Proceedings of the 10th International Conference "Reliability and Statistics in Transportation and Communication" (RelStat'10), 20–23 October 2010, Riga, Latvia, p. 207-217. ISBN 978-9984-818-34-4
Transport and Telecommunication Institute, Lomonosova 1, LV-1019, Riga, Latvia

THE REINFORCEMENT FRAMEWORK OF A DECISION SUPPORT SYSTEM FOR THE LOCALIZATION AND MONITORING OF INTELLIGENT REMOTE BIO ROBOTS

[1]Dale Dzemydiene, [2]Antanas Andrius Bielskis, [2]Arunas Andziulis, [2,3]Darius Drungilas, [1,3]Ramunas Dzindzalieta, [2,3]Gediminas Gricius

[1]Mykolas Romeris University
Ateities str. 20, LT- 08303 Vilnius, Lithuania
E-mail: daledz@mruni.eu

[2]Klaipėda University
Manto 84, 92294 Klaipėda, Lithuania
E-mail: andrius.bielskis@ik.ku.lt, arunas@ik.ku.lt,

[3]Institute of Mathematics and Informatics
Akademijos str. 4, Vilnius, Lithuania
E-mail: dorition@gmail.com, ramunas.dzindzalieta@teo.lt

This paper analyses the possibilities of the integration of different technological and knowledge representation techniques for the development of reinforcement frameworks for the remote control of multiple agents such as wheelchair-type robots. Some technological solutions are discussed regarding the recognition of localization of moving objects by using mobile technologies. Large-scale multi-dimensional recognitions of emotional diagnoses of disabled persons often generate large amounts of multi-dimensional data with complex recognition mechanisms, based on the integration of different knowledge representation techniques and complex inference models. The problem is to reveal the main components of a diagnosis as well as to construct flexible decision making models. Sensors can help to record primary data for monitoring objects; however the recognition of abnormal situations, the clustering of emotional stages and resolutions for certain types of diagnoses is an oncoming issue for bio-robot constructors. The prediction criteria of the diagnosis of the emotional situation of disabled persons are described using knowledge based models of neural networks. The research results present the development of a multi-layered framework architecture with the integration of artificial agents and support components for diagnosis recognition and control, or further actions, by using mobile technologies. The method of fuzzy neural network control of the speed of wheelchair-type robots working in real time by providing movement support for disabled individuals is presented. The fuzzy reasoning by using fuzzy logical Petri nets is described in order to define the physiological state of disabled individuals through recognizing their emotions during their different activities. Some new possibilities of the recognition of moving object location are introduced in the system.

Keywords: multiple agent system, decision support system, knowledge representation techniques, fuzzy logic, neural networks, Petri nets

1. Introduction

The development process of intelligent systems with adaptive e-services is important for providing user-friendly e-health and e-social care for people with movement disabilities. Such systems include different intellectual components for control and for monitoring sensors by supporting multi-agent activities. In addition, in accordance with the recognition of certain situations, these systems integrate the possibilities to affect and control the devices used by disabled persons [14, 16, 25]. We recognize the possibilities of developing the integration of different types of knowledge representation techniques in such bio-robot systems with working on-line sub-systems of complex mechanisms of cooperation of multi-agent activities for sensing human affect.

The framework provides intelligent accident preventive robot-based support for people with movement disabilities and includes affect sensing in Human Computer Interaction (HCI) in providing e-health care for people with movement disabilities, Human-Robot Interaction (HRI) for assisting tele-healthcare patients to remain autonomous, and Computer Mediated Communication (CMC), to provide adaptive user-robot friendly collaboration. Such systems should depend upon the possibility of extracting emotions without interrupting the user during HCI, HRI, or CMC [1, 2, 4, 6, 8, 9]. Emotion is a mind-body phenomenon accessible at different levels of observation (social, psychological, cerebral and physiological). The continuous physiological activity of a disabled person is being made accessible by use of intelligent agent-based bio-sensors coupled with computers.

The aim of this research concerns investigations into the integration of different knowledge representation techniques in order to develop a reinforcement framework of multiple cooperative agents'

activities in order to recognise the prediction criteria of the diagnoses of the emotional situations of disabled persons. The research results present further development of a multi-layered model of this framework, with integration of the evaluation of localization possibilities and decision support system constructions. The knowledge of decision support systems is represented by fuzzy neural control of the speed of two wheelchair-type robots working in real time providing movement support for disabled individuals. The method of fuzzy reasoning using fuzzy logical Petri nets [15] is described in order to define the physiological state of disabled individuals by recognition of their emotions.

2. The Framework Architecture of the Adaptive Control of a Multi-agent System Working with Robot Motion Recognition Components

The proposed reinforcement framework is based on the interaction of intelligent remote bio-robots, localization services, embedded decision support systems and data stored in a data warehouse (Fig. 1). The data warehouse is based on distributed information systems with important personal data of the patients and sensor monitoring data. The framework includes the adaptive moving wheelchair-type robot which is remotely communicating with a wearable human affect sensing bio-robot. To record, for reasons of e-health care, relevant episodes based on humans affect stages [2], the context aware sensors are incorporated into the design of the Human Affect Sensing Bio Robot-x (HASBR-*x*) for every disabled individual, and into the local Intelligent Decision Making Agent-x (IDMA-*x*) for every intelligent support providing robot.

