Hahoe KAIST Robot Theatre: Learning Rules of Interactive Robot Behavior as a Multiple-valued Logic Synthesis Problem

Marek Perkowski [1,3], Tsutomu Sasao [2], Jong-Hwan Kim [3], Martin Lukac [1], Jeff Allen [1], and Stefan Gebauer [4]

[1] Department of Electrical and Computer Engineering, Portland State University, Portland, Oregon, 97207-0751, [2] Department of Electronics, Kyushu Institute of Technology, Iizuka, Japan, [3] Department of Electronics and Computer Science, Korea Advanced Institute of Science and Technology, KAIST, Daejeon, Republic of Korea, [4] Facultaet fuer Elektrotechnik und Informationstechnik, Chemnitz University of Technology, Germany.

Abstract

The paper presents a new application of decomposition of multiple-valued relations. We developed a theatre of interactive humanoid robots, ***Hahoe KAIST Robot Theatre.*** *Version 2 includes three full body robots, equipped with vision, speech recognition, speech synthesis and natural language dialog based on machine learning abilities. The needs for this kind of project result from several research questions, especially in emotional computing and gesture generation, but the project has also educational, artistic, and entertainment values. It is a testbed to verify and integrate several algorithms in the domain of Computational Intelligence. Machine learning methods based on multiple-valued logic are used for representation of knowledge and machine learning from examples.*

1. Introduction

What is the mystery of puppet theatre? Puppets are only pieces of wood and plastic, and yet their viewers soon become immersed in the play and experience the artistic thrill of the drama. Does the art lay in the hand that animates the puppet—indeed, a human hand? Will it still be art if this hand is replaced by computer-controlled servomotors? What about animated movies? Children laugh and cry while perceiving a fast-changing sequence of pictures as a truly live action. The movement has been recorded once for all on a tape – it never changes – and yet this does not detract from its artistic value. Can this art be recorded in a computer program, as in video games, which are also an emergent art form? Another closely related art form is an interactive installation, in which the first robotic installations appear. What we want to do is push this boundary; by experimentally analyzing issues on the boundary of art, science and engineering, we hope to build new form of art and entertainment – the interactive humanoid robot theatre. Like the movies in the early Auguste and Louis Lumiere brothers era, interactive robot theatre is not an art yet, but is definitely capable of attaining this level (see the "Artificial Intelligence" movie by Spielberg). It is only a question of time and technology. We want to start progressing in this direction.

The existing robot theatres in the world, usually in theme parks and museums, are also at a very beginning stage. They are based on programmed movement synchronized with recorded sound. They do not use any Computational Intelligence techniques. There is no interaction with the child, for instance by voice commands. They do not teach a child much, either. Current robot toys for adults are programmable but do not learn from interactions with their users, and the keyboard programming necessary to operate them is too complex for many users. Thus, such robots are not applicable in advertisement and entertainment industries. Some toys or theatres have high quality robot-puppets that are not interactive and have no voice capabilities. Other "theatres" use computer-generated dialog but have very simple, non-humanoid robots, such as wheeled carts. Yet other theatres are based on visual effects of robot arm movements and have no humanoid robots at all [1,7,8,9,10,11]. Our ideal is a robot muppet that would draw from the immortal art of Jim Henson [6].

Since 2000, we have been in the process of designing a set of next-generation theatrical robots and technologies, that, when taken together, will create a puppet theatre of seeing, listening, talking and moving humanoids. Such robots can be used for many applications, such as video-kiosks, interactive presentations, historical recreations, assistive robots for children and elderly, foreign language instruction, etc. Thus, we can categorize them as "intelligent educational robots." These robots will truly learn from examples, and the user will be able to reprogram their behavior with a combination of three techniques: vision-based gesture recognition, voice-recognition and keyboard-typing of natural language dialog texts. Thus, programming various behaviors based on

Proceedings of the 35th International Symposium on Multiple-Valued Logic (ISMVL'05)
0195-623X/05 $ 20.00 **IEEE**

multi-robot interaction will be relatively easy and will lead to the development of "robot performances" for advertising and entertainment.

Thanks to a generous grant from KAIST, in 2003/2004 we were able to complete version 2 of the theatre with three robots. We related our robots to Korean culture and tradition. Several methods of machine learning, human robot interaction and animation were combined to build these robots, as embedded in their controlling and dialog software. The original machine learning method that has been developed by the PSU team [3,4,5,17,18,19,23,24,25,26,27] and is based on the induction of multiple-valued relations from data has been applied to these robots to teach them all kinds of verbal and non-verbal behaviors [2,3,4,5]. This logic method compiles optimized behavior rules from the examples of robot's behavior.

Here we discuss the basic technology and various versions of our theatre and especially its software. The long-term goal of this project is to perform both the theoretical research and the practical development leading to a commercial product.

