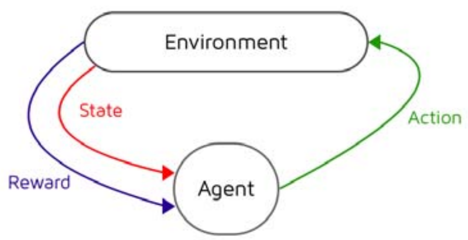
Accelerating Reinforcement Learning in Engineering Systems

Tham Chen Khong
with contributions from Zhou Chongyu and Le Van Duc
Department of Electrical & Computer Engineering
National University of Singapore

Reinforcement Learning

- An agent learns to perform actions that maximize long term rewards
- Discovers solutions through interaction with the environment



- Different from supervised learning which a lot of deep learning is used for
- Deep learning can still be applied for RL, but target value changes
- RL is a powerful technique, but is usually computationally expensive and may be impractical since it requires many interactions with the environment

ARTICLE

doi:10.1038/nature24270

19 Oct 2017

Mastering the game of Go without human knowledge

David Silver[1*], Julian Schrittwieser[1*], Karen Simonyan[1*], Ioannis Antonoglou[1], Aja Huang[1], Arthur Guez[1], Thomas Hubert[1], Lucas Baker[1], Matthew Lai[1], Adrian Bolton[1], Yutian Chen[1], Timothy Lillicrap[1], Fan Hui[1], Laurent Sifre[1], George van den Driessche[1], Thore Graepel[1] & Demis Hassabis[1]

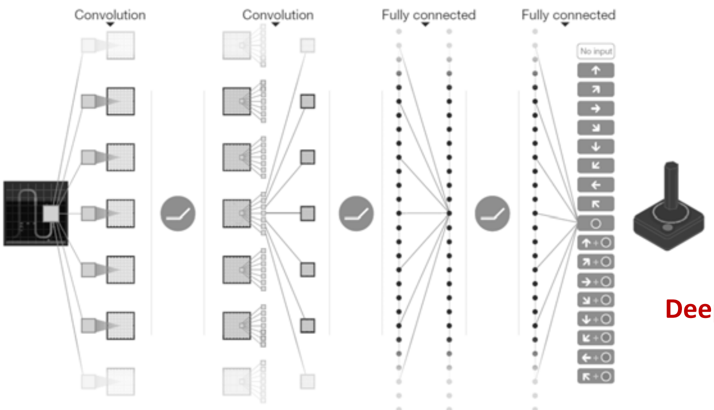
A long-standing goal of artificial intelligence is an algorithm that learns, *tabula rasa*, superhuman proficiency in challenging domains. Recently, AlphaGo became the first program to defeat a world champion in the game of Go. The tree search in AlphaGo evaluated positions and selected moves using deep neural networks. These neural networks were trained by supervised learning from human expert moves, and by reinforcement learning from self-play. Here we introduce an algorithm based solely on reinforcement learning, without human data, guidance or domain knowledge beyond game rules. AlphaGo becomes its own teacher: a neural network is trained to predict AlphaGo's own move selections and also the winner of AlphaGo's games. This neural network improves the strength of the tree search, resulting in higher quality move selection and stronger self-play in the next iteration. Starting *tabula rasa*, our new program AlphaGo Zero achieved superhuman performance, winning 100–0 against the previously published, champion-defeating AlphaGo.

Much progress towards artificial intelligence has been made using supervised learning systems that are trained to replicate the decisions of human experts[1–4]. However, expert data sets are often expensive, unreliable or simply unavailable. Even when reliable data sets are available, they may impose a ceiling on the performance of systems trained in this manner[5]. By contrast, reinforcement learning systems are trained from their own experience, in principle allowing them to exceed human capabilities, and to operate in domains where human expertise is lacking. Recently, there has been rapid progress towards this goal, using deep neural networks trained by reinforcement learning. trained solely by self-play reinforcement learning, starting from random play, without any supervision or use of human data. Second, it uses only the black and white stones from the board as input features. Third, it uses a single neural network, rather than separate policy and value networks. Finally, it uses a simpler tree search that relies upon this single neural network to evaluate positions and sample moves, without performing any Monte Carlo rollouts. To achieve these results, we introduce a new reinforcement learning algorithm that incorporates lookahead search inside the training loop, resulting in rapid improvement and precise and stable learning. Further technical differences in

