Effect of emotional feedback in a decision-making system for an autonomous agent

Javier G. Rázuri[1], David Sundgren[1], Rahim Rahmani[1], Aron Larsson[1] and Antonio Moran Cardenas[2]

[1] Dept. of Computer and Systems Sciences, Stockholm University
Stockholm, SE-164 40, Sweden

[2] Pontifical Catholic University of Perú PUCP,
Lima, Lima 32, Perú

Abstract

The point of view of Isaac Asimov is unlikely in a close future, but machines that develop tasks in a sensible manner are already a fact. In light of this remark, recent research tries to understand the requirements and design options that imply providing an autonomous agent with means for detecting emotions. If we think about of exporting this model to machines, it is possible that they become capable to evolve emotionally according to such models and would take part in the society more or less cooperatively, according to the perceived emotional state. The main purpose of this research is the implementation of a decision model affected by emotional feedback in a cognitive robotic assistant that can capture information about the world around it. The robot will use multi-modal communication to assist the societal participation of persons deprived of conventional modes of communication. The aim is a machine that can predict what the user will do next and be ready to give the best possible assistance, taking in account the emotional factor. The results indicate the benefits and importance of emotional feedback in the closed loop human-robot interaction framework. Cognitive agents are shown to be capable of adapting to emotional information from humans.

Keywords: *Detection of Emotional Information, Affective Computing, Facial Expression Recognition, Artificial Neural Network, Adversarial Risk Analysis, Broaden and Build Theory*

1. Introduction

Over the years machines have incorporated emotional acquisition into their architectures [1], [2], [3], [4], [5], which makes it possible to solve tasks that go further than performing accurate and quick calculations with rational, rigorous and logical behavior. Looking forward, humans might be partakers of a new Human-Machine Interaction with entities that can develop human emotions, such as kindness, loyalty, friendship and even love. It is possible to think, that future generations of machines have some skills to understand human emotions and these emotions can serve as a guide for their actions. The new machines could have some kind of functionalities to respond more flexibly, foretelling and adjusting to what humans want.

In the present article, we address the problem of designing a model which will support making decisions affected by human's emotions in an autonomous agent, capable to interact with an individual. We thus aim at supporting the decision making of an autonomous agent so as to improve the human machine interaction, that it still is far from fluent [6].

We tackle the problem with the proposal of combining two models: an emotional model and a decision-making model from an agent. The combination will give rise to a new evolution from agent's behavior and that might experience close affinities to humans. In light of this remark, the decision agent will be able to make decisions influenced by the emotional information conveyed through the human face, using frameworks of Adversarial Risk Analysis (ARA) [7] and neural network based methods in facial emotion recognition [8], [9]. We illustrate the implementation of the model with a robot endowed with several sensors to infer the user's actions and environment's states. We include some computational experience implemented in MATLAB.

This paper is organized as follows. Section 2 describes the general scenario where the interactions between the agent and human are performed. Subsections 2.1 and 2.2 describe the proposed models that allow an agent to make choices and to capture human's emotions. Section 3 describes a rule-based design of emotional feedback for self-regulation of positive behavior. Section 4 describes the experimental results of simulations examining the effects of emotional feedback in the behavioral model of the autonomous agent are shown. Discussion and future work are presented in Section 5.

2. Interaction model

As we can see in Fig. 1, within the scenario the interaction between an agent and human is being carried out. The arrows show the direction of the input and output data that the agent will use. The emotional input will comprise the current emotion provided by the facial expression. The agent will behave taking into account the information of the human adversary and the information of the environment in which the agent evolves. The next action selected by the agent will be affected by the perceived emotion of the adversary. This will change the behavior of the agent in a manner that can be suitable for a future learning process in a social context. Part of the model is essentially multiattribute decision analytic, [10], [11], but the agent also manages forecasts models of the evolution of its adversary taking into account the emotional feedback and the environment surrounding all of them.
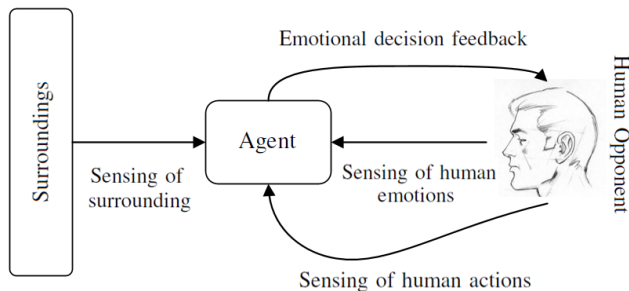

Fig. 1 Interaction general model

2.1 Emotion Detection Model

The agent constantly stores the information provided by the opponent's face, this information will be the emotional feedback input that will support the final decision. The model is based on two modules: Facial Image Treatment and an ANN propagation algorithm to recognize facial expressions. Basically, this facial emotion detection loop starts when the agent recognizes the face of opponent, it passes through a series of phases, turning it into a new image (merged image), that will be interpreted by the ANN algorithm. The ANN has trained previously with a series of images that the agent has stored in the memory. The group of images are the training set related with six different facial expressions (six for each individual), with the same treatment as the input image. Finally, once the group to which the image belongs is found, the system reports about the emotional state of the opponent's face, as you can see in Fig. 2.
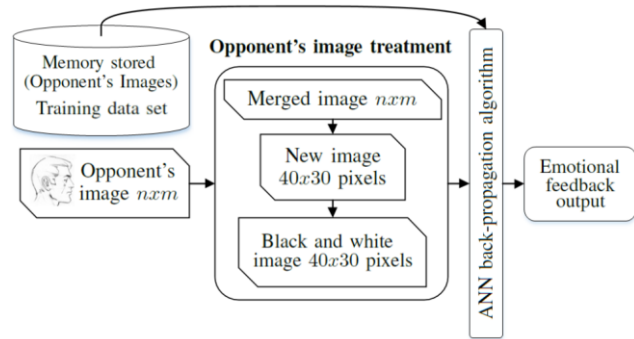

Fig. 2 Emotion Detection Model

An example of the total process of the facial image treatment can be seen in Fig. 3. The algorithm works with the complete zone of the eyes and the mouth, for the sole purpose of merging the two extractions into a single new image.
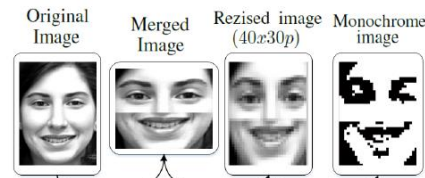

Fig. 3 Facial Image Treatment

The resizing of the merged image is covered by the Nearest Neighbor Interpolation method, it requires less computation, using a nearest neighbor's pixel to fill interpolated point. For the new matrix, a new value is calculated from a neighborhood of samples and replaces these values in the minimized image; this technique is applied to all the images in the training set of the Artificial Neural Network. It is important to apply some type of binarization over the images. For this, we apply a simple formula related to the threshold from the pixel,

$$V_{n_{pix}} = \begin{cases} 0, & \text{if } V_{c_{pix}} < 230 \\ |\frac{V_{c_{pix}}-255}{255}|, & \text{otherwise,} \end{cases} \qquad (1)$$

The algorithm works with an input layer containing 1200 neurons that are the pixels of 40 x 30 image's size. The merged image is constructed based on a part of the forehead and the mouth zone. The input variables x_i (each pixel) is multiplied by a weight w_i and are added after. We used a neuron bias b in the initialization process. Now we have the neuron's activation z. So,

$$z = \sum_{i=1}^{p} w_i x_i + b = \begin{bmatrix} W^T & b \end{bmatrix} \begin{bmatrix} X \\ 1 \end{bmatrix} \qquad (2)$$

IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 5, No 2, September 2014
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org

