Check for updates

A Novel Model for Arbitration Between Planning and Habitual Control Systems

Farzaneh Sheikhnezhad Fard and Thomas P. Trappenberg*

Faculty of Computer Science, Dalhousie University, Halifax, NS, Canada

It is well-established that human decision making and instrumental control uses multiple systems, some which use habitual action selection and some which require deliberate planning. Deliberate planning systems use predictions of action-outcomes using an internal model of the agent's environment, while habitual action selection systems learn to automate by repeating previously rewarded actions. Habitual control is computationally efficient but are not very flexible in changing environments. Conversely, deliberate planning may be computationally expensive, but flexible in dynamic environments. This paper proposes a general architecture comprising both control paradigms by introducing an arbitrator that controls which subsystem is used at any time. This system is implemented for a target-reaching task with a simulated two-joint robotic arm that comprises a supervised internal model and deep reinforcement learning. Through permutation of target-reaching conditions, we demonstrate that the proposed is capable of rapidly learning kinematics of the system without a priori knowledge, and is robust to (A) changing environmental reward and kinematics, and (B) occluded vision. The arbitrator model is compared to exclusive deliberate planning with the internal model and exclusive habitual control instances of the model. The results show how such a model can harness the benefits of both systems, using fast decisions in reliable circumstances while optimizing performance in changing environments. In addition, the proposed model learns very fast. Finally, the system which includes internal models is able to reach the target under the visual occlusion, while the pure habitual system is unable to operate sufficiently under such conditions.

Keywords: machine learning, reinforcement learning, supervised learning, habitual controller, planning, internal models, decision making

1. INTRODUCTION

Much of the current reinforcement learning (RL) literature implements model-free control. Such a learning agent learns a value function from interacting with the environment, usually updating a proposed value function from a temporal difference between the previous expectation and a new experience (Mnih et al., 2013, 2015). The value function is like a big lookup-table that can quickly supply evaluations for possible actions and hence provide guidance for actions in a fast and somewhat automated way. Such a decision system can be characterized as habitual. While habitual action selection takes time to learn and requires that similar previous situations have been encountered sufficiently, the advantage is that decisions and corresponding actions can be generated in a timely manner.

In contrast, a system that has some internal models of the environment can be used to derive a value function on demand for a specific situation. A prime example is a Markov decision problem where the reward function and transition function of the agent are known so that the Bellman equations can be used to calculate the optimal value function for every state action pair without the need to explore the environment. Of course, this system requires learning of the internal models, which requires previous interactions with the environment. The learning of internal models can be achieved through some form of supervised learning. Once the models have been learned, the model-based system is able to calculate a value function on the fly. This resembles some form of internal deliberation. The advantage of such a system is its flexibility to new situations. However, deliberations take time so that a habitual system is preferable when it comes to situations that benefit from a higher degree of automation.

In this paper, we introduce a learning system that we call the Arbitrated Predictive Actor-Critic (APAC) that combines a habitual reinforcement learning system with a supervised learning system of internal models. Most importantly, we introduce an arbitration system that mediates between their usage. We specifically discuss a situation in which both systems alone can solve an exemplary task so that we can study the consequences of their direct interactions in relation to their exclusive use. We show that this system is responsive to changes in the environment and that it can learn the reward function very fast. Our results demonstrate how the learning paradigm tend to rely on habits after learning the reward function. Our results are in line with evidence of human behavior mentioned above.

2. THEORETICAL PREMISES

There is a lot of behavioral and neurophysicological evidence for different types of control systems in the brain that are usually termed habitual or model-free and goal-directed or model-based (Balleine and Dickinson, 1998; Gläscher et al., 2010; Daw et al., 2011). In particular, one control system associated with the prefrontal cortex (Miller and Cohen, 2001) predicts action-outcomes using an internal model of the agent's environment and hence can be associated with a control system that uses deliberative planning. We will use in this paper the term deliberative planning instead of goal directed model-based control to minimize the possible confusion between the models of the environment from the models of the value function. Another control pathway in the brain is associated with the dorsolateral basal ganglia (Houk and Barto, 1995) learns to repeat previously rewarded actions that resemble a habitual system.

Some research showed that the two different control systems are used in different situations and can be simultaneously active (Poldrack et al., 2001; Lengyel and Dayan, 2008). For example, in the brain, the cortical system represents a generalized mapping of input distributions while hippocampal learning is an instance-based system (Lengyel and Dayan, 2008). Moreover, when the model of the environment is known and there is sufficient time to plan, the best strategy is deliberate planning (Daw et al., 2005),

but when the decision should be taken very fast the habitual controller is used (Blundell et al., 2016). Other work shows that cooperation and competition between different control systems in the brain happens especially when outcomes of each control system disagree, that is, if a deliberated planning task does not align with a habitual skill (Daw et al., 2005, 2011; Daw and O'Doherty, 2013; Lee et al., 2014).

Moreover, feedback to learning systems can differ in different situations and can be provided from different modalities such as vision or auditory input. In machine learning, it is common to distinguish different learning paradigms. One is supervised learning where a teacher gives feedback from the knowledge of a desired behavior. The system can be trained by comparing the actual output of a leaner to the desired output provided by the teacher. The teacher is basically a critic who can quantify an objective function that a leaner needs to optimize. Another slightly more general learning paradigm is reinforcement learning where the environment only provides some indication of value, often only after a series of actions have been chosen by a learning agent. Reinforcement learning is thus more general in that it can be applied to a lot of more typical learning situations of an agent in an environment. The subsystems in our model align in our implementation with a supervised paradigm to learn internal models and a reinforcement learning paradigm to learn habitual control.

Habitual reinforcement learning which is based on TD learning (Sutton, 1985) has been very successful in explaining experimental evidence from the animal learning literature and dopamine-based learning in the brain (Barto, 1995; Schultz et al., 1997). Such models which have originally been formulated with tubular methods based on discrete state action spaces are now commonly combined with neural networks as a function approximator that broadens the range of practical applications to be solved using RL, especially for control problems with continuous states/actions spaces (Waltz and Fu, 1965; Barto et al., 1990). Barto, Sutton and Anderson introduced the Actor-Critic architecture that was implemented by neural networks (Barto et al., 1983). Later, Barto (1995) represented an adaptive critic which has similar behavior to the dopamine neurons projection to the Striatum and frontal cortex. The adaptive critic uses the internal sensory information to learn an effective reinforcement signal.

It has long been hypothesized that the brain builds an internal representation of the world and its body (Miall et al., 1993; Miall and Wolpert, 1996; Wolpert et al., 1998; Kawato et al., 2003), and evidence shows that "forward" and "inverse models" exist in the brain (Miall et al., 1993; Kawato et al., 2003). The internal model is used to perform in the environment and learn a new task. Flanagan and Wing (1997) showed that the internal model can predict the load force and the kinematics of a hand movement that depends on the load. Moreover, when learning how to use a new tool, humans make a transient change in the internal model of the arm as well as making an internal model of the tool (Kluzik et al., 2008). Furthermore, imitation experiments show that a direct mapping develops between observation and the internal model (Iacoboni et al., 1999). Another advantage of having an internal representation

FIGURE 1 | Overview of the Arbitrated Predictive Actor-Critic (APAC) model.
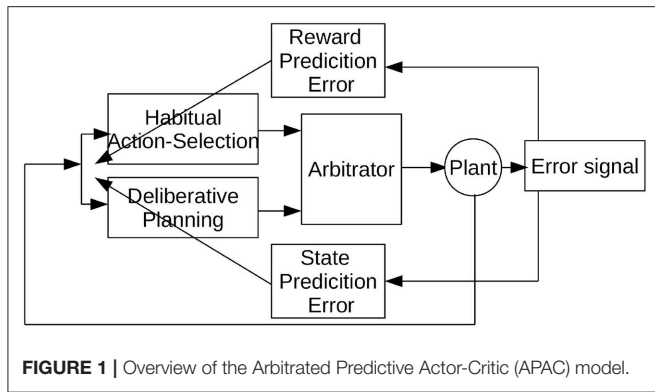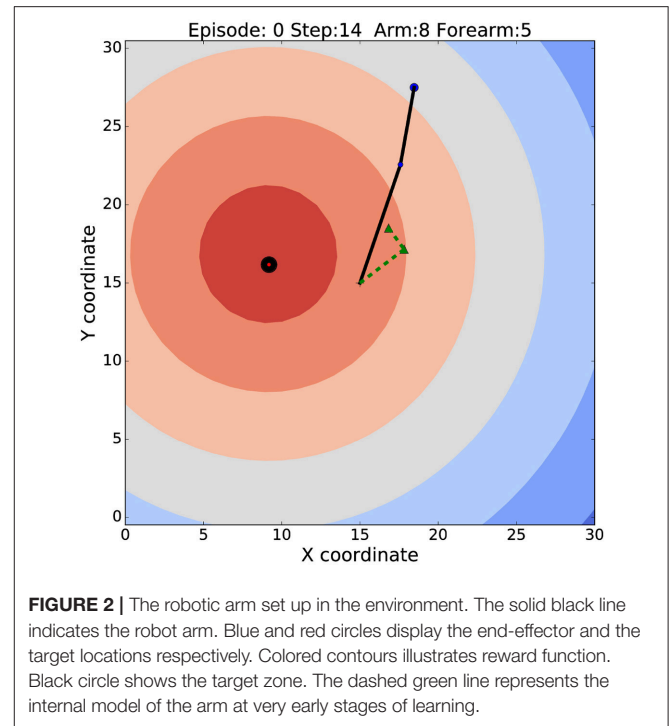
is obtaining a reliable source of information for the agent to perform accurately even if there are no other sources of information (e.g., visual information) available (Wolpert et al., 1998; Kawato et al., 2003).