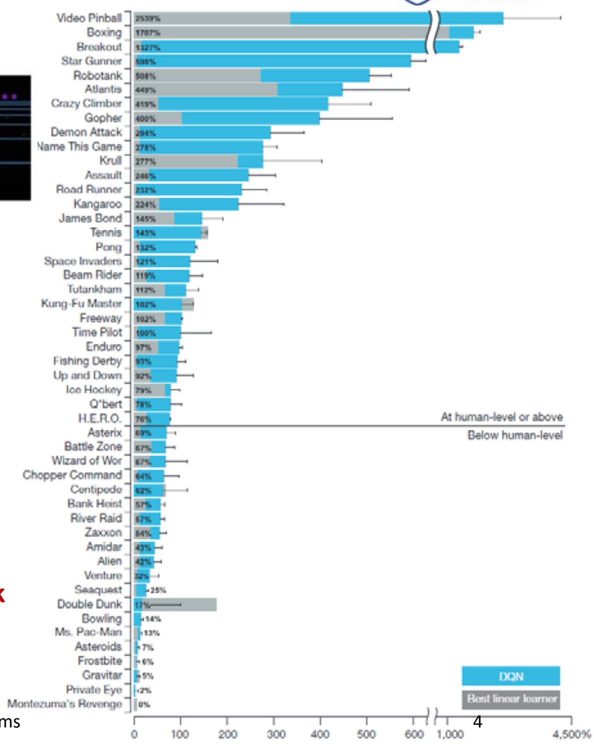
Human-level Control in Atari 2600 Games

Mnih et al, 2013/15

Pong Breakout Space Invaders Seaquest Beam Rider



Deep Q-Network (DQN)



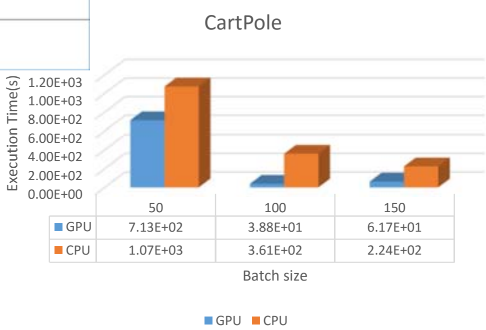
Engineering Systems

- Complex systems
  - behaviour described by differential equations
  - may have a number of operational modes
- Real-time control is required
- Need to operate reliably and safely
- Examples
  - vehicles, robots, manufacturing and communication and computing systems

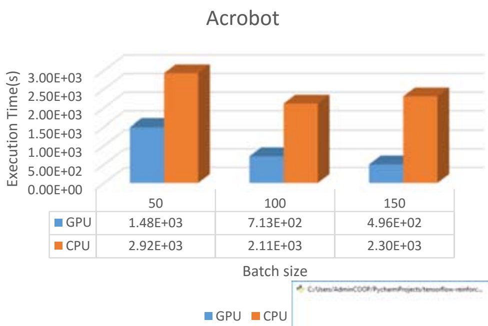


Execution Time: CPU vs GPU on OpenAI Gym Tasks

CPU: Intel Core i7-3770 CPU 3.40 GHz (4 cores, GB RAM) (power consumption: 47-100 W)
GPU: NVIDIA GeForce GTX 1060 (1,280 cores, 6 GB) (~S$350; power consumption: 120 W)



