AN EXHAUSTIVE SURVEY OF REINFORCEMENT LEARNING WITH HIERARCHICAL STRUCTURE

Anshul Chaturvedi[1], Nidhi Mishra[2]

[1]*Research Scholar, Madhya Pradesh, India,* ***anshulchaturvedi03@gmail.com***
[2]*Research Scholar, Madhya Pradesh, India,* ***ndh.mishra1@gmail.com***

Abstract

Today reinforcement learning (RL) is holding the attention in research area under Machine Learning and AI. Hierarchical Reinforcement Learning (HRL) that break down the RL problem into sub-problems where solving of each sub-problem will be more powerful than solving the whole problem will be present in this paper. A review of the characteristics of HRL has been investigated as well as different domains have been highlighted those are based on HRL. Different domains must have different problems; some proposed solutions have been addressed. It has been discovered that HRL has not yet been that much discussed in the current existing research; the reason that motivated to work on this scenario. Some ideas have been come out into view during the work on this research and have been proposed for follow in future research.

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

----------***----------

1. INTRODUCTION

Reinforcement Learning has been an arousing research field in the domain of Machine Learning and AI. Due to self-adaptation and self-learning feature of RL that received many attentions from the fields of operations research [7]. RL algorithms work on enhancing the learning by agent while directly interacting with its environment [34]. HRL works on the principle of breaking down the RL problem into sub-problems where solving each sub-problem will be more powerful than solving the whole problem [8]. According to the recent years studies it is stated that the problem of "Curse of Dimensionality" (meaning that memory needs grow exponentially with the number of state variables) has been solved via HRL [17], [19], [14]. Then HRL works on reducing dimensionality by breaking down it into several levels. HRL overcome the agent-learning complexities at some extent that are considered as one of the typical issues in the learning environments [27]. Different domains must have different problems; some proposed solutions have been addressed.

The paper is organized as follows: section 2 lightens over the background on RL, HRL and Q-learning. Section 3 expresses the main contribution in the area of HRL. Section 4 has conclusion and the ideas that follow in future research.

2. BACKGROUND

2.1 Reinforcement Learning

RL come under the machine learning areas in which an agent and environment plays main role where these two has to interact with each other in order to achieve a goal as shown in Figure 1. Reinforcement learning based on the structure of Markov Decision Processes (MDPs); an agent learning structure interacting with its environment to receive rewards and drawbacks [30], [6], [16]. States, actions and reinforcements are the basic and fundamental elements of RL [24]. The agent recognizes the environment via agent's sensors and implements actions based on a policy that's the cause of environmental change. As per these changes, the agent receives rewards as per the taken actions [4], [25]. Learning through trial and error RL improves strategy by interacting with the environment and perceive the best actions at each state to reach toward the goal and obtain the best rewards [17], [3]. RL tries to find the best policy that enhances the total reward. Reinforcement Learning algorithms work on how the agent can learn to get an optimal strategy while to interact with its environment [34].

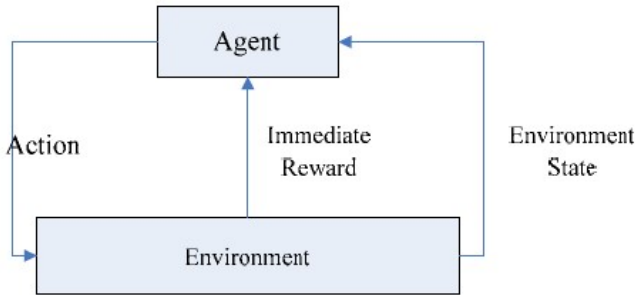

Fig -1: Reinforcement Learning Basic Model. [32]

