Fast Adaptation of Behavior to Changing Goals with a Gamma Ensemble

Chris Reinke
Neural Computation Unit
Okinawa Institute of Science and Technology
Okinawa 904-0495, Japan
chris.reinke@oist.jp

Eiji Uchibe
Department of Brain Robot Interface
ATR Computational Neuroscience Laboratories
Kyoto 619-0288, Japan
uchibe@atr.jp

Kenji Doya
Neural Computation Unit
Okinawa Institute of Science and Technology
Okinawa 904-0495, Japan
doya@oist.jp

Abstract

Humans and artificial agents not only have to cope with changes in their environments, but also with changes in the goals that they want to achieve in those environments. For example, during foraging the goal could change from obtaining the most desirable food to securing food as rapidly as possible if there is time pressure. In reinforcement learning, the goal is defined by the reward function and how strongly rewards are discounted over time. If the goal changes, model-free value-based methods need to adapt their values to the new reward function or discounting strategy. This relearning is time-intensive and does not allow quick adaptation.

We propose a new model-free algorithm, the Independent Gamma-Ensemble (IGE). It is inspired by the finding that the striatum has distinct regions to encode values computed by different discount factors. Similarly, the IGE has a set of distinct modules, which are Q-functions with a different discount factors. This allows the IGE to learn and store a repertoire of different behaviors. Furthermore, it allows information about the outcome of each behavior to be decoded, making it possible to choose the best behavior for a new goal without relearning values. In a task with changing goals, the IGE outperformed a classical Q-learning agent.

The IGE is a step toward adaptive artificial agents that can cope with dynamic environments in which goals also change. Furthermore, the IGE provides a model for the potential function of the modular structure in the striatum. The striatum, which is involved in habit learning, may learn different habits in its distinct regions with different discounting factors. Depending on the context, which could be indicated by the stress level, for example, the most appropriate habit could be used without the need to relearn. This may mean that the striatum is able to learn and retain several habits for the same environment and to select them in a context-dependent manner.

Keywords: goal adaptation, model-free, Q-learning, transfer learning, value-based

1 Introduction & Problem Definition

An adaptive agent should not only be able to adapt to changes in its environment, but also to the goal that it wants to achieve in that environment. Consider, for example, one of our daily routines: going out for lunch. We have to learn the way to a good restaurant. We explore our surroundings, find several restaurants, and learn the shortest path to them. Then we decide between the different restaurants based on their reward value, e.g. the food quality, and the time to get there. We can model such tasks as episodic MDPs: (S, A, T, R) (see Fig. 1 for an example). The state space S is our position. The actions A are the directions in which we can go. We have a transition function $T(s, a, s')$ which defines what position s' we can reach from position s with action a in a fixed time, e.g. $\Delta t = 30$ seconds. If we reach a restaurant, i.e. a terminal state $G \subset S$, the reward function $R(s \in G)$ defines its food quality. We consider these factors to be external factors, which usually do not change significantly over time.

Nonetheless, there can be rapid changes in the goal that we want to achieve in this environment. One day we may want to go to the best restaurant in the area, i.e. the one that gives the highest reward. On another day, we may be experiencing stress at work and we have to judge between the restaurant quality and the time to get there. Another day we may have to entertain guests and we want to go to the closest restaurant that provides food with a certain minimum level of quality.

Current reinforcement methods include such goal formulations in the reward function R and their discounting of values over time. Model-free methods, such as Q-learning, suffer from the drawback that they need to relearn their values from new observations, i.e. experiencing the environment with the new reward function or value discounting. Learning takes time and the behavior cannot adapt from one day to the next. Model-based methods can solve this problem by maintaining a model of the external environment. The model can be used to find the best behavior for the current goal. Nonetheless, depending on the task complexity, this procedure can be computationally demanding.

We propose a new model-free algorithm, the Independent γ-Ensemble (IGE), to circumvent such problems. The algorithm is inspired by research about human intertemporal decision-making by Tanaka et al. [3]. They identified distinct brain regions in the striatum, each learning values for choices with a different discount factor. The IGE employs this concept by having several γ-modules representing Q-functions with different discount factors γ. This allows the IGE to learn and store different behaviors, e.g. going to different restaurants, and to choose the appropriate behavior, depending on the current goal.