In this paper, we propose a model to study the cooperation and competition between a habitual and planning-based control components with an arbitrator component. The general architecture of the proposed *Arbitrated Predictive Actor-Critic* (APAC) is shown in **Figure 1**. In this model, each control paradigm implies a specific type of teaching feedback. The deliberative planning controller incorporates internal models that are usually trained with supervised errors so that we consider here an explicit state predictions error. In contrast, the habitual action selection system is a common deep reinforcement learner which learns from reward prediction errors. The new component here is an arbitrator that mediates between these systems that can select the command given to the controlled system, the agent or in the plant in the common language of control theory. Of course, it is possible that both decision systems are trained with a combination of supervised and reinforcement learning, but this is not the crucial point in this paper. The model is designed to study how a combined control system behaves in different environmental situations. In the following section we apply this general model to a specific motor control task in which both systems can be trained on the same feedback signal, but in which the execution would follow a habitual or planning implementation.

3. APAC FOR TARGET REACHING

In this section, we apply the APAC model to the motor control task of target reaching. We choose this task as it is a good example of a minimal control task while being complex enough to highlight the advantages and disadvantages of the two principle control architectures discussed in this paper. Target reaching lives in a continuous state and action space with 6 degrees of freedom when considering a shoulder and elbow yaw, pitch and roll, although we simplify this here even more to a 2-dimensional system with only one angle for the elbow and one the shoulder. Learning the reaching task in this 2D environment is learning a non-linear mapping function that

FIGURE 2 | The robotic arm set up in the environment. The solid black line indicates the robot arm. Blue and red circles display the end-effector and the target locations respectively. Colored contours illustrates reward function. Black circle shows the target zone. The dashed green line represents the internal model of the arm at very early stages of learning.

maps joint angles of the robot arm onto a location of the end-effector in the environment. An example image of our simulated robot arm is shown in **Figure 2** with the black line. The contour plot shows the distance to the target while the dotted green line shows an internal model of the robot arm early in learning.

The refined control architecture of our APAC model for the reaching tasks is shown in **Figure 3**. For this application, the state is defined as the position of the elbow, the end-effector, and the target. The planning component is now implemented as a combination of deep forward and inverse models, while the habitual system is implemented as a deep actor-critic model. An integrator is used to derive the training signals that are used for the feedback. In the following, we specify each subsystem in detail.

3.1. Habit Learning Control System

The habitual controller is implemented as a deep deterministic policy gradient model (DDPG) following the work of Lillicrap et al. (2015). The arm position is given by the vector X with two vector components, the position of the end-effector X^{end} and the position of the elbow X^{elbow}. The arm position together with the target location X_{target} defines the current state $s_t = [X^{end}, X^{elbow}, X_{target}]$ of the agent.

The critic $\mathcal{Q}(s_t, a_t; \theta^{\mathcal{Q}})$ is implemented as a deep neural network, where s_t is the current state at time t, a_t is the action taken at time t, and $\theta^{\mathcal{Q}}$ are the parameters of the critic network. The goal of the critic is to approximate the accumulation of the environmental reward (sometimes called return) that can be expected from a certain state action
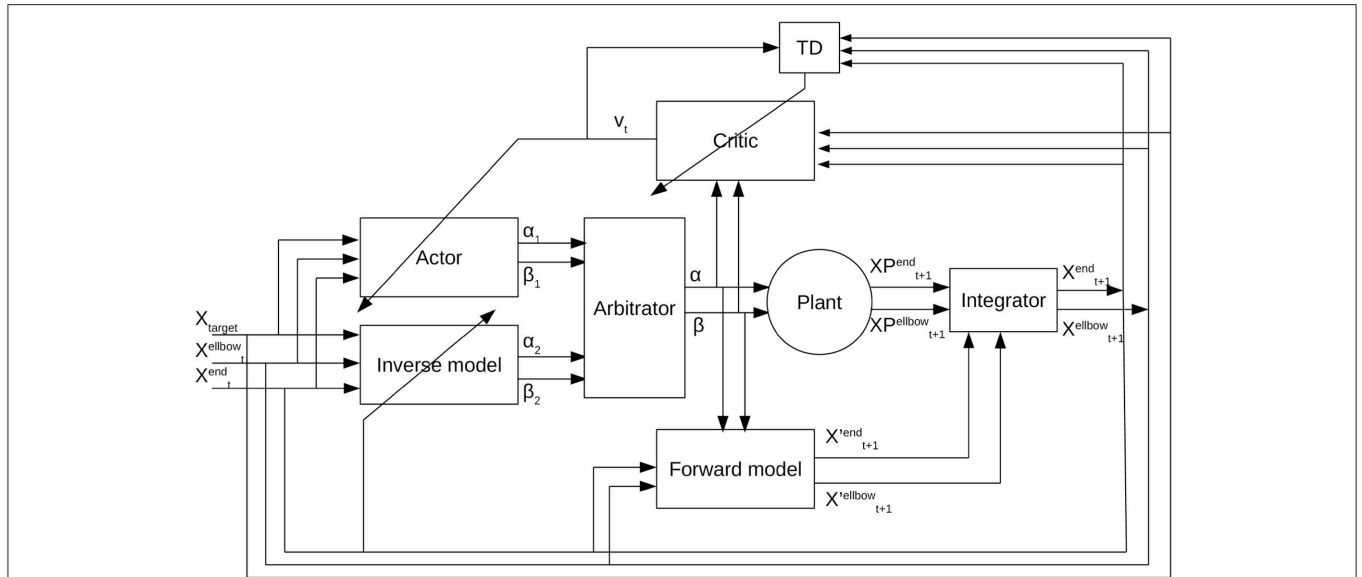
FIGURE 3 | Arbitrated predictive actor critic: The actor receives the current state defined as the current location of the robot arm and the target location, and provides an action that is the change in angles for shoulder and elbow. The inverse model takes the current state and predicts another action. The output of the inverse model and the actor goes to the arbitrator. Then the arbitrator selects one of these actions. The output of the arbitrator along with the current state is the input to the critic from which the critic predicts a reward. The forward model receives the selected action and the current position of the agent and predicts the future state of the agent. The agent takes the selected actions and transfers to the next state. The predicted future state from the forward model is integrated with the estimation of the actual state of the plant after taking the action and obtains a new current state for the system.

combination. The critic is learned through temporal difference (TD) learning (Sutton, 1988; Schultz et al., 1997).

$$\mathcal{Q}(s_t, a_t; \theta^{\mathcal{Q}}) \leftarrow \mathcal{Q}(s_t, a_t; \theta^{\mathcal{Q}}) + l_1 \delta, \tag{1}$$

$$\delta = \underbrace{r_t + \gamma \max_{a'} \mathcal{Q}'(s_{t+1}, a'; \theta^{\mathcal{Q}})}_{\text{estimated reward}} - \underbrace{\mathcal{Q}(s_t, a_t; \theta^{\mathcal{Q}})}_{\text{actual reward}}, \tag{2}$$

where l_1 is the learning rate of the critic network, r_t is the actual immediate reward received from the environment at time t, γ is a discount factor, and $\mathcal{Q}'$ represents the estimation of the value of a state-action pair. As in DDPG, we use the main network for training but we use a *target network* for predicting, which is a less frequently updated copy of the main network to avoid oscillation. More precisely, DDPG actually has two target networks, one for the critic network and one for the actor network. We follow directly the smooth update for the target networks as in DDPG (Lillicrap et al., 2015),

$$\theta' = \theta' \times (1 - \tau) + \theta \times \tau, \tag{3}$$

with change parameter $\tau \ll 1$. The parameters θ' represents the target network parameters, and θ is the parameter of the main network, either the actor or the critic.