2.2 Hierarchical Reinforcement Learning

HRL refers to the idea in which RL problem is divided into sub-problems where solving each of sub-problems will be more powerful than solving the whole problem [3], [13], [21], [20] and [28]. HRL defined as a set of computational techniques that enlarge the RL process to involve temporarily abstract actions [12] and [15]. That hierarchical break down has some benefits like: reducing the computational complexity of sub-problem, individually manage the sub-problems that will maximize its reusability

by which the learning process will speed up [3]. Different forms of abstraction are used by HRL techniques that handle the exponentially increasing number of parameters which are necessary to be learned specifically in large problems for perfectly reducing the search space which allows the agent to find the optimal solution [18]. "Curse of Dimensionality" problem has been resolved by HRL [29]. HRL with well-designed reward function can decrease the number of impractical acts of exploration which allows the agent to interact with the ease and in quick manner with the environment [19]. Natural Language Generation (NLG)'s utterance planning and content selection can be optimized via HRL along with Bayesian Networks [14]. Various HRL models are available that scale RL to large state spaces problems by breaking down them into sub-problems like MAXQ, Hierarchical abstract machines (HAMs), ALisp and options [31], [33];

2.3 Q-Learning

Q-learning is one of the well known RL algorithms that have been successfully used in different domains [22]. Q-learning tries to estimate an optimal action policy by finding the optimal state-action function Q(s, a) where s is the state from the set of the possible states S, a is known as action from the set of the possible actions A. The Q function shown the maximum reward received when the action 'a' is received by state 's' that is executed over the state 's' [4]. 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 learning rate is 'α', the discount factor is 'γ' and the reward is represented by 'r', noticed by execution of action 'a' over the state's'.

3. TECHNICAL PART

Different problems under different domains based on HRL will be focused and explored in this section:

3.1 Control Architecture Based on HRL for Semi-Autonomous Rescue Robots in Cluttered Environments

Urban search and rescue (USAR) scenes are disordered and the information about that kind of environment is already unknown due to their desolation. That's why finding victims in that kind of environments by human teleportation of rescue robots is a tedious task. To resolve the USAR problems number of different solutions has been given, some of them are as follows:

- Wirelessly teleported control: The search task has been very tedious for the robot under this technique because due to the environment nature the communication between the human and robot will be lost [36].
- Fully autonomous controllers: This technique is completely robot-based; though humans could not trust the robot in critical tasks. The fact that dust and debris in environments will affect the sensors so using this technique is quite challenging, hence this technique requires some more changes [36].
- Semi-autonomous control architecture based on HRL: HRL algorithm starts the robot to learn and make its own decisions on the basis of rescue tasks, identification of victim and exploration by performing these tasks in quick way and efficiently. The experiments revealed the effectiveness of the given technique by determining the ability of the robot while examine minutely the full USAR environment [18].

3.2 HRL in Computer Games

In AI and machine learning the computer games are one of the interesting topics for research. The Non-Player Characters (NPCs) behavior is one of the issues in computer games that catch researchers to work on it because of their complexity and difficulty in to be represented by typical finite state machines. At all stages the control description of NPCs are commonly hand-coded; that is the reason which makes the development task more time consuming and exposed for errors. HRL based on the Hierarchies of Abstract Machines (HAMs) helped to come out from these limitations. By using the proposed solution, system designers can discover stages within the program itself where they do not need to bother about how the code will be written while it's discovered by the robot's learning process. Experiments have been done to test the efficiency of the proposed solution under the Quake2UR (that performs as 3D Game Server) and ALisp system (that used perform as a client). Results announced that the proposed solution was quite flexible and full fills the requirement for controlling NPCs easily [21].

The MaxQ-Q HRL algorithm in the NPCs to increase the experience of user and to make better the natural humanness while interacting with computer games [1]. Results drown after comparing NPCs which is based on Finite State Machines (FSM) and the again NPCs that is based on MaxQ-Q through the game; indicated that NPCs - MaxQ-Q HRL are 52% far better than NPCs which is based on FSM.

