2013 IEEE International Conference on Systems, Man, and Cybernetics

Neural Combinatorial Learning of Goal-directed Behavior with Reservoir Critic and Reward Modulated Hebbian Plasticity

Sakyasingha Dasgupta†, Florentin Wörgötter†, Jun Morimoto‡ and Poramate Manoonpong†
†Bernstein Center for Computational Neuroscience (BCCN), Georg-August-Universität,
Friedrich Hund Platz 1, 37077, Göttingen, Germany
dasgupta@physik3.gwdg.de
‡ATR Computational Neuroscience Laboratories, 2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288, Japan

***Abstract*—Learning of goal-directed behaviors in biological systems is broadly based on associations between conditional and unconditional stimuli. This can be further classified as classical conditioning (correlation-based learning) and operant conditioning (reward-based learning). Although traditionally modeled as separate learning systems in artificial agents, numerous animal experiments point towards their co-operative role in behavioral learning. Based on this concept, the recently introduced framework of neural combinatorial learning combines the two systems where both the systems run in parallel to guide the overall learned behavior. Such a combinatorial learning demonstrates a faster and efficient learner. In this work, we further improve the framework by applying a reservoir computing network (RC) as an adaptive critic unit and reward modulated Hebbian plasticity. Using a mobile robot system for goal-directed behavior learning, we clearly demonstrate that the reservoir critic outperforms traditional radial basis function (RBF) critics in terms of stability of convergence and learning time. Furthermore the temporal memory in RC allows the system to learn partially observable markov decision process scenario, in contrast to a memoryless RBF critic.**

***Keywords*—Re-inforcement learning, Reservoir networks, Correlation learning, Temporal memory**

I. Introduction

Operant conditioning (or reinforcement learning) and classical conditioning (or correlation-based learning) form the two classes of conditioning for associative learning in biological systems. Several animal experiments provide evidence of effective learning when these two classes are combined together [14]. Inspired by this in [8] the neural combinatorial learning framework was introduced. This combined the input correlation learning (ICO) [13] and actor-critic reinforcement learning (RL) [1] for controlling artificial agents in continuous time. The learning performance of the combined system clearly outperforms the individual mechanisms for both standard benchmark learning problems as well as complex goal-directed behavior problems. However, the actor-critic learner was modelled in a traditional manner, using a feedforward radial basis function (RBF) critic network [9]. Although this works well for most standard memoryless markovian learning tasks, it fails to approximate the value function in case of non-markovian or partially observable markov decision problems (POMDP).The role of the critic within the actor-critic learning paradigm is crucial as it needs to approximate the expected cumulative future reward (value function) such that the temporal difference (TD) error can be minimised. This TD-error in turn drives the policy of the actor and guides the behavior of the controlled agent. In case of highly non-linear environments where the agent has only partial sensory capabilities, a critic with temporal memory is required. As such in this paper, we replace the previous RBF based critic with a new recurrent neural network based adaptive critic of the reservoir computing (RC) type. RC networks [4][5] make use of a randomly connected dynamic reservoir with delayed temporal memory capacity [2]. Using a recursive least squares algorithm, this type of critic can be trained very fast in an online setup. Furthermore due to internal feedback connections, the short term memory of incoming sensory information can be used to solve POMDP learning problems. The RC based critic enhances the actor-critic based learner. In order to combine it with the ICO learning component of the combinatorial framework, learning of the connection weights between the two systems (Fig. 1) is very important. We solve this problem by introducing a new learning rule based on a biologically plausible mechanism called reward modulated Hebbian plasticity (RMHP) [6]. The RMHP rule updates the connection weights between ICO and actor-critic RL by checking for correlations between a constant reward signal and the deviation from the mean output level of the respective learning mechanisms. As such depending on the learner which drives the agent towards the correct goal (i.e. positively reinforced), the weight adaptation proceeds to finally find a suitable combination between the two learning systems.

Previously in [10][11] an application of the echo-state network (specific RC network) as an adaptive critic for reinforcement learning was presented. Although the authors implemented an online learner, the training and testing data for the RC network were carried out by manually controlling a wheeled robot. Moreover these implementations were designed with the purpose of minimizing a specific utility function for obstacle avoidance, door-passing scenario or very simple learning to reach a single goal. In contrast we implement a completely continuous learner where the reservoir critic learns online without any initial manual control of the robot. Furthermore, to the best of our knowledge, this is the very first implementation that combines correlation learning with reservoir based actor-critic learning and reward modulated Hebbian learning to succesfully

978-1-4799-0652-9/13 $31.00 © 2013 IEEE
DOI 10.1109/SMC.2013.174

demonstrate a more efficient and fast learner. In addition, the RMHP learning rule is both biologically plausible and an effective mechanism to learn the contribution of competing systems modulated by constant reward signal. As proposed in our previous work [2] self-adaptation of the reservoir neurons non-linearity is carried out using a general intrinsic plasticity mechanism based on the Weibull probability distribution.
We test our combined network on a complex goal-directed behavior task with a simulated wheeled robot for both fully and partially observable scenarios. The RC based adaptive critic clearly outperforms feedforward critic networks based on RBF kernels, both in terms of stability of performance as well as the learning time. Moreover the RMHP based weight adaptation rule, by working on a very slow timescale is able to accurately combine the two learning systems in an adaptive manner. Specifically this type of a neural combinatorial learning framework based on reservoir critics can be used to solve complex control problems as well as to solve tasks with delayed reward or partially observable state space, in continuous time.
This article is organized as follows. Section II introduces the neural combinatorial learning framework in greater detail with descriptions of the new reservoir based actor-critic learner (section II A) and the reward modulated Hebbian plasticity rule (section II B). Section III presents the experimental setup with the discussion of results. This is followed by the conclusion in Section IV.