The computation in Equation (2) is done in the TD component. To train the critic using the TD rule, the error needs to be back-propagated through the critic. The error between estimated value and the actual value is used to compute the loss function of the critic (Equation 4),

$$L^{\mathcal{Q}} = 1/N \sum (\delta)^2. \tag{4}$$

DDPG takes advantage of the *experience memory replay* which is a memory to store and reuse past experiences. The memory replay is in form of $R(s_t, a_t, r_t, s_{t+1}, T_t)$, where s_t is the current state at time t, a_t is the action taken at time t, s_{t+1} is the next state, r_t is the reward received at time t, and T_t indicates whether the state at time $t + 1$ is a terminal or not. The replay memory is a queue-like buffer with a finite size. The agent will forget older experiences and it will update its parameters based on its recent experiences. At each time step, a random batch of N samples is selected from the experience memory replay, and this batch is used to train both the actor and the critic.

The actor (π) receives the current state (s_t) and predicts future actions to be taken (a_t).

$$\pi(s_t; \theta^\pi) = a_t, \tag{5}$$

where $a_t = [\alpha_1, \beta_1]$ and α_1 and β_1 are motor commands sent to the shoulder and elbow, respectively. The actor is implemented as a deep network where θ^π indicates the parameter of the actor network and is trained using the deterministic policy gradient method (Silver et al., 2014). Note that the main actor network is used for training, however, the target network of the actor is used for the action prediction.

The changes of the weights of the actor corresponded to the changes in expected reward with respect to the actor's parameters,

$$\theta_t^\pi \leftarrow \theta_t^\pi + l_2 \frac{\partial \mathcal{Q}(s_t, a_t; \theta^{\mathcal{Q}})}{\partial \pi(s_t; \theta_t^\pi)} \frac{\partial \pi(s_t; \theta_t^\pi)}{\partial \theta_t^\pi}, \tag{6}$$

where l_2 here is the learning rate of the actor. The plant, which is the simulated arm in our example, takes the action and

transitions to its new position $[Z_{t+1}^{end}, Z_{t+1}^{elbow}]$, which forms the new state s_{t+1} when combined with the target location. Like DDPG we apply noise to the environment using an Ornstein-Uhlenbeck process (Uhlenbeck and Ornstein, 1930) that results in new samples.

3.2. Internal Models for Planning

For the planning controller, we need to learn the transition function of the plant to build the model of the environment. Here we use supervised learning to learn the internal representation of the agent. More specifically, we used a supervised learning controller that uses past experiences to generalize an inverse model of the arm and a forward model of the arm. The training examples used in our implementation are obtained from the same experience replay memory that is used for the habitual controller.

A combination of a forward and an inverse model is used for planning the next actions. The forward model f_F is a neural network that receives the current position of the arm $[X_t^{end}, X_t^{elbow}]$ and the action a_t and predicts the future position of the arm $[X'^{end}_{t+1}, X'^{elbow}_{t+1}]$. We can train the network from the discrepancy between the predicted future position $[Z_{t+1}^{end}, Z_{t+1}^{elbow}]$ and the actual position from visual feedback. For training we use the loss function

$$L^{f_F} = \frac{1}{N} \sum_t ([X'^{end}_{t+1}, X'^{elbow}_{t+1}] - [Z_{t+1}^{end}, Z_{t+1}^{elbow}])^2, \qquad (7)$$

where N is the number of selected samples in a batch of experiences stored in the replay memory. The size of the batch to train the forward model and the inverse model is the same as the one used for the actor and the critic.

An inverse model is another deep network, $f_I(s_t; \theta^{f_I})$. The aim of the inverse model is to provide a proper action to reach the target by minimizing the error between predicted action (a'_t) with the actual action taken (a_t) that transfers the agent from the current position to its next position. This network is then trained on the loss function:

$$L^{f_I} = \frac{1}{N} \sum_t (a_t - a'_t)^2. \qquad (8)$$

The aim of having the forward model is learning to predict future positions of the agent by taking specific actions. Such a model enables the agent to perform the task even with occluded vision. When the inverse model has been trained well, it can be used to produce a suitable action to transfer the agent from its current state to the target location by replacing X_{target} with Z_t^{end}. Hence, the inverse model can be trained with the input $[X_{t-1}^{end}, X_{t-1}^{elbow}, Z_t^{end}]$ and predicting the proper actions on $[X_{t-1}^{end}, X_{t-1}^{elbow}, X_{target}]$. Note that $[X_{t-1}^{end}, X_{t-1}^{elbow}]$ are part of states s_t in the replay memory while Z_t^{end} is taken from s_{t+1} in the replay memory.

Another component of the overall system is "the integrator" module. In general, the integrator could be a Bayes filter such as a Kalman filter which estimates the best estimated position from the available information that combines an internal model prediction with external sensory feedback. Since we use a reliable visual feedback in our case study we simplify this to an integrator that passes the actual location of the plant in case visual information is available. With occluded vision, the prediction of the forward model is used as the estimated actual position of the agent. In our previous work (Fard et al., 2015), we showed how to implement a Kalman Filter with Dynamic Neural Fields (Amari, 1977). The integrator is the explicit critic in this example, which provides the state prediction error for the forward model (see **Figure 1**).

A training session of the system includes an infant phase that uses "motor babbling" (Iverson and Fagan, 2004; von Hofsten, 2004; Demiris and Dearden, 2005; Iverson et al., 2007; Caligiore et al., 2008). During the babbling phase, the plant produces random movements with random actions to produce actual samples to be stored in the experience memory. In the babbling phase, the actual position of the arm after taking an action is considered the target location. Therefore, all samples in this case reach the terminal state and will gain the maximum reward value. The babbling phase is used to provide valid examples to train both control systems.

3.3. Arbitration Between Habitual and Planning Controllers

A novel component of APAC is an arbitrator. The arbitrator receives action predictions from the deliberative planning module (the inverse model), and the habitual action selection module (the actor), and makes the final decision of which action to use. This selected action is transferred to the actuators of the plant to bring the agent into its new position resulting in a new state when combined with the target location. The arbitrator's decision is also fed into the forward model and the critic for training purposes. As in DDPG, noise from an Ornstein-Uhlenbeck process is added to both proposal actions provided by the inverse model and the actor.

In our implementation of the APAC, we consider discrete action steps so that both controllers (habitual and planning) create actions at each step. However, it is known that the habitual controller is faster than deliberative planning. Therefore, to imply the time constraint we set the arbitrator to give priority to the habitual controller. Moreover, the arbitrator is set to always take the action that is provided by the habitual system for the first two steps of every episode. However, from the third step on, the actor's prediction is taken if the habitual controller is reliable, meaning that the reward prediction error for the last experience is smaller than a threshold. We use $abs(\delta) < 1$ in the following experiments. Otherwise, the action from the inverse model is selected.

The implementation of the arbitrator here is somewhat a minimal model suitable for our experimental setting and to highlight the consequences of such arbitration. Of course, it is possible to implement a more dynamic realization of such an arbitrator. For example, the threshold could itself be modulated according to the tasks and in this way produces a more rich speed-accuracy trade-off (Satel et al., 2005). Indeed, such modeling will open the possibility to discuss behavioral consequences with different system settings a ultimately compare them to variations in populations or psychiatric disorders such

as eduction and eating disorders (Huys et al., 2016). However, the simple implementation discussed in this paper captures the minimal assumptions as outlined above and is sufficient for the following simulations.

3.4. Experimental Conditions and Environment

To test the APAC model on a simulated robot arm with a target reaching task (**Figure 2**), we simulated a two-joint robotic arm whose range of motion at each joint was constrained to 180 degrees. The arm's motion was limited to a 2D plane of width 30 and height 30, upon which the arm's "shoulder" was fixed in the center (15,15). The initial length segment from the shoulder to the elbow was set to $l_1 = 5$, and the initial length of the lower segment ("hand" to "elbow") is $l_2 = 8$.

All experiments described herein had an episodic trial structure. At the beginning, the arm's position was set to zero-degree angle at the shoulder and 180-degree angle at the elbow. Time was discretized in the simulations, and the learning agent was given only 30 action-steps per trial to achieve the designated goal. We define a "target zone" as a circle centered at the target location with a radius $r_{target} = 0.5$. The target is defined as "reached" once the robot arm is inside the target zone. If the goal was not reached within 30 time-steps, the trial was aborted and a new trial was started.

Importantly, the reward function is defined as the negative Euclidean distance between the end-effector and the center of the target area. This is for this example tasks the same information as is given to the supervised learner. This was deliberatively done so that the different systems are compared on the same feedback situation. It is possible to learn this task from much simpler feedback such as some reward if the target area is reached vs. no reward otherwise, although this would then also need more time to train the habitual system. The point of our study here is rather the direct comparison of decision components based on a value lookup vs. learning internal models.

Within this environment we define several conditions that defined the variety of the different target-reaching tasks studied here. These conditions include the target position (**static target** vs. **changing target** at each episode), kinematics (arm dimensions as **static kinematics** vs. **changing kinematics**), and vision (**occluded vision** vs. **perfect vision**). We tested all combinations of these factors.

