The 23rd IEEE International Symposium on Robot and
Human Interactive Communication
August 25-29, 2014. Edinburgh, Scotland, UK,

Path-finding Using Reinforcement Learning and Affective States

Johannes Feldmaier[1] and Klaus Diepold[1]

Abstract— **During decision making and acting in the environment humans appraise decisions and observations with feelings and emotions. In this paper we propose a framework to incorporate an emotional model into the decision making process of a machine learning agent. We use a hierarchical structure to combine reinforcement learning with a dimensional emotional model. The dimensional model calculates two dimensions representing the actual affective state of the autonomous agent. For the evaluation of this combination, we use a reinforcement learning experiment (called *Dyna Maze*) in which, the agent has to find an optimal path through a maze. Our first results show that the agent is able to appraise the situation in terms of emotions and react according to them.**

I. Introduction

The integration of affective states and emotions into next generation artificial intelligence (AI) could be the next consequent step in the development of robotic systems. In modern psychology emotions are an important component in decision making. Humans judge situations with emotions and the reactions depends on them [1], [2]. Emotions can also be seen as a way to simplify actual decisions, enabling one to compare different events and actions against each other [3]. Thereby, emotions can be used as a utility function for action selection and for event classification. Using objective utility functions for decision making is a common basis in AI systems. In the future, it would be desirable to use affective utility functions instead. This could improve the believability and behaviour of AI systems. At a first glance, emotions could be seen as a direct result of the state and previous actions and hence there is no difference compared to objective utility functions. But generally emotions incorporate more than objective features. They depend on personality (especially the mood), the attachment to things and previous experiences. Especially knowledge and experience are main components of emotions. In psychology there is a phenomenon called *mental time travel* describing a situation in which the current action promises an immediate positive emotion but also simultaneously an expected negative consequence in the future (e.g. stealing something). Remembering consequences requires a memory process and the ability to learn. Particularly reinforcement learning (RL) combines rewards with previous actions and is a commonly used strategy in artificial intelligence. However, not many existing systems combine reinforcement learning with rewards calculated using artificial emotions. One approach integrating emotional attachment and decision making into an agent is described in [4]. This approach shows some similarities to our system, but focuses mainly on the human-machine interaction part. However, our approach focuses on the affective appraisal of decisions made by the agent itself. Therefore, we integrate affective states into reinforcement learning. Furthermore, we describe why such a combination is a contribution to social robotics. Social robotics means in this context that a robotic system should act in a way that can be understood by humans. Today, many robots act like big black boxes with some actuators and thereby elicit feelings of fear and anxiety in their human operators. On the one hand, this is a result of their often cryptic textual outputs and signals. On the other hand, there exists only few output devices displaying the current emotional state of the system using profound integrative and functional frameworks of the displayed emotion and mood. A display showing the user the actual emotional state (e.g. the feelings and mood) enables the AI agent more transparent to non experts and therefore more attractive and less frightening.

The primary goal of this work is the combination and integration of affective states and models with reinforcement learning. We use a reinforcement learning framework (*Dyna Maze*) simulating a small maze, which is bounded with walls and contains some obstacles. In Section II we point out some basic principles of reinforcement learning. Section III describes the main points of our dimensional emotion model which is then, in Section IV, used in combination with the reinforcement learning framework to appraise the learning progress of the agent. The last two Sections discuss the results and conclude the work.

II. Reinforcement Learning

For the evaluation of our emotional model and the combination with reinforcement learning we use the well known gridworld example *Dyna Maze* [5]. This example represents a rectangular maze with some walls and obstacles. The agent has in each state four possible actions, *up, down, right,* and *left* (cf. Fig. 1). Selecting one of the actions takes the agent to the corresponding neighbouring state. In cases when the movement is blocked by a wall or the boundaries of the maze, the agent remains in its current position. The agent begins in the start state S and has to reach the goal state G. We define the state space $S = s^1, s^2, ..., s^n$ and the available set of actions as $A(s)$. Hence, $s_t \in S$ denotes the actual state of the agent at time t and $a_t \in A(s)$ denotes the action executed at time t. Executing action a_t causes a transition change from state s_t to state s_{t+1}. Additionally, we can assume a deterministic world so that each transition or action is executed properly.

