IEEE/RSJ International Conference on Intelligent Robot and System (IROS06)
Beijing, China, Oct. 9-15, 2006

Q-RAN: A Constructive Reinforcement Learning Approach for Robot Behavior Learning

Jun Li, **Achim Lilienthal**
Department of Technology
Örebro University
Sweden

Tomás Martínez-Marín
Department of Physics, System Engineering and Signal Theory
University of Alicante, Spain

Tom Duckett
Department of Computing and Informatics
University of Lincoln, UK

Outline

- Background – acquiring a robot behavior
  - by engineering design
  - by learning from robot's own experiences
- A layered learning system – QRAN
  - Main ideas of our learning system
  - Architecture of our learning system
  - Implementation of QRAN learning
  - Comments on QRAN learning
- Experimental results and analysis
  - Docking behavior
  - Learning by the QRAN system
- Conclusion and future work

Background – acquiring behaviors

- A reactive behavior
  - a sequence of sensory states and their corresponding motor actions for different tasks
  - some example behaviors

Robot docking

Moving-object following

Doorway crossing

Background– acquiring behaviors

- Engineering design
  - Linear control[1], Fuzzy control[2], and Symbolic-based planing[3].

[1] R. Siegwart and I. R. Nourbakhsh. Introduction to Autonomous Mobile Robots. The MIT Press, Cambridge, Massachusetts, 2004.

[2] A. Saffiotti. Autonomous Robot Navigation: a fuzzy logic approach PhD thesis, Université Libre de Bruxelles, 1998.

[3] N. J. Nilsson. *Artificial Intelligence: A New Synthesis*. Morgan Kaufmann, CA 94104, USA, 1998.

- Learning from experiences
  - Learning by demonstration[1], shaping[2], and development [3]

[1] A. Billard and R. Siegwart. Robot learning from demonstration. Robotics and Autonomous Systems, 47:65–67, 2004.

[2] M. Dorigo and M. Colombetti. Robot Shaping: An Experiment in Behavior Engineering. MIT Press/Bradford Books, 1998.

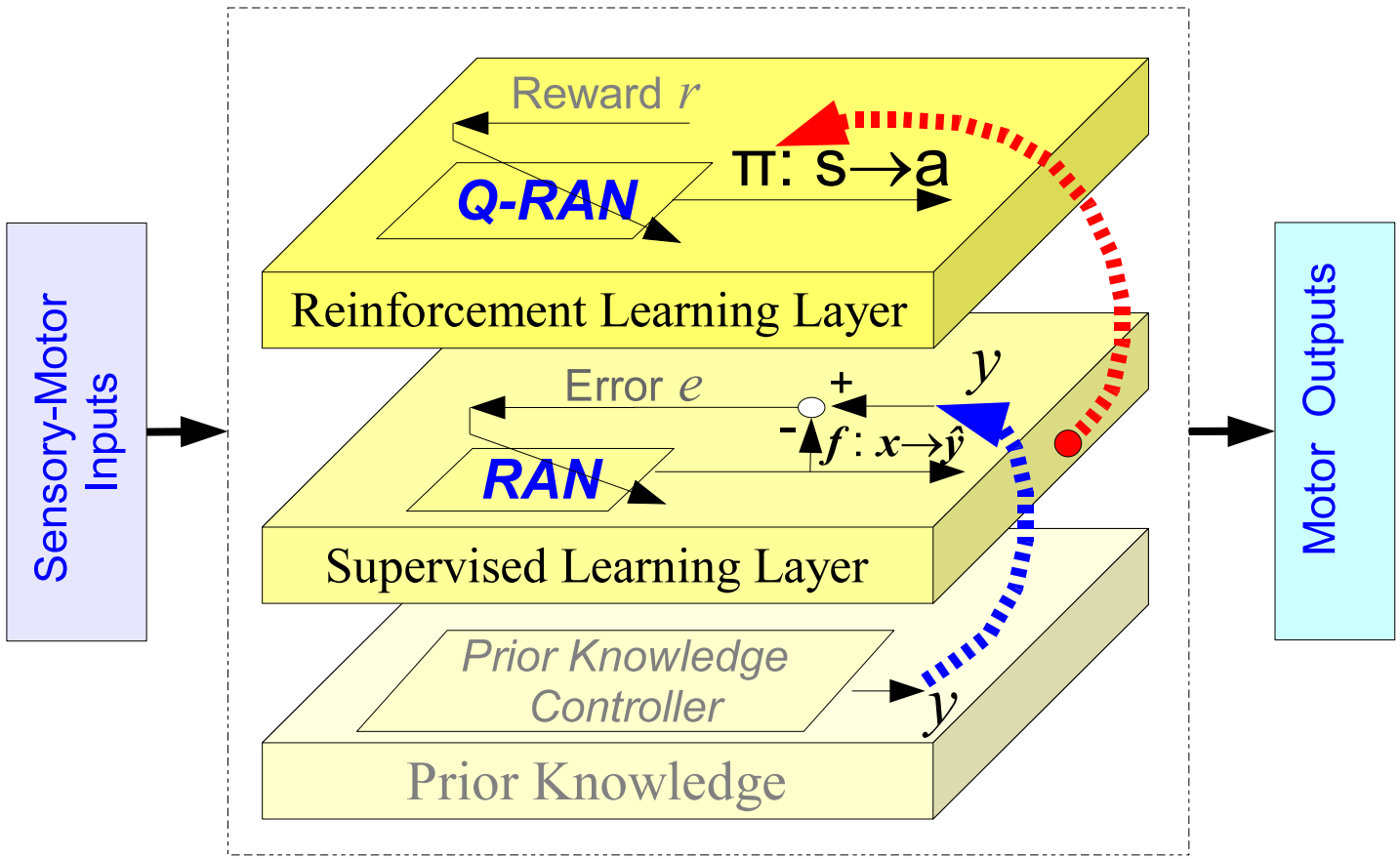
[3] J.Weng. Developmental robotics: Theory and experiments. International Journal of Humanoid Robotics, 1(2):199–236, 2004.

