International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 4, No.2 (Apr-2015)
http://dx.doi.org/10.12785/ijcds/040207

Hierarchical Reinforcement Learning: A Survey

Mostafa Al-Emran

Admission & Registration Department, Al-Buraimi, Oman

Received 29 Dec. 2014, Revised 7 Feb. 2015, Accepted 7 Mar. 2015, Published 1 Apr. 2015

Abstract: Reinforcement Learning (RL) has been an interesting research area in Machine Learning and AI. Hierarchical Reinforcement Learning (HRL) that decomposes the RL problem into sub-problems where solving each of which will be more powerful than solving the entire problem will be our concern in this paper. A review of the state-of-the-art of HRL has been investigated. Different HRL-based domains have been highlighted. Different problems in such different domains along with some proposed solutions have been addressed. It has been observed that HRL has not yet been surveyed in the current existing research; the reason that motivated us to work on this paper. Concluding remarks are presented. Some ideas have been emerged during the work on this research and have been proposed for pursuing a future research.

Keywords: Reinforcement Learning; Hierarchical Reinforcement Learning; Q-learning.

1. INTRODUCTION

Reinforcement Learning (RL) has been an interesting research field in the community of Machine Learning and AI that received many attentions from the fields of operations research due to its self-adaptation and self-learning [30]. RL algorithms work on maximizing the agent learning while interacting with its environment directly [17]. Hierarchical Reinforcement Learning (HRL) works on decomposing the RL problem into sub-problems where solving each of which will be more powerful than solving the entire problem [20]. In the recent years, [3], [22] and [23] stated that the problem of "*Curse of Dimensionality*" (which is the exponential growth of memory requirements with the number of state variables) has been solved via HRL. The RL then works on reducing dimensionality through decomposing it into several levels. HRL helps to overcome the agent-learning complexities that are considered as one of the typical issues in the learning environments [18]. Different HRL-based domains have been investigated within this work. Different problems in such different domains along with some proposed solutions have been addressed.

The paper is organized as follows: section 2 demonstrates a background on RL, HRL and Q-learning. Section 3 addresses the main contribution in the area of HRL. Section 4 presents the conclusion and the ideas that have been emerged while conducting this research.

2. BACKGROUND

A. Reinforcement Learning

Reinforcement learning (RL) is one of the machine learning areas in which an agent has to interact with its environment in order to achieve a goal (as in Figure 1). RL based on the structure of Markov Decision Processes (MDPs); a reliable structure for the agent learning while interacting with its environment in order to receive rewards and drawbacks [28], [29], [35]. The essential elements of RL are the states, actions and reinforcements [34]. Via the agent's sensors, the agent recognizes the environment and implements actions (according to a policy) in which leads to changes in the environment. According to these changes, the agent obtains rewards based on the taken actions [1], [2]. RL improves strategy through the learning via trial and error by interacting with the environment and recognize the best actions at each state in order to reach the goal and gain the best rewards [3], [4]. RL tries to find the best policy that increases the total reward. [17] indicated that RL algorithms work on how the agent can learn to estimate an optimal strategy while to interact with its environment.

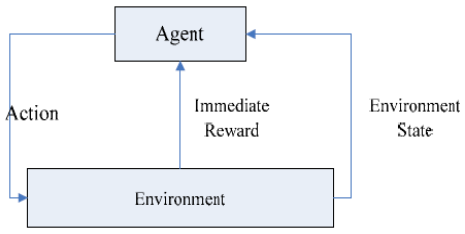

Figure 1. Reinforcement Learning Basic Model. [3]

B. Hierarchical Reinforcement Learning

Hierarchical Reinforcement Learning (HRL) refers to the notion in which RL problem is decomposed into sub-problems (sub-tasks) where solving each of which will be more powerful than solving the entire problem [4], [5], [6] and [27], [36]. [25] and [31] has defined HRL as the set of