Before describing the IGE in more detail, we introduce a distinction between external and internal reward. External reward is provided by the environment and is defined by the reward function $r_t = R(s_t)$, e.g. the food quality of a restaurant. Internal reward $\phi(r_t, t)$ depends on the goal that the agent has to achieve and modulates the external reward depending on the time t that was needed to reach it. For example, under stress we might want to reduce the external reward linearly by the time we needed to reach it: $\phi(r_t, t) = r_t - 10t$. The final objective of the agent is to maximize the sum over internal rewards. In episodic environments, the number of steps T needed to attain the maximum reward sum should also be minimized:

$$\min_{T} \max_{\pi} E_{\pi} \left[\sum_{t=0}^{T} \phi(r_t, t) \right]. \tag{1}$$

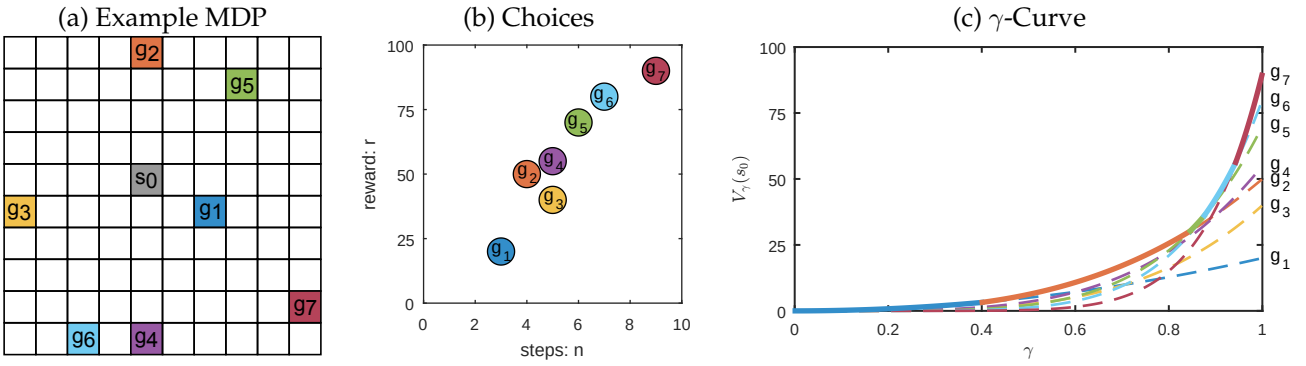

Figure 1: (a) A 2D grid world MDP with 7 terminal states $g_1, \ldots, g_7$. The agent starts in state s_0. It can move in 4 directions (north, east, south, west). Transitions result in zero reward until a terminal state is reached. (b) If the agent starts in state s_0 it has several choices to choose from. Each choice represents the minimal number of steps n and the reward r of reaching one of the possible terminal states. (c) Discounted values $V_\gamma(s_0, g) = \gamma^{n_g - 1} r_g$ for each choice for state s_0 (broken lines). The solid line represents the values the γ-Ensemble would learn ($\max_g V_\gamma(s_0, g)$). For state s_0 the ensemble will learn policies to go to terminal states g_1, g_2, g_5, g_6, g_7. It will not learn policies to go to g_3, g_4.

2 The Independent γ-Ensemble

The IGE is composed of γ-modules. Each module is a Q-function $Q_\gamma(s,a)$ with a different discount factor $\gamma \in (0,1)$. The functions are independent of each other and learn their Q-values based on the external reward $r = R(s')$. Their values are updated in parallel with the standard Q-learning rule after a transition (s,a,r,s') is observed:

$$Q_\gamma(s,a) \leftarrow Q_\gamma(s,a) + \alpha \left(r + \gamma \max_{a'} Q_\gamma(s',a') - Q_\gamma(s,a) \right). \tag{2}$$

Over time, each γ-module learns its optimal policy to maximize the discounted reward sum: $E\left[\sum_{t=0}^{T} \gamma^t r_t\right]$. Because γ differs between modules, their optimal policies differ. Modules with large γ's have a slow discounting and learn policies that result in big rewards, but that require more steps to reach. Smaller γ's have a steeper discounting and prefer smaller rewards that are reached with fewer steps. Fig. 1 (c) shows how γ-modules discount the reward for each terminal state at state s_0 for the example MDP. Modules with $\gamma < 0.4$ prefer the nearby terminal state g_1 ($r = 25$, $n_{s_0} = 3$). Whereas modules with $\gamma > 0.95$ prefer the more distant terminal state g_7 ($r = 90$, $n_{s_0} = 9$) that gives a higher reward. Therefore the IGE holds a set of policies.