A layered learning system

- Main ideas of our learning system
  - A prior knowledge controller (rough controller)
    - derived from the engineering design, or
    - derived from the demonstration of a "teacher"
  - Lower layer with supervised learning
    - improving the prior knowledge controller
    - in the sense of smooth control
  - Upper layer with reinforcement learning
    - improving the lower layer's controller
    - in the sense of optimal control

A layered learning system

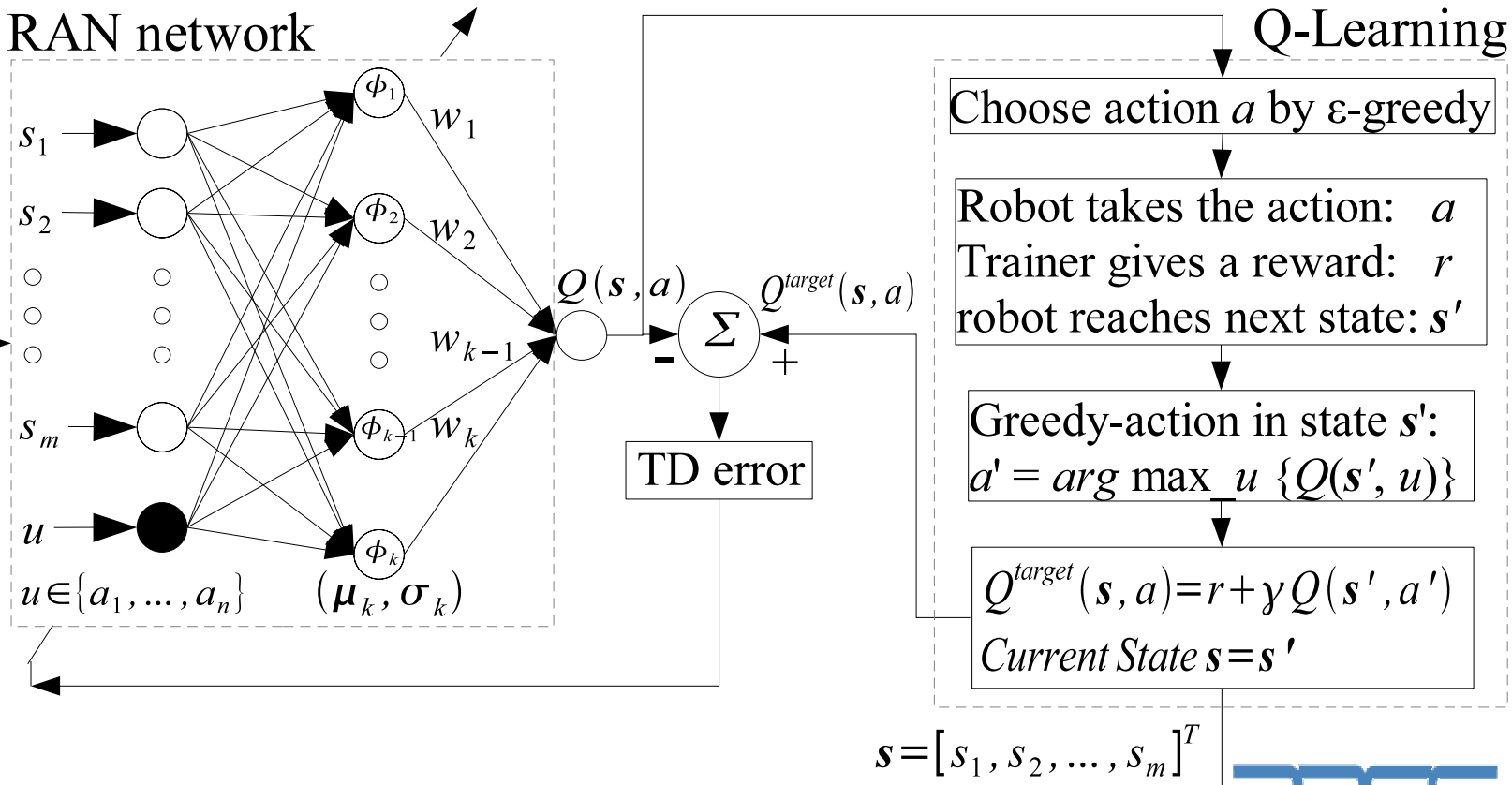
- Learning system's architecture



A layered learning system

- QRAN learning: $Q(s_t,a_t)=Q(s_t,a_t)+\alpha[r_{t+1}+\gamma\max_a Q(s_{t+1},a)-Q(s_t,a_t)]$

where $s_t \in S$, and $a_t \in A$



A layered learning system

- Comments on QRAN learning
  - Applicable constraints of Q-learning
    - Discrete state-action representation
    - Infinite visits of (s, a) guarantee the optimal mapping
  - Related work

    Rivest et. al, Combining TD-learning with cascade-correlation networks, ICML-2003
    Smart et. al, Effective reinforcement learning for mobile robot, ICRA-2002
    Santos et. al, Exploration tuned reinforcement learning for mobile robot, Neucomp. 1999
    Martínez et. al, Fast reinforcement learning for vision-guided mobile robots. ICRA-2005
    ……
  - What is new in QRAN-learning
    - RAN – A constructive ANN for continuous states representation
    - Off-policy of Q-learning speeds up the learning process on real robots
    - Easy to use and simple to implement due to the simple structure