	50	100	150
GPU	7.13E+02	3.88E+01	6.17E+01
CPU	1.07E+03	3.61E+02	2.24E+02



	50	100	150
GPU	1.48E+03	7.13E+02	4.96E+02
CPU	2.92E+03	2.11E+03	2.30E+03

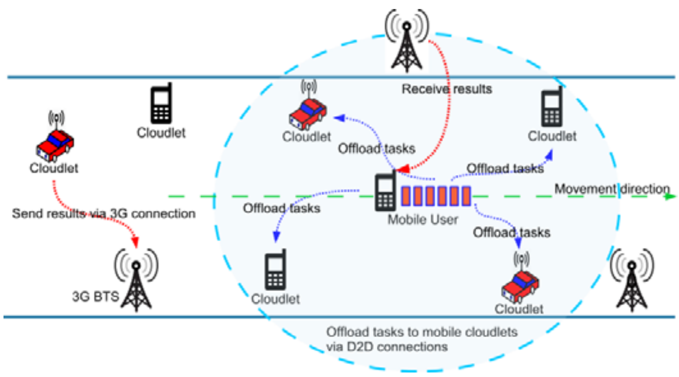
Training parameters:
Episodes: 1,000
Iterations in each Episode: 500

Engineering Case Studies

- Next Generation Distributed Mobile Computing:
  5G - IoT - Fog/Edge Computing

- Learning Skills on a Robot Manipulator

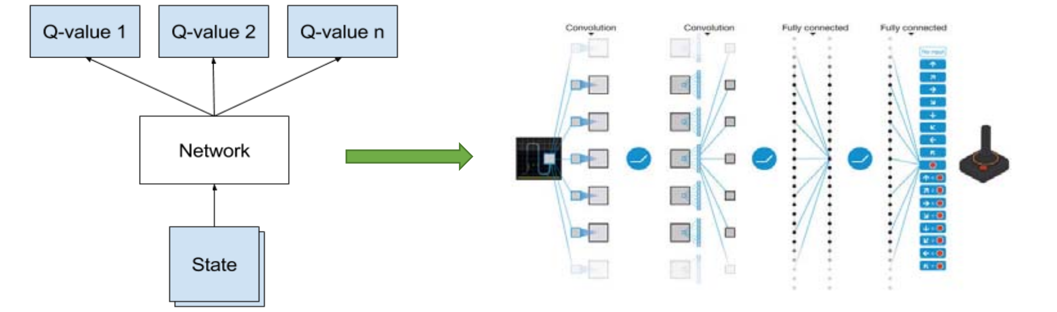
Computation Offloading in an Ad-hoc Mobile Cloud (5G - IoT - Fog Computing operation scenario)



- Consider an ad-hoc mobile cloud where a mobile user can offload its computation tasks to nearby mobile cloudlets (e.g., smart phones and vehicles) with available computation resources
- Computation tasks are offloaded to the cloudlets via a device-to-device (D2D) communication-enabled cellular network
- Deep reinforcement learning (DRL) is applied for the user to learn an optimal offloading policy with the objective of maximizing the user's utility, while minimizing the required payment, energy consumption, processing delay and task loss

D V Le, and C K Tham, A Deep Reinforcement Learning (DRL)-based Offloading Scheme in Ad-hoc Mobile Clouds, submitted to *IEEE PerCom 2018*