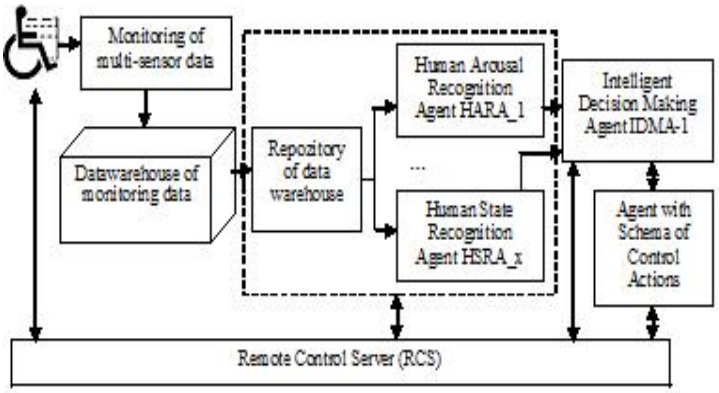


Figure 1. The reinforcement framework of an intelligent remote bio robot interaction based on distributed information systems

This framework allows a multi-sensor data fusion before the transmission of the data to the Remote Control Server (RCS) to minimize the TCP/IP (UDP) bandwidth usage. Multi-agent based adaptive motion control of both robots is based on an adaptive Fuzzy Neural Network Control (FNNC) approach. The architecture of the FNNC controller represents an approach of Adaptive Neural Fuzzy Inference System (ANFIS) that combines the fields of fuzzy logic and neural networks [3] (Fig. 2).

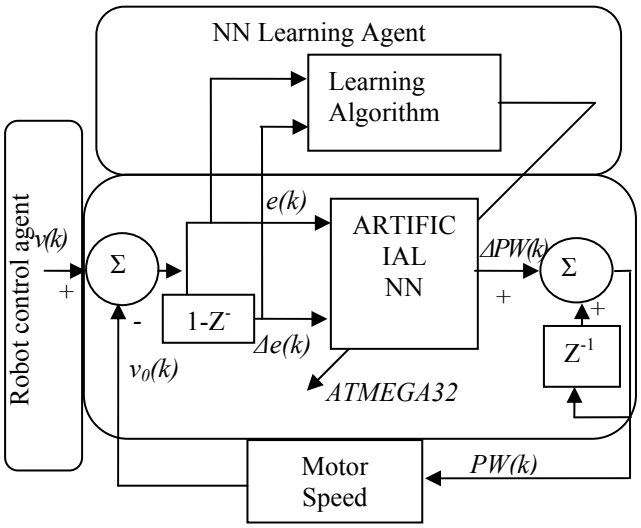


Figure 2. Modified agent based adaptive FNNC-type DC motor speed controller according to [3]

The ability to learn about the nonlinear dynamics and external disturbances of the motor speed controller with a stable output, small steady error, and fast disturbance rejection is integrated into this framework.

At the kth moment, the difference between motor speed reference value *v(k)* and motor speed output value $v_o(k)$ is split to speed error *e(k)* and speed error change *Δe(k). These* values are used as proposed in [6] *NN Learning Agent* in Fig. 3 for learning the artificial neural network *Artificial NN in* Fig. 2 as well as the 2nd order input vector of the *Artificial NN*. The output of the *Artificial NN* generates a percentage value of pulse width change *ΔPW(k)* to describe how much pulse width value *PW(k)* of the real motor speed control value at the moment *k* should be changed. This value then is generated in real time by the *ATmega32* microcontroller to perform online calculating:

$$PW(k) = PW(k-1) + \Delta PW(k). \tag{1}$$

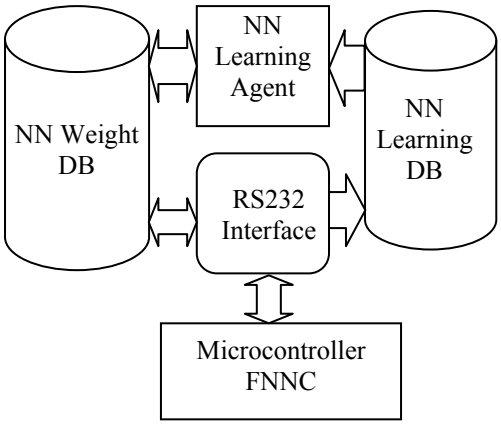


Figure 3. Multi-agent based adaptive robot motor speed control system using Agent-based NN learning system [6]

The architecture of the neural-fuzzy controller [3] for DC motor speed control of a wheelchair-type robot is presented in Fig. 4.

There layer 1 represents inputs $X = e(k)$ and $Y = \delta e(k)$ to the fuzzy neural controller, the speed error *e(k)* and the change in speed error $\delta e(k) = e(k) - e(k-1)$, respectively. Layer 2 consists of 7 input membership nodes with four membership functions, *A1, A2, A3,* and *A4,* for input *X* and three membership functions, *B1, B2,* and *B3*, for input *Y* [3]. Each node in layer 2 acts as a linguistic label of one of the input variables 1 and 2, i.e., the membership value specifying the degree to which an input value belongs to a fuzzy set is determined in this layer. The triangular membership function is chosen owing to its simplicity. For the change in motor speed error *δe(k),* the initial values of the premise parameters (the corner coordinates aj, b_j and c_j of the triangle) are chosen so that the membership functions are equally spaced along the operating range of each input variable.

The weights between input and membership level are assumed to be at unity. The output of neuron j = 1, 2, 3, and 4 for input i = 1 and j = 1, 2, and 3 for input i = 2 in the second layer can be obtained as follows.