Experimental results and analysis

- Docking behavior

A start position

Tracing a green can

Approaching table

Picking up the can

Experimental results and analysis

- LC controller solution

$$v_{trans} = k_P P$$

$$v_{rot} = k_\alpha \alpha + k_\beta \beta$$

Where

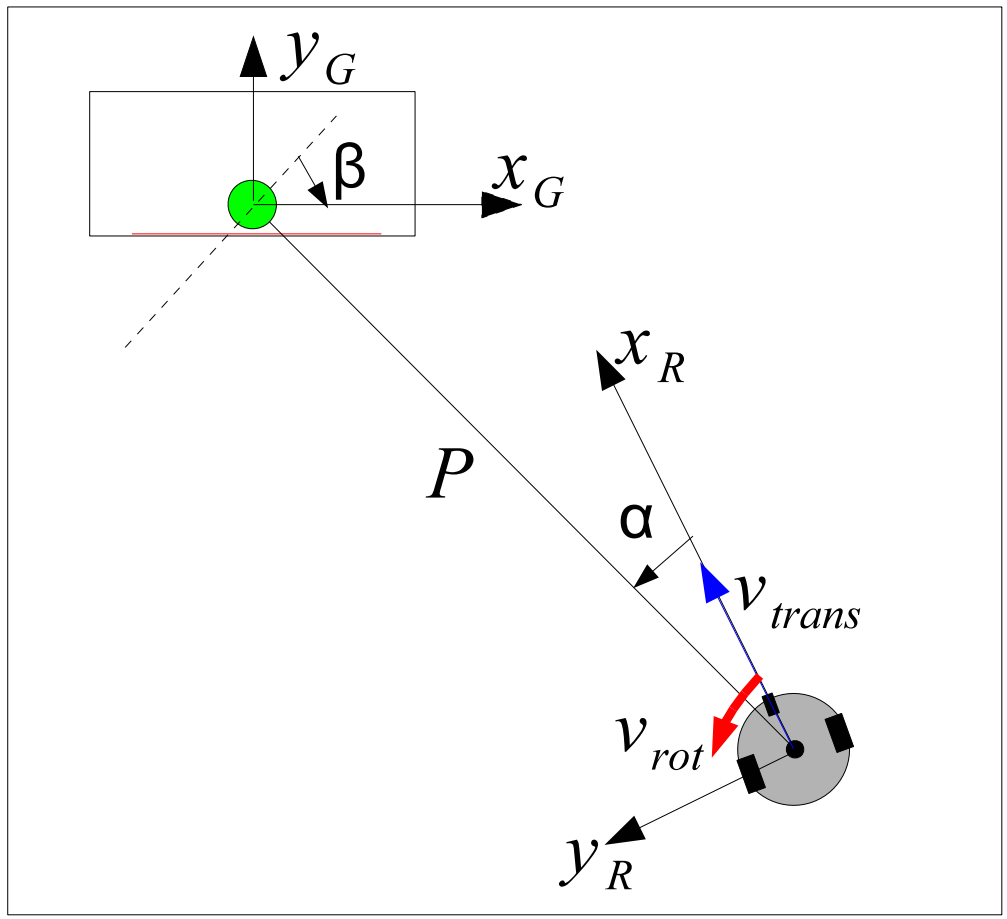
v_{trans} – translational velocity

v_{rot} – rotational velocity

α, β, and P – state variables

k_α, k_β, and k_P – gains

(x_G, y_G) – the global coordinates



Experimental results and analysis

- Docking becomes a complex behavior

1. $\{a_{tilt,}\ a_{pan}, a_{edge}\}$ is in local coordinate (clip: LC_chattering)
   a. LC controller is not applicable (overshooting and not robust)
   b. dependence time lag and momentum of robot and camera

2. fully reactive docking behavior
   a. the visual servoing – stabilizing and synchronizing
   b. the object tracking – robust

3. precise positioning at the goal pose

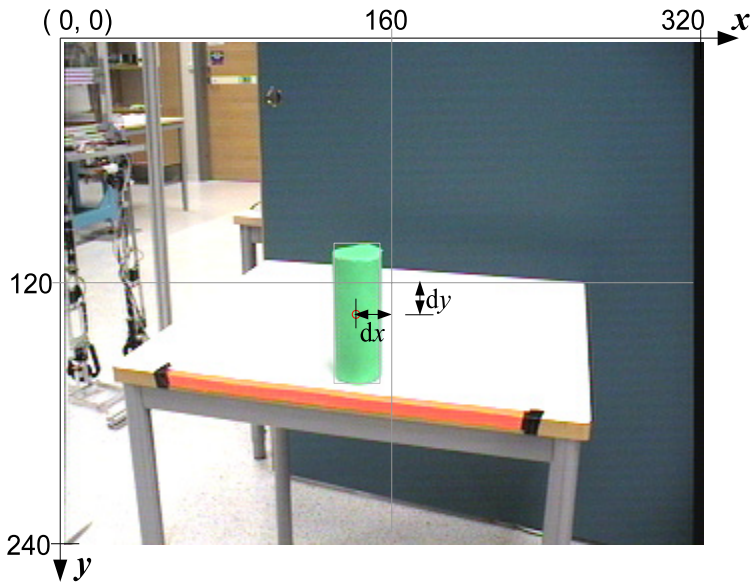
4. time-optimal trajectory

Experimental results and analysis

- Object tracking and visual servoing
  - Estimating the table edge's angle a_{edge}
    1. computing edge slope b_r by a LS model
    2. $a_{edge} = \arctan b_r$
  - Estimating a_{pan} and a_{tilt} by PD controllers