[1]J. Feldmaier and K. Diepold are with Department of Electrical Engineering, Institute for Data Processing, Technische Universität München, Germany johannes.feldmaier@tum.de, klaus.diepold@tum.de

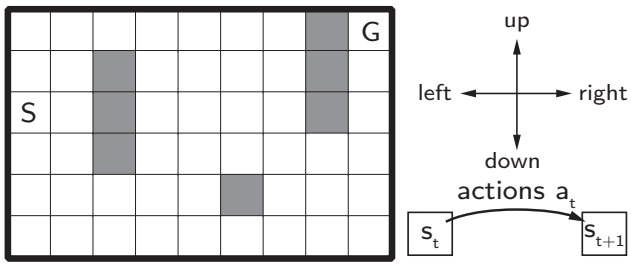

Fig. 1. Gridworld example and the four possible actions a_t. Each field of the grid corresponds to a possible state s_t of the agent.

The experiment is conducted in episodes. In each episode the agent has to find its way from the starting point to the goal. The maze always remains unchanged. In the first episode the agent is placed at the start state *S* without any knowledge about the structure of the maze. The agent first selects an action (randomly) and performs the movement within this new environment. After performing each action the agent receives a reward r_t from the environment. The first steps or episodes can be seen as an exploration process. During the exploration the reward is either negative or relatively low. If the agent reaches the goal state *G*, it receives a high positive reward. After some episodes, the cumulative reward at the end of each episodes increases because the agent gains knowledge about the maze.

In our experiment we use the *Dyna-Q* algorithm [5] for learning and selecting the actions in each state. *Dyna-Q* includes planning, acting, model-learning and direct reinforcement learning. Each process occurs continually during the exploration and path-finding.

First, the agent selects an action a_t and performs the transition $s_t, a_t \rightarrow s_{t+1}, r_{t+1}$ to the next state s_{t+1}. After the state transition the agent gets the reward which depends only on state s_{t+1}. With the reward and the performed action the model records in its table entry for the previous state and action pair s_t, a_t the prediction for the reward in the following state s_{t+1} taking action a_{t+1}. Thus, the model is able to predict the next reward using the last-observed next state and reward. The model update step is followed by the planning step. While planning the *Dyna-Q* algorithm randomly samples previously observed state-action pairs improving the action-value (*Q*-) function. The action-value function *Q* is learned or updated after each state transition. The *Q*-function estimates the future expected reward taking an action in a specific state. This enables the agent to determine *how good* it is to be in a given state and perform a given action [5]. We use *Q-learning* to update the action-value function. The update of the *Q*-function

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \gamma max_a Q(s_{t+1}, a) - Q(s_t, a_t)] \quad (1)$$

is independent of the policy being followed (*off-policy*) [6] and depends on the step-size α and discount-rate γ (cf. Table I). After planning, updating the *Q*-function and the model, the agent uses the action-value function to select the next action. For the action selection we used the ϵ-greedy policy. This policy selects most of the time the *greedy* action (the one with the highest *Q* value), but once in a while with a small probability ϵ it selects a random action.

We use the described *Dyna-Q* learning algorithm for the path-finding task in a small maze environment because this experiment provides a manageable complexity, sufficient events for the stimulation of our emotional model and additionally, the structure can be integrated in the affect design structure of Norman et al. [7].

III. Emotional Model

There is no unique definition of emotions but rather there are several different models of emotional systems. One theory describes the emotional system of humans with so-called *basic emotions*. This theory was mainly influenced by Paul Ekman [8] and Jaak Panksepp [9]. The basic emotion theory describes emotions as discrete states disregarding some intermediate states and the integration of the theory into an overarching framework. Such a complete and continuous structure of emotions is offered by so called *dimensional theories* [10], [11], [12]. Commonly, dimensional models represent emotions as a point in a two- or three-dimensional space. The different dimensions were determined in empirical studies in which subjects had to appraise emotional scenes.