II. Neural Combinatorial learning framework

In this section we briefly describe the neural combinatorial learning framework (CLF), as introduced in our previous work [8]. The CLF combines two classes of associative learning, namely classical conditioning and operant conditioning, as a dual learning system. It is used for goal directed behaviors in continuous state-action spaces. Classical conditioning involves the presentation of two different stimuli often termed as a conditional stimulus (CS) and an unconditional stimulus (US), leading to corresponding responses. The agent learns the association between the US and CS such that after learning completes, it now responds to the CS rather than the original unconditioned response (an innate reflex action) to the US. In general the CS acts as a predictor signal (occuring earlier in time) for the US, e.g. the famous Pavlovian dog [12] initially salivates (unconditioned response) at the sight of food (US) and after learning salivates at the ring of a bell (CS) much prior to the sight of food. However this type of learning occurs in the absence of any explicit future positive or negative feedback (other than the immediate reflex signal) for a particular action. In contrast, Operant conditioning based learning involves an explicit reinforcer or reward signal that provides positive or negative feedback to the agent for every corresponding action. Over time the agent learns to respond with the desired action such that it maximises (for the positive case) the total accumulated reward. As such this type of conditioning is popularly termed as Reinforcement learning (RL).
Although these two mechanisms are distinct from each other they seem to occur in combination as suggested from several animal behavioral studies. For a more clearer understanding let us consider the example of Pavlov's dog once again. Say, once the bell is rung, the dog is now required to perform a specific

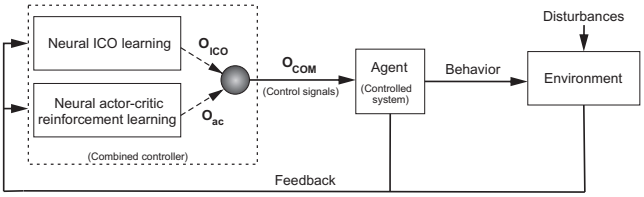

Fig. 1. Combinatorial learning framework with parallel combination of ICO learning and actor-critic reinforcement learning. Individual learning mechanisms adapt their weights independantly and then their final weighted outputs (O_{ico} and O_{ac}) are combined into O_{com} using a reward modulated Hebbian plasticity rule (dotted arrows represent plastic synapses). O_{com} controls the agent behavior (policy) while sensory feedback from the agent is sent back to both the learning mechanisms in parallel.

task (e.g. stand on two legs) and only then it receives food. In this slightly modified scenario, the bell is still the conditional stimulus; however, now the food acts as the reinforcer or reward signal. The dog learns to associate the sound of bell and food and starts to salivate based on classical conditioning. Interestingly after sufficient repetitions, the dog would learn to perform the desired action of standing on two legs as soon as it hears the bell and expect to receive the food as reward. Thus the overall behaviour is shaped through a combined learning system.
Inspired by such biological systems the CLF acts as a neural learning system that succesfully combines classical conditional (CC) with operant or reward based conditioning. Input correlation learning (ICO)[1] [13] was implemented as an example of CC, while a continuous actor-critic learner [1] was implemented as an example of reward based conditioning. Taking advantage of the individual learning mechanisms, the combined framework can learn the appropiate control policy for the agent in a fast and robust manner outperforming the singular implementation of the individual components.

The input correlation learning (ICO) and actor-critic RL subsystems can either be combined in series or in parallel. Previously in [7] serial combination was presented, where the ICO learner was used for reward related feature space extraction and provide prior knowledge to the actor-critic learner. Although this considerably improved the performance of the combined learning system, it suffered from the drawback of technical inconvinience of running the learning systems separately. This is also biologically less plausible. We extended this to a parallel combination in [8], however with a memoryless radial basis function critic network. Furthermore the subsystems were combined with equally weighted contribution in a non-adaptive manner to control the overall action of the Agent. As such in this work, we start with a parallel combination (Fig. 1) of the two individual learning systems. The actor-critic reward based learner is extended with a dynamic adaptive reservoir based critic with delay temporal memory capability [2] that can handle partially observable markov decision process problems (POMDP) in continuous time. Furthermore we implement a new reward modulated hebbian plasticity rule that learns the degree of contribution of the two learning systems.
The learning goal of the ICO learning system is to use a predictive signal (CS) in order to predict the occurence of the

[1]ICO learning is implemented as a differntial Hebbian learner. For more details refer to [8]

reflex signal (US). This in general enables the agent to react earlier and avoid the reflex altogether. Here the synaptic adaptation takes place by changes via heterosynaptic interactions as a consequence of the order of the arriving inputs. If the predictive inputs (agents sensory signals) are followed by the reflex input, the plastic synapses of the predictive inputs get strengthened and if the order is reveresed, it weakens based on a differential Hebbian learning method. For further details of the ICO learning system, the reader is refered to [13] [16].

A. Actor-critic Learning with Dynamic Reservoir

The continuous actor-critic reinforcement learning scheme is particularly suited for complex continuous state-action problems while at the same time being based on a biological learning model [3]. The basic learning model can be divided into two sub-mechanisms popularly termed as the actor and the adaptive critic (Fig. 2). The actor behaves as the main controller of an agent, while the critic provides an evaluative feedback or reinforcement signal to the actor by observing the consequences of its behaviour in the environment (controlled system). This evaluative feedback in general acts as a measure of goodness of behaviour i.e. overtime the agent learns to anticipate reinforcing events.