$$\Delta Pan = k_{pp}(x_o^{cur} - x_I) + k_{dp}dx$$

$$\Delta Tilt = k_{pt}(y_o^{cur} - y_I) + k_{dt}dy$$



  - State variables (α, β, P) is estimated by visual servoing variables

$$\alpha = a_{pan}, \beta = a_{edge}, P = 80 - a_{tilt}$$

Experimental results and analysis

- Learning with QRAN
  - State inputs: $x = [\alpha, \beta, u]^{\mathrm{T}}$
  - Action output:
    - Rotational velocity $v_{\mathbf{rot}}$ learned by QRAN
    - Translational velocity is determined by $v_{\mathrm{tran}} = k_P P$
  - Training QRAN network

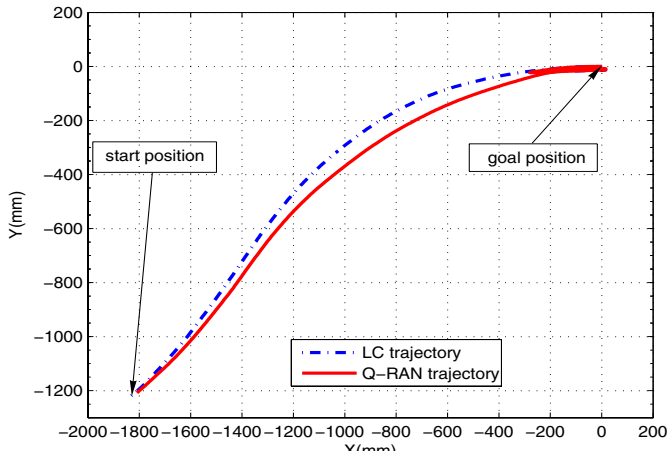
1. estimate the control variable $\{\alpha, \beta, P\}$ by visual servoing

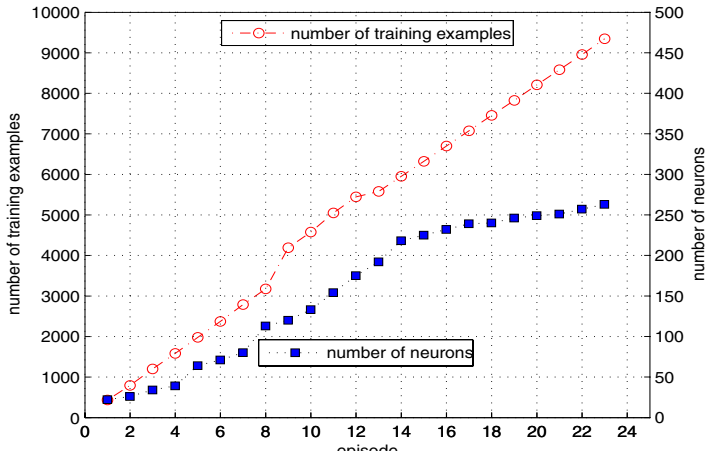
2. if goal or failure state, then end this episode, move the robot to a new starting position, goto 1 to start a new episode

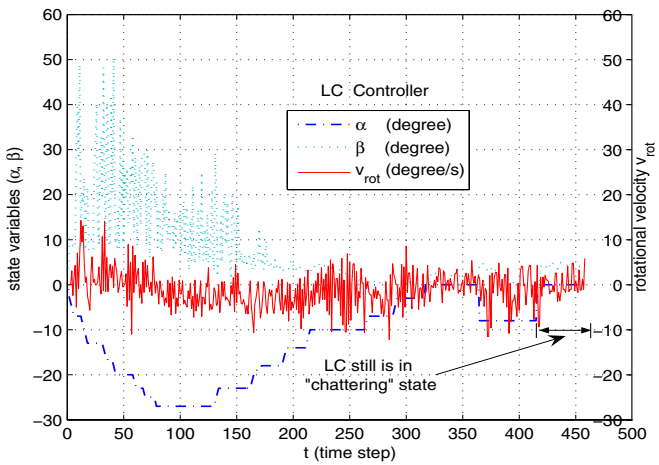
3. else train the QRAN network, and goto 1