There is an ongoing discussion [13], [14] between the dimensional and the basic emotion views of emotion, but it is not our objective to support one model explicitly. Furthermore, we consider the different models regarding their usability in technical systems. Therefore, we decided to implement and evaluate an emotional model based on the dimensional theory of emotions.

VA-Space Model

In dimensional models the affective state of an organism is represented in a space of two- or three-dimensions. This space is called core affective state space. Most computational models build on the three-dimensional *PAD* model of Mehrabian and Russell [10]. In the *PAD* model the three dimensions of the core affective state are denoted as pleasure *P*, arousal *A*, and dominance *D*. Dropping dominance *D* is a common simplification of the model. The dimension of dominance is not significant in the present scenario involving only a single agent. Pleasure denotes a measure of valence and indicates whether an emotion, an event, or an experience is good or bad for you (the general perception of positivity or negativity of a situation). In psychology, the terms pleasure and valence are interchangeable. In the following, we will use valence *V*. The second fundamental dimension is called arousal *A* and indicates the level of affective activation of an agent. Arousal can be defined as the mental alertness and physical activity [15]. The core affective state space spanned by the two dimensions of valence and arousal is further described by the discrete emotions which subjects have reported for each quadrant (Q1 – Q4 in Figure 2) [16]. They serve as additional discrete labels for each core affective state. Placing a point in this two-dimensional space and *pushing it around* by a

continuous time-varying process appraising eliciting events is a commonly used implementation for the core affect of an agent [17], [18]. Besides direct influences of eliciting events, the core affect could additionally be modified by incorporating the impact of dispositional tendencies such as mood state and personality. There is some disagreement within psychology to use dimensional models as emotion eliciting process [13], [14], [18]. However, in technical systems the representation of events within the two-dimensional affective state space is a convenient and economical way to appraise events in terms of emotions [18], [19]. For this reason, we use a two-dimensional emotion model for calculating the affective state during a reinforcement learning experiment.

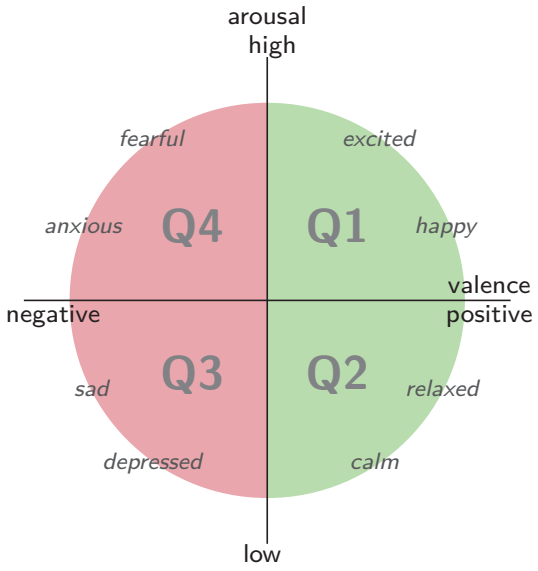

Fig. 2. Core affect represented in two-dimensional space – valence and arousal. Possible reported affective states (emotions) are stated in each quadrant (words in italics). The left half-plane of the VA-space is related to negative (red) experiences and the right half-plane to positive (green) ones (adapted from [16]).

IV. Integration of the Emotional Model into Dyna Maze

Norman et al. published a paper about affect and machine design describing a model of affect and cognition. They propose a three-level theory of human behaviour basically applicable to the architecture of affective computer systems [7].

The three levels are the *reaction* level, the *routine* level, and the *reflection* level. These levels enable a processing of the surrounding world in terms of affective evaluation and cognitive interpretation of the environment.

Figure 3 depicts the three levels including their assigned components of the Dyna Maze example and the emotional model. Lower levels perform fast computations and processes and higher levels involve more information resulting in increased effort.