182

The neuron bias b works the same as adding an additional input with weight b and value 1, so we have,

$$z = \sum_{i=1}^{p} w_i x_i = W^T X \qquad (3)$$

Having the neuron's activation z, it is fed into the activation function $\sigma(z)$. This function returns a value on the interval [0, 1]. The activation function that we used is the sigmoidal function, Thus, we shall assume that

$$\sigma(z) = \begin{cases} 0, & \text{for } z < 0 \\ 1, & \text{for } z \geq 0 \end{cases} \quad \text{and} \quad \sigma(z) = \frac{1}{1 + e^{-sz}} \qquad (4)$$

The parameter z determines the steep grade of the sigmoid function. If z tends towards to 1, the threshold function is again obtained but in our case, we choose $s = 1$ that will define the output of the activation as the output of the neuron. The number of neurons in the hidden layer depends on the number of training samples. In order to fix the number of hidden neurons in the hidden layer, we will use the approach developed in [12],

$$N_h = \frac{(N_{in}) + \sqrt{N_P}}{L}$$

where L is the number of hidden layers, N_{in} is the number of input neurons and N_p is the number of input samples. The output layer includes six nodes that will be the output variables. The number of neurons is fixed as the number of emotions that we are going to consider (anger, disgust, surprise, happiness, sadness and fear) into six combinations per opponent. Fig. 4 shows a part of the images from Cohn-Kanade database that belong to the training data set. In [13] results of the algorithm are shown, along with the structure of the Articial Neural Network and the percentage of accuracy from the system.

Fig. 4 Samples from the Cohn-Kanade (CK) database

The structure has one hidden layer and we start with input X_i. The neurons between layers are fully interconnected with weight V_{ij} and W_{jl}, where i is the "emitting" or preceding layer of nodes, j is the receiving or "subsequent" layer of nodes, l is the layer of nodes that follows j, ij is the layer of weights between node layers i and j, jl is the layer of weights between node layers j and l.

The activation of one hidden neuron j can then be found using $z_j = W_j^{hT} X$, So, we can define

$$X = \begin{bmatrix} X_1^T \\ X_2^T \\ \cdot \\ \cdot \\ \cdot \\ X_N^T \end{bmatrix}, \quad W^h = \begin{bmatrix} W_1^{hT} \\ W_2^{hT} \\ \cdot \\ \cdot \\ \cdot \\ W_p^{hT} \end{bmatrix} \qquad (5)$$

$$\text{and} \quad Z = \begin{bmatrix} z_{11} & z_{12} & \cdots & z_{1p} \\ z_{21} & z_{22} & \cdots & z_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ z_{N1} & z_{N2} & \cdots & z_{Np} \end{bmatrix}$$

The elements of the matrix Z define the activation of the hidden neuron j to input sample x_i. We can find the hidden layer activation Z, the hidden layer output V and the system output Y, in such case,

$$Z = XW^h, \quad V = \sigma(z) \quad \text{and} \quad Y = \sigma(VW^0) \qquad (6)$$

Where $Y = [y_1 \ y_2 \ldots y_N]^T$, and the output layer weights W^0 are defined equal to hidden layer weights W^h. The back-propagation training covers: the feed-forward of the input training pattern, the calculation and back-propagation of associated error and the adjustment of the weights. The ANN is initialized randomly, the input x is taken and the next step will locate the resulting output y. The desired output d serves to calculate the back-propagation of associated error $e = d - y$, the goal now is to minimize the cost function,

$$J = \frac{1}{2} \sum_l e_l^2 \qquad (7)$$

We assume that the output layer has no activation function. So we have $y_l = \sum_j w_{jl}^0 v_j$. Now, we need to adjust the weights of the output layer using the update rule,

$$w_{jl}^0(n+1) = w_{jl}^0(n) - \alpha(n)\frac{\partial J}{\partial w_{jl}^0} \qquad (8)$$

In this equation the learning rate is $\alpha(n)$, and affects the speed at which the ANN arrives at the minimum. To find the Jacobian $\frac{\partial J}{\partial w_{jl}^0}$ we need use the chain rule, that is,

IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 5, No 2, September 2014
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org

183

$$\frac{\partial J}{\partial w_{jl}^0} = \frac{\partial J}{\partial e_l}\frac{\partial e_l}{\partial y_l}\frac{\partial y_l}{\partial w_{jl}^0} \tag{9}$$

These three partial derivatives are all relatively easy to find. We have,

$$\frac{\partial J}{\partial e_l} = e_l, \quad \frac{\partial e_l}{\partial y_l} = -1, \quad \frac{\partial y_l}{\partial w_{jl}^0} = v_j \tag{10}$$

The output layer weights are updating through,

$$w_{jl}^0(n+1) = w_{jl}^0(n) + \alpha(n)v_j e_l \tag{11}$$

In the hidden layer we apply a similar principle; the Jacobian is obtained through,

$$\frac{\partial J}{\partial w_{ij}^h} = \frac{\partial J}{\partial v_j}\frac{\partial v_j}{\partial z_j}\frac{\partial z_j}{\partial w_{ij}^h} \tag{12}$$

After some computation, we obtain,

$$\frac{\partial J}{\partial v_j} = \sum_l \left(\frac{\partial J}{\partial e_l}\frac{\partial e_l}{\partial y_l}\frac{\partial y_l}{\partial v_j}\right) = \sum_l -e_l w_{jl}^0 \tag{13}$$

with,

$$\frac{\partial v_j}{\partial z_j} = \sigma'_j(z_j) \quad \text{and} \quad \frac{\partial z_j}{\partial w_{ij}^h} = x_i \tag{14}$$

The data is actualized through law for hidden neuron weights, so that,

$$w_{ij}^h(n+1) = w_{ij}^h(n) + \alpha(n)x_i\sigma'_j(z_j)\sum_l e_l w_{jl}^0 \tag{15}$$

Based on the error correction computed by backpropagation using gradient descent step, the weights for all layers are adjusted simultaneously, for this, is necessary use of N test samples x_i, d_i in order to adjust the weights. The training data set are each of six output expressions per individual in a merged image. We have decided to have 3 layers in order to train the ANN and the "Bias Neurons" option and "Sigmoid" for transfer function is important because our data set is normalized. Once the weights are calculated the ANN is ready to select the emotion in a merged image output, the algorithm will provide six outputs related with each set of emotions.

2.2 Interactive decision model

The agent maintains Interaction prediction modules that assess probabilities when the opponent has performed an action that triggers surrounding reactions, given the past history of agent's actions, opponent's actions and the evolution of the surroundings state, as was the case in [14]. The model describes the dynamics between agent, human and surroundings. The dynamics entails different levels of interactions per each entity. We will assume that the interaction of the agent $\mathcal{R} = \{r_1, \ldots, r_m\}$ and the opponent $\mathcal{O} = \{o_1, \ldots, o_n\}$ have being developed indoors of a surrounding state $\mathcal{S} = \{s_1, \ldots, s_n\}$. The agent will confront diverse surrounding states changing with the actions of the opponent and that will affect the own agent's behavior. We will assume that the agent has a group of sensors which in each time t gather information in a vector $\phi_t = (\phi_t^1, \ldots, \phi_t^q)$. The agent will infer the surrounding state based on a transformation function $s_t = f(\phi_t)$, likewise, it uses the sensors to infer the actions performed by the opponent $\hat{o}_t = g(\phi_t)$. At the decision level the agent follows its planning activities according to Fig. 5
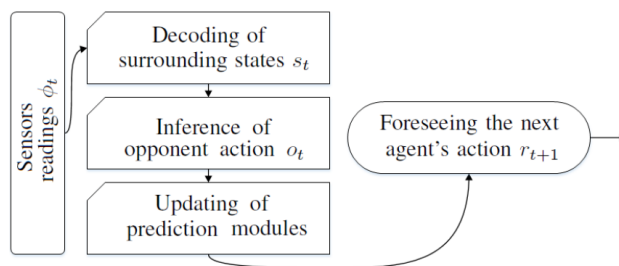

Fig. 5 Decision Model

The agent will make use of its repertoire of 12 actions $\mathcal{R}$ to interact with the opponent. The set of agent's actions are listed in Table 1.