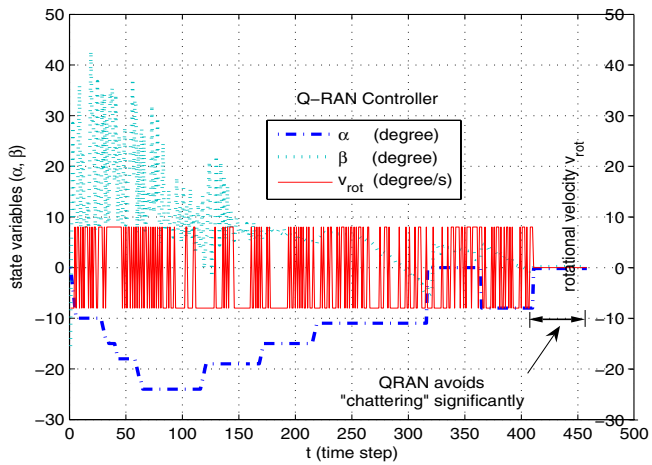
Experimental results and analysis

- Comparison: QRAN and LC controllers









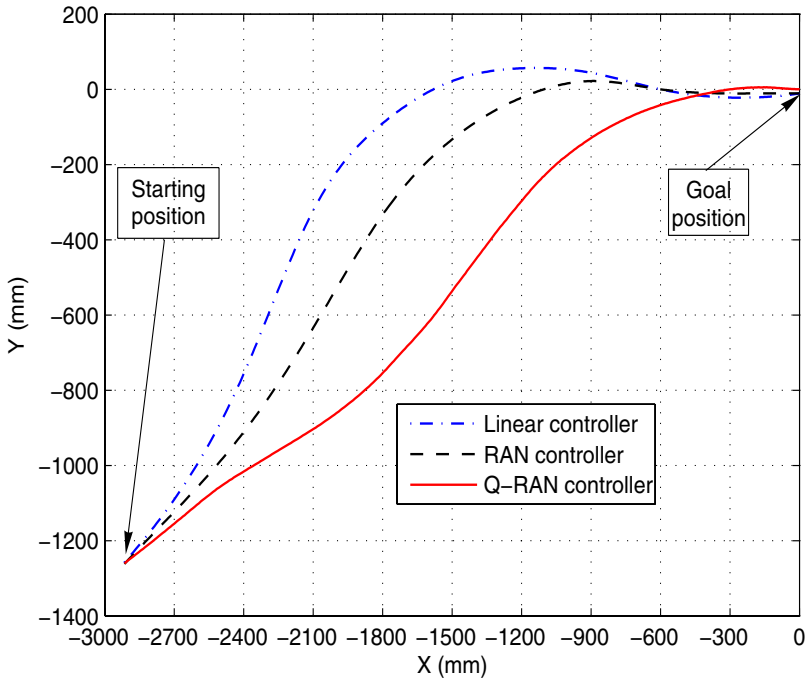
Experimental results and analysis

- Comparison: QRAN and LC controllers

Learning with only upper layer (Approx. 2m away from the goal)				
controller	Successful trials	Average steps	Number of neurons	Training episodes
QRAN	10 of 10	405 ± 12	263	23
LC	8 of 10	458 ± 14	*	*