In addition to providing a set of policies, the γ-ensemble can also provide information about the expected reward trajectories for those policies. This information can help to determine which policy is best for a given goal. As seen in Fig. 1 (c), modules with similar discount factors $\gamma_a \approx \gamma_b$ often result in the same trajectory, if they start from the same state s. Therefore their values $V_\gamma(s)$ are computed based on the same expected reward trajectory $R_\tau = (E[r_0], E[r_1], \dots, E[r_T])$ with $V_\gamma(s) = E[r_0] + E[r_1]\gamma^1 + \cdots + E[r_T]\gamma^T$. Having $T+1$ modules that follow the same trajectory, their values can be defined by the matrix multiplication:

$$V = \Gamma \times R_\tau, \text{ with } V = \begin{bmatrix} V_{\gamma_0}(s) \\ \vdots \\ V_{\gamma_T}(s) \end{bmatrix}, \ \Gamma = \begin{bmatrix} 1 & \gamma_0 & \gamma_0^2 & \cdots & \gamma_0^T \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \gamma_T & \gamma_T^2 & \cdots & \gamma_T^T \end{bmatrix} \text{ and } R_\tau = \begin{bmatrix} E[r_0] \\ \vdots \\ E[r_T] \end{bmatrix}.$$

Based on this formulation it is theoretically possible to decode the expected reward trajectory from the learned values with $R_\tau = \Gamma^{-1} V$. Unfortunately, this method has practical problems if applied to general MDPs. First, it is difficult to determine which γ-modules will result in the same trajectory. Furthermore, using the inverse Γ^{-1} to compute R_τ is theoretically possible, but numerically unstable because the inverse becomes ill-conditioned for large T.

Nonetheless, the idea of using several modules to decode information about their expected reward trajectory can be used in the more restricted set of deterministic, goal-only-reward MDPs. Such MDPs have deterministic state transitions $s' = T(s,a)$ and reward is only given if a terminal state is reached: $\forall s \notin G : \ R(s) = 0$. Although these are strict restrictions, many natural decision tasks, such as the given restaurant example, can be modeled with them. In such MDPs, the value definition is simplified to $Q_\gamma(s,a) = \gamma^{n_s} r$ with $V(s) = \max_a Q(s,a)$, where r is the expected reward gained from ending in terminal state g and n_s is the number of steps to reach it from state s. For two modules (γ_a, γ_b), which prefer the same terminal state g, their expected reward r and the number of steps n_s to reach g can be computed by solving the linear equation:

$$\begin{aligned} V_{\gamma_a}(s) &= \gamma_a^{n_s} \cdot r \\ V_{\gamma_b}(s) &= \gamma_b^{n_s} \cdot r \end{aligned} \quad \Leftrightarrow \quad V_{\gamma_a}(s) - \gamma_a^{n_s} = V_{\gamma_b}(s) - \gamma_b^{n_s} \quad \Leftrightarrow \quad \begin{aligned} n_s &= \tfrac{\log(V_{\gamma_a}(s)) - \log(V_{\gamma_b}(s))}{\log(\gamma_a) - \log(\gamma_b)} \\ r &= \tfrac{V_{\gamma_a}(s)}{\gamma_a^{n_s}} \end{aligned} \tag{3}$$

Therefore, the IGE can compute $r(\gamma_i, \gamma_{i+1})$ and $n_s(\gamma_i, \gamma_{i+1})$ for the policies of neighboring γ-modules (γ_i, γ_{i+1}), because they usually have the same trajectories. A problem occurs if the trajectories are different, as for $(\gamma_a = 0.4 - \epsilon, \gamma_b = 0.4 + \epsilon)$ in Fig.1(c). In this case, the computed r and n_s are wrong. Nonetheless, these cases can be detected by comparing them to neighboring pairs. If $n_s(\gamma_a, \gamma_b) \neq n_s(\gamma_b, \gamma_c) \neq n_s(\gamma_c, \gamma_d)$, then the result of pair (γ_b, γ_c) is ignored. As a result, the IGE not only learns and stores different policies, but it also provides information about their expected rewards and the number of steps required.

