*Proceedings of the International Conference on*
*Information Technologies (InfoTech-2017)*
*20-21 September 2017, Bulgaria*

# GRAPH GENERATION APPROACH FOR AGENT BEHAVIOUR

**Vanya Markova, Ventseslav Shopov**

*Institute of Robotics – BAS*
*e-mail: markovavanya@yahoo.com*
*Bulgaria*

**Abstract:** This paper deals with the problem for automatic graph generation and clustering for construction of curriculum learning. The proposed approach is based on reinforcement learning framework. In addition we use transfer of full transferred reward (Reward FT) knowledge to achieve better reward. Inseparable part of this methods is a graph generation and graph clustering.
**Key words:** reinforcement learning, graph clustering, graph generation, knowledge transfer

## 1. INTRODUCTION

In recent years, building the behaviour of autonomous agents in dynamically changed environment became a widespread among the machine learning scientists. From one side reinforcement learning is well suited to many sequential decision problems, from the other hand problems characterized by delayed reword are difficult to solve. In such cases an appropriate solution is the use of knowledge transfer of knowledge by learning methods.

The agent have to learn from prior tasks and after that to transfer that experience to improve future performance. This is a key aspect of intelligence, and is critical for building lifelong learning agents. Recently, multi-task and transfer learning received much attention in the supervised and reinforcement learning (RL) setting encouraging results.

During the 90's Elman put forward the idea that a curriculum learning of progressively harder tasks could significantly accelerate a neural network's training [4]. However the lack of computational resources delays development curriculum

learning methods for years. So this approach has only recently become popular in the field [3] of the machine learning, due in part to the greater complexity of problems now being considered. In particular, recent work on learning programs with neural networks has relied on curricula to scale up to longer or more complicated programs [5, 12, 16].

Another issue is the decision about which task to study next. One solution is to treat this decision as a stochastic policy, continuously adapted to optimise some notion of what [11] termed learning progress. So various indicators of learning progress could be used as reward signals to encourage exploration, including compression progress [1], information acquisition [15], Bayesian surprise [7], prediction gain [2] and variation information maximisation [6]. Lastly, there are recent effort on using Bayesian optimisation to find the best order in which to train the embedding network [17].

The reward function is a very import in regard of transferring the knowledge gain. Ng investigates conditions under which modifications to the reward function of a MDP preserve the optimal policy [10]. Konidaris introduces shaping rewards in reinforcement learning tasks that result in accelerated learning in late tasks that are related but distinct [8]. Svetllik addresses framework for automatic construction of curriculum [14]. Lee and Popovic introduce learning behaviours styles with inverse reinforcement learning and show that the discovered reward function can be applied to different environments and scenarios [13, 18].

We focus on variants of prediction gain, and also introduce a class of progress reward signals which we refer to as complexity gain. So the agent is capable to transfer from multiple tasks into a single target task and, as a result curriculum as a Directed Acyclic Graph is generated. This is an approach to use heuristic transfer potential as a measure for usefulness of transferring from a task. The aim in this paper is to automatic framework that generate curriculum graph for autonomous agents which work together.

## 2. MATERIAL AND METHODS

In sequential decision problems the 'task' is representing by the reward function. Reinforcement learning is the problem of getting an agent to act in the world so as to maximize its rewards. The agent's goal is to find a policy and state-update function so as to maximize the expected sum of discounted rewards. [10] Given the reward function and a model of the domain the optimal policy is determinate. The method proposed in this paper uses reward shaping [13] as a means to transfer knowledge. Here, we use formalization of training strategies in the context of machine learning and base on hypothesize that curriculum learning can be seen as continuation methods with optimization strategies for dealing with minimizing non-convex criteria.

### 2.1 Reinforcement learning

The problem with RL can be formatted as follows:

The environment is modelled as a stochastic machine with end states with inputs (actions sent by the agent) and outputs (observations and awards sent to the agent.

The state transition function $P(X_t / X_{t-1}), A_t$

Observation function (output) $P(Y_t / X_t, A_t)$

The prize function $E(R_t / X_t, A_t)$

The agent is also modelled as stochastic FLM with inputs (observations / awards sent from the environment) and outputs (actions sent to the environment).

The transition state function: $S_t = f(S_{t-1}, Y_t, R_t, A_t)$

Policy / Output Function: $A_t = \pi S_t$

The MDP is just like the Markov chain, except for the transition matrix, which depends on the actions taken by the decision maker, which is the policy that determines what action to take in each country so as to maximize Some function of the sequence of awards. One can formalize this with regard to Belman's equation, which can be iteratively decided through political iteration. The unique fixed point of this equation is the optimal policy.