DQN-based Offloading Decision Learning



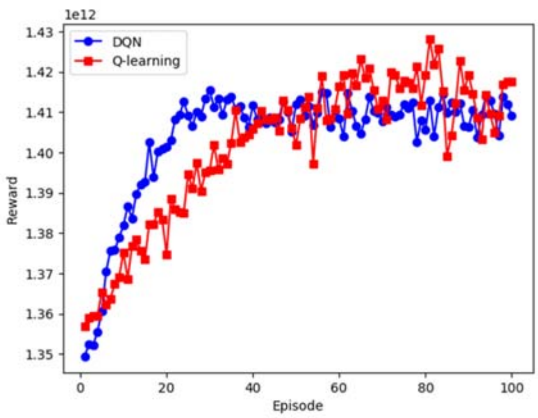
Deep Q-network

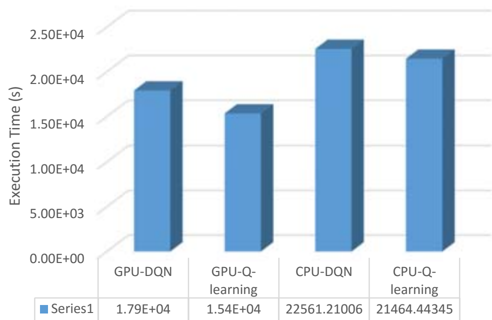
- The user maintains a neural network to estimate Q-values for all pairs of states and actions
- In the deep Q-network, the experience replay and target network techniques are used to train the neural network at every learning step
- The ϵ-greedy policy is adopted to select an action for each state based on the estimated Q-values

Offloading Results

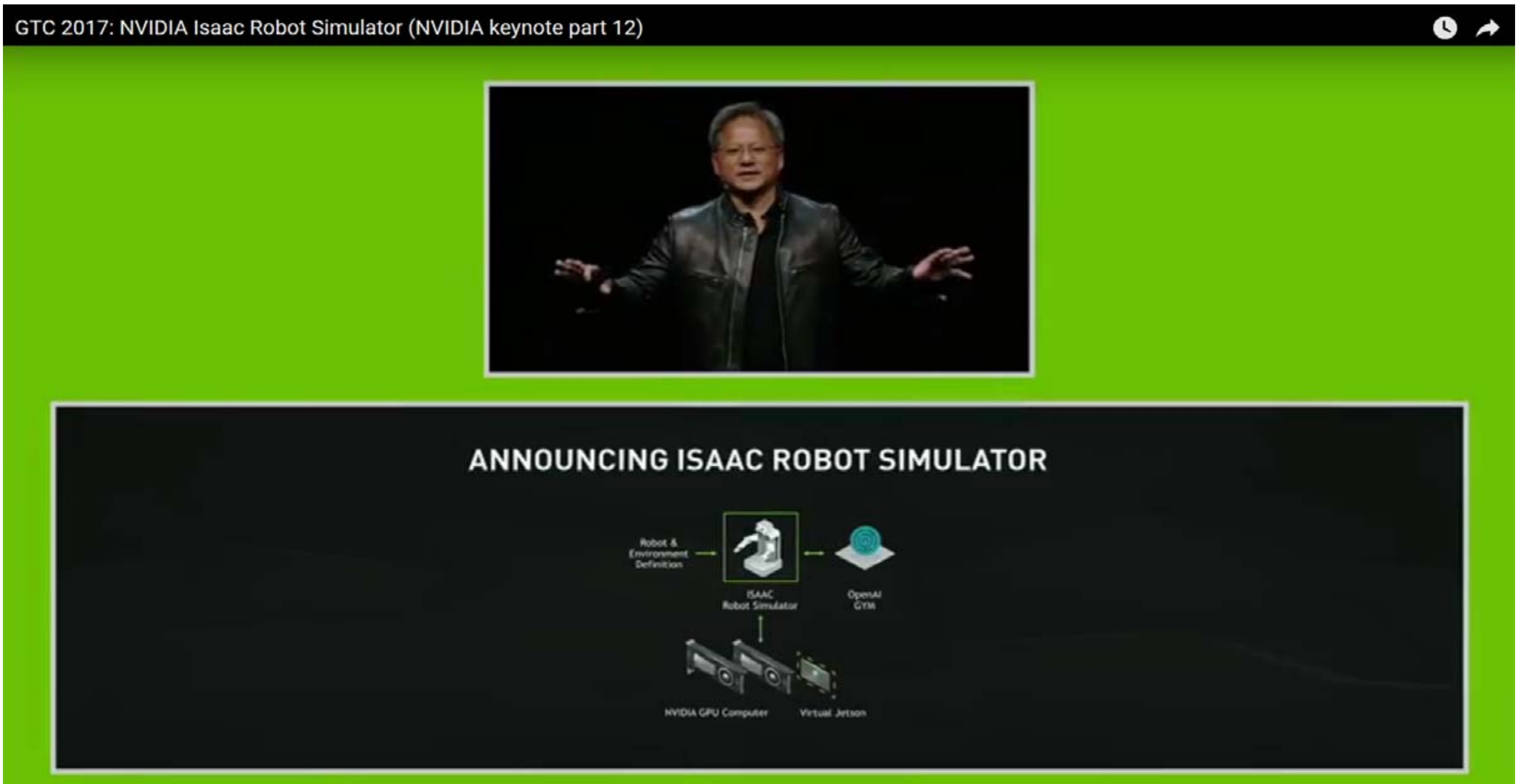
Achieves good payment, energy consumption, processing delay and task loss performance