This allows the IGE to adapt quickly to different task goals (Eq. 1). Given a certain goal formulation ϕ_k, it can compute the outcome, i.e. the expected internal reward sum, for each of its learned policies. Then it can select the policy that maximizes that reward sum. This is done in the beginning of an episode after the initial state s_0 is observed. First, the reward $r(\gamma_i, \gamma_{i+1})$ and the number of steps $n_{s_0}(\gamma_i, \gamma_{i+1})$ for each neighboring module pair are computed (Eq. 3). Then the internal reward sum is calculated by $\sum_{t=0}^{n_{s_0}-1} \phi_k(0,t) + \phi_k(r, n_{s_0})$. The IGE chooses then one of the modules that generates the highest internal reward sum with the fewest steps (Eq. 1). The policy of this module is then used throughout the whole episode to control the agent. If in the next episode a different goal $\phi_{j \neq k}$ is used, the IGE can immediately adapt by again choosing a module with the most appropriate policy.

3 Experiments

We tested the goal adaptation of the IGE and compared it to a time-dependent Q-learning algorithm in the MDP defined in Fig. 1. Both agents had to adapt to 8 different goals $\Phi = (\phi_1(r,n), \ldots, \phi_8(r,n))$ (Fig. 2). For convenience, we assumed that the internal reward function only gave reward when a terminal state was reached with n steps and an external reward of r. The agents had 3000 episodes to learn the first goal ϕ_1. Then the goal switched to ϕ_2 and the agents were given 1000 episodes to adapt. For each of the following goals, the agents also had 1000 episodes to adapt.

The first goal ϕ_1 is to receive the maximum external reward in an episode. The second goal ϕ_2 also maximizes external reward, but a punishment of -10 is given for each step beyond the third. ϕ_3 gives exponentially increasing punishment for more than 3 steps. The goal of ϕ_4 is to find the shortest path to the closest terminal state. For ϕ_5, the goal is the shortest path to a terminal state that gives at least an external reward of 65. Reaching a terminal state with less external reward will result in a strong negative internal reward. For ϕ_6, the goal is to find the highest external reward in less or equal 7 steps. For ϕ_7, the agent has only a limit of 5 steps. In ϕ_8, the goal is to maximize average reward.

The IGE used the procedure described in the previous section to learn different policies based on the external reward and to chose the most appropriate policy for the current goal ϕ_k. The time-dependent Q-learning learned values directly based on the internal reward. It held values for each possible time step $t = (0, \ldots, 50)$ during an episode. During learning, the values for all possible time points t were updated after an observation of (s, a, r, s') with:

$$Q(t,s,a) \leftarrow Q(t,s,a) + \alpha \left(\phi_k(r,t) + \gamma \max_{a'} Q(t+1, s', a') - Q(t,s,a) \right). \tag{4}$$

For both agents, an ϵ-Greedy action selection was used with a decaying epsilon over episodes: $\epsilon(j) = \max\left(\epsilon_0 \cdot d^j, 1\right)$.

Learning parameters of both agents were manually optimized. Both agents used the learning rate $\alpha = 1$. The IGE used 100 modules with logarithmically spaced γ parameters between 0 and 1. The parameters have a higher density toward 1 to allow better coverage of policies that result in longer paths. For the first goal ϕ_1, both agents used the same action selection parameters: $\epsilon_0 = 2, d = 0.9975$. For the IGE, action selection in all other episodes had no exploration ($\epsilon = 0$) because it could adapt immediately to a changed goal formulation. The Q-learning required exploration ($\epsilon_0 = 1, d = 0.99$) to adapt its Q-values to find the new optimal policy.

