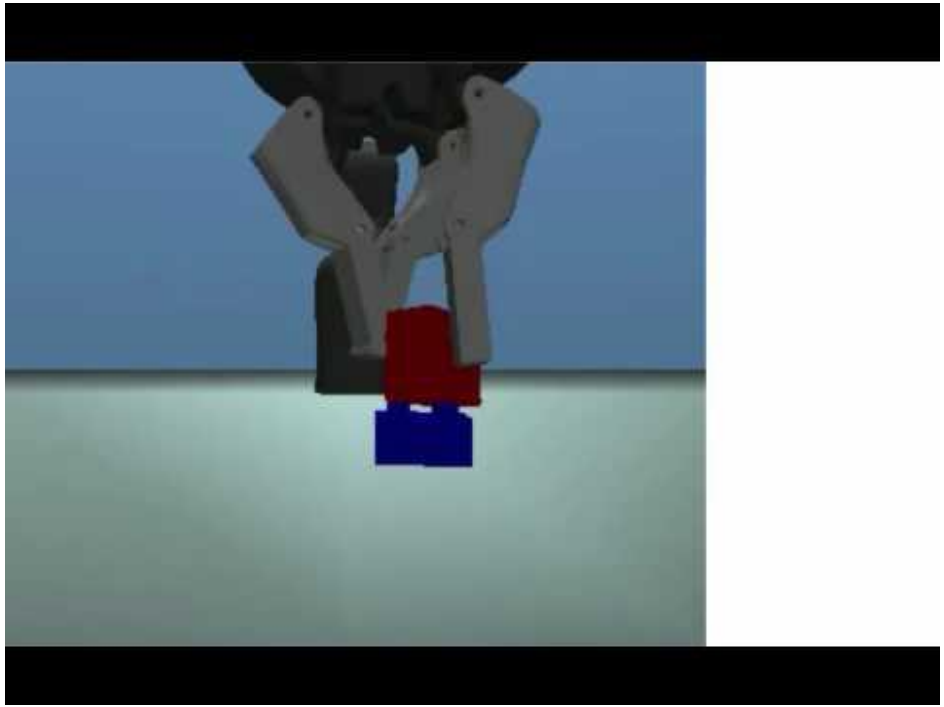
Situation Agent sees first reward after 1 million steps of exploration

Problem Most algorithms waste all this previous experience

Solutions Store all experience in replay memory
Perform a lot off-policy training before next environment interaction step

ocado TECHNOLOGY

End-to-end stacking with DDPG



Vanilla DDPG algorithm

+

Asynchronous agent (16x)

Large number of replay steps

Sub-task shaping rewards

Instructive states

+

4 days of training

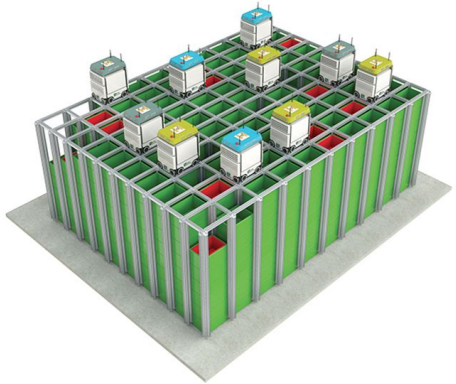
(4 weeks from pixels)

Popov et al., 2017. Data-efficient Deep Reinforcement Learning for Dexterous Manipulation

ocado TECHNOLOGY

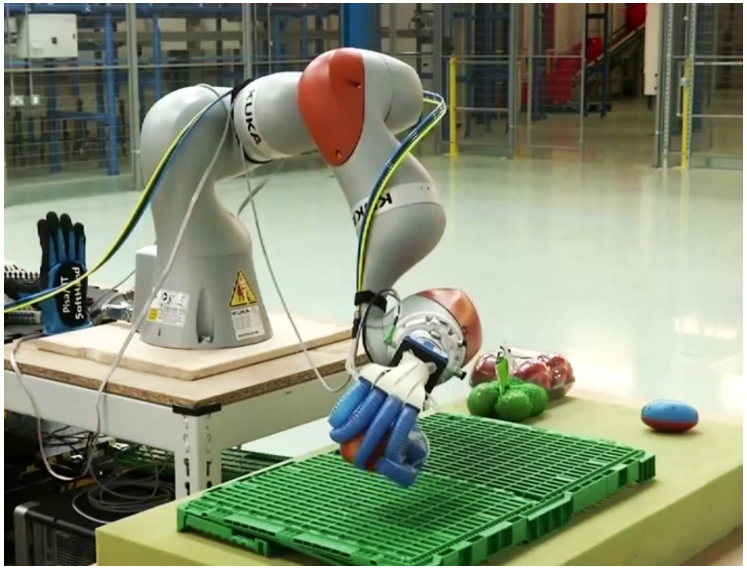
Reinforcement learning in Ocado

- Robotic picking of food items
- OSP bot control
- OSP full grid control
- Product recommendation
- Chatbot systems
- Self-driving vehicles
- Many other...