The lowest level – the reaction level – processes low level information, performs rapid reactions to the current state (such as reflexes), and controls motor actions. Therefore, we have used the reaction level to perform the state transitions

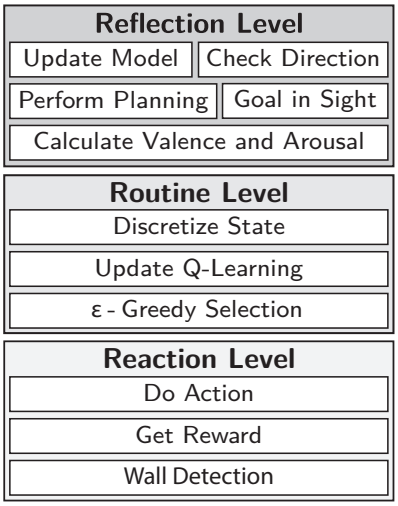

Fig. 3. Integration of reinforcement learning and Dyna Maze into the three level model of affect and cognition.

of the agent within the maze, detect collisions with walls and obstacles and receive the reward from the environment.

In the routine level, *routinized* actions are performed. In humans, the routine level is responsible for most motor skills, language generation, and other skilled and well-learned behaviours [7]. It has access to working memory and more permanent memory to guide decisions and update planning mechanisms. Therefore, it is the perfect level to perform two important steps of reinforcement learning: updating the action-value function (*Update Q-learning*) and selecting the following action according to a *routinized* policy (e.g. ϵ-*Greedy Action Selection*). In the present maze scenario, the discretization of the state is not a complex step, but in alternative scenarios with continuous states and a non-deterministic world, the *Discretize State* step would be a challenging and computationally expensive step and thus has to be performed in the routine level.

The highest level – the reflection level – calculates the most computationally intensive steps. Reflection is a *meta-process* involving all available data to build own internal representations about the surrounding environments. It reasons about the environment and interprets the pre-processed sensory inputs of lower levels. The main tasks of this level are planning, problem solving, and reasoning about facts. This suggests to implement the update and planning functions of reinforcement learning into the reflection level. Updating the model means to use the current and previous state, the actual performed action, and the earned reward to update the previously saved model with this new information. Similarly, the planning step of reinforcement learning uses the model (the *long-term* memory of the agent) and selects random previously observed states and performs *virtually* a random action at this state. The model outputs a reward for this random action which in turn is used to update the model itself. This kind of planning internally simulates the environment and produces *simulated experience* [5].

Besides these high-level functions of reinforcement learning the reflection level accommodates the calculation of the core affect of the agent.

Calculation of Arousal and Valence

The calculation of the arousal and valence value requires information of lower levels and, additionally, reasoning steps and thus takes place in the reflection level. The reaction level monitors changes in direction, collisions with walls, and obstacles. This triggers short term memory processes in the reflection level. Each memory process is modeled by a frequency variable. The frequency is high, if the event occurs frequently and decays to zero, if the event does not occur anymore.

If a collision occurs ($c = 1$) this is reported to the reflection level. The reflection level uses this information and computes a frequency of collisions given by

$$f_{col}(t) = \begin{cases} f_{col}(t-1) + (1 - f_{col}(t-1)) \cdot \nu & \text{if } c = 1 \\ f_{col}(t-1) + (-1 - f_{col}(t-1)) \cdot \nu & \text{otherwise,} \end{cases} \quad (2)$$

using a discrete dynamical system where ν describes the amount of residue that is added or subtracted from the frequency value. The system responds to collision in an exponential way. The more collisions occur the faster (exponential) the frequency value converges to one. If there are no collisions, the frequency value decays exponentially and converges to minus one.