Inspired by the reservoir computing framework, here we use a large recurrent neural network (dynamic reservoir) as the critic. This provides a dynamic network with a large repertoire of reservoir signals that can be used to approximate the value function $v(t)$. It approximates the accumulated sum of the future rewards $r(t)$ with the discount factor γ where, $0 \leq \gamma < 1$.

$$v(t) = \sum_{i=1}^{\infty} \gamma^{i-1} r(t+i). \tag{1}$$

The primary goal of the critic is to predict $v(t)$ such that the temporal-difference error δ (TD-error) is minimized over time. The TD-error δ is computed from the predictions as follows:

$$\delta(t) = r(t) + \gamma v(t) - v(t-1). \tag{2}$$

The reservoir network (Fig. 2 bottom) is constructed as a random RNN with N internal neurons and fixed synaptic connectivity. The recurrent neural activity within the dynamic reservoir varies as a function of it's previous activity and the current driving input signal. As such, the discrete time state dynamics of reservoir neurons is given as:

$$\mathbf{x}(t+1) = (1-\lambda)\mathbf{x}(t) + \lambda f_{sys}(\mathbf{W}_{in}\mathbf{u}(t+1) + \mathbf{W}_{sys}\mathbf{x}(t)), \tag{3}$$

$$\mathbf{y}(t) = f_{out}(\mathbf{W}_{out}\mathbf{x}(t)), \tag{4}$$

where $\mathbf{x}(t)$ is the N dimensional vector of reservoir state activations, $\mathbf{u}(t)$ is the input to the reservoir, consisting of the agent's states (sensory inputs) and $\mathbf{y}(t)$ is the vector of output neurons. Here the predicted value function $v(t) = \mathbf{y}(t)$. The reservoir time scale is controlled by the parameter λ, where $0 < \lambda \leq 1$. $\mathbf{W}_{in}$ and $\mathbf{W}_{sys}$ are the input to reservoir weights and the internal reservoir recurrent connection weights, respectively.

The output weights $\mathbf{W}_{out}$ are calculated using the recursive least squares (RLS) algorithm at each time step, while the training inputs $\mathbf{u}(t)$ are being fed into the reservoir. $\mathbf{W}_{out}$ are calculated such that the overall TD-error is minimized. We implement the RLS algorithm using a fixed forgetting factor ($\lambda_{RLS} < 1$) as follows:

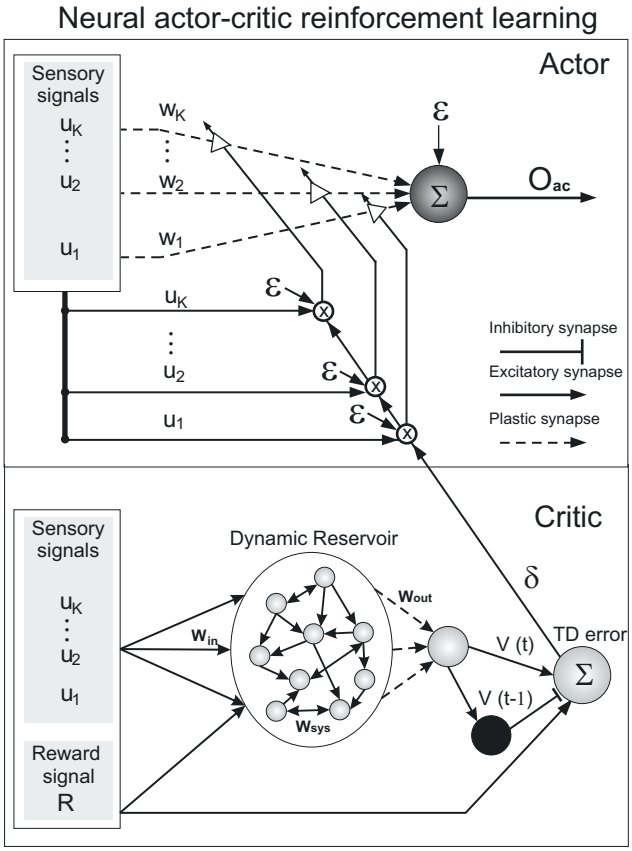

Fig. 2. The Neural circuit of actor-critic RL based on TD learning. (Top) The actor modeled as a stochastic neural network. (Below) The critic modeled using a dynamic reservoir network (details in text).

RLS algorithm for self-adaptive reservoir training:
Initialize: $\mathbf{W}_{out} = 0$, exponential forgetting factor (λ_{RLS}) is set to a value less than 1 (we use 0.85) and the auto-correlation matrix ρ is initialized as $\rho(0) = \mathbf{I}/\beta$, where $\mathbf{I}$ is unit matrix and β is a small constant.

Repeat: At time step t
Step 1: For each input signal $\mathbf{u}(t)$, the reservoir state $\mathbf{x}(t)$ and network output $\mathbf{y}(t)$ are calculated using Eq. 3 and Eq. 4.

Step 2: Online error $e(t)$ calculated as:
$e(t) \leftarrow \delta(t)$

Step 3: Gain vector $\mathbf{K}(t)$ is updated as:
$\mathbf{K}(t) \leftarrow \frac{\rho(t-1)\mathbf{x}(t)}{\lambda_{RLS} + \mathbf{x}^T(t)\rho(t-1)\mathbf{x}(t)}$

Step 4: Update the auto-correlation matrix $\rho(t)$
$\rho(t) \leftarrow \frac{1}{\lambda_{RLS}}\Big[\rho(t-1) - K(t)\mathbf{x}^T(t)\rho(t-1)\Big]$

Step 5: Update the instantaneous output weights $\mathbf{W}_{out}(t)$
$\mathbf{W}_{out}(t) \leftarrow \mathbf{W}_{out}(t-1) + K(t)e(t)$

Step 6: $t \leftarrow t+1$

Until: Maximum number of time steps is reached.