Namely, to define the matrix of transition and the function of remuneration as follows:

$T(s,a,s') = \Pr[S_{t+1} = s'', S_t]$

$T_{t+1} S_t = a, A_t = a, S_{t+1}$

We assume that states, actions and time are discreet and continuous tasks can be defined, but they are usually solved by sampling. The agent's goal is to find a policy and status update feature so as to maximize the expected amount of reduced wages

$E(R_0 + gR_1 + g^2 R_2 + .....) = E\sum_{t=0} g^t R_t \quad g \le 1$ is a discount factor that models the fact that the future prize is less than the immediate prize

We determine the value of the executing action a in state s as follows:

$Q(s,a) = \sum s'' T(s.a.s') R(s,a,s')$

Where 0 <g <= 1 is the sum by which we reward future prizes, and V (s) is the total value of state s given by the Bellman equation:

$V(s) = \max a$ and $Q(s,a) = \max \sum_{s'} T(s,a,s')$

In words, the state's value is the maximum expected prize we will receive in this country, plus the expected discounted value of all possible inheritance states, s'. If we determine

$R(s,a,s') = \sum_{s'} T(s,a,s')$

The above equation has been simplified to the more common form
$V(s) = \max aR(s,a) + \sum_{s'} T(s,a,s')gV(s')$

For fixed policy and table (non-parametric) representation of V / Q / T / R functions can be rewritten in matrix vector form as $V = R + gTV$. The solution of these n simultaneous equations are called determination of the N value is the number of states). In a special case that Y (t) = X (t), we say that the world is fully visible and the model turns into the Markov Decision Process (MDP).

MDP $M = (S, A, T, g, R)$

We will transform MDP $M = (S, A, T, g, R')$,

where $R' = R + F$ and $F : SxAxS' \to R$ is a limited rating function called rewarding at each step of the time.

The agent receives a reward that depends on the action and the condition. The goal is to put a function called a policy that defines what action to take in each country so as to maximize some function of the sequence of prizes. One can formalize this with regard to Belman's equation, which can be iteratively decided through political iteration. The unique fixed point of this equation is the optimal policy.

Our aim is to find a suitable curriculum for the task which uses only task from T. The curriculum is the directed, acyclic graph, weakly connected.

### 2.2. Graph generation and clustering for agent's curriculum

The unordered set of tasks T and final task $t_f$ are given. We group tasks which share certain features. After that we apply graph clustering to partition the task set. As a result the clusters with more "alike" features set are formed.

To ensure the effective knowledge transfer one could choose the tasks with high potencial first. Instead of that we prefer to eliminate the tasks with low potential. Given these observations, we define transfer potential as:

$$v(s,t) = \frac{\left|D_{Q_s}(t)\right|}{1 + \left|S_t\right| - \left|S_s\right|}$$

Hence transfer potential is defined between pairs of tasks [14 ]. To allow more than one source task we define experience as a group attribute. So that experience could be calculated on sets of source tasks.

$$E_t(C) = E\left[\left(\sum_{t\in V, s, tt\neq t_f} e(X_t, t\right) + e_f(X_f, l)\right], l - threshold$$

In intra-group transfer, we consider transfer between tasks within the same group. From every group we choose only the small fraction of tasks with greater potential with respect to the transfer potential to the final task. So we iterate over the group, assigning an edge between tasks that have the highest potential among the tasks in the group. We return the resulting set of edges. So intra-group transfer constructs sub-graphs on each task group.

Inter-group transfer is then carried out amongst the task groups to further assign edges. While any given group contains tasks that are themselves related, there may be tasks in separate groups that would also benefit from transfer. This is the goal of Inter-group transfer. Two task groups are compared directly for transfer. The group g is considered the source group and the other, h, is considered the target group. The algorithm iterates over g, and adds an edge to a task in h so that edge $(g,h)$ between tasks that have the highest potential among the tasks in the two groups.

## 3. EXPERIMENTS AND RESULTS

We illustrate the proposed method for curriculum generation graph in the following case study:

Block Dude, performed at BURLAP [9], is a 2-D puzzle in which the agent must reach the exit by arranging blocks to climb over obstacles made of bricks.

The agent can move left and right, pick up and put blocks and climb only if there is a block or brick in front of him. For each action, one is taken out of the prize, and a payout of +50 is given to reach the exit.