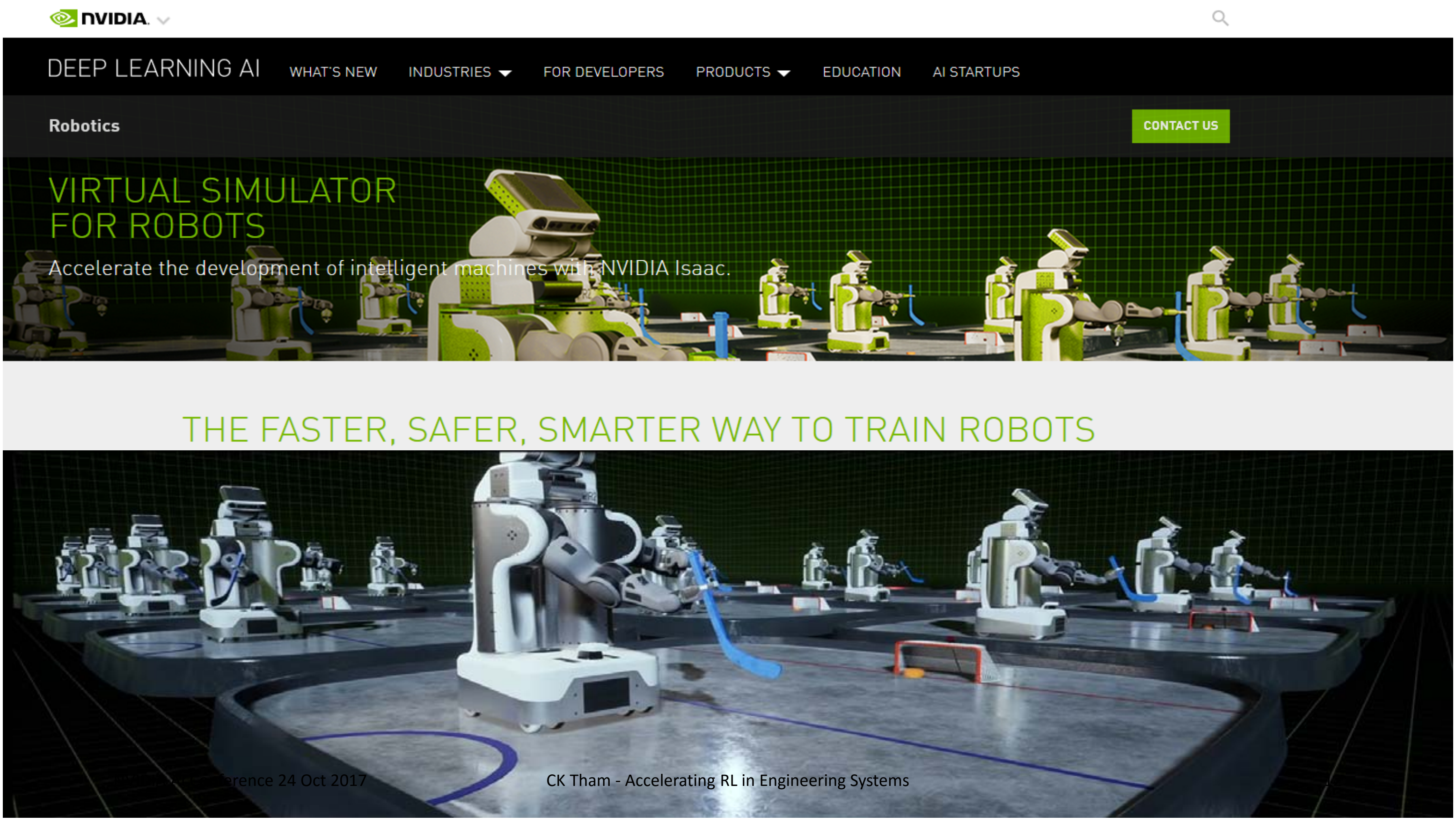
CPU: Intel Core i7-3770 CPU 3.40 GHz (4 cores, 16 GB RAM)
GPU: NVIDIA GeForce GTX 1060 (1,280 cores, 6 GB)