E-mail: malemran@buc.edu.om

computational techniques that enlarge RL procedures in order to involve temporarily abstract actions. The hierarchical decomposition has some advantages such as: reducing the sub-problem's computational complexity, managing the sub-problems individually will maximize its reusability which in turn will speed up the learning process [4]. HRL techniques use several forms of abstractions that have the ability to handle the exponentially increasing number of parameters which are required to be learned specifically in big problems for effectively reducing the search space which in turn allows the agent to determine the optimal solution [7]. HRL is one of the well-known methods to resolve the problem of "*Curse of Dimensionality*" [32]. [22] stated that a well-designed reward function along with HRL can decrease the number of impractical acts of exploration which in turn allows the agent to interact easily and quickly with the environment. [23] presented that utterance planning and content selection in Natural Language Generation (NLG) can be optimized via HRL along with Bayesian Networks. Various HRL models are available like MAXQ, Hierarchical abstract machines (HAMs), ALisp and options [26], [33]; the models that scale RL to large state spaces problems by decomposing them into sub-problems.

C. Q-Learning

Q-learning is one of the RL algorithms that has been successfully used in many domains such as: face recognition, simple toys, web-based education and many others [8]. *Q-learning* tries to find an optimal action policy by estimating the optimal state-action function *Q(s, a)* where *s* → state from the set of the possible states *S*, *a* → action from the set of the possible actions *A*. The *Q* function described the maximum reward achieved when an action *a* is executed over the state *s* [1]. The *Q-learning* equation is described as follows:

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha(r+\gamma \max_{a'} Q(s',a'))$$

Where α refers to the learning rate, γ refers to the discount factor and *r* refers to the reward of executing the action *a* over the state *s*.

3. TECHNICAL PART

Different domains with different problems based on HRL will be discussed and explored within this section:

A. HRL Based Control Architecture for Semi-Autonomous Rescue Robots in Cluttered Environments

Urban search and rescue (USAR) scenes are cluttered enough and the information about such environment is already unknown due to their desolation. Therefore searching for victims in such environments using human teleoperation of rescue robots is a difficult task. Different solutions have been proposed to resolve the USAR problem as in the following table:

TABLE I. TECHNIQUES USED FOR RESOLVING USAR PROBLEM.

Technique	Description
Wirelessly teleoperated control	Where the communication between the human and robot will be lost due to the environment nature [9]. Hence this technique makes the search task very difficult for the robot.
Fully autonomous controllers	Since this technique is fully robot-based; humans couldn't trust the robot in such critical tasks. Moreover, using this technique is challenging due to the fact that dust, debris in such environments will affect the sensors, hence this technique remains ineffective and needs more improvements. Another alternative technique by [9].
HRL-based semi-autonomous control architecture	HRL algorithm enables the robot to learn and make his own decisions based on the rescue tasks, victim identification and exploration by performing these tasks very quickly and efficiently. The experiments revealed the effectiveness of the proposed technique by observing the ability of the robot while exploring the whole USAR environment [7].

B. HRL in Computer Games

Computer games are one of the hot topics for research in AI and machine learning. One of the issues in computer games that attract researchers to work on is the Non-Player Characters (NPCs) behaviors due to their complexity and difficulty in to be represented by typical finite state machines. The control details of NPCs at all stages are commonly hand-coded; the reason that makes the development task consumes more time and exposed for errors. To overcome these limitations, HRL has been used based on the Hierarchies of Abstract Machines (HAMs). Through applying the proposed solution, system designers can determine points within the program itself where they don't care about how the code will be written while it's determined through the robot's learning process. Experiments have been conducted to test the efficiency of the proposed solution under the *Quake2UR* (which performs as 3D Game Server) and *ALisp system* (which performs as a client). Results revealed that the proposed solution was very flexible and satisfies the need for controlling NPCs easily [6].

Similarly, [21] has proposed the MaxQ-Q HRL algorithm in the NPCs in order to enhance the user's experience and to improve the natural humanness while interacting with computer games. Experiments were performed using "Capture the Flag" strategic game. A comparison was made between the NPCs based on Finite State Machines (FSM) and the NPCs based on MaxQ-Q through the game. Results indicated that NPCs based on MaxQ-Q HRL are 52% much better than NPCs based on FSM.

Moreover, the *Infinite Mario* game is one of the interesting action games that got popularity in the area of AI and machine learning. The domain of this game is very complex and contains huge state-action spaces. [19] has integrated the HRL along with the object-oriented representation in order to reduce the state-action spaces in the game domain. Accordingly, involving HRL has increased the agent-learning performance.