Moreover, in the area of AI and machine learning the Infinite Mario game is the interesting and popular action based game. The domain of this game is very complex and contains large state-action spaces. HRL integrated along with the object-oriented representation to make lesser the state-action spaces in the game domain [11].

3.3 Option-based HRL's Course-Scheduling Algorithm

Traditional timetable scheduling system implements Reinforcement Learning algorithm. Whereas the reward of RL algorithm is not come out immediately; due to this reason algorithm suffers from the oscillation period. This will create impact on the RL algorithm to show that the timetable state dimension is very large while scheduling the course. An option-based HRL algorithm is applying to the timetable scheduling strategy to increase the performance of

traditional RL [17]. 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' shows the reward, 'γ' shows the discount rate factor, 'α' show the learning rate and't' is the time taken by the option. There are some environmental parameters such as; the instructor, course, college, major, semester, grade and classroom showing that the agent has no prior information about the environment before learning. Experiments result determined that the proposed algorithm is able to reduce the oscillation period. Moreover, as HRL is included in this algorithm, the course-scheduling actions are break down into sub-tasks; this will help the agent to learn in very quick manner and select the prominent strategy. Furthermore, the results shown that the Q-value update equation is far smoother than the regular Q-learning algorithm.

3.4 HRL Approach in Motion Planning in Mobile Robotics

In mobile robotics motion planning is one of the interesting tasks that plan for generating a collision free path from the beginning to the goal point for the robot. RL with the use of Neural Network is applied to avoid all the obstacles in mobile robotics [35]. However this became an old technique as [4] introduced an option-based HRL in which basic behaviors are used. In the learning process each behavior is learned individually; to solve the problem of the motion planning this individual learning process allows the robot to organize all the basic behaviors. Semi-Markov Q-learning has been used to calculate the state-option function values Q(s, o) by taking an 'o' as option in the state's' on the basis of policy 'μ'. After implementing option 'o', the final state's' 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' shows the discount rate, 'α' shows the learning rate and 'k' shows to number of steps between's' and 's''. Results said that in the unknown environment; the proposed algorithm has the ability to work effectively as well as in the task of motion planning with no use of Neural Networks it perfectly avoid all the obstacles come along the path by the robot.

3.5 HRL using Path Clustering

Small and medium scales RL problems resolve through the use of path clustering in order to enable its hierarchical decomposition [3]. HRL path clustering method has been introduced which allows the robot to gain the knowledge about the state's sequences which lead to the goal and propose those states at the end of the sequences as sub-goals. Taxi-problem (one of the standards in Reinforcement Learning and is being used the HRL solutions for testing) has been used. In this issue, sub-goals increase the learning speed by getting good results faster than the old traditional *Q-learning* due to the concept that problem scale is very small. It has been proposed to put the sub-goals into the process of learning. Results declared that the early involvement of sub-goals will get a sub-optimal learning.

3.6 HRL as Web Service Composition method

Web services composition make the easier combination of the single web-services into featured services it could satisfy the user's requirements as the individual single web-services could not so.

The dynamic web service composition model is presented in Figure 2. As "service agency i" presented in "task acceptor" obtained the data and then the flow chart is produced by "composed service engine" correspondingly.

The problem of optimization (that is how to find an optimal policy) is one of the main problems of the dynamic web-service composition. Number of solutions has been proposed to put an optimal policy for dynamic web service composition. RL based algorithm; however suffers from the "*Curse of dimensionality*" mainly in the problems of large-scale of web-service composition [26]. On the other side, a continuous time integrated HRL-MAXQ algorithm proposed to handle the problems of large-scale in the context of Semi-Markov decision process (SMDP) [10]. The comparison of this algorithm with the *Q-learning* algorithm has made also. Simulation results shown that the performance of MAXQ algorithm is far better than the *Q-learning* algorithm through their result of comparison both with a discount factor a = 0.01 because of this reason the MAXQ algorithm has the quality to increase the learning speed. Moreover, by the comparison of both algorithms with number of different tasks, it has been seen that as the number of tasks increase, the success rate of the *Q-learning* decreases very faster than MAXQ. The proposed algorithm is far better than the *Q-learning* algorithm to deal with the problem of *Curse of dimensionality* appearing in large-scale issues of web-service composition [10].