For the first goal ϕ_1, the performance of both algorithms is identical (Fig. 2), because both have to learn the Q-values for the MDP. For most of the successive tasks ($\phi_2, \ldots \phi_6, \phi_8$), the IGE can directly select the optimal policy. In contrast,

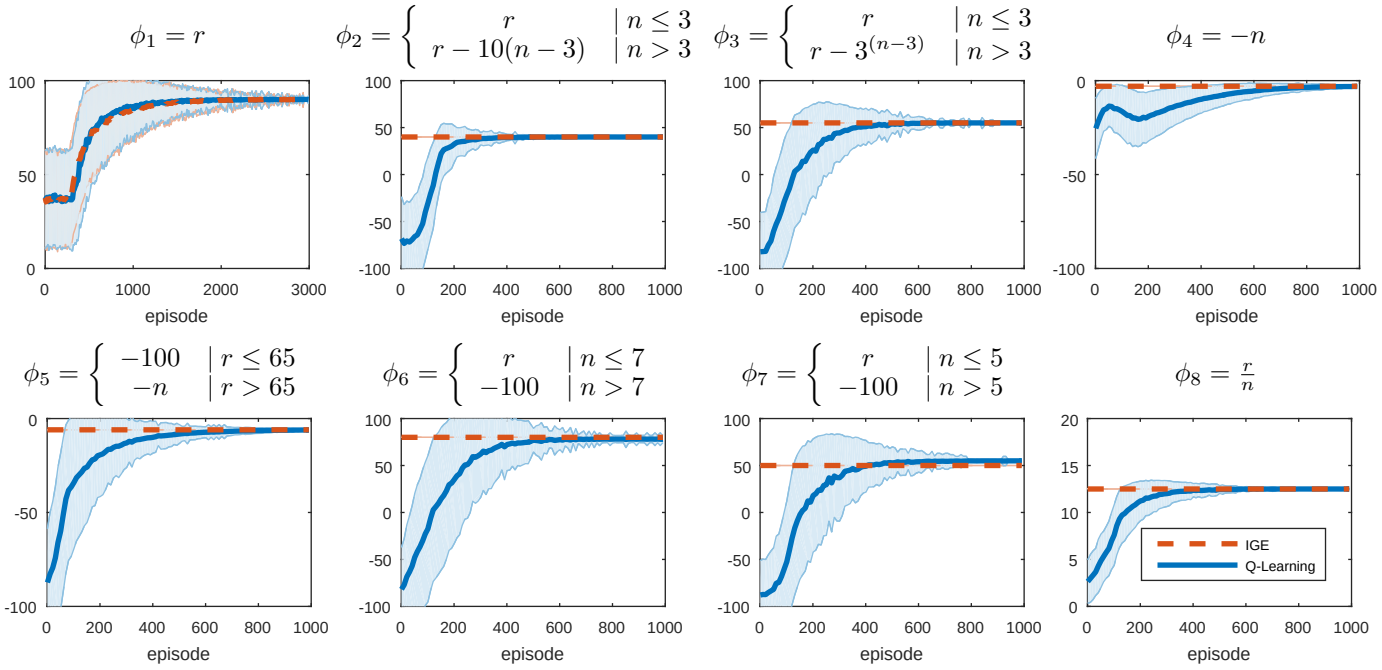

Figure 2: The learning performance of the IGE and Q-learning for all goal-formulations. The IGE can adapt directly to changes in goal-formulations ϕ, whereas Q-learning needs time to adjust its values. Performance is measured by the internal reward $\phi_k(r,n)$ that the agent achieved per episode. The plots show the mean and standard deviation of 1000 learning runs per algorithm. The minimal reward per episode was limited to -100 to make the plots more readable, because some goal formulations can result in a large negative reward during exploration.

Q-learning needed to relearn each task to solve it optimally. Task ϕ_7 is an exception. The IGE is not able to solve the task optimally, because the optimal behavior (go to g_4) is not in its policy set (Fig. 2,c).