C. Course-Scheduling Algorithm of Option-based HRL

Traditional timetable scheduling system implements RL algorithm. Whereas this algorithm suffers from the oscillation period due to the reason that the reward of RL algorithm is not immediately obtained. This will affect the RL algorithm to indicate that the timetable state dimension is extremely large while scheduling the course. [3] proposed applying an option-based HRL algorithm to the timetable scheduling strategy in order to enhance the performance of traditional RL. The *Q*-value update of option-based HRL algorithm is as follows:

$$Q_{k+1}(s,o)=(1-a_k)Q_k(s,o)+a_k[r+\gamma^t \max_{o' \in O_{s'}} Q_k(s',o')]$$

Where *r* denotes the reward, γ denotes the discount rate factor, *a* denotes the learning rate and *t* is the time that the option takes. The environment parameters are the instructor, course, college, major, semester, grade and classroom indicating that the agent has no prior knowledge about the environment before learning. Experiments results revealed that the proposed algorithm has the ability to reduce the oscillation period. Moreover, while HRL is involved in this algorithm, the course-scheduling actions are divided into sub-tasks; this will allow the agent to learn quickly and select the optimal strategy. Furthermore, the results demonstrated that the *Q*-value update equation is much smoother than the regular *Q-learning* algorithm.

D. HRL Approach for Motion Planning in Mobile Robotics

Motion planning task is one of the interesting tasks in mobile robotics that looks for generating a free-collision path from the initial point to the goal point for the robot. [10] applied RL in order to avoid all the obstacles in mobile robotics through the use of Neural Networks, However this became an old technique where [1] proposed an option-based HRL in which basic behaviors are used. Each behavior is individually learned in the learning process; this allows the robot to organize all the basic behaviors to solve the problem of the motion planning. Semi-Markov *Q-learning* has been used to estimate the state-option function values *Q(s, o)* via choosing an option *o* in the state *s* based on the policy μ. After implementing *o*, the final state *s'* along with the *Q-value* will be updated based on the equation:

$$Q(s,o) \leftarrow Q(s,o)+\alpha\left\{r+\gamma^k \max_{o' \in O(s')} Q(s',o')-Q(s,o)\right\}$$

Where γ denotes the discount rate, *α* denotes the learning rate and *k* refers to number of steps between *s* and *s'*. Experimental tests have been implemented in simulation. Results revealed that the proposed algorithm has the ability to work effectively in unknown environments as well as to avoid all the encountered obstacles by the robot in the motion planning task without the use of Neural Networks.

E. HRL using Path Clustering

[4] intended to resolve small and medium scales RL problems through the use of path clustering in order to enable its hierarchical decomposition. Moreover, how to enhance the *Q-learning* algorithm performance via automatically finding sub-goals and making better usage of the knowledge acquired. HRL path clustering method has been proposed which allows the robot to acquire the knowledge about the states' sequences which lead to the goal and introduce those states at the end of the sequences as sub-goals. Taxi-problem (as one of the standards in RL and is being used for testing the HRL solutions) has been used. In this problem, sub-goals enhance the learning speed by achieving good results faster than the traditional *Q-learning* due to the fact that the problem scale is very small. It has been proposed to insert the sub-goals into the learning process. Results revealed that the early involvement of sub-goals will achieve a sub-optimal learning.

F. Web Service Composition method using HRL

Web services composition facilitates the combination of single web-services into featured/valued services that could satisfy the users' needs while the individual single web-services couldn't so. The dynamic web service composition model is shown in Figure 2. When the "task acceptor" of the "service agency i" receives the data, the corresponding flow chart will be generated through the "composed service engine". Accordingly, each simple service will be executed by the "business execution engine". "Service agency i" will receive the results and the system continues to invoke "service agency j" that delivers other services [13].