Another issue in dynamic web-service composition is that how to merge a collection of simple web-services based on the user's functional requirements and how to pick such services based on user's quality of service (QoS) requirements among all the available services. An HRL based algorithm proposed and Logic of Preference; the algorithm that efficiently handles with both user's functional and quality of service (QoS) requirements and has the quality to work in large-scale problems [23]. The algorithm is divided into two parts: Logic of Preference (for choosing the service) and MAXQ (for the service composition). An experiment has been done using 180 states and 500 web services. Experiments results declared that the computation cost is significantly going down (decrease) as the number of execution times is increasing. Moreover, results declared that using HRL can effectively speed up the composition task.

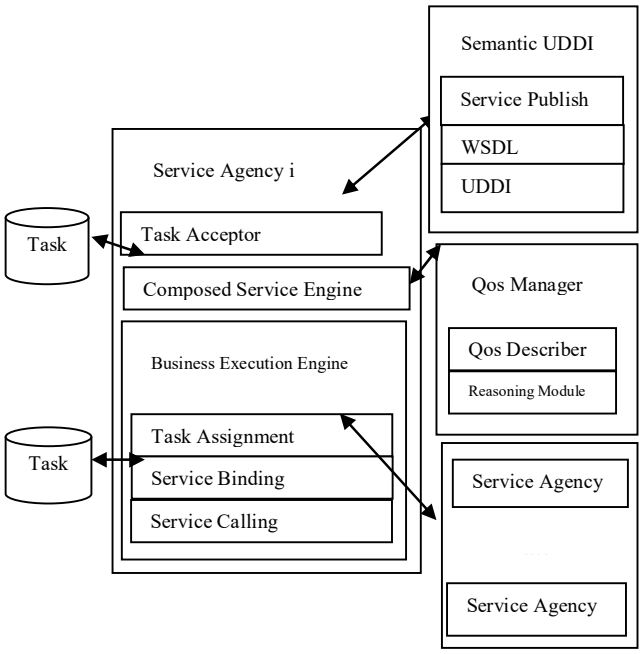

Fig -2: Dynamic Web Service Composition Model. [23]

3.7 Multi-robot Cooperative Target Searching in Complex Unknown Environments with the help of Combined HRL Based Approach

The fundamental fact in various applications like searching of target and environment exploration is the collaboration of multi-robots in unknown environments. The learning quality in many Reinforcement Learning approaches is temporary and it is one of the main weaknesses; this is due to the reason that it is environment-based; the quality to handle new environments and specifically dynamic environments. A joint approach of both Option and MAXQ algorithms in which the knowledge and the hierarchical structure are proposed and created respectively by both algorithms [32]. However, this solution still lacks the systematic consideration of the unnecessary parts of the environments. An effective HRL algorithm that joins both the Option algorithms and MAXQ as shown in Figure 3 where all the necessary parameters will be automatically collected through the learning, while other algorithmic approach selects parameters via trial and error [5]. The solution has proposed with the quality to estimate the feedback and tries to obtain featured parameters for coming up processes; because of this, that solution is unique for that kind of environments as comparing with the others. The simulation results declared that the given solution has the quality to allow a team of robots to collectively get target searching in unknown environments.