Table 1: Agent's Actions

	R
r_1	cry
r_2	alert
r_3	warn
r_4	ask for help
r_5	salute
r_6	play
r_7	speak
r_8	claim for ludic interaction
r_9	claim for energy
r_{10}	doze
r_{11}	funny chatting
r_{12}	do nothing

Regarding the opponent, the agent is able to detect a set of 12 actions $\mathcal{O}$ issued by the opponent as affective actions, aggressive actions, and interactive actions. The set of opponent's actions are listed in Table 2.

Table 2: Opponent's Actions

O	
o_1	recharge
o_2	stroke
o_3	flatter
o_4	attack
o_5	offend
o_6	move
o_7	update
o_8	speak
o_9	play
o_{10}	order
o_{11}	ignore
o_{12}	do nothing

2.2.1 Interaction prediction modules

The agent maintains prediction modules that assess probabilities when the opponent performed an action which will trigger surrounding reactions, given the past history of agent's actions, opponent's actions and the evolution of the surroundings state, so that we are interested in computing,

$$p(s_t, o_t | r_t, (s_{t-1}, r_{t-1}, o_{t-1}), (s_{t-2}, r_{t-2}, o_{t-2})). \quad (16)$$

Extensions to k instants of memory or forecasting m steps ahead follow a similar path. (16) may be divided in

$$p(s_t | o_t, r_t, (s_{t-1}, r_{t-1}, o_{t-1}), (s_{t-2}, r_{t-2}, o_{t-2})) \times$$
$$\times p(o_t | r_t, (s_{t-1}, r_{t-1}, o_{t-1}), (s_{t-2}, r_{t-2}, o_{t-2})).$$

We assume that the surrounding states are under the control of opponent, e.g., the individual can control the heating or tenuity of light. Only the latest of the opponent's actions will affect the evolution of the surrounding states. This means should lead to the **surrounding module state**,

$$p(s_t | o_t, r_t, (s_{t-1}, r_{t-1}, o_{t-1}), (s_{t-2}, r_{t-2}, o_{t-2})) =$$
$$= p(s_t | o_t, s_{t-1}, s_{t-2}).$$

We shall actually consider four surrounding states attached at time t, $s_t^1(energy)$, $s_t^3(position)$, $s_t^4(detection)$, $s_t^2(temperature)$. Assuming conditional independence for the four surrounding states, so that

$$p(s_t | o_t, s_{t-1}, s_{t-2}) = \prod_{i=1}^{3} p(s_t^i | o_t, s_{t-1}^i, s_{t-2}^i).$$

Each surrounding state takes the form of:

a. *Energy:*

$$p(s_t^1 | o_t, s_{t-1}^1, s_{t-2}^1) = p(s_t^1 | o_t, s_{t-1}^1).$$

The current energy level and the action of the opponent will determine the forecast of energy level,

- if $o_t \neq o_1 = recharge$,
  $s_t^1 = s_{t-1}^1 - k_1 \times \Delta t$ where k_1 is the energy consumption per unit time.
- if $o_t = o_1 = recharge$,
  $s_t^1 = s_{t-1}^1 + k_2 \times \Delta t$ where k_2 is the energy consumption per unit time.

b. *Temperature:*

$$p(s_t^2 | o_t, s_{t-1}^2, s_{t-2}^2) = p(s_t^2 | s_{t-1}^2, s_{t-2}^2).$$

In this case we shall assume a temperature differential, $s_t^2 = s_{t-1}^2 + (s_{t-1}^2 - s_{t-2}^2) \times \Delta t$.

c. *Position:*

$$p(s_t^3 | o_t, s_{t-1}^3).$$

Where $o_t = attack$ is the opponent's action. The position sensor detects only whether (1) or not (0) the agent is in vertical position. As we can see in Table 3.

Table 3: Variation of the Position

s_{t-1}^3	Attack	Not attack
0	0	0
1	0	1

d. *Detection :*

$$p(s_t^4 | o_t, s_{t-1}^4).$$

The detection surrounding state shows only whether (1) or not (0) the agent perceives the opponent's presence. In this case o_t is part of opponent's interacting actions (OIA). According to the Table 4, p_1 is a probability that determine the opponent's presence when o_t not being part of the interacting actions. It follows a Beta-binomial distribution [15],

$$p_1 | data \sim Beta(\alpha_1 + qo_1, \beta_1 + no_1 - qo_1),$$

Where $no_1 =$ number of occurrences and $qo_1 =$ opponent not detected. It is summarized through

$$\frac{\alpha_1 + qo_1}{\alpha_1 + \beta_1 - no_1}$$

Table 4: Detection Surrounding State

s_{t-1}^4	$o_t \in OIA$	$o_t \notin OIA$
0	1	0
1	1	p_1

Furthermore, the opponent has an evolution of its own behavior, that could be affected by how he reacts to the agent's actions, thus incorporating the adversarial risk analysis principle, where risks stem from deliberate actions of intelligent adversaries, so that

$$p(o_t \mid r_t, (s_{t-1}, r_{t-1}, o_{t-1}), (s_{t-2}, r_{t-2}, o_{t-2})) = \quad (17)$$
$$= p(o_t \mid r_t, o_{t-1}, o_{t-2}).$$

In connection with (17) the module is shaped by two modules, one describes the own evolution of the opponent that controls whole surrounding state, we call it the **opponent's evolution module** and it describe through

$$p(o_t \mid o_{t-1}, o_{t-2}).$$

The opponent's evolution module uses the information that the opponent has done two time steps before, in order to predict the opponent's current action, as you can see in a matrix-beta model Fig. 6. D_t will designate the data available until time t

For $i, j \in \{1, 2, \ldots, n\}$ we have a priori

$$p(o_t \mid o_{t-1} = o_i, o_{t-2} =$$
$$= o_j) \sim Dir(\beta_{1ij}, \ldots, \beta_{nij}), \; o_t \in \{o_1, o_2, ..., o_n\}.$$

If h_{kij} designates the number of occurrences that the opponent did o_k after having done o_i and o_j, we have that the posterior is

$$p(o_t \mid o_{t-1} = i, o_{t-2} = j, D_t) \sim$$
$$\sim Dir(\beta_{1ij} + h_{1ij}, \ldots, \beta_{nij} + h_{nij}), \; o_t \in \{o_1, o_2, ..., o_n\},$$

which we may summarize through

$$\hat{p}_{kij} = \frac{\beta_{kij} + h_{kij}}{\sum_k (\beta_{kij} + h_{kij})}, \; k \in \{1, 2, ..., n\}.$$
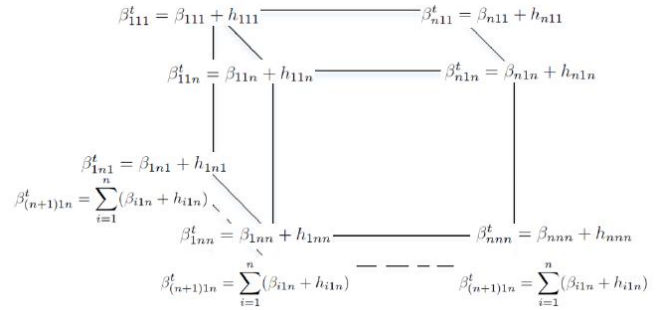

Fig. 6 3D Matrix-beta model for the opponent

At each time instant, we update the corresponding kij-th element and the corresponding last row of the 3D Matrix. That is, if the sequence was $o_{t-2} = i, o_{t-1} = j, o_t = k$, we shall update $\beta_{(n+1)j}^{t+1} = \beta_{(n+1)j}^t + 1$ and $\beta_{kij}^{t+1} = \beta_{kij}^t + 1$ entries satisfying $\beta_{kij}^{t+1} = \beta_{kij}^t$.