A similar approach was chosen to model the frequency of changes in direction $f_{dir}(t)$. If the agent changes its direction in every step, a value should go up indicating these frequent changes. For this the reaction level reports the actual action to the reflection level. A comparison with the previous action results in setting a variable $d = 1$ indicating the direction change. A second frequency value is calculated by

$$f_{dir}(t) = \begin{cases} f_{dir}(t-1) + (1 - f_{dir}(t-1)) \cdot \theta_1 & \text{if } d = 1 \\ f_{dir}(t-1) + (-1 - f_{dir}(t-1)) \cdot \theta_1 & \text{otherwise,} \end{cases} \quad (3)$$

with θ_1 determining the amount of $f_{dir}(t-1)$ added to or subtracted from the previous frequency value. The response to changes in direction is also exponential and increases or decreases at the beginning very fast and slowly converges then to its maximal value of minus or plus one. We additionally modified θ_1 if the agent follows the policy and selects the optimal selection or if the agent performs random exploration steps.

After these calculations, the agent checks if the goal state is in sight. Therefore, we used the Bresenham line algorithm [20] to calculate the states between the actual state and the goal state and compare the resulting states with the positions of walls and obstacles. If there is no obstacle in the line of sight, the agent is able to see the goal state ($g = 1$). This information is used to calculate a third frequency value $f_{goal}(t)$ which is high if the goal is regularly in sight and decaying if the goal gets out of sight. For this calculation we use the same type of equation as in Equation 2 and 3 with different variables $f_{goal}(t)$, θ_2, and g.

Finally we use these three frequencies to calculate the arousal value $A(t)$ of the agent as a weighted average given by

$$A(t) = \begin{cases} A(t) = \frac{f_{dir}(t) + f_{col}(t)}{2} & \text{if } g = 0 \\ A(t) = \frac{f_{dir}(t) + f_{col}(t) + f_{goal}(t)}{3} & \text{if } g = 1. \end{cases} \quad (4)$$

The valence value $V(t)$ is calculated using $f_{goal}(t)$ and the information if the agent is moving away from the starting point. This information is derived by calculating the distance between starting point and actual position. If this distance increases ($i = 1$) the agent assumes to be on the right way, otherwise, it assumes something is going wrong ($i = -1$). The variable i is used in a discrete dynamical system to calculate a frequency equivalent $f_{right}(t)$ (i is the limit the frequency value converges to and κ the factor determining the slope) as

$$f_{right}(t) = f_{right}(t-1) + (i - f_{right}(t-1)) \cdot \kappa. \quad (5)$$

Using these two frequencies, $f_{goal}(t)$ and $f_{right}(t)$, the valence component $V(t)$ for the actual core affect state is calculated as

$$V(t) = \frac{f_{goal}(t) + f_{right}(t)}{2}. \quad (6)$$

All these calculations are performed in the reflection level at each time step during the reinforcement learning experiment. The resulting arousal and valence values are used to affectively appraise the learning process.

V. Simulation and Results

In the following we show some results of this appraisal and explain them according to the elicited emotions.

In our experiments the agent is placed at the beginning of each episode in the start state *S* and has to find the goal state *G*. Each episode consists of a maximum of 2000 steps. Within this period the agent has to find the goal state, otherwise the episode is terminated. At the beginning of an experiment the model and Q-table is reset to the initial value of zero and the agent is placed at its starting position. All other variables and temporary tables are set to their initial values (Table I). Those initial values are depended on the episode length and the desired behavior of the agent. The slope factors ν, θ_1, and θ_2 determine how fast the agent reacts on events and how long they have an effect. After each episode, the agent's position is reset to the start state, but the already learned model and Q-tables are reused in the following episode(s). One experiment consists of several episodes, in which the agent advances its performance and reduces the steps needed to find the goal.