Each experiment consisted of 1,000 episodes of maximal 30 action steps each. In the static target condition, the target is initialized randomly and stays unchanged for all 1,000 episodes. For the changing target condition, the target is located at a random location at the beginning of every episode. As discussed above, each episode was terminated when either (A) the target was reached, or (B) 30 time steps had elapsed. Targets were only placed within a reachable distance for the arm. The arm dimensions were kept fixed in the case of static kinematics; however, the length of both the upper and lower arm segments were increased by 0.001 at each time step for the changing kinematics condition. These changes were only started after the 100th episode of target reaching to provide some time for basic
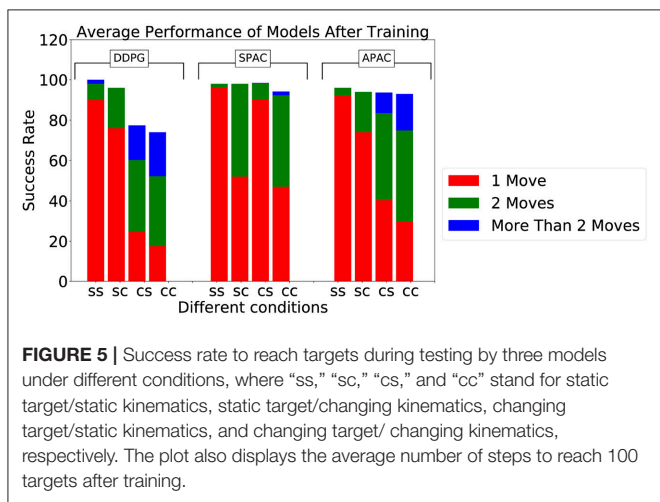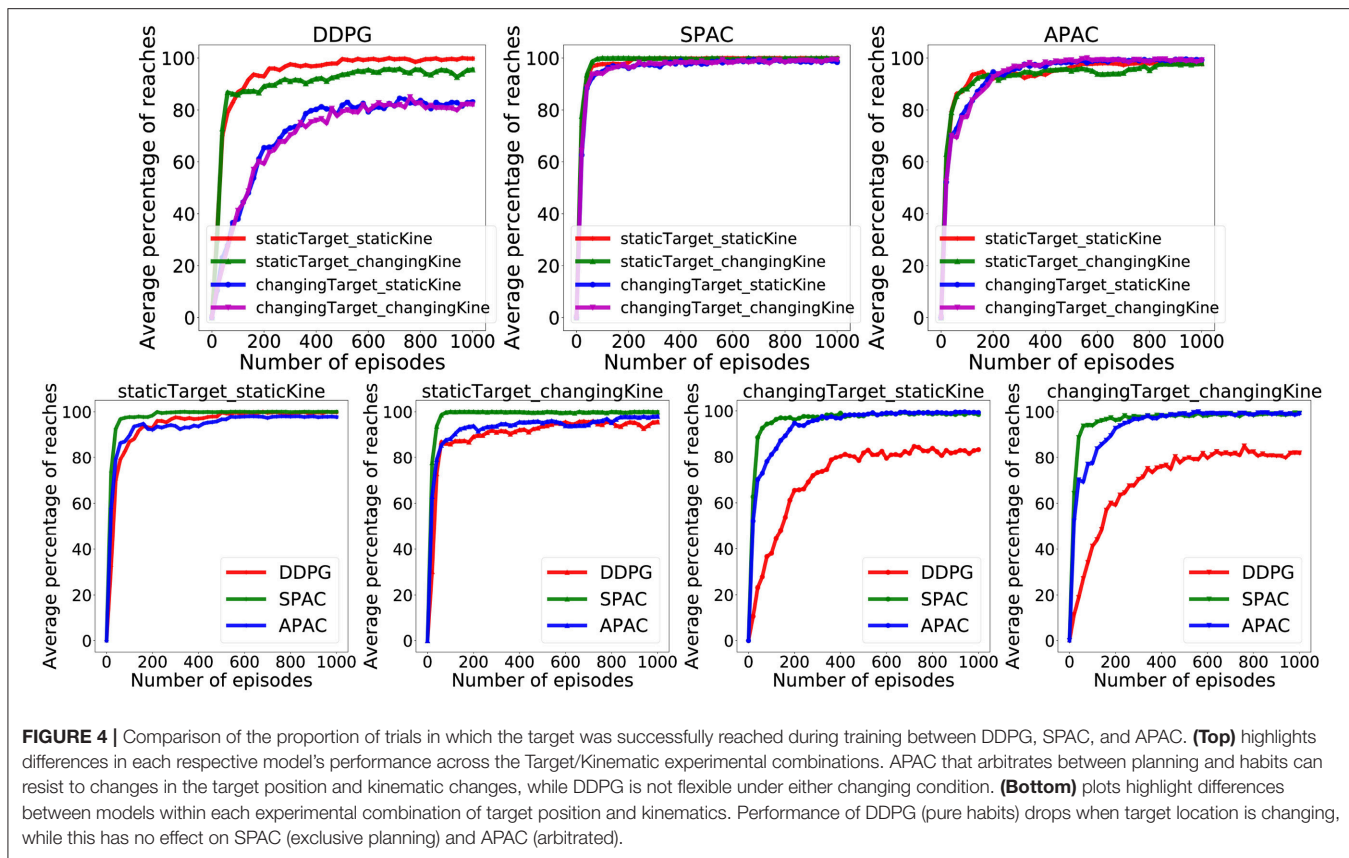
training. As already mentioned, an environmental noise was included in all experiments. In the occluded vision condition, the location of the arm and the target was unavailable for the agent during the movement. This task is also known as memory guided target reaching (Westwood et al., 2003; Heath et al., 2004). We repeated all static/changing kinematics and static/changing target conditions with our proposed models for the reaching-target task in the occluded vision condition. To examined the generalization of the models under each condition, we trained each model when targets are located only in a specific area that represents 2/3 of whole reaching area, and tested with targets located in the other part of the environment, which is the rest 1/3 of the reaching area.

4. RESULTS

We considered three versions of APAC that represent (a) exclusive habits, (b) exclusive deliberate planning, and (c) arbitration between habit and planning. Exclusive habit is when the arbitrator is set to always pick the action from the habitual system. In this case, the APAC behaves exactly like DDPG. If the arbitrator always selects the action from the inverse model for each step, then the APAC becomes an exclusive deliberate planning controller which we call supervised predictive actor-critic (SPAC) (Fard et al., 2017). The third model is when the APAC is able to arbitrate between the actions provided by the inverse model and the actor.

For each condition, we trained 50 independent instances of each model for a total of 1,000 episodes. At the end of the 1000th episode, all network parameters were frozen and no more training was applied. Subsequently, each of the independent model instances performed 100 target reaching episodes under the respective training conditions. In the case of the occluded vision, sensory input (i.e., visual target position) was initially presented at time step 0, and subsequently rendered unavailable. Since DDPG has no internal model and requires visual input throughout the task, we only compare APAC with SPAC in the experiments with occluded vision. All experiments were implemented and tested in Python (3.5) using the TensorFlow (1.3) package (Abadi et al., 2016) on an NVIDIA GeForce GTX 960 graphical processing unit.

Figure 4 illustrate the percentage of trials that reach the target within 30 action steps during training under different conditions for 1,000 episodes each. The pure deliberate planning model SPAC reaches almost near perfect performance very fast after 100 episodes. Furthermore, SPAC's performance is very robust under different conditions and neither changes in reward function nor in kinematics effects the performance of SPAC very much. DDPG learns to reach almost 100% of the targets only under static target/static kinematics condition. Performance of DDPG drops slightly under changing kinematics compared to static kinematics; however, its performance drops dramatically (about 20%) under changing target conditions. This is of course expected as habits become invalid solutions under changing environments. Our point here is that APAC can reach almost all of the targets both under static and changing targets as good as SPAC, although it tends to use more habits than planning after a few trials of

FIGURE 4 | Comparison of the proportion of trials in which the target was successfully reached during training between DDPG, SPAC, and APAC. **(Top)** highlights differences in each respective model's performance across the Target/Kinematic experimental combinations. APAC that arbitrates between planning and habits can resist to changes in the target position and kinematic changes, while DDPG is not flexible under either changing condition. **(Bottom)** plots highlight differences between models within each experimental combination of target position and kinematics. Performance of DDPG (pure habits) drops when target location is changing, while this has no effect on SPAC (exclusive planning) and APAC (arbitrated).

FIGURE 5 | Success rate to reach targets during testing by three models under different conditions, where "ss," "sc," "cs," and "cc" stand for static target/static kinematics, static target/changing kinematics, changing target/static kinematics, and changing target/ changing kinematics, respectively. The plot also displays the average number of steps to reach 100 targets after training.

learning(see **Figure 6**). The speed of learning in APAC is also very high and comparable to SPAC. In this sense it combines the benefits of DDPG and SPAC.