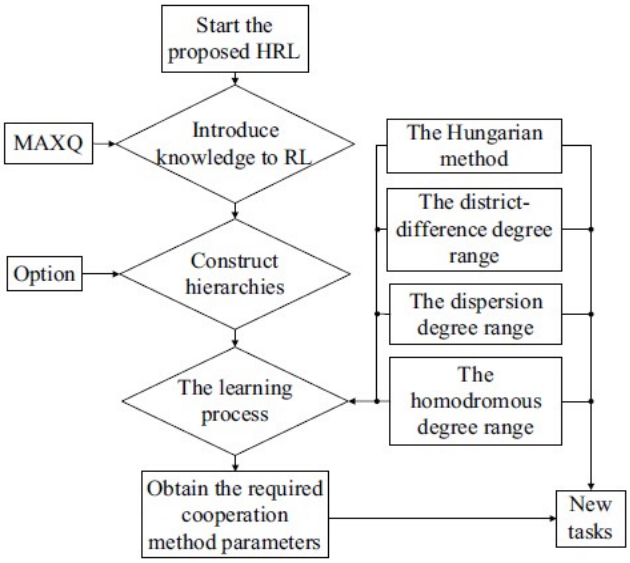

Fig-3: MAXQ and Option algorithms combination. [5]

3.8 Modeling HRL policies with Deep Belief Network

Intelligent robots worked on number of tasks during their entire lifetime that needs concurrent modeling as well as control the complexity in unknown environments. A major issue named as Policy Learning has to go from the problem of "*Curse of Dimensionality"* that is the main fact of scaling problems for regular RL. To deal with this issue, the robot should perfectly capture and reuse potential knowledge. A latest learning method for HRL on the basis of Conditional Restricted Boltzmann Machines (CRBMs) to deal with the growing learning and scaling problems for regular RL [2]. A simple Taxi domain was proposed to discover the learning efficiency and show the HRL policies. The proposed taxi domain shows a car in 1D space that chooses a packet from a state and leaves it at a destination as shown in Figure 4. HRL based-CRBMs have capability to provide a uniform means to concurrently learn policies and add abstract state features under a reliable network structure.

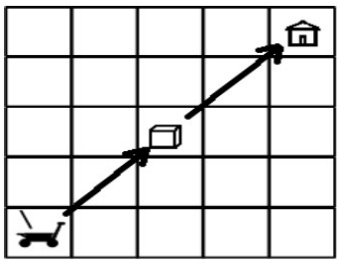

Fig-4: Simplified Taxi Domain. [2]

4. CONCLUSION & FUTURE WORK

In the area of Machine Learning and AI, RL plays an important role. HRL concentrates on dividing the RL problems into sub-problems where solving each sub-problem separately is simpler and more powerful than solving the whole problem. State-of-the-art of HRL has been reviewed and investigated. Number of different research

areas with different problems on the basis of HRL has been surveyed under this paper such as: rescue robots from cluttered environments, games of computer, scheduling of course, planning of motion in mobile robotics, web service composition, clustering of path, multi-robot co-operation and intelligent robots.

During the work on this survey, some ideas have been discovered and can be carried out as a future work; the ideas that need more focus from researchers who are interested in the field of HRL. Those ideas could be summarized as under:

- Will multi-robot cooperation be a capable way to find victims in cluttered USAR environments? As in comparison with the discussed solutions by [36], [18]. Moreover, these solutions have been used in environment of small scale so future research requires focusing on large-scale environments to judge their efficiency.
- How multi-robot cooperation will support the web-service composition problems? As to compare with [10].
- A smaller discrete RL issue by the path clustering is focused in [3]. Further research may concentrate on larger continues discrete RL issues.
- MAXQ and Q-learning algorithms in the area of web-service compositions are compared in [10]. Further work could make the comparison of the two algorithms in robots race and observe which one is going to learn and reach the goal state faster?

REFERENCES

[1] 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.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

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

[13] 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.

[14] 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.

[15] 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.

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

[17] 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.

[18] 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.

[19] 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.

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

[21] 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.

[22] 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.

[23] 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.

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

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

[26] 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.

[27] 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.

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

[29] 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.

[30] 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.

[31] 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.

[32] 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.

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

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

[35] 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.

[36] 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.