The other module refers to the opponent's reactions to the agent's actions. We call it **classical conditioning module** and it will be of the form

$$p(o_t \mid r_t).$$

As before, D_t will designate the data available until time t. For each r_t, the prior distribution will be Dirichlet, so that

$$p(o_t \mid r_t = r_j) \sim$$
$$\sim Dir(\beta_{1j}, \ldots, \beta_{nj}), \; o_t \in \{o_1, o_2, ..., o_n\}.$$

Now, if h_{ij} designates the number of occurrences of opponent doing o_i, when the agent has made r_j, the posterior distribution will be

$$p(o_t \mid r_t = r_j, D_t) \sim$$
$$\sim Dir(\beta_{1j} + h_{1j}, \ldots, \beta_{nj} + h_{nj}), \; o_t \in \{o_1, o_2, ..., o_n\}.$$

which we may summarize through

$$\hat{p}_{ij} = \frac{\beta_{ij} + h_{ij}}{\sum_i (\beta_{ij} + h_{ij})}, \; i \in \{1, 2, ..., n\}, j \in \{1, 2, ..., m\}.$$

The data are stored in a 2D matrix structure as you can see in Fig. 7. The last row accumulates the sum of row's values for each column.

IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 5, No 2, September 2014
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org

186

$$\begin{bmatrix} \beta_{11}^t = \beta_{11} + h_{11} & \cdots & \beta_{1m}^t = \beta_{1m} + h_{1m} \\ \vdots & \ddots & \vdots \\ \vdots & & \vdots \\ \beta_{n1}^t = \beta_{n1} + h_{n1} & \cdots & \beta_{nm}^t = \beta_{nm} + h_{nm} \\ \beta_{(n+1)1}^t = \sum_{i=1}^{n}(\beta_{i1} + h_{i1}) & \cdots & \beta_{(n+1)m}^t = \sum_{i=1}^{n}(\beta_{im} + h_{im}) \end{bmatrix}$$

Fig. 7 Classical conditioning module 2D matrix structure

As we mentioned earlier, at each time instant, we update the corresponding ij-th element and the corresponding last row of the 2D matrix. The β_{ij}'s elements are similarly evaluated. In this module we may be facing the incompatibility of a pair of actions, for this reason, if $r_t = r_j$ and $o_t = o_i$ are compatible and $\beta_{ij} = 1$ in other case $\beta_{ij} = 0$.

2.2.2 Weighting of posterior model probability

We describe now how the actualization takes place in the model based on the average of the posterior distributions, using the opponents evolution module and the classical conditioning module, [16].

$$p(o_t|r_t, o_{t-1}, o_{t-2}) = p(M_2)p(o_t|o_{t-1}, o_{t-2}) + p(M_1)p(o_t|r_t),$$

Assuming that $p(M_i)$ designates the probability that the agent follow the model i with $p(M_1) + p(M_2) = 1$, $p(M_i) \geq 0$, This model captures the agent's reactive behavior facing the opponent actions.

$$p(o_t|r_t, o_{t-1}, o_{t-2}, D_t) = p(M_1|D_t)p(o_t|a_t, D_t) + \\ + p(M_2|D_t)p(o_t|o_{t-1}, o_{t-2}, D_t),$$

with

$$p(M_i|D_t) = \frac{p(D_t|M_i)p(M_i)}{\sum_{i=1}^{2} p(D_t|M_i)p(M_i)}, \quad i = 1, 2.$$

by the premise $p(M_1) = p(M_2) = \frac{1}{2}$,

$$p(M_i|D_t) = \frac{p(D_t|M_i)}{\sum_{i=1}^{2} p(D_t|M_i)},$$

with

$$p(D_t|M_i) = \int p(D_t|\theta_i, M_i)p(\theta_i|M_i)d\theta_i$$

The modules can then be summarized as follows:

✓ **M1: Having**

$$p(D_t|M_1) =$$

$$= \int \cdots \int \left(\prod_{i,j} p_{ij}^{h_{ij}} \right) k \left(\prod_{i,j} p_{ij}^{\beta_{ij}-1} \right) dp_{ij},$$

Where $k = Normalizing\ constant$. We have to

$$p(D_t|M_1) = \left[\frac{\prod_{i=1}^{n} \Gamma(\beta_{i1})}{\Gamma(\sum_{i=1}^{n} \beta_{i1})} \cdots \frac{\prod_{i=1}^{n} \Gamma(\beta_{im})}{\Gamma(\sum_{i=1}^{n} \beta_{im})} \right]$$
$$\left[\frac{\Gamma(\sum_{i=1}^{n}(\beta_{i1} + h_{i1}))}{\prod_{i=1}^{n} \Gamma(\beta_{i1} + h_{i1})} \cdots \frac{\Gamma(\sum_{i=1}^{n}(\beta_{im} + h_{im}))}{\prod_{i=1}^{n} \Gamma(\beta_{im} + h_{im})} \right].$$

Now according to $p(D_t|M_1) = p_t^1$ we can see that if at iteration $t+1$ the bot performed r_j and the opponent performed o_i, the new probability model is updated by

$$p_{t+1}^1 = p_t^1 \times \frac{\beta_{(n+1)j}^t}{\beta_{ij}^t}$$

✓ **M2: Having**

$$p(D_t|M_2) =$$

$$= \int \cdots \int \left(\prod_{i,j,k} p_{ijk}^{h_{ijk}} \right) k' \left(\prod_{i,j,k} p_{ijk}^{\beta_{ijk}-1} \right) dp_{ijk}$$

Where $k = Normalizing\ constant$. We have to

$$p(D_t|M_2) =$$

$$= \left[\prod_{i=1}^{n} \frac{\Gamma(\beta_{i11})}{\Gamma(\beta_{i11} + h_{i11})} \cdots \prod_{i=1}^{n} \frac{\Gamma(\beta_{inn})}{\Gamma(\beta_{inn} + h_{inn})} \right]$$
$$\left[\frac{\Gamma(\sum_{i=1}^{n}(\beta_{i11} + h_{i11}))}{\Gamma(\sum_{i=1}^{n} \beta_{i11})} \cdots \frac{\Gamma(\sum_{i=1}^{n}(\beta_{inn} + h_{inn}))}{\Gamma(\sum_{i=1}^{n} \beta_{inn})} \right].$$

Here again as in M_1 the new probability model is updated by

$$p(D_t|M_2) = p_t^2,$$

then

$$p_{t+1}^2 = p_t^2 \times \frac{\beta_{(n+1)jk}^t}{\beta_{ijk}^t},$$

assuming that, at iteration $(t+1)$, the opponent performed o_k, after having performed o_i and o_j.