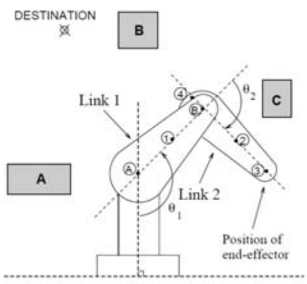
Training parameters:
Episodes: 100
Iterations in each Episode: 5,000
epsilon is decreased from 0.99 to 0.1 with decay rate 0.9999


GTC 2017: NVIDIA Isaac Robot Simulator (NVIDIA keynote part 12)
ANNOUNCING ISAAC ROBOT SIMULATOR
Robot & Environment Definition
ISAAC Robot Simulator
OpenAI GYM
NVIDIA GPU Computer
Virtual Jetson

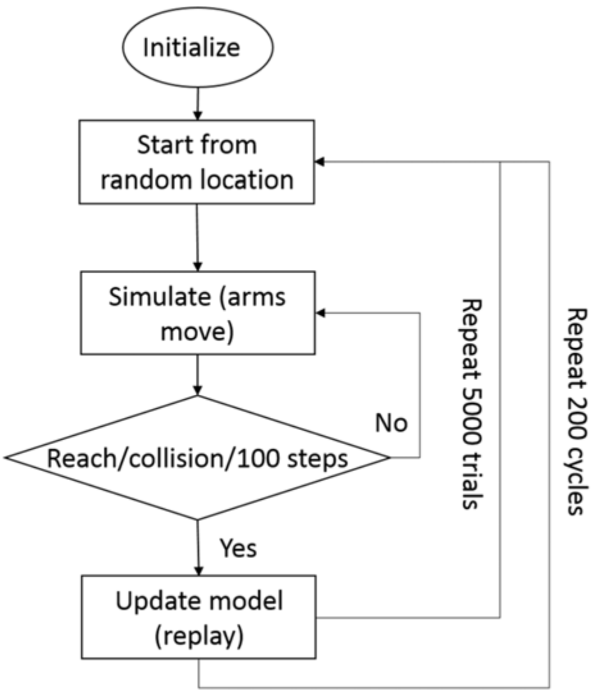

NVIDIA
DEEP LEARNING AI
WHAT'S NEW
INDUSTRIES
FOR DEVELOPERS
PRODUCTS
EDUCATION
AI STARTUPS
Robotics
CONTACT US
VIRTUAL SIMULATOR FOR ROBOTS
Accelerate the development of intelligent machines with NVIDIA Isaac.
THE FASTER, SAFER, SMARTER WAY TO TRAIN ROBOTS

