# Emotional Evaluation of Bandit Problems

Johannes Feldmaier, Klaus Diepold
Institute for Data Processing
Technische Universität München
Munich, Germany
{johannes.feldmaier, kldi}@tum.de

***Abstract*— In this paper, we discuss an approach to evaluate decisions made during a multi-armed bandit learning experiment. Usually, the results of machine learning algorithms applied on multi-armed bandit scenarios are rated in terms of earned reward and optimal decisions taken. These criteria are valuable for objective comparison in finite experiments. But learning algorithms used in real scenarios, for example in robotics, need to have instantaneous criteria to evaluate their actual decisions taken. [1] To overcome this problem, in our approach each decision updates the Zürich model which emulates the human sense of feeling secure and aroused. Combining these two feelings results in an emotional evaluation of decision policies and could be used to model the emotional state of an intelligent agent.**

## I. Introduction

Cognitive systems are often biologically inspired. Cognition means the mental process to acquire knowledge and understanding through observations, experiences, and sensing the environment. One aspect of cognition are emotions. They are triggered and in turn influenced by the environment through cognitive processes.

This work concentrates on emotion aware systems and presents one way to model psychologically correct two basic feelings and express them in a high-level manner as human emotions. It could be used to design inner mental states for future generations of robots or other cognitive systems.

Up to now, most systems try to model emotional states of surrounding people or robots instead of having an own internal system of emotions and feelings only depending on their own observations and decisions. Being equipped with an own personality and will is considered to be one of the most important aspect of human-like behaviour of artificial agents. In future, it is getting more important to make machines more believable so that the user is able to trust them [2].

Another reason for turning machines with artificial intelligence into intelligent emotion aware systems is uncertainty. There are two categories of uncertainty - aleatory and epistemic uncertainty. The second, epistemic uncertainty is the result "of the human's lack of knowledge" [3] of the surrounding environment. Currently, the measurement of uncertainty is done using Bayesian probabilities, fuzzy sets, fuzzy logic, possibility theory, belief functions and many more [4], [5]. Humans usually deal with uncertainty through relying on their experiences - commonly known as gut feeling or intuition - but machines cannot [6].

## II. Evaluation of Multi-armed Bandit Problems

In our scenarios, an artificial agent is confronted with a multi-armed bandit decision problem. We have chosen the multi-armed bandit problems as a representative example of many decision problems. Multi-armed bandit problems model decisions in a large variety of research topics such as online advertising, news article selection, network routing, and medicinal trials, to name a few.

Currently, the decisions made by learning algorithms on bandit problems are commonly rated in terms of the cumulated reward, rate of optimal decisions, and regret. Usually, these values are calculated after each experiment. In our work, we present a way to evaluate actual decisions in a biologically inspired way. Therefore, we have implemented a psychological correct model which states the two feelings of security and arousal for an artificial agent. The model relies on the three input features for each object. These features are relevancy, familiarity and psychological distance. We use eligibility traces to calculate the relevancy and familiarity of the bandit arms.

A combination of the two resulting values for security and arousal provides a natural and more convenient way to evaluate decision processes or rate decision uncertainty.

## III. Related Work

The research issue of artificial emotions and feelings can be divided into two parts: on the one hand, techniques used for emotion recognition, and on the other hand, those techniques modelling (and expressing) artificial emotions and feelings.

Emotion recognition is an important aspect for affective systems to recognize the mood of the user and adapt its behaviour. Two great examples can be found in [7] and [8]. In case of robots, the emotion dependant adaptation is used to express artificial emotions improving the social behaviour and acceptance of the robot [9], [10]. There are many ways of reacting to recognized emotions and therefore there exists many ways to implement a correct recognition system. The difficulty lays in the correct combination of the feature acquisition and psychological modelling of human emotions.

Besides emotion recognition, the second important aspect of artificial emotions is the design of emotional models [11], [12]. These models can be integrated into artificial agents (e.g. robots) or computer games. Normally, those agents

or artificial intelligences try to imitate human behaviour according to fuzzy logics, self organizing maps or other psychologically or biologically inspired models.

The mentioned systems are focused on robotics and social behaviour of distributed systems. To the best of our knowledge, there exist no concepts for modelling emotion aware systems based on a fixed psychological model which utilizes only parameters related to decisions. Fixed means in this case that the model works without any training. So, the main goal of our work is to conduct an internal evaluation of decisions in terms of emotions or feelings. This is our attempt to give artificial agents the ability to experience something like the well-known gut feelings of humans.