2.2.3 Multiobjective preference module

The agent's behavior provides utilities and probabilities, in which, we can apply the Maximum Expected Utility (MEU) criterion [17],[18], to select the optimal action based on all current information. The agent faces constantly multiple synergy states $\lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_l\}$. The consequences are provided by the synergy states that depend on the agent's actions r_t, the opponent's actions o_t and the future surrounding state s_t that happened after of r_t and o_t. The synergy states take the form of

$$\lambda_i(r_t, o_t, s_t), \; i = 1, \ldots, l.$$

Through a multi-attribute utility function [19], we will evaluate the synergy states without much loss of generality [20]. The model will adopt an additive form,

$$u(\lambda_1, \lambda_2, \ldots, \lambda_l) = \sum_{i=1}^{l} w_i u_i(\lambda_i), \text{with } w_i \geq 0, \; \sum_{i=1}^{l} w_i = 1.$$

In order to compute the maximum expected utility, we plan several instants ahead assuming additive utilities over time,

$$\max_{(r_t, \ldots, r_{t+r})} \psi(r_t, \ldots, r_{t+\eta}) =$$

$$= \sum_{(o_t, s_t), \ldots, (o_{t+\eta}, s_{t+\eta})} \left[\sum_{i=D}^{\eta} u(r_{t+i}, o_{t+i}, s_{t+i}) \right] \times$$

$$\times p((o_t, s_t), \ldots, (o_{t+\eta}, s_{t+\eta}) \mid$$
$$\mid (r_t, r_{t+1}, \ldots, r_{t+\eta}, (r_{t-1}, o_{t-1}, s_{t-1}), (r_{t-2}, o_{t-2}, s_{t-2}))).$$

For purposes of simulation, we could plan just one period ahead,

$$\max_{r_t \in \mathcal{R}} \psi(r_t) = \sum_{o_t, s_t} u(r_t, o_t, s_t) \times$$

$$\times \left[p(o_t, s_t \mid r_t, (r_{t-1}, o_{t-1}, s_{t-1}), (r_{t-2}, o_{t-2}, s_{t-2})) \right].$$

The decision is probabilistic and defines a probability model that randomly generates alternatives with probability proportional to the expected utility [21], therefore, it increases unpredictability of the agent's decisions under similar circumstances.

$$P(r_t) \propto \psi(r_t), \text{ with } P(r_t) = \text{probability of choosing } r_t.$$

    a. Preference structure:

        The agent aims at satisfying five objectives, as you can see in Fig. 8. This hierarchy entails that the agent will invest most resources in achieving

a sufficient level in the lowest objective, because of its higher weight, see [22]. Once it has attained a sufficient value in that level, it will redistribute it's his resources to achieve the next level, and so on. We describe now the global utility function, the component utility functions, and the objectives related to energy and security with the intention of making some simulations, we shall assume that
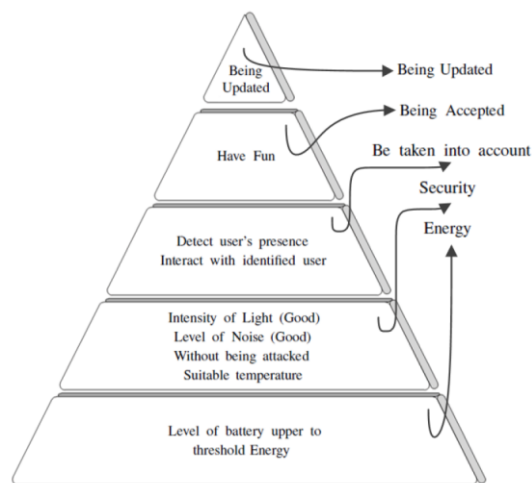

Fig. 8 Pyramidal distribution of self-objectives

    b. Basic Objectives:

- *Self-objective of Energy:* For the most part, energy needs are the literal requirements for agent survival. If these requirements are not met, the agent simply cannot continue with the normal course of its activities. Low levels are perceived as bad indicators by the agent. The objective is given by

$$u_1(Energy) = \begin{cases} 0, & \text{if } Energy \leq Lth \\ 1, & \text{if } Energy \geq Uth \\ \left(\frac{Energy - Lth}{Uth - Lth}\right), & \text{otherwise,} \end{cases}$$

with Uth (Upper threshold) = 0.5 and Lth (lower threshold) = 0.1

- *Self-objective of Security:* Agent's security, essentially, the way that the agent takes in account the risk of injury when it is attacked (Atk) by any human. At the same time the agent needs to remain at an appropriate level of

IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 5, No 2, September 2014
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org

188

temperature (Tp) for its proper operating. It is represented through

$$u_2(Atk, Tp) = w_{21} \times u_{21}(Atk) + w_{22} \times u_{22}(Tp),$$

with $w_{2i} \geq 0$, $\sum_{i=1}^{2} w_{2i} = 1$ and weights ordered in importance as follows: $w_{21} > w_{22}$
The utility functions are given by

$$u_{21}(Atk) = \begin{cases} 0, & \text{if no attack occurred} \\ 1, & \text{otherwise,} \end{cases}$$

$$u_{22}(Tp) = \begin{cases} 0, & \text{if } Tp < LTh \text{ or } Tp > UTh \\ 1, & \text{if } LTcth < Tp < UTcth \\ 1 - (\frac{(LTcth - Tp)}{LTcth}), & \text{if } Tp < LTcth \\ \frac{(UTh - Tp)}{(UTh - UTcth)}, & \text{if } Tp > UTcth, \end{cases}$$

with LTh (Lower thermal threshold) = 0°C, UTh (Upper thermal threshold) = 35° C, LTcth (Lower thermal comfort) = 20°C and UTcth (Upper thermal comfort) = 25°C.

- **Global utility function:** The global utility function based in the two lowest level objectives would be

$$w_1 \times u_1(Energy) + w_2 \times u_2(Atk, Tp),$$

With

$$w_1 >> w_2 \text{ and } w_1 + w_2 = 1.$$

3. Rule-based design of emotional feedback

Through the procedure described above the agent has the capability to evolve with a positive behavior facing the opponent by implicit communication, e.g., in case of a robot caregiver as an intelligent agent, its positive actions can influence the patients that in turn would be more inclined to engage in a closed loop human-robot emotional interaction [23]. The idea is that the agent can recognize the affective state of the human captured by the facial expression implicitly and adapt to it modifying its actions appropriately, reflecting some degree of emotional intelligence [24]. Authors such as those of [25] and [26] have raised doubts on whether machines can exhibit intelligent behavior without sign of emotion.
In human communication, emotions play an important role, and the ability to display and understand the human emotions and affective states are essential for effective communication [6]. A machine that expresses, recognizes and understands emotions similar to human ones could be a good collaborator. This is indeed an area of growing interest [27], and many designs have demonstrated how human-robot interaction in cognitive robotics can be augmented by a closed loop that implies an implicit emotional communication, [28], [29], [30], [31], [32]. The connection between decision making and emotions is very thin [33]. In this case, emotions must be seen as features that can influence the decision level. Some emotion models take into account theories which connect decision making and emotions using weighing criteria and evaluation of alternatives, [34], the mood congruency of memories that affect predictions [35], a model of intensity from the visceral factors involved in decision making [36], the weighted average of decayed base emotion [37] and numerous descriptive findings on such relations [38]. In the future we would like to extend the capabilities of the model presented in this paper using weight categorization from the integration of two or more emotions; it could be an internal emotional model that will use secondary emotions. The acquisition of emotions by a machine is a promising paradigm that allows human-like social interaction due to their ability to give emotional feedback. The appropriate responses to the emotions of others is a contextually sensitive ability that humans are particularly skilled at, given this advantage, machines could learn how to modify actions and behaviors based on emotional feedback from our system of emotional signals, [39], [40], [41]. In humans emotions motivate future behavior, the connection of actions to specific emotions is often hard-wired. Emotions help humans to overcome obstacles, and in fact pave the way to select the most appropriate behavior.
Our agent will show increasingly positive behavior and will keep a self-regulation between actions and emotions. To achieve this goal, we will base the rules on principles of the broaden and build theory, [42], [43]. In this case, the agent's reactions when it is facing the opponent's positive behavior, could trigger a cycle of positive emotions connected to the relevant action. Through positive emotions humans can develop various types of skills and capacities which negative emotions are not able to support. Negative emotions tend to favour some preferences [44], [45], [46], e.g., anger tends to maintain a course of action or in the most extreme case attack, while fear can evade the immediate context or the course of action. Disgust is associated as a specific reaction to something that is offensive, and the action that it will generate pursues to expel the trigger stimulus. As opposed to negative emotions, positive emotions help the body move from a narrow set of actions to a broader one, allowing pursuit of a wider array of thoughts and alternatives. Table 5 shows