4 Discussion

The IGE has two key features. First, it is able to learn and store different policies by using different discount factors γ for each of its modules. This allows it to switch between policies, depending on the current goal. Second, it is possible to gain information about the expected reward trajectory of the policies by using values of modules that result in the same behavior. The information about expected rewards can be used to select the most appropriate policy for a goal. The experimental results demonstrate the ability of the IGE to adapt immediately to a different goal formulation, using its stored policies and information about them. Classical model-free algorithms, such as Q-learning, need several observations to adapt to such changes.

One disadvantage of the IGE is that it can not guarantee to find the optimal policy for each goal formulation. For goal ϕ_7, the IGE fails to find the optimal policy (Fig. 2). Nonetheless, optimality can be guaranteed for a subset of goal formulations. The IGE is optimal to maximize the exponentially discounted reward sum for each of its γ factors because it uses Q-learning, which is guaranteed to converge to the optimal policy. Furthermore, it is optimal in the case of finding the shortest path or the solution with the highest external reward in episodic MDPs. Most interestingly, its convergence to the optimal policy for average reward $\frac{r}{n}$ in episodic, deterministic, goal-only-reward MDPs can be proven. For other goal formulations, the IGE can be viewed as a heuristic that does not guarantee optimality, but it often produces good results with the ability to adapt immediately to a new goal.

The IGE can be categorized as a transfer learning approach [4]. In transfer learning, a set of tasks (MDPs) M is given. The goal is to transfer knowledge from solved source tasks $M_S \subset M$ to new, unfamiliar target tasks $M_T \in M$, to improve learning performance. In our current approach, the different goal formulations Φ can be interpreted as the task set M. All tasks share the same state space, action space, and the same transition probabilities. They differ in their internal reward function ϕ_k. The IGE is able to transfer the policies learned from one task to another because the internal reward function ϕ_k between all task in M is based on a constant feature, the external reward function that all tasks in M share.

In addition to its potential use for artificial agents that need to adapt quickly to goal changes, the IGE also provides a possible model for the discount factor mapping found in the striatum by Tanaka et al. [3]. An existing model by Kurth-Nelson & Redish [2] showed that such a modular architecture can replicate hyperbolic discounting. Such discounting is often observed during intertemporal decision-making by humans. Their model is also based on modules with different discount factors, but in contrast to the IGE, those modules depend on each other and learn values for a single policy.

In contrast, the IGE proposes the idea that the striatum is composed of independent modules. The striatum, which is involved in habit learning [1], could therefore have the ability to learn and store different habitual behavior in parallel, similar to the policy set of the IGE. Furhermore, Tanaka et al. [3] showed that serotonin regulates which regions are active during a task. A low serotonin level resulted in activation of regions that have steep discounting, i.e. regions that prefer nearby rewards. Serotonin could be understood as a context signal that specifies which regions and habits are used. Because serotonin is linked to stress, it is possible that we learn different habits in parallel for different stress levels. Under low stress, we can use habits that strive for large rewards, but that require a longer time to reach; whereas under high stress we can quickly adapt by activating habits that go for short-term rewards. The advantage of such a system is that it does not need to relearn habits if its stress level changes, but to use a repertoire of habits for different stress levels.

Although the IGE is composed of independent modules, it can also replicate hyperbolic discounting as observed in the model of Kurth-Nelson & Redish [2]. This can be achieved by optimizing the average reward $\frac{r}{n}$, but instead of selecting the active module that controls the behavior in the beginning of an episode, it gets reselected after each transition.

References

[1] N. D. Daw, Y. Niv, and P. Dayan. Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nature Neuroscience*, 8(12):1704–1711, 2005.

[2] Z. Kurth-Nelson and A. D. Redish. Temporal-difference reinforcement learning with distributed representations. *PloS One*, 4(10):e7362, 2009.

[3] S. C. Tanaka, N. Schweighofer, S. Asahi, K. Shishida, Y. Okamoto, S. Yamawaki, and K. Doya. Serotonin differentially regulates short-and long-term prediction of rewards in the ventral and dorsal striatum. *PLoS One*, 2(12):e1333, 2007.

[4] M. Taylor and P. Stone. Transfer learning for reinforcement learning domains: A survey. *The Journal of Machine Learning Research*, 10:1633–1685, 2009.