## IV. Multi-armed Bandits

One-armed bandits are also known as slot machines and can be found in almost all casinos around the world. Those slot machines pay a reward from an unknown probability distribution at each play. If a player is confronted with a row of slot machines and has to decide which machine he likes to play, then he is confronted with a multi-armed bandit problem. The multi-armed bandit (MAB) research topic goes back until 1952 when Herbert Robbins considered his clinical trials as a bandit problem with more than two arms [13].

In a MAB experiment several trials are played on the same bandit machine. The bandit machine consists of more than one arm. Each arm has a specific probability distribution which determines the success probability for winning. At each trial the agent tries to maximize its reward by selecting the arm with the highest chance of success. During a game, the agent tries to find out which arm has the highest success probability and then will keep playing this arm to maximize its reward. To find out the right arm, the agent first has to check out each arm several times. This is called exploration. Always playing the arm with the supposedly highest success probability is called exploitation. Both together is called the exploration/exploitation trade off, which is an important research problem in the machine learning area [14].

The success of each play is evaluated in terms of cumulative reward and the rate of optimal decisions. The performance of an agent (or learning algorithm) is evaluated in the long run by calculating the mean values of the cumulative reward and optimal decision rate over several plays with several trials. These mean values are also used for comparison among different algorithms.

Besides many different learning algorithms for multi-armed bandit problems, there are also many variations of the MAB problem. Bernoulli, Exponential or Poisson probability distributions are used as reward structures.

For our experiments we have chosen the class of Bernoulli distributed bandit problems. This class represents decisions with only two possibilities - success or failure. Both cases with a distinct probability. The extension to multiple Bernoulli distributed experiments (MAB problem) corresponds to a situation where each move can be successful, but only one move has the highest success probability to win. An efficient implementation of the Bernoulli multi-armed bandit problem and several learning algorithms was done by Olivier Cappé et al. [15]. We use this implementation to conduct our experiments generating the decision policies and reward processes.

## V. Zürich Model

The German psychologist Norbert Bischof proposed his "Zürich Model of Social Motivation" in 1975 [16], [17]. It is the result of his research in the fields of ethology and evolutionary theory. Basically, it consists of three negative feedback loops connected to so called detectors delivering the actual state. This actual state is compared to reference values resulting in impulses (momenta) compensating these discrepancies.

The Zürich Model of Social Motivation can be considered as one of the most applicable psychological models of social motivation, since Bischof describes his model in a psychological manner and in a systems theoretic way.

In the following, we describe Bischof's model, but restrict it to two basic feelings - security and arousal - omitting the third, the autonomy claim which is not relevant to our scenario. Because, the autonomy claim model would influence both the arousal and the security system [18]. Based on empirical studies, the model describes the behaviour and actions performed by a children in the present of surrounding objects. These objects could either be things like an ordinary ball or other humans. The recognition and classification of these objects thereby is not a part of the original model [18]. Instead, so called detectors were assumed to assign two values - relevancy $R_i$ and familiarity $F_i$ - to each object $i$. In addition to these two values each object has a position $z_i$. The complete Zürich model is drawn in Figure 1.

It can be divided into three coupled subsystems, namely the security system, the arousal system, and the detectors.

### *Detectors*

In the present bandit scenario we have implemented two detectors ($Det_F$ and $Det_R$) observing the bandit process. The first detector calculates a familiarity value $F_i$ of each bandit arm $i$ and the second determines the relevancy $R_i$ of each arm. Both familiarity and relevancy are calculated using eligibility traces. The basic idea behind eligibility traces is that each occurrence of an event triggers a short-term memory process which gradually fades out afterwards [19]. If each new event accumulates to the existing trace, we call this the accumulating trace defined by

$$e_{t+1}(s) = \begin{cases} \lambda\gamma e_t(s) & \text{if } s \neq s_t \\ \lambda\gamma e_t(s) + 1 & \text{if } s = s_t \end{cases},$$

where $e_t(s)$ models the memory process and $s_t$ the actual state $s$ at time $t$. Each time $s_t$ equals to a specific state $s$ it is added 1 to the actual state.