The above curves give an example of behavior of the models during one learning trials. To study how these results generalize we tested the performance of all three models after learning over 50 different learning trials with random initial conditions for the networks. For the static target location we tested on the target location, that was randomly chosen

for each learning trial. However, with the changing target location we decided to cover the possible target locations more systematically and set target points on a regular grid in angle space. **Figure 5** displays average success rate over the 50 learning trials to reach these targets. All three models under the static target/static kinematics condition reach 100% of the targets. DDPG and APAC have slightly less success under static target/changing kinematics, while SPAC stays flexible under this condition. The major difference between DDPG and APAC become clear under changing target conditions, where DDPG's performance drops dramatically, while APAC obtains very good performance. SPAC is still very flexible to reach targets under changing target conditions. Overall it is remarkable how close APAC stays to the overall performance of SPAC in a situation where deliberative planning is the better choice.

The overall success rate does show the entire range of the solutions. We thus included the individual performances in terms of the average number of steps to reach the target. As can be seen, APAC needs to take sometimes more corrective steps to reach the target while an exclusive planning system can optimize the number of steps. This is interesting as this allows for different strategies in solving the task, that of relying somewhat on habitual control when the cost of the movement initiation might be small vs. more deliberate planning when the number of action steps might matter. This can explain a form of speed-accuracy trade-off.
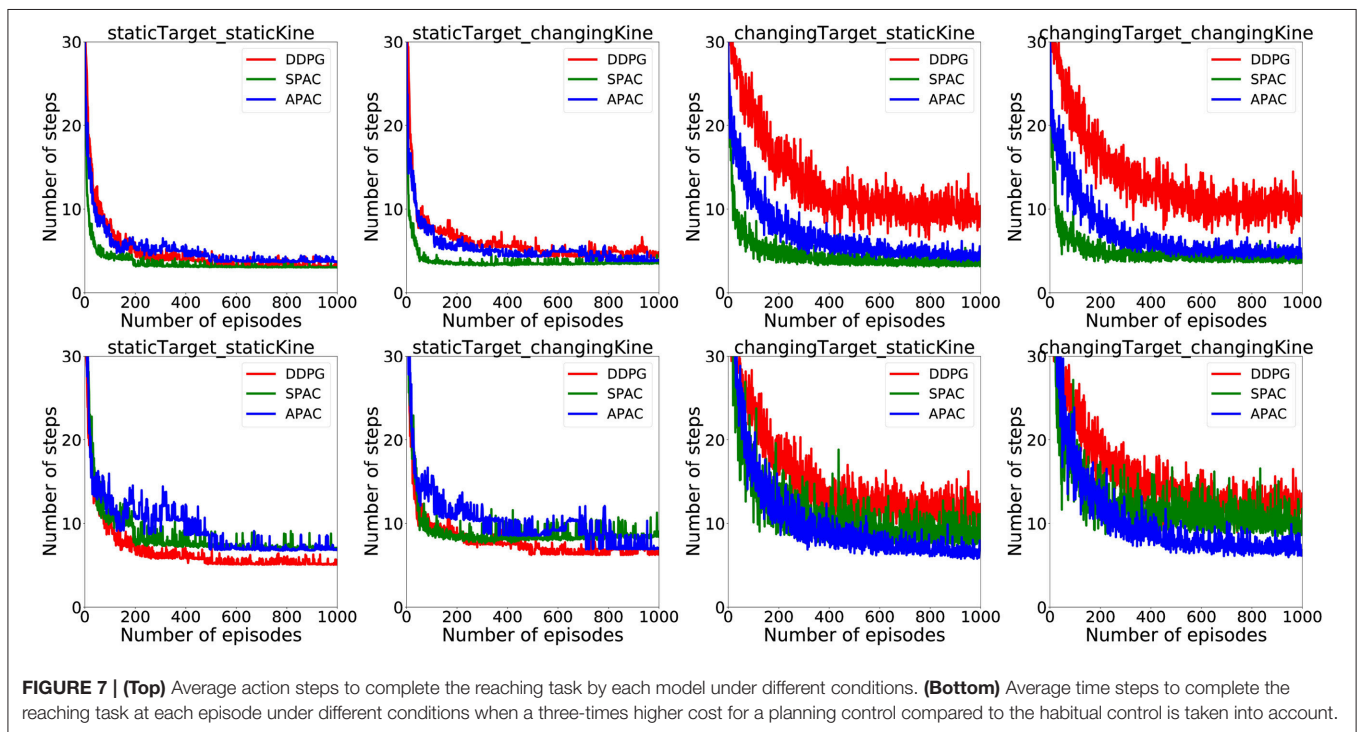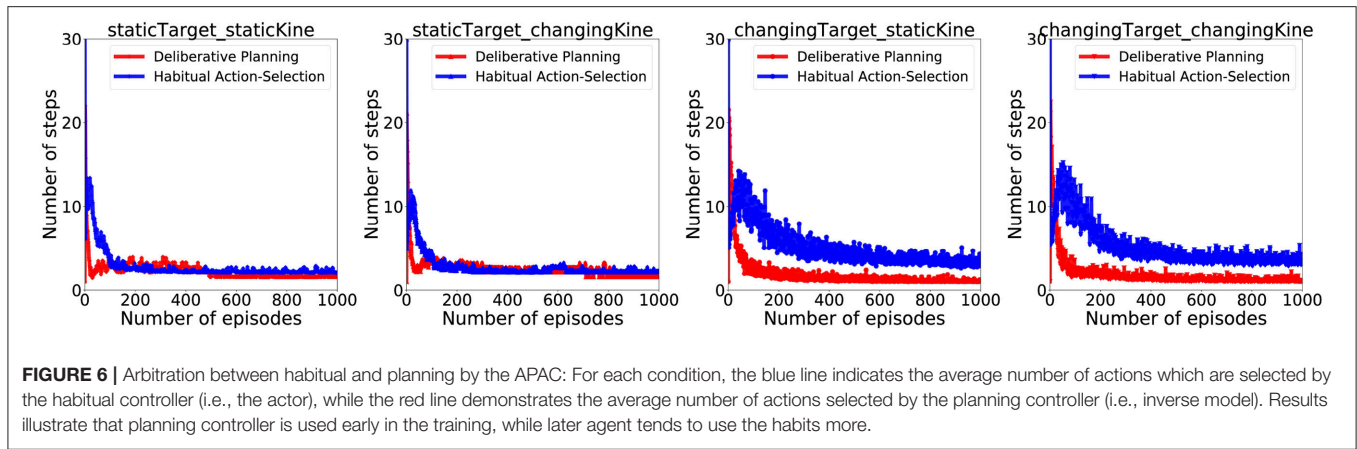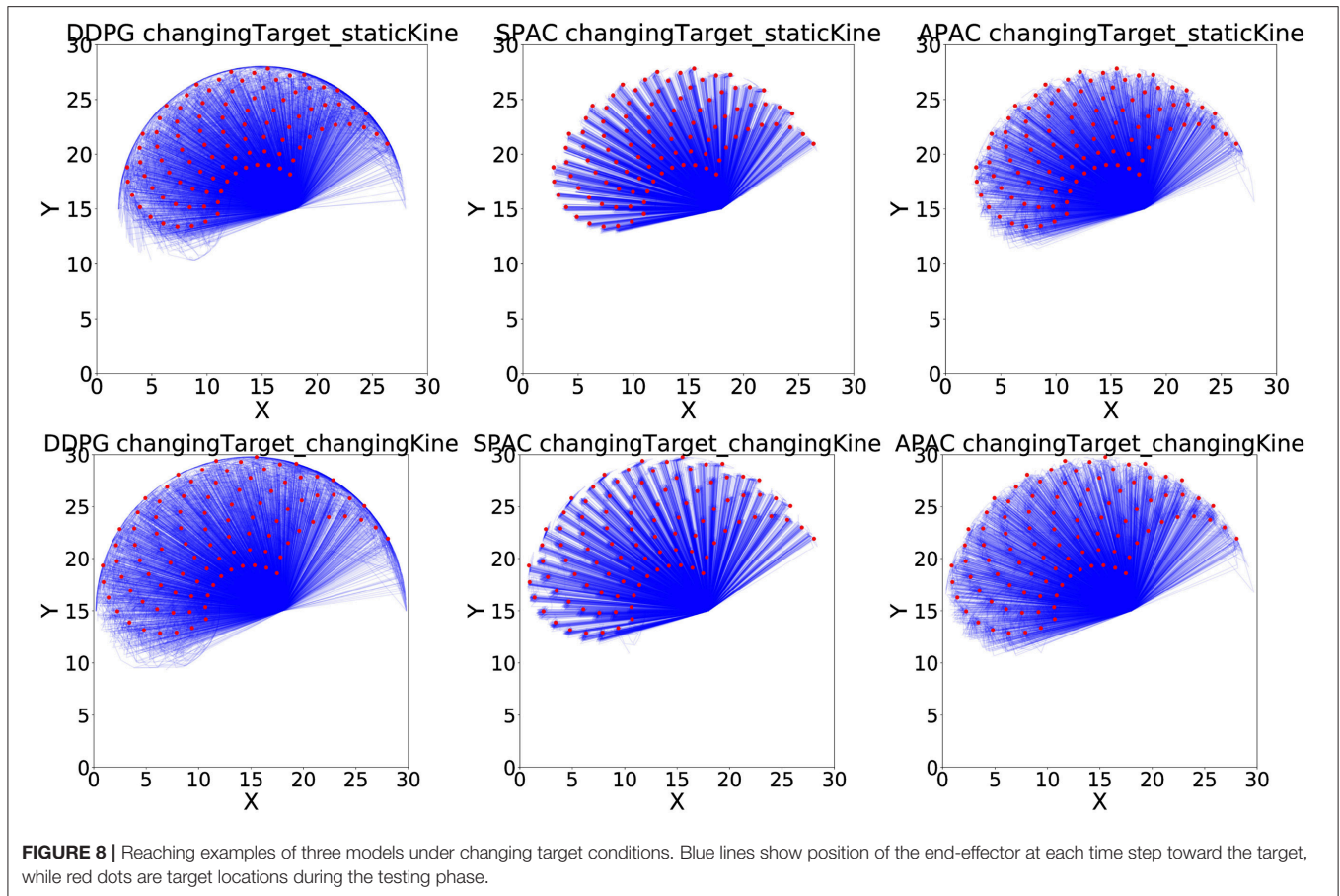
FIGURE 6 | Arbitration between habitual and planning by the APAC: For each condition, the blue line indicates the average number of actions which are selected by the habitual controller (i.e., the actor), while the red line demonstrates the average number of actions selected by the planning controller (i.e., inverse model). Results illustrate that planning controller is used early in the training, while later agent tends to use the habits more.