As proposed in [15][2] we also implement a generic intrinsic plasticity mechanism based on the Weibull distribution for unsupervised adaptation of the reservoir neuron nonlinearity. This allows the reservoir to homoeostatically maintain a stable firing rate while at the same time prevent unwanted chaotic neural activity. The reservoir neurons and the output neurons are updated using a tanh nonlinear activation function i.e. $f_{sys} = f_{out} = \texttt{tanh}$.

The actor is designed as a stochastic unit, such that for a one dimensional action setup the output (O_{ac}) is given as:

$$o_{ac}(t) = \epsilon(t) + \sum_{i=1}^{K} w_i(t)u_i(t) \tag{5}$$

where K denotes the number of sensory inputs ($\mathbf{u}(t) = u_1(t), u_2(t), .., u_K(t)$) to the agent being controlled. w_i represent the synaptic weights for the different sensory inputs. $\epsilon(t)$ is the exploration quantity updated at every time step such that the agent should explore the environment more if the expected cummulative future reward v is suboptimal and decrease the exploration as v is maximised. As a result one should expect the exploration to tend towards zero as the agent starts to learn the desired behavior. Using a gaussian white noise σ (zero mean and standard deviation one) bounded by the minimum and maximum limits of the value function (v_{min} and v_{max}), the exploration term is modulated as follows (Ω is constant scale factor):

$$\epsilon(t) = \Omega\sigma(t)\Big[\texttt{min}\Big[0.5, \texttt{max}\Big(0, \frac{v_{max} - v(t)}{v_{max} - v_{min}}\Big)\Big]\Big] \tag{6}$$

The actor learns by an online adaptation (Fig. 2 above) of its synaptic weights w_i at each time step modulated by the TD-error $\delta(t)$ from the Critic network (Equation (2)) as follows:

$$\Delta w_i(t) = \alpha\delta(t)u_i(t)\epsilon(t) \tag{7}$$

Where α is the learning rate such that $0 < \alpha < 1$.
Instead of using direct reward to update the actor weights, using TD-error (i.e. error of an internal reward) allows the system to handle even delayed reward control problems. In general once the agent learns the desired behavior, the exploration term ($\epsilon(t)$) should become zero, as a result of which no further weight change (Eq. (7)) occurs and $o_{ac}(t)$ gives the desired action without any noise. The reservoir network being an input driven dynamical system, endows the critic with long temporal memory in contrast to traditional feedforward critic networks (RBF kernels). Specifically in order to solve POMDP scenarios, temporal memory is crucial to propagate the knowledge of previously visited state space (sensory signals) for expected reward in the future. As a result unlike RBF based critics our network can effectively deal with such problems in continuous time.

B. Combinatorial learning with reward modulated Hebbian plasticity

In the previous subsections we provided an overview of the combinatorial learning framework along with the description of the new dynamic reservoir based actor-critic reinforcement learning network. We now elaborate on the parallel combination of the correlation-based learner (ICO) and the reward-based learner (actor-critic) as depicted in Fig. 1. The system works as a dual learner where the individual learning mechanisms run in parallel to guide the behavior of the agent. Both the systems adapt their weights independently while receiving sensory feedback from the agent (system state) in parallel. The final action that drives the agent is calculated as a weighted sum of the individual components. This can be described as follows:

$$o_{com}(t) = \xi_{ico}o_{ico}(t) + \xi_{ac}o_{ac}(t) \tag{8}$$

where, $o_{ico}(t)$ and $o_{ac}(t)$ are the t time step outputs of the input correlation-based learner and the actor-critic learner, respectively. $o_{com}(t)$ represents the t time step combinatorial action. The important parameter here is the weights of the individual components (ξ_{ico} and ξ_{ac}) that govern their degree of influence on the net action of the agent. A simple and straight forward approach [8] is to provide equal contribution ($\xi_{ico} = \xi_{ac} = 0.5$) for controlling the agent. Although this leads to successful solutions, they are sub-optimal. Intuitively for associative learning problems with immediate rewards the ICO system learns quickly as compared to distal reward based goal-directed problems where the ICO learner provides guidance to actor-critic learner. In general depending on the type of problem, the interaction between the two learning systems differs and needs to be taken into account. We solve this problem by introducing a new plasticity rule called reward modulated hebbian plasticity [6] in order to learn the individual synaptic weights. Based on this plasticity rule the ICO and actor-critic RL weights are learnt at each time step as follows :

$$\Delta\xi_{ico}(t) = \eta r(t)(o_{ico}(t) - \bar{o}_{ico}(t))o_{ac}(t), \tag{9}$$

$$\Delta\xi_{ac}(t) = \eta r(t)(o_{ac}(t) - \bar{o}_{ac}(t))o_{ico}(t). \tag{10}$$

Here $r(t)$ is the current time step reward signal received by the agent, while $\bar{o}_{ico}(t)$ and $\bar{o}_{ac}(t)$ denote the low-pass filtered version ($\bar{o}_{ico,ac}(t) = 0.9\bar{o}_{ico,ac}(t-1) + 0.1o_{ico,ac}(t)$) of the output from the ICO learner and the actor-critic learner, respectively. The plasticity model used here is based on the assumption that the net policy performance (agents behavior) is influenced by a single global neuromodulatory signal. The learning rule measures correlations between the reward signal and the deviations of the ICO and actor-critic learner outputs from their mean values and accordingly adjusts the respective weights. In order to prevent uncontrolled divergence in the learnt weights (ξ_{ico} and ξ_{ac}), synaptic normalization is introduced by dividing the individual weights by the total sum of weights. This ensures that the weights always add up to one and $0 < \xi_{ico}, \xi_{ac} < 1$. In general this plasticty rule occurs on a slow time scale which is governed by the learning rate parameter η. Typically η is set much less compared to the learning rate of the two individual learning systems (ICO and actor-critic).