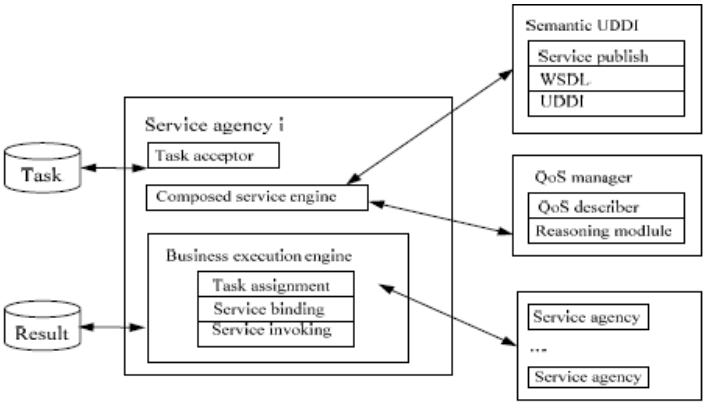

Figure 2. Dynamic Web Service Composition Model. [13]

One of the main problems of the dynamic web-service composition is the optimization problem (i.e. how to find an optimal policy). Different solutions have been proposed to accomplish an optimal policy for dynamic web service composition. [11] proposed an algorithm based on RL; however this algorithm suffers from the "*Curse of dimensionality*" specifically in large-scale problems of web-service composition. On the other hand, [12] introduces the HRL, a continuous time integrated MAXQ algorithm to handle the large-scale problems in the context of Semi-Markov decision process (SMDP). This algorithm has been compared to the *Q-learning* algorithm. Simulation results revealed that the performance of MAXQ algorithm is much better than the *Q-learning* algorithm by comparing the both with a discount factor $a = 0.01$ due to the reason that the MAXQ algorithm has the ability to accelerate the learning speed. Moreover, by comparing both algorithms with different number of tasks, it has been observed that as much as the number of tasks is increasing, the success rate of the *Q-learning* decreases much faster than MAXQ. [12] proved that the proposed algorithm is much better than the *Q-learning* algorithm to handle the problem of *Curse of dimensionality* in large-scale problems of web-service composition.

Another problem in dynamic web-service composition is that how to combine a collection of simple web-services based on the users' functional needs and how to choose such services based on users' QoS needs among all the available services. [13] proposed an algorithm based on HRL and Logic of Preference; the algorithm that efficiently deals with both users' functional and QoS needs and has the ability to work in large-scale problems. The algorithm is decomposed into two parts: MAXQ (for service composition) and Logic of Preference (for choosing the service).

An experiment has been conducted using 500 web services and 180 states. Experiments results revealed that the computation cost is significantly decreasing as long as the number of execution times is increasing. Moreover, results showed that utilizing HRL can effectively speed up the composition task.

G. A Combined HRL Based Approach for Multi-robot Cooperative Target Searching in Complex Unknown Environments

Collaboration of multi-robots in unknown environments is one of the fundamentals in various applications like target searching and exploring the environments. One of the main weaknesses is that the learning ability in many RL approaches is temporary due to the reason that it is environment-based; the ability to deal with new environments and specifically dynamic environments. [14] suggested a combination of both Option and MAXQ algorithms in which the knowledge and the hierarchical structure are introduced and constructed respectively by both algorithms. However, this solution still lacks the exploration of the unnecessary parts of the environments. [15] proposed an effective HRL algorithm that combines both the MAXQ and Option algorithms (as in Figure 3) where all the required parameters will be automatically obtained through the learning, unlike other algorithms that selects parameters via trial and error. The proposed solution has the ability to evaluate the feedback and tries to get featured parameters for future processes; the reason that makes this solution is unique for such environments as comparing with the others. The simulation results revealed that the proposed solution has the ability to allow a team of robots to collaboratively achieve target searching in unknown environments.