For positive triangle slope if $X_i \geq a_j$ and $X_i \leq b_j$

$$O_{2j} = (X_i - a_j)/(b_j - a_j),$$

Or for triangle negative slope if $X_i \geq b_j$ and $X_i \leq c_j$

$$O_{2j} = (X_i - c_j)/(b_j - c_j),$$

Where a_j, b_j, and c_j are the corners of the jth triangle type membership function in layer 2 and X_i is the ith input variable to the node of layer 2, which could be either the value of the error or the change in error. The layer 1 in Fig. 3 represents inputs $X = e(k)$ and $Y = \delta e(k\}$ to the fuzzy neural controller, the speed error *e(k)* and the change in speed error $\delta e(k) = e(k) - e(k-1)$, respectively.

Layer 2 consists of 7 input membership nodes with four membership functions, *A1, A2, A3,* and *A4*, for input *X* and three membership functions, *B1, B2*, and *B3*, for input *Y*. The weights between the input and membership levels are assumed to be at unity. Each node in Rule layer 3 multiplies the incoming signal and outputs the result of the product representing one fuzzy control rule. It takes two

inputs, one from nodes *A1–A4* and the other from nodes *B1–B3* of layer 2. Nodes *A1–A4* define the membership values for the motor speed error and nodes *B1–B3* define the membership values for the change in speed error. Accordingly, there are 12 nodes in layer 3 to form a fuzzy rule base for two input variables, with four linguistic variables for the input motor speed error $e(k)$ and three linguistic variables for the input change in motor speed change error $\delta e(k)$.

The input/output links of layer 3 define the preconditions and the outcome of the rule nodes, respectively. The outcome is the strength applied to the evaluation of the effect defined for each particular rule. The output of neuron k in layer 3 is obtained as $O_{3k} = W_{3jk}*y_{3j}$, where y_{3j} represents the jth input to the node of layer 3 and W_{3jk} is assumed to be at unity. Neurons in the output membership layer 4 represent fuzzy sets used in the consequent fuzzy rules. An output membership neuron receives inputs from corresponding fuzzy rule neurons and combines them by using fuzzy operation union. This was implemented by the maximum function.

Layer 4 acts upon the output of layer 3 multiplied by the connecting weights. These link weights represent the output action of the rule nodes evaluated by layer 3, and the output is given $O_{4m} = max\ (O_{3k}*W_{km})$, where the count of k depends on the links from layer 3 to the particular m^{th} output in layer 4 and the link weight W_{km} is the output action of the mth output associated with the kth rule. This level is essential in ensuring the system's stability and allowing smooth control actions.

Layer 5 is the output layer and acts as a defuzzifier. The single node in this layer takes the output fuzzy sets clipped by the respective integrated firing strengths and combines them into a single fuzzy set. The output of the neuron-fuzzy system is crisp, and thus a combined output fuzzy set must be defuzzified. The sum-product composition method was used. It calculates the crisp output as the weighted average of the cancroids of all output membership functions as $O_{5o} = Sum\ (O_{4m}*a_{cm}*b_{cm})\ /\ Sum\ (O_{4m}*b_{cm})$, where a_{Cm} and b_{Cm} for m = 1, 2, .., and 5 are the centres and widths of the output fuzzy sets, respectively. The values for the b_{Cm}'s were chosen to be at unity. This scaled output corresponds to the control signal (percent duty cycle) to be applied in order to maintain the motor speed at a constant value. The only weights that are trained are those between layer 3 and layer 4 of Fig. 3.

A back-propagation network is used to train the weights of this layer. The weights of the neural network were trained offline by using an open source type *R*-programming environment before they were used in the online real time experiment by applying the modified learning algorithm from [3]:

Step (1): Calculate the error for the change in the control signal (duty cycle) for ATmega32-based microcontroller as $E_o = T_o - O_5$, where E_o, T_o, and O_5 are the output error, the target control signal, and the actual control signal;

Step (2): Calculate the error gradient $\delta_m = (T_o - O_{5o})*(Sum\ (O_{4j}\ (a_{cm} - a_{cj})$ for j = 1 to m–1 and $j <> m)\ /\ Sum\ (O_{4j}\ for\ j = 1\ to\ m)*2$, where $a_{Ci}\ for\ i$ = 1…5 are the centres of the output fuzzy sets and O_{4j} is the firing strength from node j in layer 4;

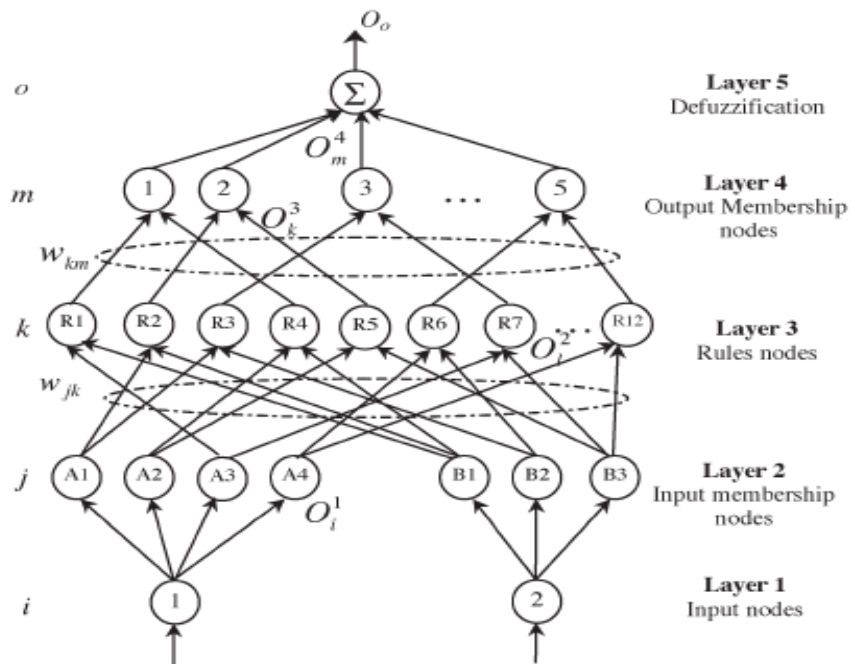


Figure 4. Architecture of the neural-fuzzy controller by [3] for DC motor speed control of wheelchair type robot