FIGURE 7 | (Top) Average action steps to complete the reaching task by each model under different conditions. **(Bottom)** Average time steps to complete the reaching task at each episode under different conditions when a three-times higher cost for a planning control compared to the habitual control is taken into account.

Figure 6 illustrates how APAC gradually shifts from a planning to a habitual control approach with increasing experience. After around 300 episode, more than 80% of APAC actions were taken from the habitual controller. Of course, SPAC uses planning control throughout the entirety of the task, so no commensurate figure was generated for it.

Since a habitual system should be faster than deliberate planning, this figure also illustrates that APAC would be less time-consuming than the SPAC at the same task and under the same condition. To visualize average time consumption by each model under different conditions, we assumed that each action selected by the deliberative planning takes three times longer than an action selected by the habitual controller. The number here is arbitrary and only chosen visualize the general effect. The top row of plots in **Figure 7** show the average number of action steps

that each of the three models need to complete the task at each episode under different conditions. These plots demonstrate that the number of steps are almost the same under static target/static kinematics conditions. Under changing target conditions DDPG needs more steps to complete the task than APAC and SPAC. However, when including a higher cost for deliberative planning in the plots shown in the bottom row of the **Figure 7**, the picture for the average number of time that is needed to complete the task changes. In this case, DDPG needs shorter time under static target conditions. However, under changing target conditions, APAC completes the task of reaching targets faster. DDPG takes longer to finish the task as it needs more corrective actions.

Figure 8 shows all 50 runs to reach 100 targets of a reaching test under changing target conditions, with (bottom row) and without (top row) changing kinematics, for all three models.

FIGURE 8 | Reaching examples of three models under changing target conditions. Blue lines show position of the end-effector at each time step toward the target, while red dots are target locations during the testing phase.

The plots illustrate that DDPG has some difficulty reaching the locations at the edges of the possible target area due to non-linearity of the mapping between angles and locations. SPAC can learn the mapping function much better, and the quality of APAC is similar to SPAC. Interestingly, although APAC tends to use more habits than deliberate planning, this model can still reach many more targets than DDPG, almost as good as SPAC.

We also tested a form of generalization of each model where a whole area of the target zone was not seen during training. This is a case of extrapolation compared to the interpolation trials in the previous generalization experiments. More specifically, we trained each model to reach the target located at a specific region in the environment and we tested each model to reach targets that are located on the unseen area of the environment (see the left most plot in the **Figure 9**). The same distribution of target locations has been used here and only those that are located in the blue area are set as targets for this experiments. Therefore, there are about 39 targets under static kinematics conditions and 42 targets under changing kinematics conditions (because of changing kinematics more targets will locate in the testing area). The results show that under static target training, neither model can reach even half of the targets. Their performance is worse under static target and changing kinematics. However, under changing target conditions, all models have obtained a good generalization but they need to take more than one

step to reach any target. SPAC has again the best performance among other models under all conditions, while DDPG has the worst performance compared to other two models. These results indicate that learning with a static target hinders generalization as the learner overfits this specific target location.

Finally, we want to show results with occluded vision. Since the habitual controller (DDPG) requires sensory input at all times, only the SPAC and APAC models were compared under this condition. In these experiments, the arm moves toward the target when the target location is only visible at the first step. When the forward model indicates that the agent has reached the target it stops and the actual distance from the agent (here the arm) is measured. Results of these occluded vision test are summarized in **Figure 10** under changing target conditions, since the performance of both models under static target conditions are near perfect.

The target zone is marked by the red line in each plot, while the average distance from the agent to the target location is drawn as a black line. The blue area shows the range of the distances that the arm has experienced during the occluded trials when the forward model thinks that it has reached the target. SPAC shows better performance under occluded vision compared to APAC under all conditions. The average actual distance of the end-effector to the target location under changing target/ static kinematic with SPAC is only about a
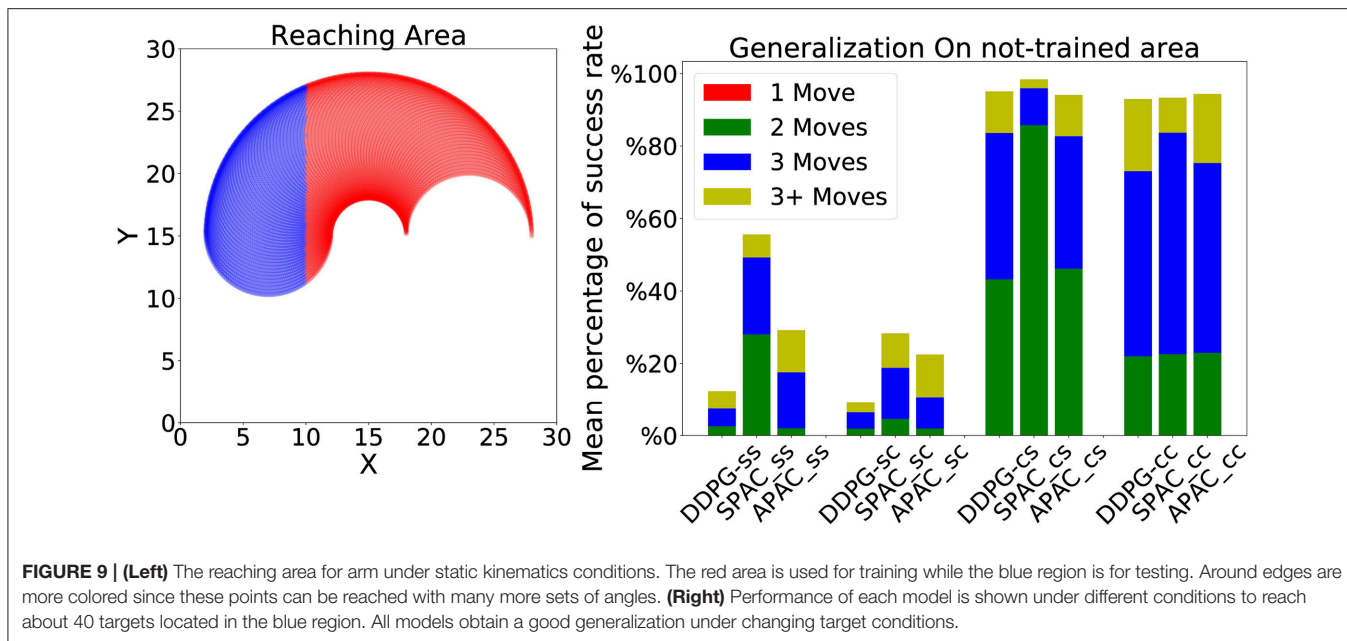
FIGURE 9 | (Left) The reaching area for arm under static kinematics conditions. The red area is used for training while the blue region is for testing. Around edges are more colored since these points can be reached with many more sets of angles. **(Right)** Performance of each model is shown under different conditions to reach about 40 targets located in the blue region. All models obtain a good generalization under changing target conditions.
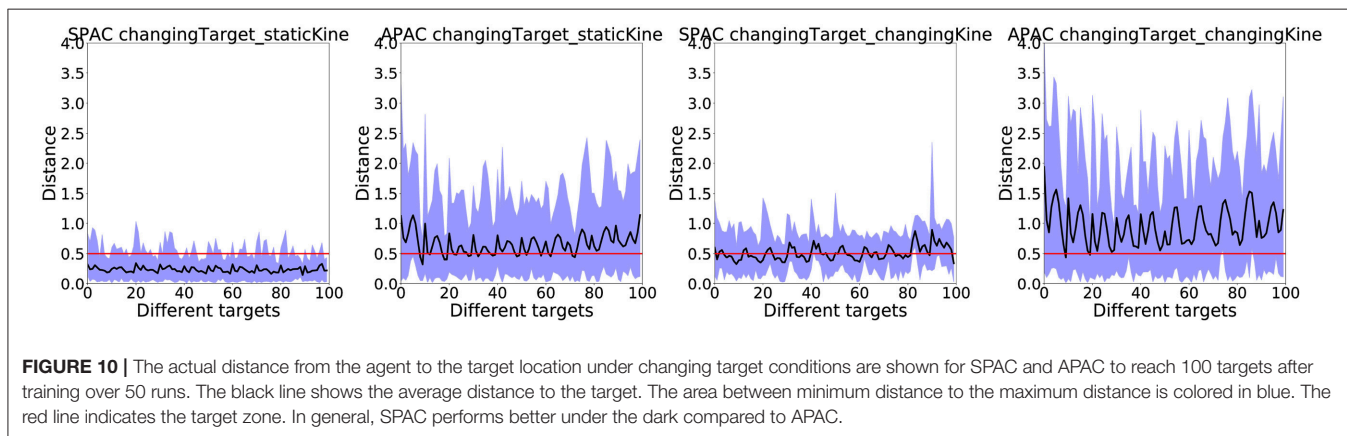