There are two variants of this experiment. In the first, the agent's performance is only appraised with the described core affect. In the second variant of the experiment the actual core affect is used to bias the reward function of the reinforcement learning framework. At the beginning of both experiments the reward function is set to $r_t = -1$ in each state except the goal state (at the goal the agent receives a reward of $r_t = 10$). In the second variant of the experiment, in which

TABLE I

PARAMETERS USED IN THE EXPERIMENTS

Parameter	Definition	Value
n	number of episodes	20
maxsteps	maximum number of steps per episode	2000
p_steps	number of planning steps	50
α	step-size	0.01
γ	discount-rate	0.95
ϵ	probability for random action in ϵ-greedy policy	0.1
ν	factor determining the slope of $f_{col}(t)$	0.2
θ_1, θ_2	factor determining the slope of $f_{dir}(t)$ and $f_{goal}(t)$	0.2
κ	factor determining the slope of $f_{right}(t)$	0.15

the reward function is biased by the actual affective state of the agent, the reward in each state (except the goal state) depends additionally on the core affect. In those states where the core affect is valenced positively (quadrant 1 and 2), the agent receives (or *perceives*) a reward $r_t = 0$ for the actual state. This should influence the learning behaviour in a way that the agent actively searches for states with a positively valenced core affect.

In Figure 4 we have plotted the average affective states of the agent during 20 episodes of the first variant of the reinforcement learning experiment. It should be noted that the reinforcement learning experiment is non-deterministic, therefore we had to average the affective state of each episode over several experiments. For each point, we have used the affective appraisal of 50 independent iterations of the experiment.

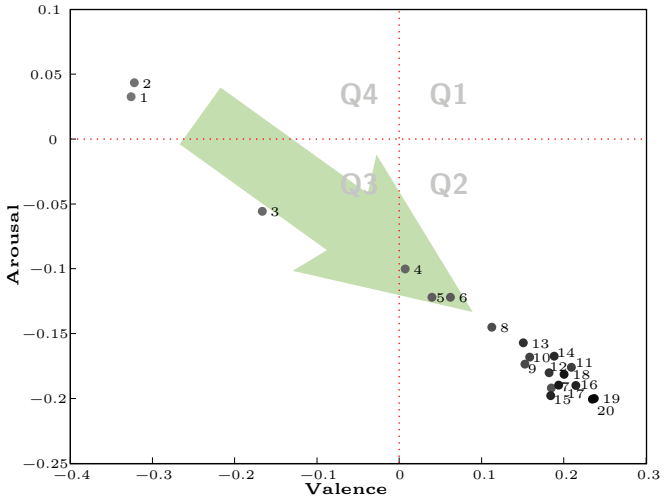

Fig. 4. Average affective states during 20 episodes of the reinforcement learning experiment.

Next to each point the episode number is plotted. The brighter the point the older the episode. There is a clear temporal movement of the core affect form quadrant four to quadrant two. During the first three episodes the agent gains knowledge about the maze. This requires a lot of steps and changes in direction resulting in a highly negative valence value and a positive arousal value, corresponding to an affective state of anxiety and fear. After about four episodes the reinforcement learning algorithm has gathered enough model knowledge about the maze enabling the agent to find a short path to the goal state. This requires less steps and directional changes and therefore the arousal is reduced and the overall feeling (valence) is positive. In the following episodes (7 to 20) the path is further optimized and the agents selects frequently the optimal action. This further reduces the arousal and the valence value increases. The result shows that the agent is able to appraise the learning progress in terms of an affective state.

Figure 5 shows the results of the second variant of the experiment which incorporates the affective state into the reinforcement learning process. Compared to the first variant of the experiment, the agent experiences a wider range and a more uniform temporal movement of affective states, which is accompanied by a slower learning progress.

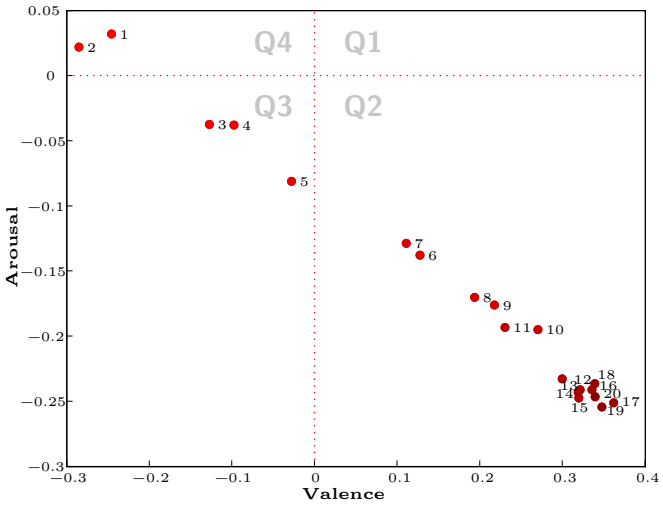

Fig. 5. Results of the second experiment which incorporates the core affect into the learning process.