Step (3): Calculate the weight correction $\Delta_{wkm} = \eta\delta_m O_{3k}$ to increase the learning rate. Here a Sejnowski – Rosenberg updating mechanism was used, which takes into account the effect of past weight and changes in the current direction of the movement in the weight space. This is given by $\Delta_{wkm}(t) = \eta(1 - \alpha)\delta_m O_{3m} + \alpha\Delta_{wkm}(t - 1)$, where α is a smoothing coefficient in the range of 0…1,0, and η is the learning rate;

Step (4): Update the weights $w_{km}(t + 1) = w_{km}(t) + \Delta w_{km}(t)$, where t is the iteration number. The weights linking the rule layer (layer 3) and the output membership layer (layer 4) are trained to capture the system dynamics and therefore minimize the ripples around the operating point.

3. Localization Possibilities of Moving Objects

In order to identity the location of a moving object, we will use the package *javax.microedition.location* in J2ME's location (JSR 179) [18]. Location service: LDS answers basic questions: where the object is (e.g. coordinates), and how you can get to the object. The location detection service enables us to obtain information about the insecurity of the locality, or, if an accident happens, we will be able to inform the appropriate institutions about the incident. The package *javax.microedition.location* enables us to write application programs (APP) to identify the location to be equipped with limited resources [22, 24]. J2ME programming interface specification can be implemented dynamically by any of our listed local methods.

The main J2ME programming interface provides the current physical location with mobile devices. Hardware equipment for the platform determines which methods are suitable for a location and which of them it supports. APP may require certain properties from the supplier, such as a degree with the minimum precision. APP should warn the user before he starts using one of the methods.

JSR 179 requirement: "Connected Device Configuration (CDC) or Connected Limited Device Configuration (CLDC) with 1.1 versions". The Connected Limited Device Configuration (CLDC) is not supported with version 1.0, because there are no floating-point numbers, which are used in APP to show the coordinates and other dimensions [18].

The package *javax.microedition.location* is designed as the main class in which other classes are included:

- Criteria class (cr) with the parameters of accuracy, response time, height and speed; if we specify the start of recognition of the object, then we create a new criteria class: *Criteria cr = new Criteria ()*; and define the horizontal accuracy: *cr.setHorizontalAccuracy(500)* (in this case 500 meters).
- Location class extracts the local results. Its object contains coordinates, speed if reached, and the address where the text is reached and the time marker, which shows the dimensions of the space.

The coordinates are designed for two classes: Coordinates of the object represent points of latitude and longitude in degrees, and the height in meters.

- Qualified Coordinates of the object contain latitude, longitude, and also their accuracy , which are depicted as the radius of the area.

Examples of this specification are presented as follows:

```
Criteria cr = new Criteria ();
cr.setHorizontalAccuracy (500);
LocationProvider locpro = LocationProvider.getInstance (crit);
Location loc = locpro.getLocation (60);
Coordinates loc.getQualifiedCoordinates cor = ();
if (cor! = null) (
double lat = cor.getLatitude ();
double lon = cor.getLongitude ();
```

The session initiation protocol (SIP) is a signalling protocol for applications that create, modify and complete sessions between one or more participants [21]. SIP clients use TCP or UDP ports (usually port 5060) to connect to the SIP server or to other terminal systems. SIP is programmed to be applied to J2ME. Some supplements are needed in order to adapt the session initiation protocol (SIP) technology to mobile devices [19, 20]. The supplemented *javax.microedition.Connection* package is created as a specific *javax.microedition.sip* package, which provides the connection sessions between the SIP clients [21].

Connections between the concepts of open and complete client and server connections are made by sending the necessary streams of data. It is necessary to view the classes, requiring the use of SIP technology. The main programming interface classes of the package are presented and described in Fig. 5.

The SIP-based system must be designed according to the requirements that enable us to send the required data: the coordinates of the location, the object identification, the status of the object and other

relevant parameters. SIP-based location information is ordered using the *SUBSCRIBE* message, and it notifies about status changes of the object by means of a *NOTIFY* message. Different sources of information such as a mobile phone or other equipment can give additional information provided to a server about time moments if the sensor information is used. This function is performed by the SIP *PUBLISH* message functional interoperability.

The notification is a user agent that generates *NOTIFY* requests for the purpose of notifying subscribers about the state of a resource. Typically notifiers also accept *SUBSCRIBE* requests to create subscriptions.