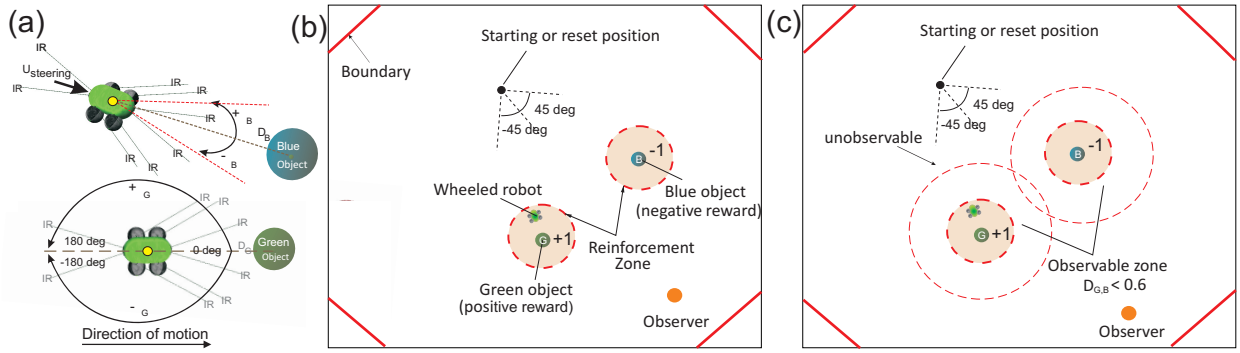

Fig. 3. Simulated mobile robot system for goal-directed behavior task. (a) (Top) The mobile robot NIMM4 with different types of sensors. The relative orientation sensor ϕ is used as state information for the robot. (Bottom) Variation of the relative orientation ϕ_G to the green goal. (b) Environmental setup for the fully observable case. The robot continuously senses its relative orientation to both the green and blue objects. Only whithin the reinforcement zone (shaded circle) the robot receives positive reward when near the green goal and negative reward when near the blue goal. (c) Environmental setup for the partially observable case. The robot can sense its relative orientation to the goals only when within the observable zone (outer dotted circles). Reinforcement is received similar to the fully observable case. Here the orange object represents an external observer.

III. Experiments and Results

In order to test the performance of the combinatorial learning framework with a reservoir critic and reward modulated Hebbian plasticity, we employ a goal-directed behavior control task using a simulated wheeled robot system (Fig. 3 (a)). The task is to let the wheeled robot NIMM4 learn to steer itself towards a desired goal (green ball, Figs. 3(b) and (c)) within a given time. As the robot approaches the desired goal, it receives positive reinforcement. Additionally an undesired goal (blue spherical ball) with negative reinforcement was also placed within the same arena. NIMM4 is provided with two relative orientation sensors (ϕ_G - green ball, ϕ_B - blue ball) that can measure angle of deviations from the two goals. They can take values in the interval $[-180^o,180^o]$ with the $\phi_{G,B} = 0^o$ when the respective goal is directly in front of the robot. In addition NIMM4 also consists of two relative position sensors ($D_{G,B}$) that can calculate it's relative distance to a goal in the interval [0,1] with the respective sensor reading tending to zero, as the robot gets closer to a goal. The task is further divided into fully and partially obervable scenarios. In the first case, the robot can continuously sense its angle of deviation to the two goals with $\phi_{G,B}$ always active. For the later case, the robot cannot sense direction to either of the goals ($\phi_{G,B}$ inactive) untill it reaches the half way distance to either of the goals i.e. $D_{G,B} < 0.6$. In both the cases when the robot gets very close to either of the goals, within a distance of ($D_{G,B} = 0.2$) it receives a positive or negative reward. Within this boundary for the green goal it receives a continuous reward of +1 at every time step and a continuous reward of -1 in case of the blue goal, respectively. This distance is also used as the zone of reflex to trigger a reflex signal for the ICO learner. It is important to note that only the relative orientation sensory data is used as state input for both the ICO learner and the actor-critic learner. Furthermore as $\phi_{G,B}$ signals overlap with each other (i.e., the robot simultaneously senses its relative orientation to both the goals in the whole arena). NIMM4 is also supplied with eight infra-red sensors that are used only to reset it to the starting location if it hits a boundary before reaching either of the