Robot Manipulator Control

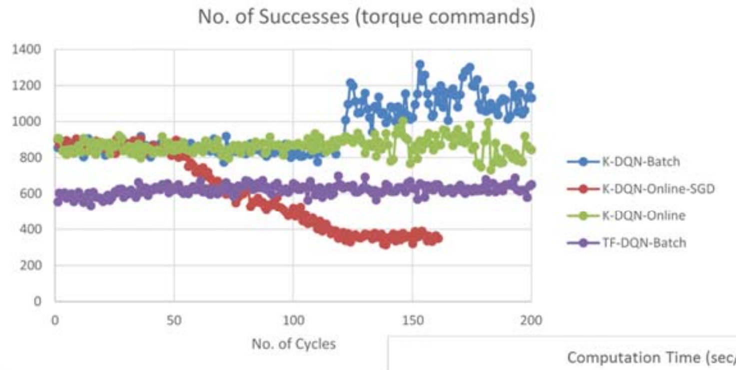
- Goal: move end-effector from any starting position to the destination position without colliding with any obstacle
- Generate torque commands to drive each link
- A dynamical system whose model is governed by differential equations

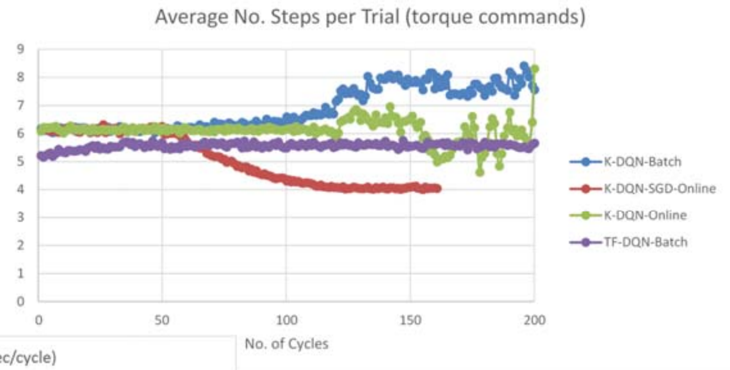


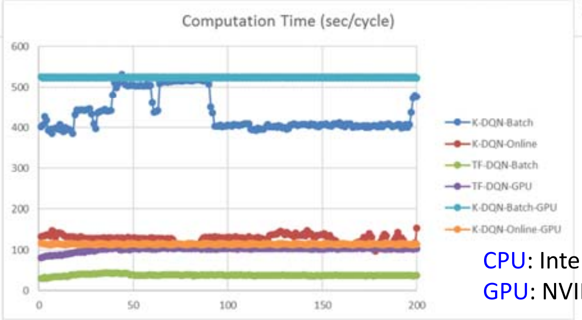
Train DQN



Robot Manipulator Results

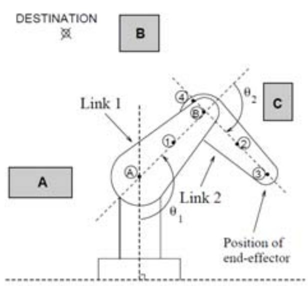




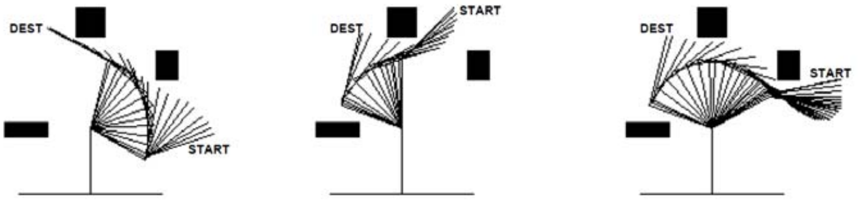


CPU: Intel Core i7-6700, 3.40 GHz (4 cores, 16 GB RAM)
GPU: NVIDIA GeForce GTX 1060 (1,280 cores, 6 GB)

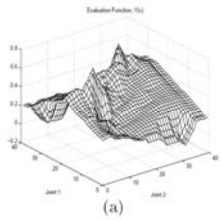
*Is this task **too difficult** for Reinforcement Learning?*