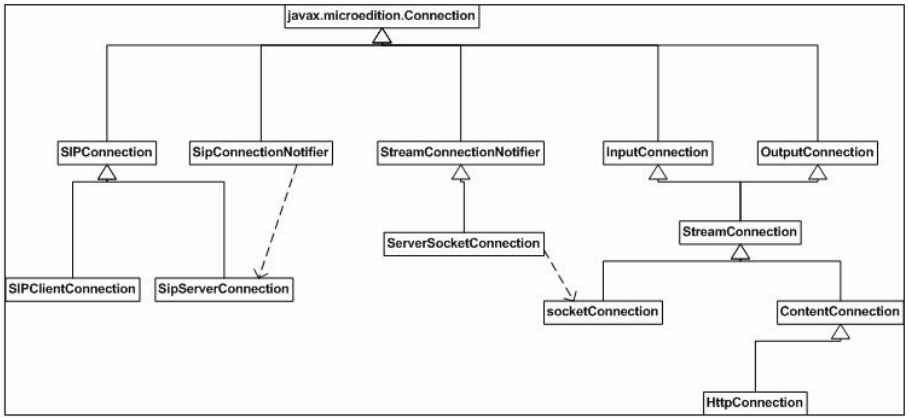


Figure 4. Classes of extension package *javax.microedition.Connection* (according to [28, 29])

Notification is the act of sending a *NOTIFY* message to a subscriber to inform the subscriber of the state of a resource. A subscriber is an agent that receives *NOTIFY* requests; these *NOTIFY* requests contain information about the state of a resource the subscriber is interested in. Typically subscribers also generate *SUBSCRIBE* requests and then send them to notification actions to create subscriptions. When a change in the subscribed state occurs, the notification immediately constructs and sends a *NOTIFY* request to inform subscribers of changes in the state to which the subscriber has a subscription.

Mobile services can provide data about the changing position of a user's terminal in geographical dimensions. Mobile Web services may be added to different terminals and a relationship will be possible if the interface is the same. Such a realization is inappropriate in our case because it will not have the possibility of building up sessions between moving terminals.

We have to use the mobile Web services between the terminals as P2P that can use SIP sessions. The end points of mobile Web services are SIP URI. Web service end-points are two points of the URI which consist of the IP addresses.

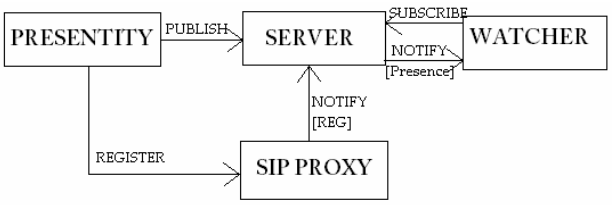


Figure 5. Common scheme of the relationship between Presentity and Watcher

3. Human Computer Interaction in the System

There are many different methods of recognizing physical state or behaviour by using data from a wearer's emotion recognition sensors [5-9]. A modified Arousal – Valence model from [6] was used to discover information in real time in order to provide some friendly advice to a person with movement disabilities.

The framework presented in Figure 1 uses four emotion recognition sensors for each disabled individual: ECG (Electrocardiogram), SCR (skin conductance response), STH (skin temperature of head), and STF (skin temperature of finger) to provide HR(heart rate), HRVH(heart rate variability for the range of 0.15 to 0.4 Hz), HRVL (heart rate variability for the range of 0.015 to 0.15 Hz), SCR, STH , and STF inputs for defining fuzzy values of arousal and valence (Fig. 4).

The system uses 67 rules to transform the 2 inputs (the arousal and the valence) into the 5 outputs (fun, challenge, boredom, frustration, and excitement). The fuzzy system model from [6] is used for recognition of five emotional states (fun, challenge, boredom, frustration, and excitement) from arousal and valence.

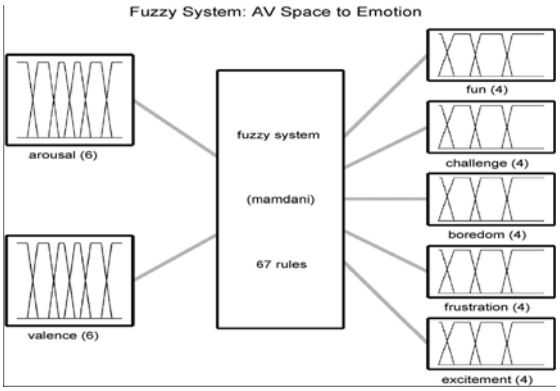


Figure 6. Fuzzy system of [6] for recognition of five emotional states (fun, challenge, boredom, frustration, and excitement) from arousal and valence

The computing results of fuzzy reasoning were obtained by the application of fuzzy logic Petri nets [13]. A classical Petri net is defined as a structure $N = <S, T, F>$ where S means set of places, T is set of transitions and F is $F \subseteq (S \times T) \cup (T \times S)$, where $(\forall\ t \in T)(\exists\ p, q \in S)(p, t), (t, q) \in F$. Graphical representation is set up by the following symbols: *places* - by rings, *transitions* - by rectangles, and *relations* – by pointers between transitions and places or places and transitions. In classical Petri nets, there is a token placed if the expression is true (1) or not if it is false (0). Any *IF-THEN* rule is given in the form of

$$\text{IF } X_1 \text{ is } A_1 \text{ AND} \ldots \text{AND } X_n \text{ is } A_n \text{ THEN } Y \text{ is } B,$$

where $A_1,.., A_n$ and B are certain predicates characterizing the variables $X_1,...,X_n$ and Y. The set of *IF-THEN* rules forms the linguistic description:

$$R_1 := \text{IF } X_1 \text{ is } A_{11} \text{ AND} \ldots \text{AND } X_n \text{ is } A_{1n} \text{ THEN } Y \text{ is } B_1$$