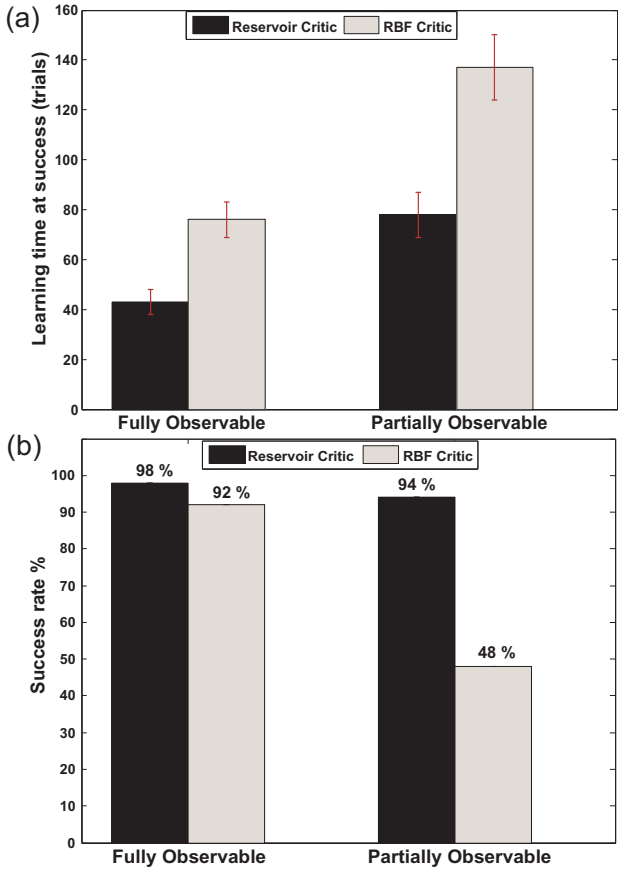

Fig. 4. Performance comparison between a reservoir based critic and RBF based critic for the fully observable and partialy observable cases (ICO and actor components remained the same). (a) Average learning time (trials) needed to succesfully complete the task, calculated over 50 experiments (error bars indicate standard deviation for 95 % confidence interval). (b) Success rate in percentage. Here "success" indicates the robots ability to correctly navigate to the green goal.

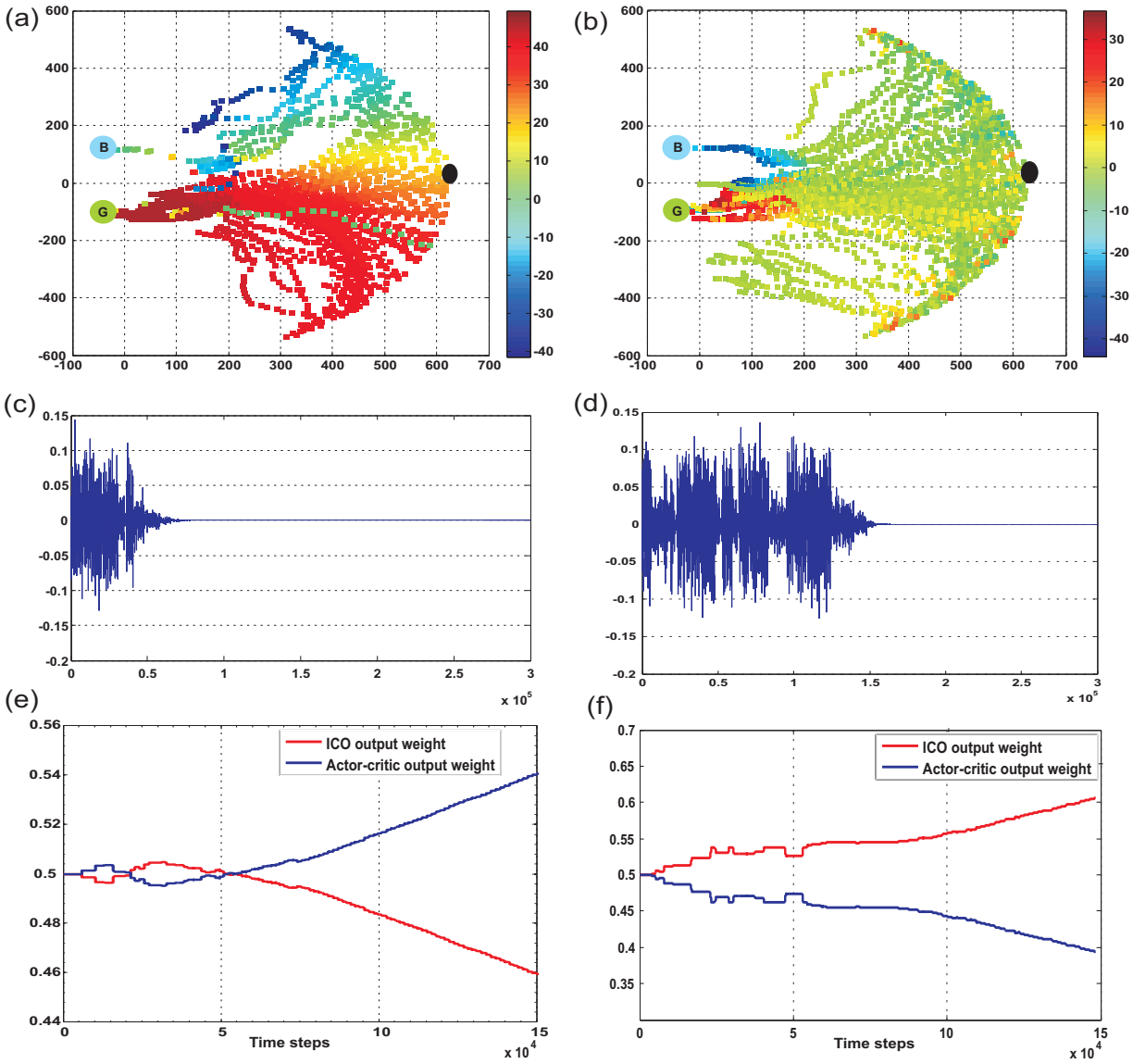

Fig. 5. (a) Estimation of the value function $v(t)$ using reservoir based critic. The $v(t)$ estimate is plotted with respect to local co-ordinates of the robot and an observer located directly opposite to the robot starting position. Colormap indicates the changing $v(t)$ values. The black ball indicates the starting position of the robot with random orientation and the curvature of the plot is resultant of the shape of view from the observer. (b) Estimation of value function $v(t)$ for the same task using the static RBF based critic. (c) Convergence of exploration term $\epsilon(t)$ using reservoir critic. (d) Convergence of exploration term using RBF critic. (e) Adaptation of ICO weights ξ_{ico} and actor-critic weights ξ_{ac} using the RMHP rule for Combined learner with reservoir critic. (f) Adaptation of ICO weights ξ_{ico} and actor-critic weights ξ_{ac} using the RMHP rule for Combined learner with RBF critic. The continuous change in the learned weights even after successful learning of task is due to the Hebbian nature of the adaptation rule. This can be easily controlled by introducing additional synaptic scaling mechanism or in this case stop the weight updation once exploration ($\epsilon(t)$) becomes zero. All the plots were generated for the same goal-directed behavior task of reaching the green goal. The plots indicate the best performance case for each setup.