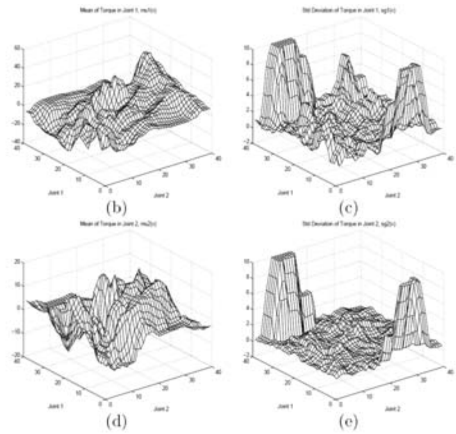


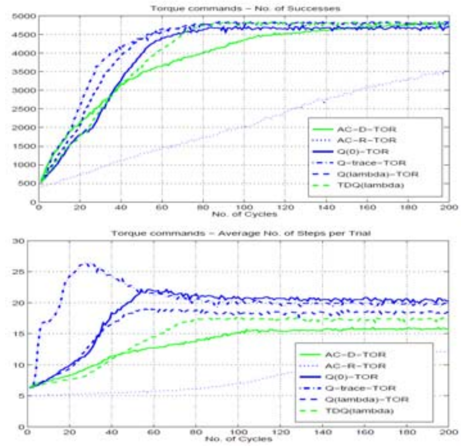
No. It can be solved … using a different function approximator



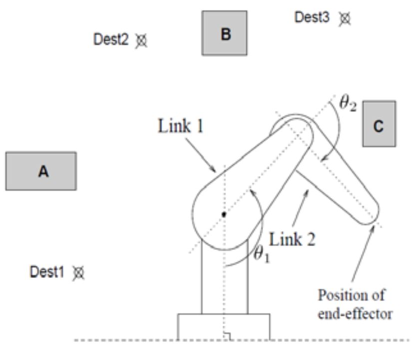
Faster
Higher resolution



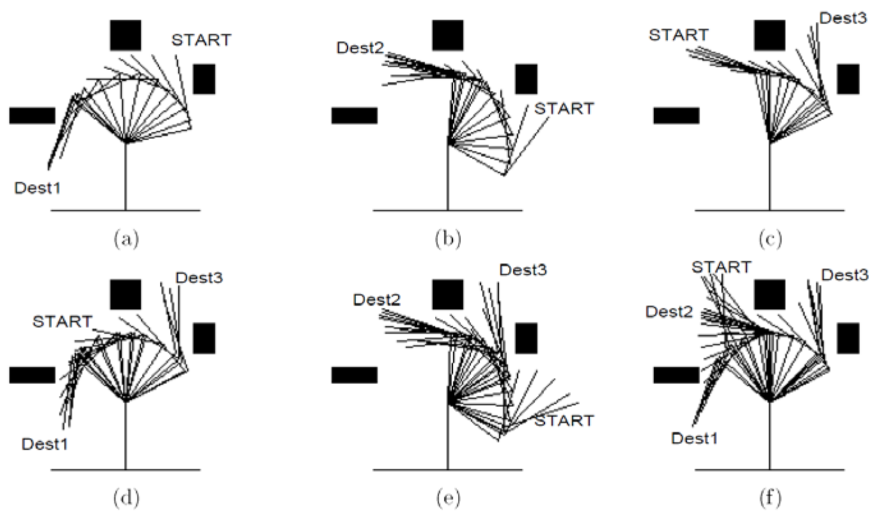




Composite Tasks



Task	Description	Decomposition
T_1	reach **1**	T_1
T_2	reach **2**	T_2
T_3	reach **3**	T_3
C_1	reach **1**, **3**	T_1T_3
C_2	reach **2**, **3**	T_2T_3
C_3	reach **1**, **2**, **3**	$T_1T_2T_3$



Conclusion

- Reinforcement Learning is capable of solving engineering tasks
- Hardware accelerators like GPUs* and algorithmic improvements have enabled RL to be accelerated in time and reduce the number of iterations or interactions with the environment that are required
- However, direct application of deep learning techniques may not yield the best results
- Further research into various aspects is needed to enhance RL's ability to better solve engineering problems

*

Thank You

Questions?

Contact: Tham Chen Khong

E-mail: eletck@nus.edu.sg