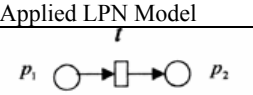
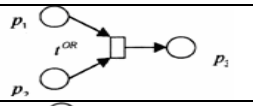
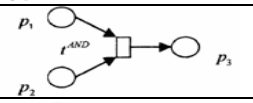
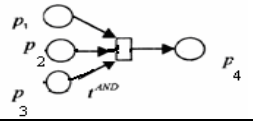
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$R_m := \text{IF } X_1 \text{ is } A_{m1} \text{ AND} \ldots \text{AND } X_n \text{ is } A_{mn} \text{ THEN } Y \text{ is } B_m$$

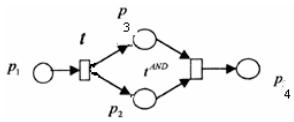
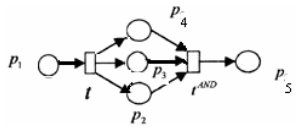
where each transition of the result fuzzy Petri net corresponds to one rule of linguistic description.

For recognition of diagnosis, 22 rules are constructed for determining the galvanic skin response (GSR), heart rate (HR), heart rate variability high (HRV_H), heart rate variability low, (HRV_L), and skin temperature (Table1).

Table 1. Models of Logical Petri Nets Applied to Transforming 89 Fuzzy Inference Rules to Constructing Support Information System for Bio Robots

No	Applied LPN Model	Fuzzy Computing	Applied Transitions
1		$\alpha_2 = \lambda_t \alpha_1 \ if\ \alpha_1 \geq \theta_t$	$T_1, T_3, T_{26}, T_{59}, T_{60}, T_{61}, T_{65}, T_{84}, T_{85}, T_{86}, T_{87}, T_{88}, T_{89}$
2		$\alpha_3 = \lambda_{t^{OR}} \max\limits_{\substack{\alpha_i \geq \theta_{t^{OR}} \\ i=1 \vee 2}} \{\alpha_1, \alpha_2\}$	$T_2^{OR}, T_4^{OR}, T_{20}^{OR}, T_{21}^{OR}, T_{22}^{OR}$
3		$\alpha_3 = \lambda_{t^{AND}} \min\limits_{\substack{\alpha_i \geq \theta_{t^{AND}} \\ i=1 \wedge 2}} \{\alpha_1, \alpha_2\}$	$T_{5\text{-}18}^{AND}, T_{23\text{-}25}^{AND}, T_{27\text{-}50}^{AND}, T_{62\text{-}64}^{AND}, T_{66\text{-}83}^{AND}$
4		$\alpha_4 = \lambda_{t^{AND}} \min\limits_{\substack{\alpha_i \geq \theta_{t^{AND}} \\ i=1 \wedge 2 \wedge 3}} \{\alpha_1, \alpha_2, \alpha_3\}$	T_{19}^{AND}

Continuation of Tabl.1

No	Applied LPN Model	Fuzzy Computing	Applied Transitions
5		$\alpha_2 = \lambda_t \alpha_1 \; if \; \alpha_1 \geq \theta_t$ $\alpha_3 = \lambda_t \alpha_1 \; if \; \alpha_1 \geq \theta_t$ $\alpha_4 = \lambda_{t^{AND}} \max\limits_{\substack{\alpha_i \geq \theta_{t^{AND}} \\ i=2 \wedge 3}} \{\alpha_2, \alpha_3\}$	$T_{51\text{-}52}$, $T_{55\text{-}58}$
6		$\alpha_2 = \lambda_t \alpha_1 \; if \; \alpha_1 \geq \theta_t$ $\alpha_3 = \lambda_t \alpha_1 \; if \; \alpha_1 \geq \theta_t$ $\alpha_4 = \lambda_t \alpha_1 \; if \; \alpha_1 \geq \theta_t$ $\alpha_5 = \lambda_{t^{AND}} m\min\limits_{\substack{\alpha_i \geq \theta_{t^{AND}} \\ i=2 \wedge 3 \wedge 4}} \{\alpha_2, \alpha_3, \alpha_4\}$	$T_{53\text{-}54}$

The set of 67 rules from [6] and corresponding transitions of logical Petri nets are used in the transforming of arousal-valence space into five modelled emotional states for converting arousal and valence into boredom, challenge, excitement, frustration, and fun. Table 1, shows some models of logical Petri nets applied to the transforming of 89 of fuzzy inference rules for constructing a real time support information system for bio robots of the model of Fig.1. A set of 22 rules and corresponding transitions is proposed for determining galvanic skin response (GSR), heart rate (HR), heart rate variability high (HRV_H), heart rate variability low (HRV_L), skin temperature of head (ST_H), and skin temperature of finger ($ST_{F)}$ into arousal and valence.

Table 2. Examples of rules and corresponding transitions proposed in concerting diagnosis from sensor's data