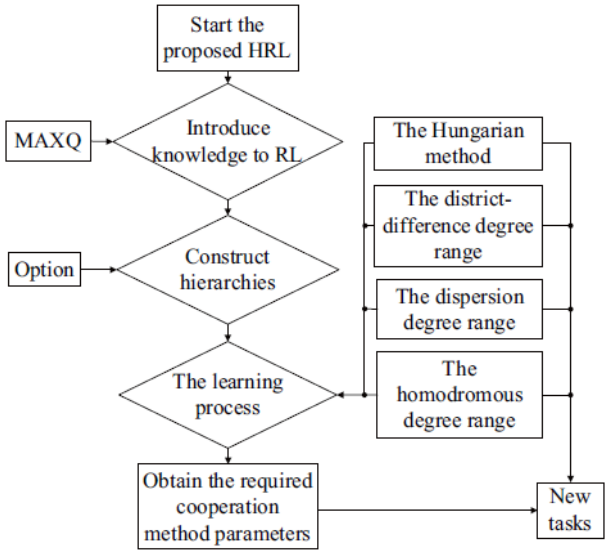

Figure 3. MAXQ and Option algorithms combination. [15]

H. Deep Belief Network for Modeling HRL policies

Intelligent robots faced multiple tasks during their lifetime that requires concurrent modeling and involves controlling the complexity in unknown environments. Policy learning is one of the major issues that suffers from the "*Curse of Dimensionality"* which leads to scaling problems for regular RL. To handle this issue, the robot should efficiently acquire and reuse potential knowledge. [16] proposed a novel learning technique for HRL based on Conditional Restricted Boltzmann Machines (CRBMs) to tackle the growing learning and scaling problems for regular RL. A simple Taxi domain was designed to investigate the learning capabilities and represent the HRL policies. The designed taxi domain represents a car in one dimensional space that picks a packet from a state and drops it at a destination as demonstrated in Figure 4. HRL based-CRBMs have potential to offer a uniform means to concurrently learn policies and associate abstract state features within a reliable network structure.

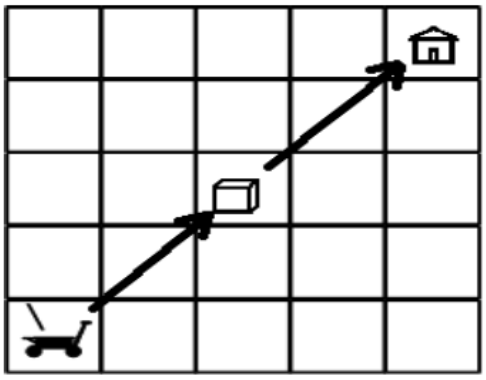

Figure 4. Simplified Taxi Domain. [16]

4. CONCLUSION & FUTURE WORK

Reinforcement Learning (RL) plays a significant role in the area of Machine Learning and AI. HRL focuses on decomposing the RL problems into sub-problems where solving each sub-problem independently is much easier and powerful than solving the entire problem. A review of the state-of-the-art of Hierarchical Reinforcement Learning has been investigated. Different research areas with different problems based on HRL have been surveyed within this paper such as: rescue robots in cluttered environments, computer games, course scheduling, motion planning in mobile robotics, web service composition, path clustering, multi-robot cooperation and intelligent robots.

While working on this survey, some ideas have been emerged and can be pursued as a future work; the ideas that require more attention from researchers who are interested in the HRL field. Such ideas could be summarized as per the following:

- Will multi-robot cooperation be an effective way to search for victims in cluttered USAR environments? As to compare with the proposed solutions by [9], [7]. Moreover, these solutions have been implemented in small-scale environments so further research needs to focus on large-scale environments to test their efficiency.
- How multi-robot cooperation will support the problems of web-service compositions? As to compare with [12].
- [4] focused on smaller discrete RL problems through the use of path clustering. Further research may focus on larger continues RL problems.
- [12] compares both MAXQ and Q-learning algorithms in the area of web-service compositions. Further work could compare the two algorithms in robots race and see which one is going to learn and reach the goal state faster?

REFERENCES

[1] Buitrago-Martinez, A., Rosa, R., & Lozano-Martinez, F. (2013, October). Hierarchical Reinforcement Learning Approach for Motion Planning in Mobile Robotics. In *Robotics Symposium and Competition (LARS/LARC), 2013 Latin American*, pp. 83-88. IEEE.

[2] Dayan, P., & Niv, Y. (2008). Reinforcement learning: the good, the bad and the ugly. *Current opinion in neurobiology*, *18*(2), pp. 185-196.

[3] Ming, G. F., & Hua, S. (2010). Course-scheduling algorithm of option-based hierarchical reinforcement learning. In *2010 Second International Workshop on Education Technology and Computer Science*, Vol. 1, pp. 288-291.