Fig. 6. Simulation screenshots showing the actual behavior of the robot after succesfully learning the task. Upon learning, it continuously steers towards the green goal and avoids movements towards the blue ball. For video of a complete learning sequence, please visit http://www.manoonpong.com/rcrl/rcrl.wmv .

goals. Keeping the ICO learner fixed for the combinatorial setup, we tested both the scenarios (Figs. 3(b) and (c)) for a reservoir based critic and a feedforward RBF critic. The combinatorial learning mechanism learns to steer the robot towards the desired goal (green object). Without control, the robot randomly moved around. The robot always starts from the same location, however with random orientation. 50 runs were carried out with each setup for both fully observable and partially-observable scenarios. Each run consisted of a maximum of 200 trials (robot resets). The robot was reset if it reached either of the goals or if it hit a boundary wall or if the maximum simulation time of 15s was reached.

ICO learning was setup as follows: $\phi_{G,B}$ were used as predictive signals. Two independent reflex signals were configured with one for blue ball and the other for the green ball. The reflex signal was designed to elicit a turn towards a ball once the robot comes close enough to it (inside the dotted circle in Figs. 3(b) and (c)). Irrespective of the kind of goal (desired or undesired) the reflex signal drives the robot towards it with a turn proportional to the deviations defined by $\phi_{G,B}$ i.e large deviations cause sharper turns. The green and the blue ball were placed such that there was no overlap between the reflex areas, hence only one reflex signal got triggered at a time. In other words, the goal of the ICO learner is simply to learn to drive towards a goal location without any knowledge of their worth (positive or negative reward).

The actor-critic learner was setup as follows: The inputs to the critic and actor networks (Fig. 2) consisted of the two relative orientation sensor data ϕ_G and ϕ_B. The reservoir network for the critic consisted of $N = 100$ neurons and one ouput neuron that estimates the value function $v(t)$ (Eq. (1)). Reservoir input weights W_{in} were drawn from an uniform distribution $[-0.5, 0.5]$ while the reservoir recurrent weights W_{sys} were drawn from the uniform distribution $[-1, 1]$. W_{sys} was subsequently scaled to a spectral radius of 0.9 with only 10% internal connectivity. The reward signal $r(t)$ (Eq. (2)) was set to +1 when the robot comes close to the green ball and to -1 when it comes close to the blue ball. A RBF feedforward network was used for comparison with the reservoir based critic. The RBF critic size was varied from 16 to 100 hidden neurons. All other combinatorial network parameters are summarised in Table 1.

The performance of the reservoir based critic as compared to the RBF critic (keeping all other components of the combinatorial learning framework the same) is compared in Fig. 4 with respect to the fully and the partially observable scenarios of the same task. As observed from Fig. 4(b), the reservoir based critic clearly outperforms the RBF critic. Moreover the difference in performance is highly significant in the POMDP scenario, where the reservoir network outperforms the RBF critic by a success rate greater than 50%. Temporal memory of incoming agent state information available to the reservoir critic is crucial for solving complex non-markovian problems, as compared to memoryless feedforward critic networks. Furthermore although both the implementations have almost similar success rate for the fully observable case, the reservoir based system converges to a solution (learned behavior of driving the robot to the green goal) faster (less than 50 trials), as observed in Fig. 4(a). However, expectedly the POMDP scenario takes longer time to learn the correct behavior, owing to the reduction in the total sensory information available to the system. Upon successfully learning the task the weights of the actor (Eq. 7) converge such that the robot gets pulled towards the desired green goal. It should be noted that although linear actors (Eq. 5) were used in this setup, the POMDP scenario is effectively solved due to the inherent trace of previous inputs in the reservoir critic. In contrast the memoryless RBF critic system works on chance and hence learns the POMDP task with less than 50% success rate.

In Figs. 5 (a) and (b) we compare the performance of the reservoir based critic with a RBF critic network in terms of the value function estimation curves for the same goal-directed behavior task (i.e. the fully observable task). It is clearly observed that the reservoir critic successfully enables the mobile robot to learn to drive towards the green goal while avoiding the blue goal. Furthermore unlike the RBF critic (Fig. 5(b)), the value function curve in Fig. 5(a) displays a strong gradient of the estimated value of $v(t)$ with high positive values towards the correct goal (green object). In contrast the memory less RBF critic estimates $v(t)$ to values closer to zero in most locations except for regions within the zone of reward. As a result our modified critic learns the task faster as indicated by the fast convergence of the exploration term $\epsilon(t)$ in Figs. 5 (c) and (d). In Figs. 5(e) and (f) we plot the development of the ICO ξ_{ico} and actor-critic weights ξ_{ac} via the RMHP learning rule (Eqs. (9) and (10)). In case of the reservoir critic, the actor-critic learner component is seen to dominate over the ICO learner. In contrast when the RBF critic was used for the same task, the learnt behavior is dominated by the ICO component. This can be explained in terms of the memory of sensory state present in the reservoir network that successfully guides the agents behavior in contrast to the

TABLE I.