No	Rules	Transitions
1	If (GSR is *high*) then (arousal is *high*)	T_1
2	If (GSR is *high*) or (HR is *high*) then (arousal is *high*)	T_2^{OR}
3	If (GSR is *mid-low*) then (arousal is *mid-low*)	T_3
4	If (GSR is *low*) or (HR is *low*) then (arousal is *low*)	T_4^{OR}
5	If (GSR is *low*) and (HR is *high*) then (arousal is *mid-low*)	T_5^{AND}
6	If (GSR is *high*) and (HR is *low)* then (arousal is *mid-high*)	T_6^{AND}
7	If (GSR is *high*) and (HR is *mid*) then (arousal is *high*)	T_7^{AND}
8	If (GSR is *mid-high*) and (HR is *mid*) then (arousal is *mid-high*)	T_8^{AND}
9	If (GSR is *mid-low*) and (HR is *mid*) then (arousal is *mid-low*)	T_9^{AND}
10	IF (HRV_H is *high*) and (HRV_L is *low*) then (valence is *very-high*)	T_{10}^{AND}
11	IF (HRV_H is *low*) and (HRV_L is *high*) then (valence is *very-low*)	T_{11}^{AND}
12	IF (HRV_H is *medium*) and (HRV_L is *medium*) then (valence is *neutral*)	T_{12}^{AND}
13	IF (HRV_H is *high*) and (HRV_L is *medium*) then (valence is *high*)	T_{13}^{AND}
14	IF (HRV_H is *medium*) and (HRV_L is *high*) then (valence is *low*)	T_{14}^{AND}
15	IF (HRV_H is *medium*) and (HRV_L is *low*) then (valence is *high*)	T_{15}^{AND}
16	IF (HRV_H is *low*) and (HRV_L is *medium*) then (valence is *low*)	T_{16}^{AND}
17	IF (HRV_H is *high*) and (HRV_L is *high*) then (valence is *neutral*)	T_{17}^{AND}
18	IF (HRV_H is l*ow*) and (HRV_L is *low*) then (valence is *neutral*)	T_{18}^{AND}
19	IF (HR is *high*) and (HRV_H is *high*) and (HRV_L is *high*) then (valence is *high*)	T_{19}^{AND}
20	IF (ST_H is *high*) or (ST_L is *low)* then (valence is *low*)	T_{20}^{OR}
21	IF (ST_H is *low*) or (ST_L is *high*) then (valence is *high*	T_{21}^{OR}
22	IF (ST_H is *medium*) or (ST_L is *medium*) then (valence is *medium*)	T_{22}^{OR}

The set of 67 rules from [6] and corresponding transitions of logical Petri nets are proposed for determining the transformation of arousal-valence space into five modelled emotional states to convert arousal and valence into boredom, challenge, excitement, frustration, and fun (Table 3).

Table 3. Examples of description of rules from the set of 67 rules from [6]

No	Rules	Transitions
23	If (arousal is *not very low*) and (valence is *mid-high*) then (fun is *low*)	T_{23}^{AND}
24	If (arousal is *not low*) and (valence is *mid-high*) then (fun is *low*)	T_{24}^{AND}
25	If (arousal is *not very low*) and (valence is *high*) then (fun is *medium*)	T_{25}^{AND}
26	If (valence is *very high*) then (fun is high)	T_{26}
27	If (arousal is *mid-high*) and (valence is *mid-low*) then (challenge is *low*)	T_{27}^{AND}
28	If (arousal is *mid-high*) and (valence is *mid-high*) then (challenge is *low*)	T_{28}^{AND}
29	If (arousal is *high*) and (valence is *mid-low*) then (challenge is *medium*)	T_{29}^{AND}
30	If (arousal is *high*) and (valence is *mid-high*) then (challenge is *medium*)	T_{30}^{AND}
31	If (arousal is *very high*) and (valence is *mid-low*) then (challenge is *high*)	T_{31}^{AND}

Such rules are constructed as the schema of transitions of Logical Petri Nets proposed for determining the transformation of arousal-valence space into five modelled emotional states to convert arousal and valence into boredom, challenge, excitement, frustration, and fun.

To determine the emotions of users during their relaxation state, agents HARA-1, HARA-2, IDMA-1, and IDMA-2, presented in Fig. 1, were programmed using the following reasoning algorithm of fuzzy logical Petri nets. This algorithm receives a fuzzy Petri net as an input and creates a set of linguistic descriptions corresponding to each output place of a fuzzy Petri net. Human arousal recognition agents *HARA-1* and *HARA-2* from Fig. 1 were programmed to use these reasoning algorithms to create some friendly advice for disabled individuals.

```
input  : fuzzy Petri net: fpn
output: set of linguistic descriptions: lfln
lfln = ∅;
foreach output place op of fpn do // create linguistic description
    // create set of input variables (places) on whose op depends
    inputs = ∅;
    foreach input transition it of op do
        // add all inputs of transition it to inputs set
        inputs = inputs ∪ it.inputs;
    end
    // construct linguistic description (set of rules)
    rb = ∅;
    foreach input transition it of op do
        // construct rule corresponding to transition it
        rule = ∅;
        foreach element in from inputs do
            if rule ≠ ∅ then rule = rule + AND;
            if in ∈ it.inputs then
                rule = rule + in.name is edge(in, it).value;
            else
                rule = rule + in.name is UNDEF;
            end
        end
        rule = rule + THEN op.name is edge(it, op).value;
        rb = rb ∪ rule ; // add rule to rule base
    end
    lfln = lfln ∪ rb ; // add rule base to set of linguistic descriptions
end
```

4. Conclusions

An approach for developing the interaction architecture of mobile devices and remote server systems with additional functionalities for contextual information transmission is proposed. Some mobile

solutions are included for recognition of the location of moving objects in the process of monitoring remote agents. The recognition of the diagnosis of the emotional situation of disabled persons is based on a multi-layered model which integrates several techniques of knowledge representation: neural networks, fuzzy logic Petri nets, and evaluation of fuzzy neural control of speed of wheelchair type-robots working in real time. This was done by implementing movement support for disabled individuals using information based on the emotional state of the disabled persons.