[4] Gil, P., & Nunes, L. (2013, June). Hierarchical reinforcement learning using path clustering. In *Information Systems and Technologies (CISTI), 2013 8th Iberian Conference on* (pp. 1-6). IEEE.

[5] Stulp, F., & Schaal, S. (2011, October). Hierarchical reinforcement learning with movement primitives. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on* (pp. 231-238). IEEE.

[6] Xiaoqin, D., Qinghua, L., & Jianjun, H. (2009, August). Applying hierarchical reinforcement learning to computer games. In *Automation and Logistics, 2009. ICAL'09. IEEE International Conference on* (pp. 929-932). IEEE.

[7] Doroodgar, B., & Nejat, G. (2010, August). A hierarchical reinforcement learning based control architecture for semi-autonomous rescue robots in cluttered environments. In *Automation Science and Engineering (CASE), 2010 IEEE Conference on* (pp. 948-953). IEEE.

[8] Rodrigues Gomes, E., & Kowalczyk, R. (2009, June). Dynamic analysis of multiagent Q-learning with ε-greedy exploration. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 369-376. ACM.

[9] Murphy, R. (2004). Activities of the Rescue Robots at the World Trade Center from 11–21 September 2001, *IEEE Robotics & Automation Magazine*, pp. 50-61, 2004.

[10] Macek, K., PetroviC, I., & Peric, N. (2002). A reinforcement learning approach to obstacle avoidance of mobile robots. In *Advanced Motion Control, 2002. 7th International Workshop on* (pp. 462-466). IEEE.

[11] Wang, H., Tang, P., & Hung, P. (2008, September). RLPLA: A reinforcement learning Algorithm of Web service Composition with Preference Consideration. In *Congress on Services Part II, 2008. SERVICES-2. IEEE*, pp. 163-170. IEEE.

[12] Tang, H., Liu, W., & Zhou, L. (2012). Web Service Composition Method Using Hierarchical Reinforcement Learning. In *Green Communications and Networks*, pp. 1429-1438. Springer Netherlands.

[13] Wang, H., & Guo, X. (2009, September). Preference-aware web service composition using hierarchical reinforcement learning. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology-Volume 03*, pp. 315-318. IEEE Computer Society.

[14] Cheng, X., Shen, J., Liu, H., & Gu, G. (2007). Multi-robot cooperation based on hierarchical reinforcement learning. In *Computational Science–ICCS 2007*, pp. 90-97. Springer Berlin Heidelberg.

[15] Cai, Y., Yang, S. X., & Xu, X. (2013, April). A combined hierarchical reinforcement learning based approach for multi-robot cooperative target searching in complex unknown environments. In *Adaptive Dynamic Programming And Reinforcement Learning (ADPRL), 2013 IEEE Symposium on* (pp. 52-59). IEEE.

[16] Djurdjevic, P. D., & Huber, M. (2013, October). Deep Belief Network for Modeling Hierarchical Reinforcement Learning Policies. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on* (pp. 2485-2491). IEEE.

[17] Barto, A. G., & Mahadevan, S. (2003). Recent advances in hierarchical reinforcement learning. *Discrete Event Dynamic Systems*, *13*(4), 341-379.

[18] Kadlecek, D., & Nahodil, P. (2008, October). Adopting animal concepts in hierarchical reinforcement learning and control of intelligent agents. In*Biomedical Robotics and Biomechatronics, 2008. BioRob 2008. 2nd IEEE RAS & EMBS International Conference on* (pp. 924-929). IEEE.

[19] Joshi, M., Khobragade, R., Sarda, S., Deshpande, U., & Mohan, S. (2012, November). Object-Oriented Representation and Hierarchical Reinforcement Learning in Infinite Mario. In *Tools with Artificial Intelligence (ICTAI), 2012 IEEE 24th International Conference on* (Vol. 1, pp. 1076-1081). IEEE.

[20] Kawano, H. (2013, May). Hierarchical sub-task decomposition for reinforcement learning of multi-robot delivery mission. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on* (pp. 828-835). IEEE.