six basic emotions [47], [48], represented in a set of rule-based action/response-pairs. The categorization that we assumed was arbitrarily chosen, taking into account that it is possible to modify the model with another order of emotions. For this first scheme, we have ordered the emotions according to the "Basic emotion theories", where we can differentiate two groups "positive" and "negative" based on the emotional stimuli organized in a categorical manner, with innate categories found in humans [49], [50]. This table shows the possibility of deciding the agent's behavior by mixing the emotional feedback and the outputs of the decision system using a rule-based algorithm. The algorithm simply corrects the agent's behavior using the human biological emotions; in this case the human-machine emotional loop will be the closest connection. The reactive response between the emotion and action is dependent on the positive or negative influence that is shown, and will be the trigger of the corrective emotional output. In other words, we hold that priority should be attached to the opponent's emotions. For practical purposes, we need to define a hierarchy of levels according to the signs of the emotions and agent's actions, showing the relative importance of these signs, see Fig. 9. A pertinent focus for future research would be to examine the inclusion of different levels of precision in the emotional hierarchy into the "Rule-based design of emotional feedback", that could be the setting of intervals related to primary and secondary emotions [33].

Table 5: Detection Surrounding State

OS_{em}	R_1	R_2	NR_1	NR_2
Anger (-)	(+)	(-)	R_{upL}	R_{upL}
Sadness (-)	(+)	(-)	R_{upL}	R_{upL}
Fear (-)	(+)	(-)	R_{upL}	R_{upL}
Disgust (-)	(+)	(-)	R_{upL}	R_{upL}
Happiness (+)	(+)	(-)	R_{upL}	M_L
Surprised (+)	(+)	(-)	R_{upL}	M_L

Note: $R_a \{R_1, R_2\}$, $R_a \{(+),(-)\}$ are signs of the agent's actions before the emotional feedback. NR_1 and NR_2 are the modified actions of the agent taking into account the emotional feedback. R_{upL}: Reject and change the hierarchy level of agent's action (to positive action); M_L: Maintain the current hierarchy level of agent's action (maintain the negative action); OS_{em} : Signs of the opponent's emotion.
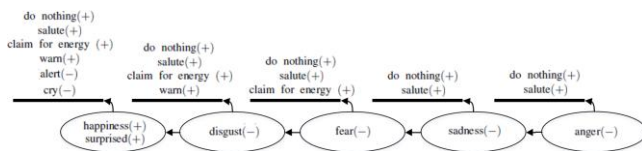

Fig. 9 Hierarchy of levels

If the opponent shows positive emotional behavior he will be willing to accept a negative behavior from outside, in this case provided by the agent's actions. Only emotions like surprise and happiness will preserve the current hierarchy level of the agent's action, e.g., "When the opponent's emotion is happiness (+) and the agent's action is cry(-), the agent preserves the negative action, in other case the agent rejects the action and increases the hierarchy level" (In real life, we are willing to manage difficult situations when we have positive emotional charge). Positive emotions may influence the generation of resources and flexible thoughts in order to choose the right action [48]. The flow chart of the agent's action modified by the positive emotion is shown in Fig. 10.
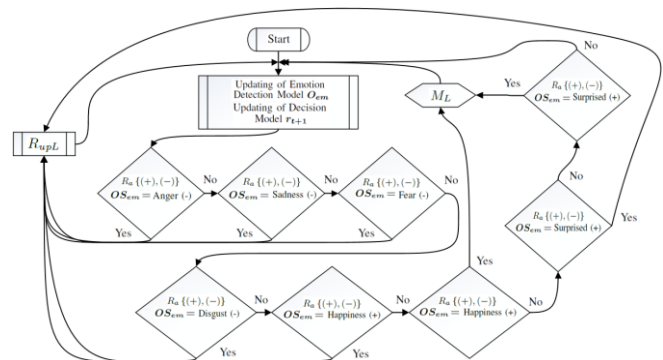

Fig. 10 Flow chart from correction of agent's action through emotional feedback

4. Experimental results

The simulations cover the evolution of the expected utility just one period ahead, the agent's reactive behavior facing the opponent's actions through the weighting of posterior model probability, the evolution of the emotional feedback provided by the opponent's face and evolution of the agent's behavior using the rule-based design of emotional feedback. We select only six agent's actions classified by sign [alert(-), cry(-), claim for energy(+), salute(+), warn(+), do nothing(+)] and six opponent's actions [attack, move, recharge, stroke and do nothing], this last set of opponent's actions do not need the signs, because all the analysis will focus on the modification of the agent's behavior. The system works per iteration facilitating capturing of the opponent's emotion through the emotion detection model. In the simulation 250 iterations were fixed, related to the training set and test set used in the Artificial neural network propagation algorithm to recognize the opponent's emotions. We used the Cohn-Kanade (CK) database [52] to construct the training set and test set. This database contains 97 individuals showing different expressions. We selected only 50 individuals

from the 97 in order to capture the set of more pronounced facial emotions. We collected 6 images per individual coming to have a total of 300 images. We trained the Artificial neural network with 50 images. The rest is comprised of 250 images and it has been used as test set. Every time that the emotion detection model has reported a misclassification, the inaccurate emotional output is replaced by the most positive agent's action, in this case do nothing(+), for the purpose of developing a positive behavior in the agent. Finally, in order to correct the agent's behavior using the emotional feedback provided by the opponent, we determine a hierarchy level of the agent's action from negative to positive; [anger(-), sadness(-), fear(-), disgust(-), happiness(+), surprised(+)]. The two simulations show interactions between the agent and the opponent during all the 250 iterations. The simulations show as expected that the agent perceives the opponent as very reactive to its actions, so the value of p(M1) rapidly achieves the maximum and p(M2) the minimum for the two cases. In simulation 1, see Fig. 11, the average success rate of emotional recognition was 86.8%, which means that 217 out of 250 test images were successfully classified. We can clearly see the influence of the emotional feedback over the agent's actions, taking into account that the emotional opponent's emotional evolution starts at a negative level. The negative component in many actions was suppressed, which is why the interaction in positive way was favored. In simulation 2, see Fig. 12, the average success rate of emotional recognition was 87.2%. Here the opponent shows more interaction with the agent, its expected utility has little random oscillation. The agent's actions are more negatively charged. After the emotional feedback a great quantity of agent's actions are allowed. Actions like cry(-) or alert(-) are more permissible when the opponent has a positive emotional charge. At the same time the agent's actions are transferred to the positive zone.
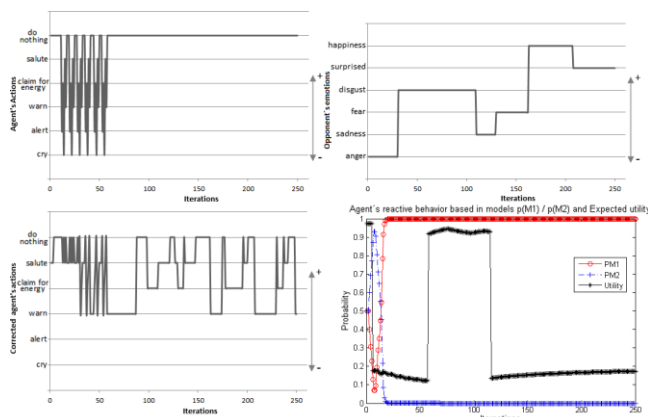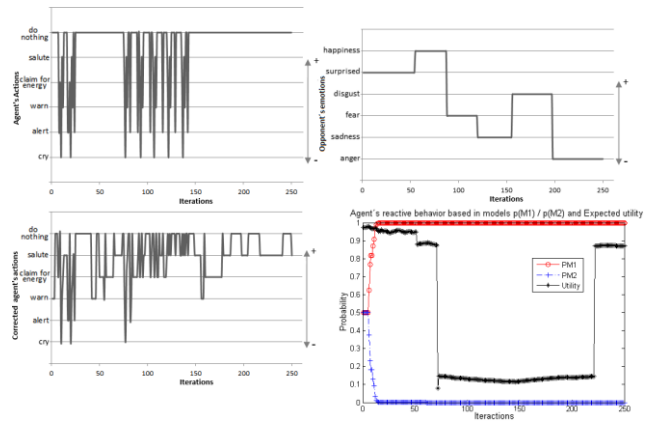