The familiarity detector calculates for each arm an eligibility trace by adding one to the trace of the actual selected arm. Then all arms are multiplied by the decaying factors $\lambda$ and $\gamma$. Whereas, the relevancy detector always adds the actual reward to each arm (could either be zero or one) before the

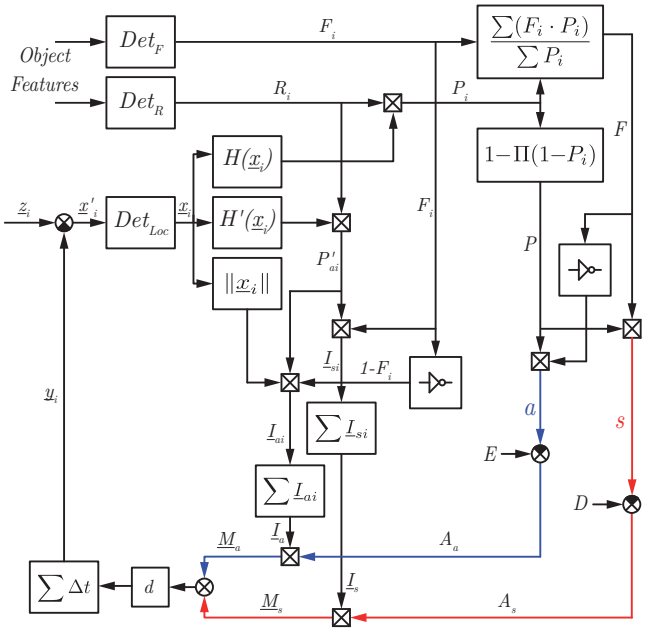


Fig. 1. Realization of the Zürich Model, the security subsystem is drawn in red and the arousal subsystem is drawn in blue

multiplication with the factors is performed. This results in high familiarity values for regularly selected arms and a high relevancy for familiar arms with high rewards.

*Security System*

The feeling of security is defined as the warmth, protection, and intimacy someone feels for example at home surrounded by a familiar person [20]. As already mentioned, the subsystems use two features and the position of objects to calculate their outputs. In a first step, the security system calculates a potency value $P_i$ from the familiarity $F_i$ and the distance value $\underline{x}_i$. The distance value in the actual implementation is determined by the Euclidean distance between the two points in a two dimensional space multiplied by a hyperbolic function $H(\underline{x}_i)$. This function is defined by

$$H(\underline{x}_i) = \begin{cases} \frac{r(x_{max} - |\underline{x}_i|)}{r(x_{max} - |\underline{x}_i|) + |\underline{x}_i| x_{max}} & \text{if } |\underline{x}_i| < x_{max} \\ 0 & \text{otherwise} \end{cases},$$

where $r$ determines the slope of decay of the hyperbolic function and $x_{max}$ the distance where it vanishes to zero. The function is used to take possible additional psychological effects influencing the purely geometric distance into account. In the actual bandit scenario, we have given each arm a fixed position, only the agent itself is able to move.

The potency value $P_i$ of each object is multiplicatively combined to the joint potency

$$P = -1 \prod_{i=1}^{n} (1 - P_i).$$

The joint potency is then multiplied by the joint familiarity $F$ of all surrounding objects $n$ by

$$F = \frac{\sum_{i=1}^{n} P_i F_i}{\sum_{i=1}^{n} P_i}$$

resulting in the security value $s = P \cdot F$. Finally, the security subsystem (drawn in red) compares the security value with the dependency $D$, a reference value determining how dependent a subject (agent) is. The result is the activation component $A_s$ of the security system.

*Arousal System*

The feeling of arousal is triggered in new and uncertain situations and can be further defined with feelings "such as interest, fascination, curiosity, as well as feelings of alarm or fear" [20]. In the Zürich model the arousal value $a$ is calculated by multiplying the inverse joint familiarity with the joint potency value:

$$a = P \cdot (1 - F).$$

The reference value of the arousal system is labelled with enterprise $E$ and denotes how initiative the subject (agent) behaves in unknown and changing environments. The arousal value and reference value is compared through subtraction resulting in the activation component $A_a$ of the arousal subsystem (drawn in blue).

*Momentum Vectors and Distance Regulation*

The activation components are used to express appetence (positive $A$ values) while negative $A$ values express aversion [18]. This means, if the actual security or arousal values fall below the corresponding reference value resulting in positive activation components, the agent enters a state called appetence. In this state, it tries to increase its feeling of security or decrease its degree of arousal through adjusting its distance to objects more familiar and relevant.