[21] Ponce, H., & Padilla, R. (2014). A Hierarchical Reinforcement Learning Based Artificial Intelligence for Non-Player Characters in Video Games. In *Nature-Inspired Computation and Machine Learning* (pp. 172-183). Springer International Publishing.

[22] Yan, Q., Liu, Q., & Hu, D. (2010, March). A hierarchical reinforcement learning algorithm based on heuristic reward function. In *Advanced Computer Control (ICACC), 2010 2nd International Conference on* (Vol. 3, pp. 371-376). IEEE.

[23] Dethlefs, N., & Cuayáhuitl, H. (2011, September). Combining hierarchical reinforcement learning and Bayesian networks for natural language generation in situated dialogue. In *Proceedings of the 13th European Workshop on Natural Language Generation* (pp. 110-120). Association for Computational Linguistics.

[24] Ichimura, T., & Igaue, D. (2013, July). Hierarchical modular reinforcement learning method and knowledge acquisition of state-action rule for multi-target problem. In *Computational Intelligence & Applications (IWCIA), 2013 IEEE Sixth International Workshop on* (pp. 125-130). IEEE.

[25] Botvinick, M. M. (2012). Hierarchical reinforcement learning and decision making. *Current opinion in neurobiology*, *22*(6), 956-962. ELSEVIER.

[26] Hengst, B. (2007). Safe state abstraction and reusable continuing subtasks in hierarchical reinforcement learning. In *AI 2007: Advances in Artificial Intelligence* (pp. 58-67). Springer Berlin Heidelberg.

[27] Hengst, B. (2010). Hierarchical Reinforcement Learning. In *Encyclopedia of Machine Learning* (pp. 495-502). Springer US.

[28] Wilson, A., Fern, A., Ray, S., & Tadepalli, P. (2007, June). Multi-task reinforcement learning: a hierarchical Bayesian approach. In *Proceedings of the 24th international conference on Machine learning* (pp. 1015-1022). ACM.

[29] Guo, Q., Zuo, L., Zheng, R., & Xu, X. (2013). A Hierarchical Path Planning Approach Based on Reinforcement Learning for Mobile Robots. In *Intelligence Science and Big Data Engineering* (pp. 393-400). Springer Berlin Heidelberg.

[30] Wang, J., Zuo, L., Xu, X., & Li, C. (2013). A hierarchical representation policy iteration algorithm for reinforcement learning. In *Intelligent Science and Intelligent Data Engineering* (pp. 735-742). Springer Berlin Heidelberg.

[31] Ribas-Fernandes, J. J., Solway, A., Diuk, C., McGuire, J. T., Barto, A. G., Niv, Y., & Botvinick, M. M. (2011). A neural signature of hierarchical reinforcement learning. *Neuron*, *71*(2), 370-379.

[32] Chen, F., Chen, S., Gao, Y., & Ma, Z. (2007, August). Connect-based subgoal discovery for options in hierarchical reinforcement learning. In *Natural Computation, 2007. ICNC 2007. Third International Conference on* (Vol. 4, pp. 698-702). IEEE.

[33] Ghavamzadeh, M., Mahadevan, S., & Makar, R. (2006). Hierarchical multi-agent reinforcement learning. *Autonomous Agents and Multi-Agent Systems*,*13*(2), 197-229.

[34] Maia, T. V. (2009). Reinforcement learning, conditioning, and the brain: Successes and challenges. *Cognitive, Affective, & Behavioral Neuroscience*,*9*(4), 343-364.

[35] Cuayáhuitl, H., & Dethlefs, N. (2011). Spatially-aware dialogue control using hierarchical reinforcement learning. *ACM Transactions on Speech and Language Processing (TSLP)*, *7*(3), 5.

[36] Mehta, N., Natarajan, S., Tadepalli, P., & Fern, A. (2008). Transfer in variable-reward hierarchical reinforcement learning. *Machine Learning*, *73*(3), 289-312.Springer.

Mostafa Al-Emran is the Head of Technical Support / Admission & Registration Department at Al Buraimi University College. Al-Emran got his BSc in Computer Science from Al Buraimi University College with the first honor level. He got his MSc in Informatics from The British University in Dubai with a distinction level. Al-Emran has published some research papers and is currently working on different research areas in Computer Science.