FIGURE 10 | The actual distance from the agent to the target location under changing target conditions are shown for SPAC and APAC to reach 100 targets after training over 50 runs. The black line shows the average distance to the target. The area between minimum distance to the maximum distance is colored in blue. The red line indicates the target zone. In general, SPAC performs better under the dark compared to APAC.

distance of 0.4, which is less than the target zone radius. However, the APAC model under the same conditions stays about a distance of 1 away from the target. Under changing target/changing kinematics, this average actual distance from end-effector to the target is around 0.7 for SPAC and about 1.5 for APAC. It thus seems that any form of habit should be suppressed in such situations, which could be achieved by a more advanced arbitrator.

5. CONCLUSION

This paper is about the study of a hybrid system with deliberate planning system and habitual control. Habitual control will, of course, be very good after long training in static environments. It was hence important to study the model in changing environments. We investigated the behavior of our proposed model (APAC) under changing target conditions (to manipulate the environmental reward function), changing kinematics of

the agent (to manipulate the learned transitional model), and with and without vision. We also tested the model under various generalization conditions to see how good they can interpolate and extrapolate. The main results are classified as below:

Adaptive to changes: Results show significant improvement in performance when planning is available compared to the pure habitual system under the conditions when of changing environments that includes changing reward conditions and changing kinematics of the agent. In comparison, SPAC and APAC are flexible under these changes. These experiment shows that having an internal model is a key to robustness on changing environment.

Moving from planning to habits: By considering that planning is costly, having another control system that is able to provide less costly solutions can be useful. In this paper, there is no inherent time constraint or computation time difference between the habitual and planning controller. However, if we

take this time constraint into account, the APAC is a better model than pure planning (SPAC). Indeed, the overall number of actions taken based on a planned action takes less time than the arbitrated model than the number of planned actions taken by the non-arbitrated model when considering some cost of planning.

Reaching under occluded vision: Since DDPG has no internal model, it can not use any sort of planning to move under occlusion. However, systems like SPAC and APAC build an internal model of the environment that enables them to anticipate and plan a target even when there is no visual information available. Results show that SPAC can perform better in the dark. This is a good example to show that a more sophisticated arbitrator could take different circumstances into account. For occluded vision, such an arbitrator should suppress habitual actions.

Our application example focused on the implementation of the habitual system as an actor-critic for reinforcement learning and a planning system with a forward and inverse model using supervised learning. There have been previous examples of combining both, some form of supervised learning with RL systems and the use of the internal model. In particular, Dyna-Q (Sutton, 1991) and supervised actor-critic (Rosenstein and Barto, 2002; Barto and Rosenstein, 2004) are examples of models that bring the capacity of planning into a model-free reinforcement learning space. For example, the supervised actor-critic (Rosenstein and Barto, 2002; Barto and Rosenstein, 2004) is able to tune the actor manually and very fast when it is needed. This solution is beneficial when dynamics of the system changes dramatically or a new policy is needed to be learned in a very short time. These authors used a gain scheduler that weights the control signal provided by the actor from the reinforcement learner and the supervised actor. In contrast to supervised actor-critic, our proposed model autonomously learns the internal model and arbitrates between the two controllers automatically and not manually. The intention of our model is to study the interaction of habitual and planning systems in form of an arbitrator and ultimately to understanding human behavior.

Sutton proposed Dyna-Q that is an integrated model for learning and planning (Sutton, 1991). With respect to our model, Dyna-Q is also a blend of model-free and model-based reinforcement learning algorithms. Dyna-Q can build a transition function and the reward function by hallucinating random samples. Therefore, although it uses a model-free paradigm at the beginning it becomes a model-based solution by learning the model of the world using the hallucination. In our model, the internal model is used to predict the future state of the agent unlike Dyna-Q that uses the model to train the critic and anticipate the future reward. Moreover, Dyna-Q starts from model-free controller and becomes a model-based controller. Hence, while Dyna-Q has focused on the utilization of internal models to learn a reinforcement controller, our model and study here is concerned with the arbitration of two control systems. However, since APAC tend to select actions from the planning controller that it learns very fast, it provides more accurate

samples in the experience replay memory. Therefore, similar to the Dyna-Q, the habitual system takes advantage of learning from more valuable samples that lead the habitual controller to a better performance compared to the time that it is trained stand alone (in pure habitual paradigm).

Another interesting approach by Uchibe and Doya (2005) explores a collection of different controllers. Their work takes also different times constrains into account. In contrast to our model, the controllers are combined probabilistically in a more collaborative way while our approach focuses more specifically on understanding the competitive decision making of a deliberative vs. habitual systems. The experiments are also somewhat different. Even though the barriers in their experiment are static, it seems that their habitual controller can not learn this task. In our experiment we made sure that the experimental task is learnable by both controllers in the static case. In addition we study the performance with changing kinematics and changing targets.

Not only have human decision-making studies supported the notion that both habitual and planning controls are used during decision-making (Daw et al., 2011; O'Doherty et al., 2015), there is evidence that arbitration may be a dynamic process involving specific brain regions Lee et al. (2014). Our APAC model results suggest that such an arbitration strategy, wherein the planning paradigm is used until the habitual system's predictions become reliable can result in performance that is non-inferior to exclusive planning control in most cases. Thus, our APAC model supports (A) the importance and value of implementing predominantly planning control early in behavioral learning and (B) the diminishing importance of planning control with greater experience in a relatively static environment.

DATA AVAILABILITY

All datasets generated for this study are included in the manuscript and/or the **Supplementary Files**.

AUTHOR CONTRIBUTIONS

All authors listed have made a substantial, direct and intellectual contribution to the work, and approved it for publication.

ACKNOWLEDGMENTS

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fnbot.2019.00052/full#supplementary-material

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2016). Tensorflow: large-scale machine learning on heterogeneous distributed systems. *arXiv [Preprint].* arXiv:1603.04467.

Amari, S.-I. (1977). Dynamics of pattern formation in lateral-inhibition type neural fields. *Biol. Cybern.* 27, 77–87. doi: 10.1007/BF00337259

Balleine, B. W., and Dickinson, A. (1998). Goal-directed instrumental action: contingency and incentive learning and their cortical substrates. *Neuropharmacology* 37, 407–419. doi: 10.1016/S0028-3908(98)00033-1

Barto, A. G. (1995). *Adaptive Critics and the Basal Ganglia.* Computer Science Department Faculty Publication Series. 7. Amherst, MA: University of Massachusetts. Retrieved from: https://scholarworks.umass.edu/cs_faculty_pubs/7

Barto, A. G., and Rosenstein, M. T. (2004). "J. 4 supervised actor-critic reinforcement learning," in *Handbook of Learning and Approximate Dynamic Programming,* Vol. 2, eds S. Jennie, A. G. Barto, and B. Warren (Powell, OH), 359.

Barto, A. G., Sutton, R. S., and Anderson, C. W. (1983). Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Trans. Syst. Man. Cybern.* 5:834–846. doi: 10.1109/TSMC.1983.6313077

Barto, A. G., Sutton, R. S., and Watkins, C. J. (1990). "Sequential decision problems and neural network," in *Advances in Neural Information Processing Systems* (Denver, CO), 686–693.