The proposed framework uses four emotion recognition sensors for each disabled individual: the ECG (Electrocardiogram), the SCR (Skin Conductance Response), the ST_H (Skin Temperature of Head), and the ST_F (Skin Temperature of Finger) to provide HR(Heart Rate), HRV_H(Heart Rate Variability for the range of 0.15 to 0.4 Hz), HRV_L (Heart Rate Variability for the range of 0.015 to 0.15 Hz), SCR, ST_H , and ST_F inputs for defining fuzzy values of arousal and valence of disabled person.

The method of fuzzy reasoning using fuzzy logical Petri nets based on transforming arousal-valence space into five modelled emotional states to convert arousal and valence into boredom, challenge, excitement, frustration, and fun is described. The method allows t physiological state of disabled individuals to be defined, and gives them online advice based on the recognition of their emotions during their activities.

References

1. Pentland A. 2004. Healthwear: Medical Technology Becomes Wearable. *IEEE Computer*, *37(5):* 42-49.
2. Villon O., Lisett, Ch. 2006. A User-Modeling Approach to Build User's Psycho-Physiological Maps of Emotions using Bio-Sensors. *Proc. of IEEE Roman*, pp. 269-276.
3. Rubaai, A., Ofoli A.R., Burge, L. and M. Garuba. 2005. Hardware Implementation of an Adaptive Network-Based Fuzzy Controller for DC–DC Converters// *IEEE Transactions on Industry Applications, Vol. 41, No. 6:* 1557-1565.
4. Bielskis, A.A., Denisov, V., Kučinskas, G., Ramašauskas, O., Romas, N. 2007. Modeling of Human Physiological Parameters in an E-Laboratory by SOM Neural Networks// *Elektronika Elektrotechnika. Vol. 3(75):* 77-82.
5. Papillo J.F., Shapiro, D. 1990. The cardiovascular system. In: Cacioppo, J.T., Tassinary, L.G. (Eds.), *Principles of Psychophysiology: Physical, Social, and Inferential Elements*. Cambridge University Press, Cambridge, England, pp. 456–512.
6. Mandryk, R.L., Atkins, M.S. 2007. A fuzzy physiological approach for continuously modeling emotion during interaction with play technologies", *Int. J. Human-Computer Studies 65,* 2007, pp.329–347.
7. Rowe, D.W., Sibert, J., Irwin, D. 1998. Heart rate variability: Indicator of user state as an aid to human-computer interaction. Proc. of the *Conf. on human factor in computer science,* pp. 480-487.
8. Wilson, G.F. 1992. Applied use of cardiac and respiration measures: Practical considerations and precautions// *Biological Psychology, 34:* 163-178.
9. Korhonen, I. 1997. *Methods for the analysis of short-term variability of heart rate and blood pressure in frequency domain: PhD thesis,* VTT Publications. http://www.vtt.fi/inf/pdf/publications/1997/P316.pdf
10. Touzet, C. F. 1999. Neural Networks and Q-Learning for Robotics. *IJCNN '99 Tutorial,* International Joint Conference on Neural Networks, Washington, DC.
11. McCallum, R.A., 1995. Instance-based State Identification for Reinforcement Learning, *Advances in Neural Information Processing Systems 7*, MIT Press, USA.
12. Santos, J.M., Touzet, C. 1999. Dynamic Update of the Reinforcement Function during Learning// *Connection Science*, *Special issue on Adaptive Robots,* (C. Torras guest ed.).
13. Jiang, C., Zheng, Y. Fuzzy Reasoning Based on Petri Nets. http://www.ici.ro/SIC/sic2000_3/art04.html)
14. Bielskis, A.A., Denisovas, V., Drungilas, D., Gricius, G., Ramašauskas. O. 2008. Modeliing of Intelligent Multi-Agent based E-health Care System for People with Movement Disabilities// *Elektronika ir elektrotechnika*. Kaunas: Technologija, 6 (86): 37-42.
15. Pavliska V. 2006. Petri Nets as Fuzzy Modeling Tool. http://irafm.osu.cz/research_report/84_rep84.pdf)
16. Dzemydienė D., Maskeliūnas S., Jacobsen K. 2008 Sustainable management of water resources based on web services and distributed data warehouses// Technological and Economic Development of Economy: Baltic journal on sustainability.Vol. 14, No. 1: 38-50.

17. Booth D., Haas H., McCabe F., E. Newcomer, M. Champion, and Orchard C. F. D. 2004 Web Service Architecture. W3C Recommendation. http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/
18. Mahmoud H. 2004. J2ME and Location-Based Services.
19. Huang, C. M., C. H. Lee, and J. R. Zheng. 2006. A novel SIP-based route optimization for network mobility. *IEEE Journal on Selected Areas in Communications* 24(9): 1682–91.
20. Rosenberg J., H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and Schooler E. 2002. SIP: Session Initiation protocol, RFC 3261. http://www.ietf.org/rfc/rfc3261.txt
21. Johnson, D., C. Perkins, and J. Arkko. 2004. Mobility Support in IPv6. IETF RFC 3775.
22. Location API for J2ME (JSR 179). Java Specification Request, September 2003.
23. Gudgin M., Hadley M., Rogers T. 2006. Web Services Addressing 1.0 – Core. W3C Recommendation. http://www.w3.org/TR/ws-addr-core/
24. Mitra N. 2003. SOAP Version 1.2 Part 0: Primer. w3C Recommendation. http://www.w3.org/TR/2003/REC-soap12-part0-20030624/ 2.2
25. Bielskis A. A., Dzemydienė D., Denisovas V., Andziulis A., Drungilas D. 2009. An approach of multi-agent control of bio-robots using intelligent recognition diagnosis of persons with moving disabilities// *Technological and Economic Development of Economy*. 15(3): 377-394.