The List of combinatorial network parameters	
Reservoir critic size (neurons)	100
RBF critic size (neurons)	16 - 100
Reservoir leak rate (λ)	0.3
RLS learning constant (β)	10^{-2}
Discount factor (γ)	0.95
Scale factor (Ω)	5
Maximum value (v_{max})	50.0
Minimum value (v_{min})	-50.0
Neuron non-linearity (f_{sys},f_{out})	tanh
RLS learning rate (λ_{RLS})	0.85
Actor learning rate (α)	0.001
RMHP learning rate (η)	0.0005

memoryless RBF network. In general the weight adaptation should occur in a task dependent manner. The actual behavior of the robot NIMM4 after succesfully learning the task of navigating towards the green goal, is depicted via screenshots of the simulation in Fig. 6.

IV. Conclusion

In this work we have successfully extended the neural combinatorial learning framework (CLF) using a reservoir network based adaptive critic, while using a stochastic linear actor unit and a basic implementation of input correlation learning. The resultant network effectively solves goal directed behavioral problems and outperforms the CLF with traditional radial basis function (feed-forward network) based critics both in terms of rate of success and the overall learning time. Furthermore due to the inherent temporal memory of reservoir networks, our modified critic enables the CLF to solve partially observable scenarios. In addition we implement a new biologically plausible reward modulated Hebbian plasticity rule which enables the CLF to learn the degree of influence of the ICO learner as compared to the actor-critic learner. This allows automatic weight adaptation between the two components working on a very slow time scale. As a future direction we plan to extend the linear actor units to reservoir based nonlinear actors which would work in conjunction with the reservoir critic. This would enable the controlled agent with memory capabilities in the action domain and thereby solve more complex goal-directed behavioral tasks like delayed reward maze navigation. Moreover behavioral analysis of the RMHP rule for evaluating the stability of learning against changing metaparameters (E.g. individual learning rates) with task independent evaluation measures will be carried out as further extension of our current work.

Acknowledgment

This research was supported by the Emmy Noether Program (DFG, MA4464/3-1), the Federal Ministry of Education and Research (BMBF) by a grant to the Bernstein Center for Computational Neuroscience II Göttingen (01GQ1005A, project D1), the Strategic Japanese-German Cooperative Program on Computational Neuroscience (JST-DFG, WO388/11-1), European Community's Seventh Framework Programme FP7/2007-2013 (Specific Programme Cooperation, Theme 3, Information and Communication Technologies) under grant agreement no. 270273 (Xperience).

References

[1] Doya, K. (2000) Reinforcement Learning In Continuous Time and Space. Neural Computation, 12, 219-245.

[2] Dasgupta, S., Wörgötter, F., and Manoonpong, P. (2013) Information Dynamics based Self-adaptive Reservoir for Delay Temporal Memory Tasks. Evolving Systems, doi: 10.1007/s12530-013-9080-y.

[3] Frémaux, N., Sprekeler, H., Gerstner, W. (2013), Reinforcement Learning Using a Continuous Time Actor-Critic Framework with Spiking Neurons. PLoS Comput Biol 9(4): e1003024. doi:10.1371/journal.pcbi.1003024.

[4] Jaeger, H., and Haas., H. (2004): Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication. Science, 304(5667), 78-80.

[5] Maass, W., Natschlger, T., and Markram, H. (2002), Real-time computing without stable states: a new framework for neural computation based on perturbations. Neural Computation. 14 (11): 253160.

[6] Legenstein, R., Chase, S.M., Schwartz, A.B., and Maass, W. (2010) A reward-modulated hebbian learning rule can explain experimentally observed network reorganization in a brain control task. J Neurosci 30:84008410.

[7] Manoonpong, P., Wörgötter, F., and Morimoto, J. (2010) Extraction of Reward-Related Feature Space Using Correlation-Based and Reward-Based Learning Methods. In Proc. 17th International Conference on Neural Information Processing, Sydney, Australia, November 22-25 (ICONIP'10),Part I, LNCS 6443, pp. 414-421.

[8] Manoonpong, P., Kolodziejski, C., Wörgötter, F., and Morimoto J. (2013) Combining Correlation-Based and Reward-Based Learning in Neural Control for Policy Improvement. Advances in Complex Systems, doi: 10.1142/S021952591350015X.

[9] Morimoto, J., and Kenji, Doya. (1998) Reinforcement learning of dynamic motor sequence: Learning to stand up. In Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 3. IEEE.

[10] Oubbati, M., Kchele, M., Koprinkova-Hristova, P., and Palm, G. (2011), Anticipating Rewards in Continuous Time and Space with Echo State Networks and Actor-Critic Design. In Proc. 19th European Symposium on Artificial Neural Networks (ESANN).

[11] Koprinkova-Hristova, P., Oubbati, M., and Palm, G. (2010). Adaptive critic design with echo state network. In Proc. IEEE International Conference on Systems, Man, and Cybernetics, 1010-1015.

[12] Pavlov, I., Conditioned reflexes (Oxford University Press, Oxford, UK, 1927)

[13] Porr, B., and Wörgötter, F. (2006), Strongly improved stability and faster convergence of temporal sequence learning by utilising input correlations only. Neural computation 18, 1380-1412.

[14] Skinner, B., The Behavior of Organisms: An Experimental Analysis (Appleton Century Croft, New York,1938)

[15] Triesch, J. (2007), Synergies between Intrinsic and Synaptic Plasticity Mechanisms. Neural Computation 4, 885-909.

[16] Wörgötter, F., and Porr, B. (2004) Temporal sequence learning, prediction and control - a review of different models and their relation to biological mechanism. Neural Computation. 17, 245-319.