Blundell, C., Uria, B., Pritzel, A., Li, Y., Ruderman, A., Leibo, J. Z., et al. (2016). Model-free episodic control. *arXiv [Preprint].* arXiv:1606.04460.

Caligiore, D., Ferrauto, T., Parisi, D., Accornero, N., Capozza, M., and Baldassarre, G. (2008). "Using motor babbling and hebb rules for modeling the development of reaching with obstacles and grasping," in *International Conference on Cognitive Systems,* E1–E8.

Daw, N. D., Gershman, S. J., Seymour, B., Dayan, P., and Dolan, R. J. (2011). Model-based influences on humans' choices and striatal prediction errors. *Neuron* 69, 1204–1215. doi: 10.1016/j.neuron.2011.02.027

Daw, N. D., Niv, Y., and Dayan, P. (2005). Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nat. Neurosci.* 8, 1704–1711. doi: 10.1038/nn1560

Daw, N. D., and O'Doherty, J. P. (2013). "Multiple systems for value learning," in *Neuroeconomics: Decision Making and the Brain, 2nd Edn,* eds W. P. Glimcher and E. Fehr (San Diego, CA: Elsevier Inc.), 393–410. doi: 10.1016/B978-0-12-416008-8.00021-8

Demiris, Y., and Dearden, A. (eds.) (2005). *From Motor Babbling to Hierarchical Learning by Imitation: A Robot Developmental Pathway.* Lund: Lund University Cognitive Studies.

Fard, F. S., Hollensen, P., Heinke, D., and Trappenberg, T. P. (2015). Modeling human target reaching with an adaptive observer implemented with dynamic neural fields. *Neural Netw.* 72, 13–30. doi: 10.1016/j.neunet.2015.10.003

Fard, F. S., Nunes, A., and Trappenberg, T. (2017). "An actor critic with an internal model," in *Inaugural Conference on Cognitive Computational Neuroscience (CCN)* (New York, NY).

Flanagan, J. R., and Wing, A. M. (1997). The role of internal models in motion planning and control: evidence from grip force adjustments during movements of hand-held loads. *J. Neurosci.* 17, 1519–1528. doi: 10.1523/JNEUROSCI.17-04-01519.1997

Gläscher, J., Daw, N., Dayan, P., and O'Doherty, J. P. (2010). States versus rewards: dissociable neural prediction error signals underlying model-based and model-free reinforcement learning. *Neuron* 66, 585–595. doi: 10.1016/j.neuron.2010.04.016

Heath, M., Westwood, D. A., and Binsted, G. (2004). The control of memory-guided reaching movements in peripersonal space. *Motor Control* 8, 76–106. doi: 10.1123/mcj.8.1.76

Houk, J. C., Adams, J. L., and Barto, A. G. (1995). "A model of how the basal ganglia generates and uses neural signals that predict reinforcement," in *Models of Information Processing in the Basal Ganglia,* 249–274.

Huys, Q. J., Maia, T. V., and Frank, M. J. (2016). Computational psychiatry as a bridge from neuroscience to clinical applications. *Nat. Neurosci.* 19, 404–413. doi: 10.1038/nn.4238

Iacoboni, M., Woods, R. P., Brass, M., Bekkering, H., Mazziotta, J. C., and Rizzolatti, G. (1999). Cortical mechanisms of human imitation. *Science* 286, 2526–2528. doi: 10.1126/science.286.5449.2526

Iverson, J. M., and Fagan, M. K. (2004). Infant vocal–motor coordination: precursor to the gesture–speech system? *Child Dev.* 75, 1053–1066. doi: 10.1111/j.1467-8624.2004.00725.x

Iverson, J. M., Hall, A. J., Nickel, L., and Wozniak, R. H. (2007). The relationship between reduplicated babble onset and laterality biases in infant rhythmic arm movements. *Brain Lang.* 101, 198–207. doi: 10.1016/j.bandl.2006.11.004

Kawato, M., Kuroda, T., Imamizu, H., Nakano, E., Miyauchi, S., and Yoshioka, T. (2003). Internal forward models in the cerebellum: fmri study on grip force and load force coupling. *Prog. Brain Res.* 142, 171–188. doi: 10.1016/S0079-6123(03)42013-X

Kluzik, J., Diedrichsen, J., Shadmehr, R., and Bastian, A. J. (2008). Reach adaptation: what determines whether we learn an internal model of the tool or adapt the model of our arm? *J. Neurophysiol.* 100, 1455–1464. doi: 10.1152/jn.90334.2008

Lee, S. W., Shimojo, S., and O'Doherty, J. P. (2014). Neural computations underlying arbitration between model-based and model-free learning. *Neuron* 81, 687–699. doi: 10.1016/j.neuron.2013.11.028

Lengyel, M., and Dayan, P. (2008). "Hippocampal contributions to control: the third way," in *NIPS,* Vol. 20 (Denver, CO), 889–896.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., et al. (2015). Continuous control with deep reinforcement learning. *arXiv [Preprint].* arXiv:1509.02971.

Miall, R., Weir, D., Wolpert, D. M., and Stein, J. (1993). Is the cerebellum a smith predictor? *J. Motor Behav.* 25, 203–216. doi: 10.1080/00222895.1993.9942050

Miall, R. C., and Wolpert, D. M. (1996). Forward models for physiological motor control. *Neural Netw.* 9, 1265–1279. doi: 10.1016/S0893-6080(96)00035-4

Miller, E. K., and Cohen, J. D. (2001). An integrative theory of prefrontal cortex function. *Annu. Rev. Neurosci.* 24, 167–202. doi: 10.1146/annurev.neuro.24.1.167

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., et al. (2013). Playing atari with deep reinforcement learning. *arXiv [Preprint].* arXiv:1312.5602.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., et al. (2015). Human-level control through deep reinforcement learning. *Nature* 518, 529–533. doi: 10.1038/nature14236

O'Doherty, J. P., Lee, S. W., Mcnamee, D., O'Doherty, J. P., Lee, S. W., and Mcnamee, D. (2015). The structure of reinforcement-learning mechanisms in the human brain. *Curr. Opin. Behav. Sci.* 1, 1–7. doi: 10.1016/j.cobeha.2014.10.004

Poldrack, R. A., Clark, J., Pare-Blagoev, E., Shohamy, D., Moyano, J. C., Myers, C., et al. (2001). Interactive memory systems in the human brain. *Nature* 414, 546–550. doi: 10.1038/35107080

Rosenstein, M. T., and Barto, A. G. (2002). *Supervised Learning Combined with an Actor-Critic Architecture.* Department of Computer Science, University of Massachusetts, Tech. Rep, 02–41.

Satel, J., Trappenberg, T., and Klein, R. (2005). "Motivational modulation of endogenous inputs to the superior colliculus," in *IJCNN'05, Proceedings of 2005 IEEE International Joint Conference on Neural Networks, 2005* (IEEE), Vol. 1 (Montreal, QC: IEEE), 262–267.

Schultz, W., Dayan, P., and Montague, P. R. (1997). A neural substrate of prediction and reward. *Science* 275, 1593–1599. doi: 10.1126/science.275.5306.1593

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., and Riedmiller, M. (2014). "Deterministic policy gradient algorithms," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14),* 387–395.

Sutton, R. S. (1985). *Temporal credit assignment in reinforcement learning* (Ph.D. dissertation). University of Massachusetts, Amherst.

Sutton, R. S. (1988). Learning to predict by the methods of temporal differences. *Mach. Learn.* 3, 9–44. doi: 10.1007/BF00115009

Sutton, R. S. (1991). Dyna, an integrated architecture for learning, planning, and reacting. *ACM SIGART Bull.* 2, 160–163. doi: 10.1145/122344.122377

Uchibe, E., and Doya, K. (2005). "Reinforcement learning with multiple heterogeneous modules: a framework for developmental robot learning," in *Proceedings of the 4th International Conference on Development and Learning* (Osaka: IEEE), 87–92.

Uhlenbeck, G. E., and Ornstein, L. S. (1930). On the theory of the brownian motion. *Phys. Rev.* 36:823. doi: 10.1103/PhysRev.36.823

von Hofsten, C. (2004). An action perspective on motor development. *Trends Cogn. Sci.* 8, 266–272. doi: 10.1016/j.tics.2004.04.002

Waltz, M., and Fu, K. (1965). A heuristic approach to reinforcement learning control systems. *IEEE Trans. Autom. Control* 10, 390–398. doi: 10.1109/TAC.1965.1098193

Westwood, D. A., Heath, M., and Roy, E. A. (2003). No evidence for accurate visuomotor memory: systematic and variable error in memory-guided reaching. *J. Motor Behav.* 35, 127–133. doi: 10.1080/0022289030 9602128

Wolpert, D. M., Miall, R. C., and Kawato, M. (1998). Internal models in the cerebellum. *Trends Cogn. Sci.* 2, 338–347. doi: 10.1016/S1364-6613(98) 01221-2