The slower learning progress of the second experiment is characterized by a slower decrease of needed steps per episode as compared to the first experiment (Figure 6). On average, the learning performance of both experiments is similar.

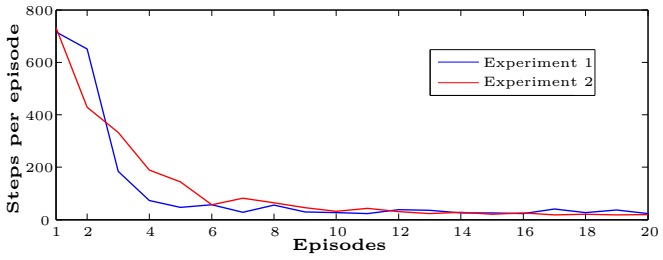

Fig. 6. Average learning curves for both experiments. Experiment 1 uses unmodified Dyna-Q-Learning and Experiment 2 incorporates the calculated affective state of the agent.

This slower learning progress is the result of the incorporation of the affective state of the agent. The agent simultaneously tries in each episode to improve the policy and the *perceived* valence of the situation. As a result,

the agent learns the optimal policy more slowly, but is able to perceive (and display) its current learning progress through its core affect. Additionally, the more diverse and uniform states enable a more believable communication of the affective states towards human observers [21].

In the first experiment the agent is only able to display sudden changes of its core affect. Whereas in the second experiment, the agent can communicate its learning progress with a more uniform transition of its affective state. A human could perceive the transition of the agent's core affect as follows: First, the agent displays fear and anxiety due to the new situation it is encountered. After these first episodes, the agent is depressed because of its bad performance. However, the improvement of the policy results in core affects of calm and satisfaction.

Overall, the results indicate that our approach for integrating affective states into reinforcement learning can be used in autonomous affective systems. The generated core affects can be used as an alternative way for communicating the actual system state towards a human observer.

VI. Conclusions and Future Work

In this paper, we present an architecture for integrating a dimensional emotional model into a virtual agent. We believe that agents or robots using affective models to guide their decisions or express their current state are highly beneficial in holding a better social interaction with humans. To prove this belief, we described a reinforcement learning framework simulating a virtual agent making decisions to find the shortest path to the goal in a maze. Further we showed how the reinforcement learning framework could be integrated into a model of affect and cognition. Finally, we extended this framework with an dimensional emotional model to calculate the core affect of the agent during the learning process. The affective appraisals of the learning progress show that the emotional model is successfully integrated into the machine learning algorithm and is able to evaluate the current state of the agent in terms of an affective state. A second experiment, which uses these affective states to guide or bias the decision behaviour of the agent was conducted. The results show that the core affect can be used to bias the reward function to guide the decisions of an agent.

However, an extensive study is necessary for a statistical prove of the system and an evaluation of its performance in different scenarios. Furthermore, in a future study we are going to use the affective state for displaying the learning progress in a more interactive way to the user and implement the whole system to a small two-wheeled robot performing the same path-finding task in a real environment.

In future implementations we will integrate free-floating moods into the system. Moods are long-term influences onto the core affective state of the system. This means that the affective state of the agent is additionally influenced through the *personality* of the system and its long term reactions on events.

Acknowledgment

This work has been supported by the German Research Foundation (DFG) funded Cluster of Excellence "CoTeSys" - Cognition for Technical Systems.