This is done by calculating a direction and magnitude of a vector superimposing so called incentive vectors $I_a$ and $I_s$. Therefore, the location, the potency value $P'_{ai}$ and the degree of familiarity $F_i$ of each object around the agent is combined. Based on the agent's location, the vectors pointing from each object to the agent's position are weighted either by the potency values $P'_{ai}$ or the inverse familiarity values $F_i$ resulting in the incentive vectors $I_a$ and $I_s$. Superimposing them would result in a single vector pointing in a direction of an area with either a higher value of security or less arousal.

Multiplying these incentive vectors with their corresponding activation components results in so called momentum vectors [18], weighted versions of the incentive vectors. The weighting with the activation component combines the demand (appentence) or reluctance (aversion) of the agent with directions of objects spending security or irritating it.

At each time step, both, the momentum vector of the security subsystem $M_s$ and the one of the arousal subsystem $M_a$ are superimposed and damped by a damping factor $d$. This damping improves the stability and smooths the movement. The result is then added to the actual agent position. This creates a self-regulating feedback loop, altering all preceding

values of the last time step, converging to a position with the highest security value and lowest arousal value.

## VI. Emotional Evaluation

In this section we describe the connection between the Zürich model and the bandit scenario in order to evaluate the decision process internally with emotions. Our objective is to give an agent the ability of evaluating an arbitrary decision process using a psychological model which does not require huge adaptations to new problems. The Zürich model fulfils this property because its adaptations to a new problem affect only the detectors. In order to keep the detectors as simple as possible we choose the multi-armed bandit scenario with only one observable variable and one possible action in each round. These two variables are the interface between the Zürich model and the bandit process. The bandit experiment is independent from the Zürich model. The emotions generated by the model are not used in the decision or learning process of the bandit. Such a combination is planned for future experiments.

Under these conditions, we conduct the bandit simulations or learning experiments generating decision processes and reward curves. This results in data streams with pairs of the decision and the gained reward. These pairs are presented one by one to the Zürich model which generates for each trial a value for the feeling of security and arousal. Then, for a better visualization we combine these values in relation to each other and plot them as human emotions.

### *Experiments*

We conduct two different bandit experiments. The first one is a complete simulation of a bandit experiment in which we predetermine the learning curve and decision policy. In the second experiment, we use a bandit simulation and a learning algorithm generating the decisions.

In the simulation, we perform 100 plays each with 1500 trials. The structure of each play is divided into five phases. The first phase represents a state in which the agent is completely untrained and does not have any knowledge about the bandits. All decisions are made randomly. After 300 trials, the agent enters the second phase. In this phase, the simulation selects randomly the two best arms resulting in higher rewards. This phase represents a state in which first training success is made by the agent. Again, after 300 trials the agent enters phase three in which the training is finished, the decisions are optimal and it selects only the best arm. After this exploitation phase, the agent enters phase four at trial 900. In this phase, a hidden disturbance in the bandit process causes that the agent is only able to select the best arm with a probability of ten percent. In the last phase, the bandit process recovers and the agent is again able to ideally select the best arm for getting the highest rewards.

We designed these phases, so that the agent is faced with five different situations. Each situation can occur during an experiment using machine learning algorithms (as used in the second experiment). But it could be difficult to clearly observe each individual phase in those experiments. Therefore, we use the simulation as described above, to present how the Zürich model evaluates predetermined situations.

The second experiment is then used to show how the Zürich model evaluates decisions made by a machine learning algorithm. In this experiment, we perform also 100 plays each with 300 trials, but now using a standard implementation of a Bernoulli bandit process and an agent autonomously learning its decisions. In order to investigate the agent's behaviour in a learning and relearning scenario we modify the bandit scenario after trial 200 and 300. The modification consists in exchanging the best arm, so that the learning algorithm has to stop exploiting the best arm in order to explore a new strategy.

During the bandit experiments, the values calculated by the Zürich model do not influence the learning algorithm.

### *Implementation*

Both, the bandit processes and the Zürich model are implemented using Matlab. A good framework we use is called *pymaBandits* [15]. It already includes several learning policies like the Gittin's index, the classical UCB policy and some variations of it, the MOSS policy and some others.