Fig. 11 Simulation 1

Fig. 12 Simulation 2

5. Discussion and Future Work

We have described a behavioral model for an autonomous agent, capable of reading information from its sensors and selecting appropriate actions based on the emotional signals expressed by a human opponent. The model uses multi-attribute decision analysis, forecasting models of the adversary and emotional feedback provided by the opponent, all supporting the final decision of the agent. The results show that the agent is capable of improving its social interaction by changing its behavior according to the affective state of the opponent. Here we can consider that there is a narrowing of human-machine emotional loop. Such interaction will have to be taken into account in the future, because the number of devices connected wirelessly with intelligent machines increases exponentially. All the multimodal sensor inputs that recover biophysiological signals from humans can be used with optimized algorithms to embed affective intelligence in synthetic agents. As a case in point, the wireless sensor networks can improve the sensory capabilities in robots. These groups of sensors are akin to usual sensors but with better monitoring and control of a wide range of information. The OpenFlow connection environment could be the link between a more precise sensing of emotional signals from humans and their proper interpretation. With all of this data collected, we can explore deeper the affective communications and make robots with more emotional capabilities that will connect with humans, offering the missing natural interaction. Following this, future work will address providing a model for an autonomous agent that makes decisions influenced by more accurate emotional factors as the emotional feedback provided by the kinetic typography, speech and facial expression in humans. This new approach could make interaction between humans and agents more fluent and natural.

IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 5, No 2, September 2014
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org

191

References

[1] R. Picard, Affective Computing, United States: the MIT Press, 1998

[2] K. Dautenhahn and I. Werry, "Towards interactive robots in autism therapy: background, motivation and challenges", Pragmatics and Cognition, Vol. 12, No. 1, 2004, pp. 1-35

[3] P. Marti, C. Moderini, L. Giusti, and A. Pollini, "A robotic toy for children with special needs: From requirements to design", IEEE 11th International Conference on Rehabilitation Robotics, Kyoto, Japan, 2009

[4] M. Lewis and L. Cañamero, "Are Discrete Emotions Useful in Human-Robot Interaction?, Feedback from Motion Capture Analysis", Affective Computing and Intelligent Interaction (ACII), Humaine Association Conference, Geneva, 2013, pp. 97-102

[5] T. Horii, Y. Nagai and M. Asada, "Touch and Emotion: Modeling of developmental differentiation of emotion lead by tactile dominance", Development and Learning and Epigenetic Robotics (ICDL), IEEE Third Joint International Conference, Osaka, Japan, 2013, pp. 1-6

[6] T. W. Fong, I. Nourbakhsh and K. Dautenhahn, "A survey of socially interactive robots", Robotics and Autonomous Systems, Vol. 42, 2003, pp. 143-166

[7] D. Ríos, J. Ríos and D. Banks, "Adversarial risk analysis," Journal of the American Statistical Association, Vol. 104, No. 486, 2009, pp. 841-854

[8] H. Kobayashi and F. Hara, "Facial Interaction between Animated 3D Face Robot and Human Beings", Proceedings of International Conference on Systems, Man, Cybernetics, Vol. 3, 1997, pp. 732-737

[9] M. Pantic, "Automatic analysis of facial expressions: The state of the art", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 12, 2000, pp. 1424-1445

[10] R. T. Clemen and T. Reilly, "Making Hard Decisions with Decision Tools", Mason, OH: South-Western Cengage Learning, 2001

[11] C. L. Hwang and K. Yoon, "Multiple Attribute Decision Making Methods and Applications: A State of the Art Survey", Springer-Verlag, Berlin, Heidelberg, New York, 1981

[12] K. Jinchuan and L. Xinzhe, "Empirical analysis of optimal hidden neurons in neural network modeling for stock prediction", Proceedings of the Pacific-Asia Workshop on Computational Intelligence and Industrial Application, Vol. 2, 2008, pp. 828-832

[13] J.G. Rázuri, D. Sundgren, R. Rahmani and A. M. Cardenas, "Automatic Emotion Recognition through Facial Expression Analysis in Merged Images Based on an Artificial Neural Network". Proceedings of 12th Mexican International Conference on Artificial Intelligence (MICAI), Mexico City, Mexico, 2013, pp. 85-96

[14] J.G. Rázuri, P.G. Esteban and D.R. Insua, "An adversarial risk analysis model for an autonomous imperfect decision agent." In: T.V. Guy, M. Kárný and D.H. Wolpert, Eds. Decision Making and Imperfection. SCI, Vol. 474, Springer, Heidelberg, 2013, pp. 165-190

[15] D. Ríos, F. Ruggeri and M. Wiper, "Bayesian Analysis of Stochastic Process Models", Wiley, 2012

[16] J. Hoeting, D. Madigan, A. Raftery and C. Volinsky, "Bayesian model averaging: A tutorial", Statistical Science, Vol. 4, 1999, pp. 382-417

[17] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann Publishers, 1988

[18] S. French and D. Ríos, Statistical Decision Theory, Oxford University Press, New York, NY, USA, 2000

[19] R. T. Clemen and T. Reilly, Making Hard Decisions with Decision Tools, Mason, OH: South-Western Cengage Learning, 2001

[20] D. Von Winterfeldt and W. Edwards, Decision Analysis and Behavioral Research, New York: Cambridge University Press, 1986

[21] C. Bielza, P. Müller and D. Ríos, "Monte Carlo Methods for Decision Analysis with Applications to Influence Diagrams", Management Science, Vol. 45, No. 7, 1999, pp. 995-1007

[22] A. H. Maslow, "A theory of human motivation," Psychological Review, Vol. 50, No. 4, 1943, pp. 370-96

[23] I. Leite, C. Castellano, A. Pereira, C. Martinho and A. Paiva, "Long-term Interactions with Empathic Robots: Evaluating Perceived Support in Children", Proceedings of 4th International Conference of Social Robotics, Springer, LNCS 2012

[24] T. Kuremoto, T. Tsurusaki, K. Kobayashi, S. Mabu and M. Obayashi, "A Model of Emotional Intelligent Agent for Cooperative Goal Exploration", Proceedings of 9th International Conference, ICIC, Vol. 7995, Nanning, China, 2013, pp. 21-30

[25] M. Minsky, The Society of Mind. Simon and Schuster, 1985

[26] L.C. Charland, "Emotion as a natural kind: Towards a computational foundation for emotion theory", Philosophical Psychology, Vol. 8, No. 1, 1995, pp. 59-85