Experimental results and analysis

- Learning with the layered learning architecture

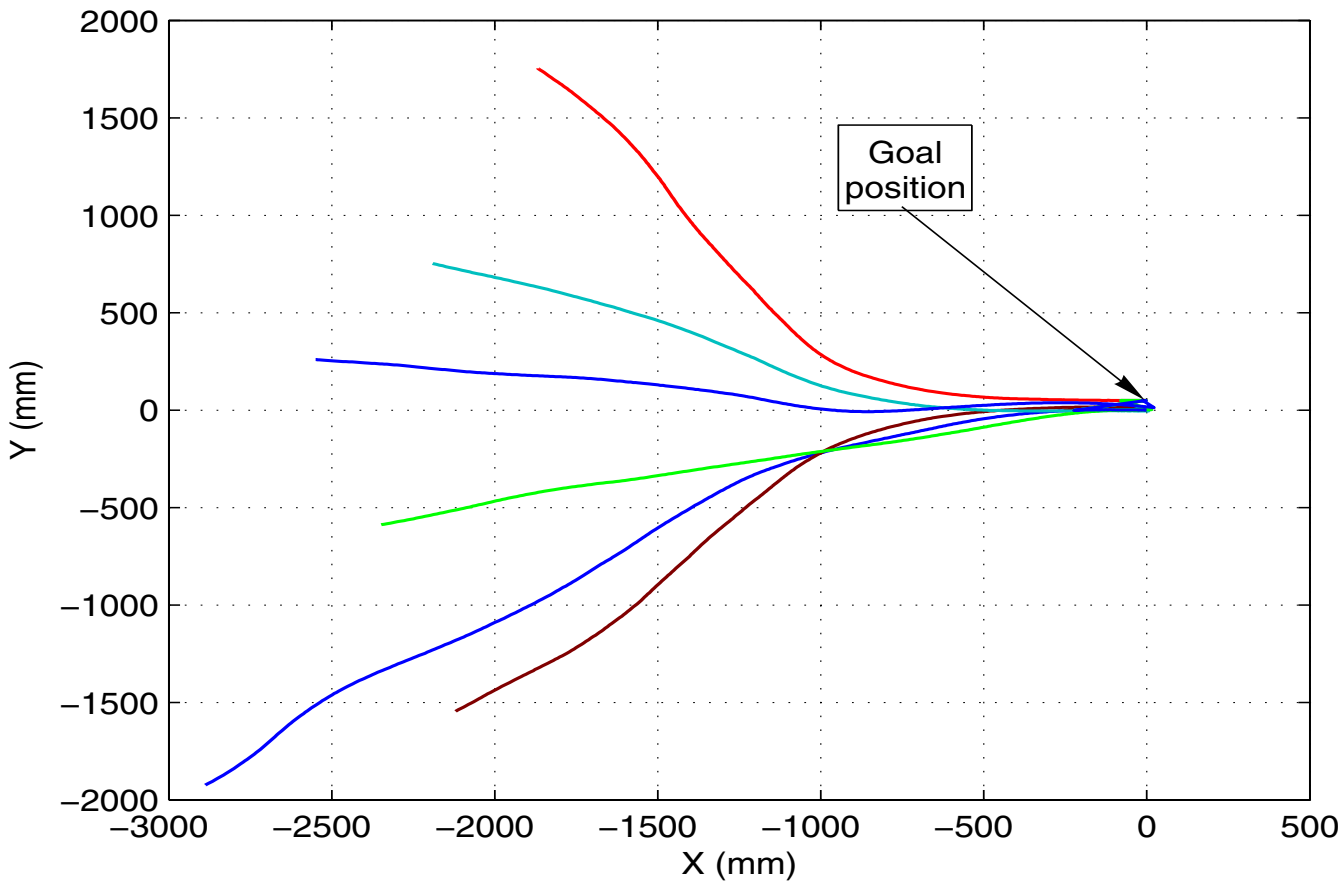


(clip: layered learning)

Learning with lower and upper layers (Approx. 4m away from the goal)				
controller	Successful trials	Average steps	Number of neurons	Training episodes
QRAN	10 of 10	518 ±19	181	18
RAN	9 of 10	618 ±28	126	100
LC	7 of 10	685 ±30	*	*

Experimental results and analysis

- Some example trajectories of layered learning



Conclusion and future work

- Conclusion
  - A layered learning architecture is proposed
    - LC is used as a prior knowledge
    - Lower layer with RAN network improves the LC controller in supervised learning fashion
    - Upper layer with QRAN improves the RAN controller
    - in reinforcement learning fashion
  - QRAN learning algorithm is proposed
    - Off-policy: incorporation of prior knowledge
    - Constructive ANN: dynamic representation of state space
- Future work
  - Automatic design of the reward function