ocado TECHNOLOGY

Dexterous manipulation for picking



Observations	Camera input Arm joint and finger positions Pressure sensors
Actions	Arm joint and finger torque or velocity
Reward	+1 for successful picks -1 for episodes terminated due to safety constraints
Episode	Fixed length (e.g. 15 sec)
Exploration strategy	Human demonstrations Curriculum

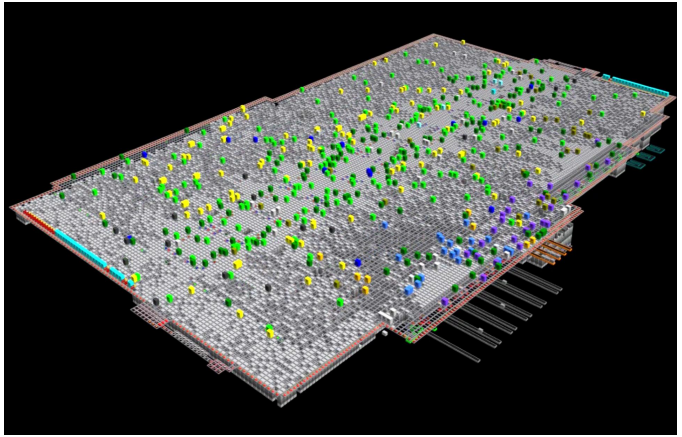
ocado TECHNOLOGY

Bot motion control

Observations	Wheel position sensors Track, torque sensors Starting absolute grid location Camera / distance sensors Accelerometers Bot state (errors, battery, etc.)
Actions	Wheel motor torques Parking motor positions
Reward	-1 x deviation from target positions -S x deviation from max speed -A x deviation from max acceleration -C_i x entering bot failure state si
Episode	Fixed length (e.g. 10 sec)
Exploration strategy	Not necessary (rewards not sparse)

ocado TECHNOLOGY

Full grid control



Observations	Current list of orders Location and state of all bots State of all stations Content of all 3D grid cells
Actions	Discrete control of all bots Discrete control of all stations
Reward	+1 for correctly picked order bag $-C_i$ for various costs: bot moves, station utilization, bot failure
Episode	Full operation cycle (hours)
Exploration strategy	Demonstrations from prior systems

ocado TECHNOLOGY

Resources

- Deep learning / Machine learning resources (see here)
- Books
  - Reinforcement Learning: An Introduction (Sutton and Barto)

    http://incompleteideas.net/sutton/book/the-book-2nd.html
- Lectures and courses
  - David Silver (UCL) http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching.html
  - Sergey Levine (UC Berkley) http://rll.berkeley.edu/deeprlcoursesp17/
  - Peter Abbeel (NIPS Tutorial) https://people.eecs.berkeley.edu/...Schulman-Abbeel.pdf
- Algorithm implementations
  - https://github.com/openai/baselines

ocado TECHNOLOGY

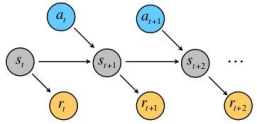
Resources–continued

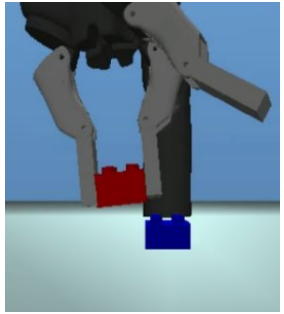
- Learning environments
  - https://deepmind.com/blog/open-sourcing-deepmind-lab/
  - https://github.com/deepmind/pysc2
  - https://github.com/openai/gym
  - https://github.com/openai/roboschool
  - https://github.com/openai/universe
- Blog posts and other
  - https://deepmind.com/blog/deep-reinforcement-learning/
  - http://karpathy.github.io/2016/05/31/rl/
  - https://github.com/aikorea/awesome-rl

ocado TECHNOLOGY

Summary

- Applications of RL
- Theory and examples
- Popular algorithms
- Advanced topics
- Ocado case studies





Thank you!

ivaylo.popov @ ocado.com

ocado TECHNOLOGY