[27] T. Ziemke and R. Lowe, "On the role of emotion in embodied cognitive architectures: From organisms to robots", Cognitive computation, Vol. 1, No. 1, 2009, pp. 104-117

[28] M.S. El-Nasr, J. Yen and T.R. Ioerger, "FLAME Fuzzy logic adaptive model of emotions", Autonomous Agents and Multi-Agent Systems, Vol. 3, No. 3, 2000, pp. 219-257

[29] J. Velasquez, "From affect programs to higher cognitive emotions: an emotion-based control approach", Proceedings Emotion-Based Agent Architecture Workshop at the International Conf. Autonomous Agents, 1999, pp. 114-120

[30] R. Kirby, J. Forlizzi and R. Simmons, "Affective social robots", Robotics and Autonomous Systems, Vol. 58, 2010, pp. 322-332

[31] D. Hanson, S. Baurmann, T. Riccio, R. Margolin, T. Dockins, M. Tavares and K. Carpenter, "Zeno: a Cognitive Character", AI Magazine, and special Proc. Of AAAI National Conference, Chicago, 2009

[32] G. Pioggia, M.L. Sica, M. Ferro, R. Igliozzi, F. Muratori, A. Ahluwalia and D. De Rossi, "Human-robot interaction in autism: FACE, an android-based social therapy", Robot and

IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 5, No 2, September 2014
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org

192

Human interactive Communication. The 16th IEEE International Symposium, 2007, pp. 605-612

[33] A. Damasio, Descartes' error: Emotion, reason, and the human brain, New York: Putmam 1994

[34] J.S. Lerner and D. Keltner, "Fear, anger, and risk. In: Journal of Personality and Social Psychology", Vol. 81, No. 1, 2001, pp. 146-159

[35] C.L. Rusting and T. DeHart, "Retrieving positive memories to regulate negative mood: Consequences for mood-congruent memory", Journal of Personality and Social Psychology, Vol. 78, 2000, pp. 737-752

[36] G. Loewenstein and J.S. Lerner, "The role of affect in decision making", Davidson, R., Scherer K., Goldsmith, H, Eds. Handbook of affective science, New York: Oxford University Press, 2003, pp. 619-642

[37] L. Moshkina, "An integrative framework for affective agent behavior", IASTED Int.Conf. on Intelligent Systems and Control, 2006

[38] B.A. Mellers, A. Schwartz, K. Ho and I. Ritov, "Decision affect theory: Emotional reactions to the outcomes of risky options", Psychological Science, Vol. 8, 1997, pp. 423-429

[39] W. Johnson, J. Rickel and J. Lester, "Animated pedagogical agents: Face-to-face interaction in interactive learning environments", International Journal of Artificial Intelligence in Education, Vol. 11, No. 1, 2000, pp. 47-78

[40] I. Leite, C. Martinho, A. Pereira and A. Paiva, "iCat: an affective game buddy based on anticipatory mechanisms", Proc. 7th Intl. joint Conf. on Autonomous agents and multiagent systems, Vol. 3, 2008, pp. 1229-1232

[41] C. Becker-Asano, P. Stahl, M. Ragni, M. Courgeon, J.C. Martin and B. Nebel, "An affective virtual agent providing embodied feedback in the paired associate task: system design and evaluation", Proc. of the 13th. Intl. Conf. on Intelligent Virtual Agents, 2013, pp. 406-415

[42] B.L. Fredrickson and R.W. Levenson, "Positive emotions speed recovery from the cardiovascular sequelae of negative emotions", Cognition and Emotion, Vol. 12, 1998, pp. 191-220

[43] R.W. Levenson, "Emotion and the autonomic nervous system: A prospectus for research on autonomic specificity. In: Social psychophysiology and emotion: Theory and clinical applications. Wagner, HL., editor. London: Wiley, 1988, pp. 17-42

[44] N.H. Frijda, The emotions. Cambridge: Cambridge University Press, 1986

[45] N.H. Frijda, P. Kuipers and E. Schure, "Relations among emotion, appraisal, and emotional action readiness", Journal of Personality and Social Psychology, Vol. 57, 1989, pp. 212-228

[46] R.S. Lazarus, Emotion and adaptation. New York: Oxford University Press 1991

[47] P. Ekman, "An argument for basic emotions", Cognition and Emotion, Vol. 6, 1992, pp. 169-200

[48] R. Plutchik, "The nature of emotions", American Scientist, Vol. 89, No. 4, 2001, pp. 344-350

[49] A. Ohman and S. Mineka,"Fears, phobias, and preparedness: Toward an evolved module of fear and fear learning", Psychological Review, Vol. 108, 2001, pp. 483-522

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

[51] B.L. Fredrickson, "What Good Are Positive Emotions?", Review of General Psychology, Vol. 2, No. 3, 1998, pp. 300-319

[52] T. Kanade, J.F. Cohn and T. Yingli, "Comprehensive database for facial expression analysis", Proceedings of Fourth IEEE Int. Conf. Automatic Face and Gesture Recognition, 2000, pp. 46-53

Javier G. Rázuri is currently a researcher at the Department of Computer and System Sciences, Stockholm University. He was a researcher and Phd student of Department of Statistics and Operations Research at Universidad Rey Juan Carlos and Acting developer of the Affective computing department in AISoy Robotics. In 1995, he received with honors the Bachelor of electronic engineering degree at Peruvian University of Applied Sciences (UPC) in Perú. In 2007, he received a Master's Degree in Business Management at IEDE - Business School of European University of Madrid. In 2008, he received a Master's Degree in Decision Systems Engineering at Universidad Rey Juan Carlos. From 2008, he was a researcher in several projects financed for the university and the European Union focused on Robotics and decision-making related to EU higher education. His research interests are related with, Affective computing, emotional decision making, Human-machine interaction, Robotics, Human Robotics, Robotics and Autonomous Systems, Neuroscience, Sentiment analysis, in order to reproducing behavior patterns similar to human and provide to the agents a type of emotional intelligence and improve the interaction experience making more close the loop human-robot emotional interaction.

David Sundgren received his Doctoral degree in Computer and Systems Sciences from Stockholm University in 2011. He worked as lecturer at University of Gavle. He defended the thesis on "Apparent Arbitrariness of Second-Order Probability Distributions", dealing with uncertain probabilities. He is currently a researcher at the Department of Computer and System Sciences, Stockholm University.

Rahim Rahmani is an Associate Professor of Computer Science, focusing on QoS and optimization for management of cloud computing infrastructure and wireless sensor networks, such as OpenFlow. He earned a technical doctorate at the Mid Sweden University in adaptive Active Queue Management (AQM) algorithms for access routers in heterogeneous networks. He has served as reviewer and in technical committees of international conferences. He is a member of the editorial board of International Journal of Wireless Networking and Communications.

Aron Larsson has his PhD degree in Computer and Systems Sciences and is currently a researcher at the Department of Information and Media, Mid Sweden University as well as the Department of Computer and System Sciences, Stockholm University. His main research interest is the use of computer-based decision analysis and process models in complex decision making in which risks, uncertainties and trade-offs exist. Aron is the coordinator for the DECIDE Research Group at Stockholm University and leads the research RDALAB at Mid Sweden University. He has developed and applied risk and decision analytical methods for the paper industry, municipal decision-making problems, international mine clearance efforts and disaster preparedness. Aron is also active in the spin-off company Preference AB, which maintains the decision analysis software DecideIT.

Antonio Moran Cardenas has his PhD degree in Engineering from TUAT, Japan. His research interests include intelligent systems design based on the integration of neural networks, fuzzy logic and genetic algorithms applied to autonomous control engineering, robotics, systems modeling and other applications.