2. The design of the Version 2 of the Theatre

2.1. Background on Hahoe KAIST Robot Theatre

The most advanced robot head of "theatrical" type, *Kismet,* is a long-term research project from MIT that uses a network of powerful computers and DSP processors. It is very expensive, and the head is much oversized, so it cannot be commercialized as a "theatre robot toy". Also, it does not use a comprehensive speech and language dialog technology, has no machine learning and no semantic understanding of spoken language. Japanese robot toys; *Memoni, dog.com* and *Aibo* have primitive language understanding and speech capabilities. They have no facial gestures at all (Memoni), they have only head movements (Aibo) or only 3 degrees of freedom (to open mouse, move eyes and head in dog.com.) The Japanese robots cost in the range of $120 (Memoni and dog.com) to $850 (the cheapest Aibo). We developed interactive humanoids with more complex behaviors than any of these toys. Although our technology involves a specialized computer network, the mechanical robots themselves, if mass-produced, could be sold for less than $1000 each.

Since 2000 we have been conducting studies of all kinds of robot puppets**,** robot theatres, head and face robots, robot-human and computer-human dialogs, speech recognition and speech synthesis, theories of learning and language acquisition, intelligent toys and especially all toys with speech recognition abilities. As the result of these studies, we developed several talking robot heads as well as a walking hexapod robot with a human face at PSU. Finally in Summer 2002 we developed a talking robot head called Professor Perky in Iizuka, Japan. In September – December 2002 a KAIST student, Mr. Taehoon Lee developed the head/neck mechanical construction of Hahoe Robot Yangban [12]. The skills gained and the experiences of past works were next used in years 2003/2004 to build the version 1 in Portland. Version 2 that includes the last act of Hahoe was done in Daejeon. The demo of version 2, integrated and mostly build by Jeff Allen, was shown in KAIST on August 27, 2004. A theatre of two lions (act 1 of Hahoe) has been built since August 2002 and is still under development by Martin Lukac. This version has two interacting lion-head robots that were demonstrated in October 2004. Martin supervised also a group of high school students and Portland Area Robot Society hobbyists on the remaining robots for the version 3 Hahoe theatre (in preparation). Currently the work towards version 3 is done by several graduate, undergraduate and high-school students. Theatre design appeals to high school volunteers who have interests in robot programming, mechanical design or art. Thanks to skills gained, each next version not only has more realistic movements and dialogs, but is also less expensive.

Building a robot theatre can be done only based on many partial experiences accumulated over years, and this experience has to include knowledge of components, design techniques, programming methods and available software tools. The successful robot design should be not costly, because our goal is that the project will be repeated by other universities and colleges. Importantly, our mechanical design is competitive since we developed a very inexpensive technology based on "radio-controlled servos" that are used in airplane models. From what we know, nobody so far in the world created a complete theatre of talking humanoid robots and heads (remember that many existing "robot heads" are not humanoid and are intended only for visual tracking—they are not related at all to robot theatre, dialog, speech and facial gestures animation). Our theatre is expensive because simultaneous computer vision, speech, dialog and control of three robots require at least three Pentium-4 class computers in the network. Our robot, however, is relatively inexpensive. We hope, therefore, then when the cost of multi-processors will go down, our complete theatre software would be able to work on a single computer which will make this project of interest to a much wider audience.

There is research in several places in Japan and USA on talking robot heads, but none of these projects is similar to the one presented here. The version 2 robot speaks only English but the version 3 robots should speak English and Korean. The robot design and behavior are strongly influenced by Korean culture and traditions (for instance, by the way how the robots look, speak, behave and sing). This is another innovative asset of our work.

2.2. Research objectives

The main research objectives of this project were the following:

1. Develop inexpensive and interactive robot puppets based on the well-known Korean mask play of Hahoe Pyolshin-gut T'al-nori as it is played in Hahoe Village (called in short Hahoe): mechanical design, artistic expressions, computer interface. The movements and speech behaviors of these puppets should be highly expressive. Develop basic level of software with all kinds of parameterized behaviors that are necessary to play a complete puppet play.
2. Develop and analyze an "animation language" to write scripts that describe both verbal and non-verbal communication of the robots interacting with themselves and with the public. The language should include: "language of emotions", language of dialog, language of interaction and language of control.
3. Apply this technology to make a complete performance of Hahoe Theatre, but with a limited set of actors (three actors in version 2).
4. On top of this technology, develop a machine-learning based methodology that will create expressions in animation language based on examples. This will allow the robots to: (1) understand a limited subset of English natural language, (2) talk in English, and (3) be involved in a meaningful verbal and non-verbal interactive natural language (English) dialog with humans, but limited mostly to subjects of the Hahoe play.

Below we will briefly discuss some issues related to meeting these objectives in the practical settings of this project.