The task is resolved successfully after the exit. If the agent falls into an unfavorable situation (for example, by building a tower with blocks that can not climb) or the number of actions taken reaches 200, the task is broken.

Domains of freedom of the domain were the width and height, the number of blocks and the number of brick columns of different height. Combining these attributes leads to 10 variants of tasks used as a set of source tasks. Once the initial conditions are generated, the agent applies the algorithm of 2.2. A function of the prize is the approximation of the Radial Basis function as applied to the BURLAP.

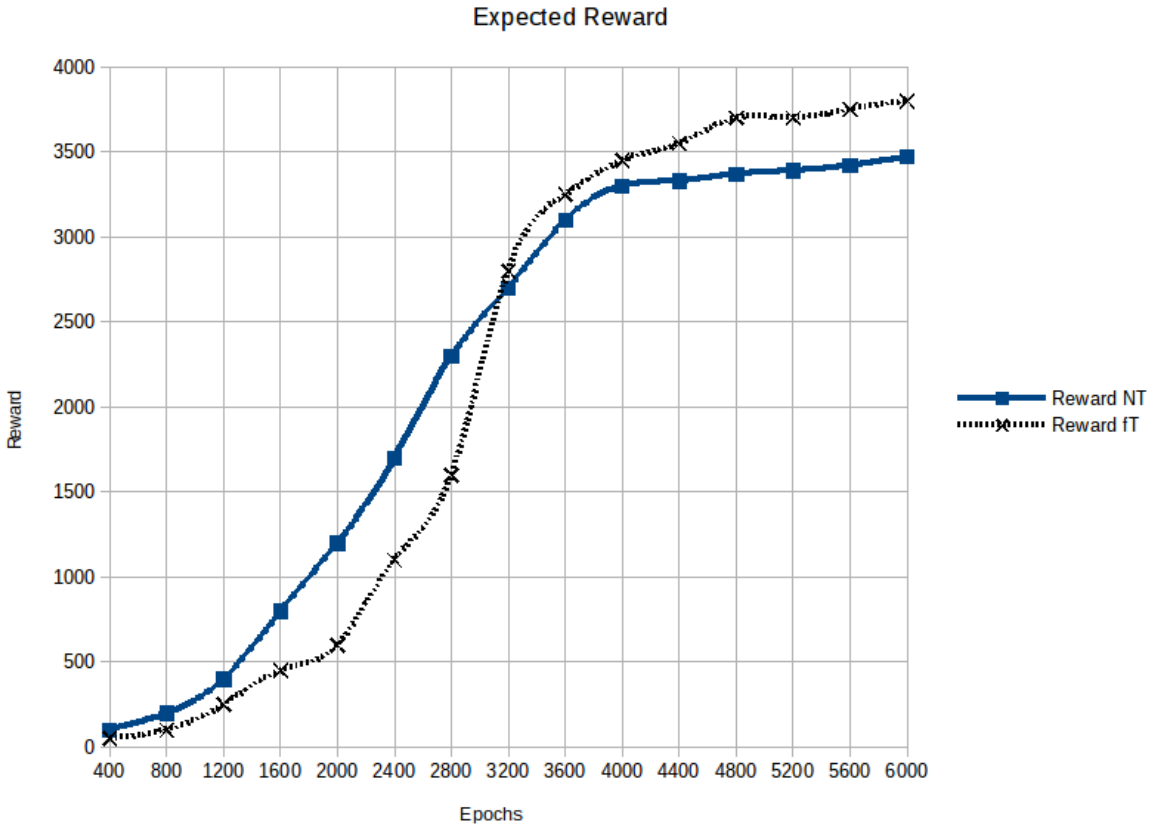


*Fig. 1. Expected rewards of learned politics*

In one case we have not transferred reward (Reward NT), in the other case we have full transferred reward (Reward FT).

From Fig 1 one can see that transfer of reward is superior when epochs grow

Table 1.

| Epochs | Reward NT | Reward fT |
|---|---|---|
| 400 | 99 | 48 |
| 800 | 211 | 112 |
| 1200 | 404 | 239 |
| 1600 | 786 | 447 |
| 2000 | 1190 | 612 |
| 2400 | 1643 | 1087 |
| 2800 | 2290 | 1603 |
| 3200 | 2687 | 2765 |
| 3600 | 3092 | 3243 |
| 4000 | 3287 | 3452 |
| 4400 | 3338 | 3551 |
| 4800 | 3374 | 3711 |
| 5200 | 3391 | 3723 |
| 5600 | 3423 | 3744 |
| 6000 | 3471 | 3804 |

It is obvious that in the case of simple worlds the direct approach without transfer of knowledge is significantly more effective. This is due to the fact that for

simple cases there is no need for knowledge transfer, because the agent can build simple and effective behavior. However, as the complexity of tasks grows and we have a lot of tasks, the efficiency of the knowledge transfer approach is considerably rising. Therefore, with the increase in the number of ages, the efficiency of the transfer of the function of the reward increases.