References

[1] H.-R. Pfister and G. Böhm, "The multiplicity of emotions: A framework of emotional functions in decision making," *Judgment and Decision Making*, vol. 3, no. 1, pp. 5–17, 2008.

[2] G. Loewenstein and J. Lerner, "The role of affect in decision making," in *Handbook of Affective Science*, R. Davidson, H. Goldsmith, and K. Scherer, Eds. Oxford, NY, USA: Oxford University Press, 2003, pp. 619–642.

[3] E. Peters, "The functions of affect in the construction of preferences," in *The construction of preference*, S. Lichtenstein and P. Slovic, Eds. New York, NY, US: Cambridge University Press, 2006, pp. 454–463.

[4] M. B. Moussa and N. Magnenat-Thalmann, "Toward socially responsible agents: integrating attachment and learning in emotional decision-making," *Computer Animation and Virtual Worlds*, vol. 24, no. 3–4, pp. 327–334, 2013.

[5] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.

[6] C. J. Watkins, "Learning from delayed rewards," PhD thesis, Cambridge University, Cambridge, England, 1989.

[7] D. A. Norman, A. Ortony, and D. M. Russell, "Affect and machine design: Lessons for the development of autonomous machines," *IBM Systems Journal*, vol. 42, no. 1, pp. 38–44, 2003.

[8] P. Ekman, "Are there basic emotions?" *Psychological Review*, vol. 99, no. 3, pp. 550–553, 1992.

[9] J. Panksepp, *Affective neuroscience: The foundations of human and animal emotions*. New York, NY, USA: Oxford University Press, 1998.

[10] A. Mehrabian and J. A. Russell, *An approach to environmental psychology*. Cambridge, MA, USA: The MIT Press, 1974.

[11] J. A. Russell, M. Lewicka, and T. Niit, "A cross-cultural study of a circumplex model of affect," *Journal of Personality and Social Psychology*, vol. 57, no. 5, pp. 848–856, 1989.

[12] D. Watson, D. Wiese, J. Vaidya, and A. Tellegen, "The two general activation systems of affect: Structural findings, evolutionary considerations, and psychobiological evidence." *Journal of Personality and Social Psychology*, vol. 76, no. 5, pp. 820–838, 1999.

[13] C. E. Izard, "Basic emotions, natural kinds, emotion schemas, and a new paradigm," *Perspectives on Psychological Science*, vol. 2, no. 3, pp. 260–280, 2007.

[14] J. Panksepp, "Neurologizing the psychology of affects: How appraisal-based constructivism and basic emotion theory can coexist," *Perspectives on Psychological Science*, vol. 2, no. 3, pp. 281–296, 2007.

[15] A. Mehrabian, "Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament," *Current Psychology*, vol. 14, no. 4, pp. 261–292, 1996.

[16] J. A. Russell and L. F. Barrett, "Core affect, prototypical emotional episodes, and other things called emotion: dissecting the elephant." *Journal of Personality and Social Psychology*, vol. 76, no. 5, pp. 805–819, 1999.

[17] P. Gebhard, "Alma: a layered model of affect," in *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*. ACM, 2005, pp. 29–36.

[18] M. Mendl, O. H. P. Burman, and E. S. Paul, "An integrative and functional framework for the study of animal emotion and mood," *Proceedings of the Royal Society B: Biological Sciences*, vol. 277, no. 1696, pp. 2895–2904, 2010.

[19] K. R. Scherer, "Emotion and emotional competence: conceptual and theoretical issues for modelling agents," in *Blueprint for Affective Computing: A Sourcebook*, K. R. Scherer, T. Bänziger, and R. Etienne, Eds. New York, USA: Oxford University Press, 2010, pp. 3–20.

[20] J. E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Systems Journal*, vol. 4, no. 1, pp. 25–30, 1965.

[21] F. D. Schönbrodt and J. B. Asendorpf, "The challenge of constructing psychologically believable agents," *Journal of Media Psychology: Theories, Methods, and Applications*, vol. 23, no. 2, pp. 100–107, 2011.