We chose the Gittins index policy for learning the decisions in our experiments, because this policy is well known and is proven to be optimal [21]. We applied it on a four-armed Bernoulli bandit experiment implemented by the existing functions of the *pymaBandit* framework.

The interface of the Zürich model consists of a function which is called every time a new pair of decision and reward is available. The current state of the model is saved in an object-orientated data format. In case of the above mentioned experiments we iteratively present the bandit decision and reward pairs to the model, which updates the objects. There exist two different classes, one representing the agent (ego) containing its position, the reference values and the actual security and arousal values. The second class models the objects around the agent, like the bandit arms and contains values for the familiarity and relevancy of the object and its position.

We took care to design the model and the experiments with as few parameters as possible. The Zürich model only needs its reference values $D$ and $E$, the detectors are parametrized with a decay parameter $\lambda$ and a discount-rate $\gamma$. For the bandit experiments, we had to set the number of arms, trials and plays, as well as the success probability $\mu_i$ for each bandit arm. We summarized these values in Table I.

A dependency value $D$ of $0.75$ and an enterprise value $E$ of $0.8$ corresponds to an individual, highly dependent on sources of security and simultaneously receptive to changing situations. We set the product of the decay and discount-rate to $0.95$, which results in a slow decay of the eligibility traces. The success probabilities of the bandit arms were selected arbitrarily, except, that one arm has a clear maximum.

### *Results*

As described in Section V, the Zürich model delivers values representing the degree of arousal and security of an

TABLE I

PARAMETER SETTING USED FOR THE ZÜRICH MODEL, THE DETECTORS AND THE EXPERIMENTS

| Zürich Model | Detectors | Bandit experiments |
|---|---|---|
| $D = 0.75$ | $\lambda \cdot \gamma = 0.95$ | $\mu_1 = 0.1$ |
| $E = 0.8$ | | $\mu_2 = 0.3$ |
| | | $\mu_3 = 0.2$ |
| | | $\mu_4 = 0.8$ |

agent. The arousal value corresponds to the human feeling of getting excited or being externally stimulated. The security value quantifies the level of how secure the agent feels. Both values have to be interpreted together, because their meanings change relatively to the higher one of the two. Therefore, we have implemented a function comparing the security and arousal value, presenting them as emotions. The exact function for each emotion is provided in Table II. But this abstraction depicts only one example of how the results of the Zürich model could be interpreted. Currently, these combinations are not based on any psychological model.

TABLE II

COMBINATORIAL RULES DETERMINING WHICH EMOTION IS TRIGGERED (A: AROUSAL, S: SECURITY)

| | Emotion | Rule |
|---|---|---|
| ● | uncertainty | $a < 0.15 \wedge s < 0.15$ |
| ● | aversion | $\nabla s < 0 \wedge \nabla a > 0$ |
| ● | anger | $(a > 0.85 \wedge s < 0.15) \vee \ldots$ $(\lvert\nabla a\rvert > 0.0075 \wedge s < 0.3)$ |
| ● | fear | $\nabla a > 0 \wedge \nabla s < 0 \wedge a > s$ |
| ● | anticipation | $\nabla a < 0 \wedge \nabla s > 0 \wedge a < s$ |
| ● | joy | $a < s \wedge \nabla a \leq 0$ |
| ● | trust | $a < s \wedge \lvert\nabla a\rvert < 0.0005$ |

In Table II the $\nabla$ operator calculates the gradient over the last five values. An emotion is outputted for each instance of time only if the corresponding rule is evaluated to be true.

Figure 2 shows the resulting security and arousal values, as well as the overlaid emotions for the simulated bandit process. The five simulation phases are separated with dotted lines. The first phase is characterized by uncertainty at the beginning followed by emotions like aversion and fear. This corresponds to the expected result in this phase. In the second phase the amount of emotions like anticipation, joy, and trust increases but is still broken by feelings of aversion. In this phase our interpretation using the combinatorial logic above lacks and needs still some improvements. The Zürich model delivers good values, i.e. the increased arousal value is an result of the changed situation and the increasing security value is a consequence of the higher rewards. Then, in phase three, the ideal case, the agent only shows emotions like anticipation, joy, and trust. The internal disturbance of the bandit process in phase four causes emotions like aversion and fear, but no uncertainty due to the remaining fraction of good decisions. Which fits to our expectations. In the last phase, the decision process recovers and the agent regains its trust. The security and arousal values also recover and show expected values.