## 4. CONCLUSION

In this paper is presented the problem for automatic graph generation for constructed of curriculum based on reinforcement learning framework. Inseparable part of these methods is a graph generation and partition approach for task separation and grouping.

The intra-group and inter-group transfer of knowledge is combined in order to choose the hugest reward transfer. The approach wit transferred reward and direct approach without transfer are compared.

In addition we use full transferred reward (Reward FT) knowledge to achieve better reward. With the increase in the number of epochs, the efficiency of the knowledge transfer of the function of the reward increases significantly.

## REFERENCES

[1] Barto, A. G. (2013). Intrinsic motivation and reinforcement learning. In Intrinsically Motivated Learning in Natural and Artificial Systems, pages 17-47. Springer.

[2] Bellemare, M. G., Srinivasan, S., Ostrovski, G., Schaul, T., Saxton, D., and Munos, R. (2016). Unifying countbased exploration and intrinsic motivation. In Advances in Neural Information Processing Systems.

[3] Bengio Y, Louradour J, Collobert R, Weston J. Curriculum learning. InProceedings of the 26th annual international conference on machine learning 2009 Jun 14 (pp. 41-48). ACM.

[4] Elman, J. L. (1993). Learning and development in neural networks: The importance of starting small. Cognition, 48(1):71-99.

[5] Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka,I., Grabska-Barwi?nska, A., Colmenarejo, S. G.,Grefenstette, E., Ramalho, T., Agapiou, J., et al. (2016).Hybrid computing using a neural network with dynamic external memory. Nature, 538(7626):471-476.

[6] Houthooft, R., Chen, X., Duan, Y., Schulman, J., De Turck,F., and Abbeel, P. (2016). Vime: Variational information maximizing exploration. In Advances In Neural Information Processing Systems, pages 1109-1117.

[7] Itti, L. and Baldi, P. (2009). Bayesian surprise attracts human attention. Vision research, 49(10):1295-1306.

[8] Konidaris, G., and Barto, A. (2006). Autonomous shaping: Knowledge transfer in reinforcement learning. In Proceedings of the 23rd international conference on Machine learning, 489–496. ACM.

[9] MacGlashan, J. (2016). Brown-UMBC reinforcement learning and planning (BURLAP).. http://burlap.cs.brown.e

[10] Ng, A. Y.; Harada, D.; and Russell, S. (1999). Policy invariance under reward transformations: Theory and application to reward shaping. In ICML, volume 99, 278–287.
[11] Oudeyer, P., Kaplan, F., and Hafner, V. (2007). Intrinsic motivation systems for autonomous mental development.IEEE Transactions on Evolutionary Computation,11(2):265-286.
[12]Reed, S. and de Freitas, N. (2015). Neural programmerinterpreters. arXiv preprint arXiv:1511.06279.
[13] Russell, S. J., and Norvig, P. (2003). Artificial Intelligence: A Modern Approach . Pearson Education, 2 edition.
[14]. Svetlick M, M. Leonetti, J. Sinapov, R. Shah,N. Walker,, P. Stone. (2017), Proceedings of the 31st AAAI Conference on Artificial Intelligence . San Francisco Automatic Curriculum Graph Generation for Reinforcement Learning Agents
[15] Storck, J., Hochreiter, J., and Schmidhuber, J. (1995). Reinforcement driven information acquisition in nondeterministic environments. In Proceedings of the International Conference on Artificial Neural Networks, vol.2.
[16] Sutskever, I. and Zaremba,W. (2014). Learning to execute. arXiv preprint arXiv:1410.4615.
[17] Tsvetkov, Yulia, F. M. L. W. M. B. D. C. (2016). Learning the curriculum with bayesian optimization for task specific word representation learning. arXiv preprint arXiv:1605.03852
[18] Lee, Y., Wampler, K., Bernstein, G., Popović, J., Popović, Z. (Feb. 1, 2010). In ACM Transactions on Graphics (Proc. SIGGRAPH Asia) , 29 (6), 138