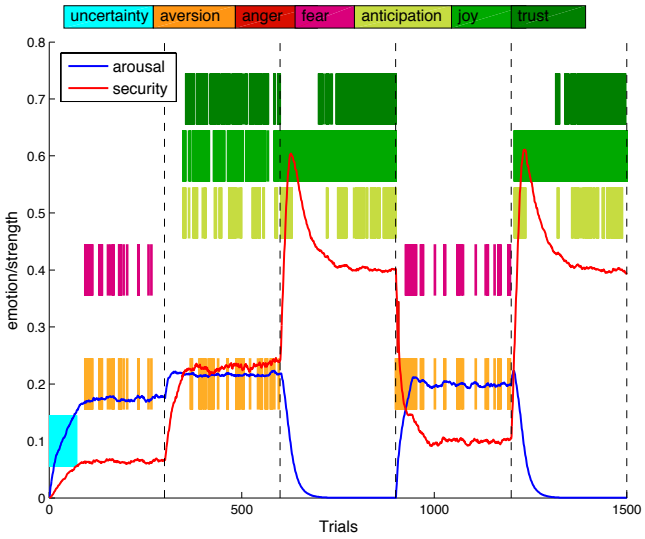


Fig. 2. Plot of the security and arousal feeling for the simulation experiment with overlayed emotions according to Table II

Applying the Zürich model on the machine learning experiment results in Figure 3. As stated above, we altered the bandit process two times while an agent learns the optimal decision policy using the Gittins index. So, there are three phases in which learning and relearning occurs. We expect for the learning phase increasing security values and decreasing arousal values and corresponding emotions. The relearning phase should invert this upto the point when the new policy is learned.

In the first phase, the agent learns for the first time and tries to find an optimal policy. The Gittins index policy discovers the best arm fast, and then stops exploring. During this initial exploring, the agent feels uncertain. Having found the best arm, results in high rewards, so the agent begins to feel joy and trust. The model works as expected. Then, at trial 100 the arm configuration changes and the agent has to relearn. Right after the change, the agent does not like its decisions resulting in the emotion of aversion followed by a phase of uncertainty, during which it discovers the new optimal choice. Then, it shows emotions like anticipation and joy again. Changing the configuration for the second time, again results in phases characterized by emotions of aversion and uncertainty. But now the relearning process lasts remarkably longer because the Gittins index policy is not designed to adapt regularly to new arm configurations. But, the Zürich model still delivers interesting results. Although the policy does not deliver optimal decisions, the agent shows emotions of anticipation, joy and trust, which is a result of the previous experiences.

The results show, that the Zürich model is able to evaluate decisions of an artificial agent based on psychological principles. We see, that the model produces values representing the feeling of security and arousal corresponding to situations

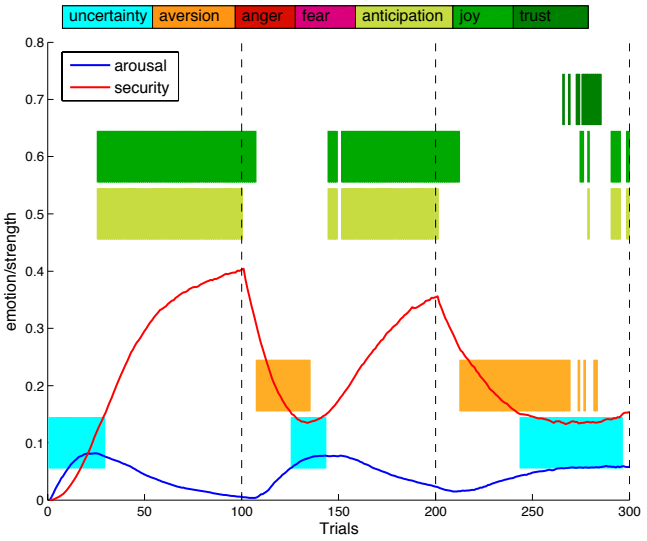


Fig. 3. Emotional evaluation of a multi-armed bandit problem using the Gittins index policy

with uncertainty and learning progress.

## VII. Conclusions and Future Work

This paper began by dividing the research topic of artificial emotions into two parts and then considering the modelling aspect of this research area. We presented the Zürich model which models the human feelings of security and arousal. As yet, machine learning algorithms were evaluated by objective figures like reward or regret, but in future emotion aware systems an emotional evaluation of decisions made by the artificial intelligence are considered to be important. The results show that the model indeed can be used for the evaluation of a machine learning process extending the common objective figures with values representing feelings.

At the same time, the emotional evaluation could be an helpful indicator for uncertainty which is often required by machine learning algorithms (e.g. reinforcement learning). Another reason for modelling feelings and emotions based on decisions instead of user feedback is to improve the behaviour of artificial agents and make them more believable [2].

Future work will concern the implementation of an autonomy system as suggested by Bischof, giving the agent the feeling of being competent and respected by others. Furthermore, it will be interesting to apply the model on further scenarios and combine it with reinforcement learning.

## Acknowledgement

The authors would like to thank M. Zehetleitner for the great and helpful introduction into the Zürich model and his support.

## References

[1] "PDCA12-70 data sheet," Opto Speed SA, Mezzovico, Switzerland.
[2] 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.
[3] Y. Li, J. Chen, and L. Feng, "Dealing with uncertainty: A survey of theories and practices," *IEEE Transactions on Knowledge and Data Engineering*, vol. preprint, 2012.
[4] P. Walley, "Measures of uncertainty in expert systems," *Artificial Intelligence*, vol. 83, no. 1, pp. 1 – 58, 1996.
[5] A. T. Berztiss, *Handbook of Software Engineering & Knowledge Engineering*. Singapore: World Scientific Pub. Co., 2002, vol. 2, ch. Uncertainty management.
[6] H. Dreyfus, S. Dreyfus, and T. Athanasiou, *Mind over machine: the power of human intuition and expertise in the era of the computer*. New York: Free Press, 1988.
[7] R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz, and J. Taylor, "Emotion recognition in human-computer interaction," *IEEE Signal Processing Magazine*, vol. 18, no. 1, pp. 32–80, 2001.
[8] J. Tao, T. Tan, and R. Picard, *Affective Computing and Intelligent Interaction*. Berlin/Heidelberg: Springer, 2005.
[9] C. Breazeal, "Emotion and sociable humanoid robots," *International Journal of Human-Computer Studies*, vol. 59, no. 1, pp. 119–155, 2003.
[10] K. Suzuki, A. Camurri, P. Ferrentino, and S. Hashimoto, "Intelligent agent system for human-robot interaction through artificial emotion," in *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 2, 1998, pp. 1055–1060.
[11] D. Cañamero, *Designing emotions for activity selection in autonomous agents*. Cambridge, MA: MIT Press, 2003, vol. 115, pp. 115–148.
[12] M. Salichs and M. Malfaz, "A new approach to modeling emotions and their use on a decision-making system for artificial agents," *IEEE Transactions on Affective Computing*, vol. 3, no. 1, pp. 56 –68, 2012.
[13] H. Robbins, "Some aspects of the sequential design of experiments," *Bulletin of the American Mathematical Society*, vol. 58, no. 5, pp. 527–535, 1952.
[14] W. Macready, D. Wolpert, *et al.*, "Bandit problems and the exploration/exploitation tradeoff," *IEEE Transactions on Evolutionary Computation*, vol. 2, no. 1, pp. 2–22, 1998.
[15] O. Cappé, A. Garivier, and E. Kaufmann, "pymabandits," http://mloss.org/software/view/415/, 2012, [Online; accessed February 4, 2013].
[16] N. Bischof, *Das Rätsel Ödipus*. München: Piper, 1989.
[17] ——, "A systems approach toward the functional connections of attachment and fear," *Child Development*, vol. 46, no. 4, pp. 801–817, 1975.
[18] M. E. Lamb and H. Keller, Eds., *Infant Development: Perspectives from German Speaking Countries*. Taylor & Francis Group, 1991, ch. 3 A Systems Theory Perspective.
[19] S. Singh and R. Sutton, "Reinforcement learning with replacing eligibility traces," *Recent Advances in Reinforcement Learning*, pp. 123–158, 1996.
[20] M. E. Schneider, "Systems theory of motivational development," in *International Encyclopedia of the Social & Behavioral Sciences*, N. J. Smelser and P. B. Baltes, Eds. Pergamon, 2001, pp. 10 120 – 10 125.
[21] J. C. Gittins, "Bandit processes and dynamic allocation indices," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 41, no. 2, pp. 148–177, 1979.