2.3. Mechanical design of Hahoe Theatre

How should the robots look like? How to create a technology that is both inexpensive and well-suited for robot theatre and interactive displays? In contrast to current toys that are either robot arms, animals with non-animated faces or mobile robots, our "theatre robot" is a set of three synchronized humanoid robots with computer vision, speech recognition and speech synthesis abilities and a distributed computer network. In the next phase, version 3, the theatre will be expanded to all twelve Hahoe humanoids. Mechanically, building a robot theatre is both a scientific and artistic task. So far, the best robot puppets were built in Hollywood, by resident "robot-artitsts" in top US Universities and in some Japanese universities. For version 2 we built three full body robots: Jin Kyung Lee – the Narrator, Sonbi – the Confucian Scholar, and Paekchong – the bad butcher. We built also: Yangban – the aristocrat - a talking head, and Pune – the flirtatious girl (mask/head only). All robots except for the Narrator have wooden mask faces. The Narrator is a latex face robot with more degrees of freedom. Sixteen degrees of freedom allows to show more emotions and facial gestures. This robot has been added to the original Hahoe play by us to aid the ease of understanding the mediaeval Korean culture and increase the artistic value of the performance. With the experience gained, we will soon write detailed instructions how to mechanically complete the remaining robots for the Version 3 Hahoe performance. The mechanical construction of the corresponding heads is described in reports [13,15,16]. The hand construction is in papers [20, 22]. Let us observe that the Hahoe robots' faces look exactly the same as masks sold in many shops in Korea, TV performances and human performances in Hahoe Village. Few small modifications have been done to masks for more DOF animation (like enlarging eyes). The advantage of this approach is that the image of these faces is very familiar to Korean people and to interested theatrical audiences worldwide. The faces of Paekchong (Fig. 1) and Sonbi (Figure 2) are quite expressive even without movement animation. Also, these masks are known to show different "emotions" even with very simple animation of only the jaw and using different angles and colors of script-controlled lights.

Figure 1. Paekchong the bad butcher robot. Note the moveable jaw

Many more effects can be achieved having at least 10 degrees of freedom for a robot. In facial animation, we added many more expressions, but these facial expressions that are well known from the fixed sculptures of the traditional masks have been preserved. The robots have built-in microphones and cameras. The degrees of freedom (DOF) differ from robot to robot but may include: jaw opening (also asymmetric jaw movements, at least 2 DOF), eyes movements (2 to 4 DOF), 2 DOF eyebrows, 2 DOF eyelids (for latex robot only), three DOF for the neck, 6 DOF for hands, 4 DOF for legs, and one DOF (up-down) for the body. Inexpensive servo motors and servos with increased torque used in air models (Hitec, Futaba) were applied.

2.4. The Hahoe Play and its adaptation.

Hahoe Pyolshin-gut T'al-nori, a combination of the mask dance, drama, and shaman rituals, consists of 9 different acts (i.e. episode, madang). It originated in thirteen century in Korea and is one of the oldest, if not the oldest in the world, continuous tradition of a folk theatre. Therefore it is well known world-wide among the specialists in ethnography, anthropology, puppet theatre and folk art. The play made fun of a ruling class, using humor and satire, and thus tried to ease tensions among the different social classes. In addition, the shaman rituals performed prayers for a prosperous year and were supposed to prevent natural disasters. This play is played at Hahoe Maul (Hahoe Folk Village) in Andong Province of South Korea. People from all over the world come to this village to watch performances. Historically, these masks were used in the village ritual of Pyolshin-gut T'al-nori, a shaman rite for exorcising evil spirits, which was performed in early January in the lunar calendar in order to pray for an abundant harvest and peace. The original eleven masks still exist and are now preserved in the Korean National Museum in Seoul. In 1964, the masks were designated as National Treasure No. 121 for their highly artistic value.

Figure 2. Sonbi the Confucian Scholar robot. Note eyes enlargement to make the eye movement more visible. There are color LEDs in the eyes.

The Hahoe play has been adapted by us from various original Korean and English-translation sources to robots that are stationary. They do not walk. When we write "dance" in the script language, the robots just shake and perform dancing movements to the music, but they bodies stay in the same place. The shamanic, ceremonial and religious parts are removed from this robot performance, dance of the original performance is replaced with music and songs sung by robots. Because we want to perform this play for US children audiences, we are rewriting now the original script to actualize the play and remove obscenities. There will be at least two variants of the play.

Figure 3. Kyng Jin Lee robot without skin. Please note the sensors on cheeks and nose and a camera on top of the head.

Figure 4. The Narrator robot with skin.

In our theatre there are two interaction types in every performance. The first type is like a theatre performance where the sentences spoken by the actors are completely "mechanically" automated using the XML-based Common Robotics Language (CRL) that we developed [28]. The same is true for body gestures. The individual robot actions are programmed or/and graphically edited (section 4.4). All robot movements, speech, lights and other effects are controlled by a computer network. Every performance is the same, it is "recorded" once for all as a controlling software script, as in Disney World or similar theme parks.

The second type, much more interesting and innovative, is an interactive theatre in which every performance is different, there is much improvisation and interaction of robots with the public. (The software scripts are not fixed but are learnt in interaction processes of humans and robots). In case of human theatre such elite experimental performances are known from the "Happening Movement", Grotowski's Theatre, Peter Brooks' Theatre, and others top theatre reformers in the world. In this part, the public is able to talk to the robot-actors, ask them questions, play games, and ask to imitate gestures. This is when the robots demonstrate language understanding, improvisational behaviors, "emergent" emotions and properties of their characters. Autonomous behavior and automatic speech recognition is demonstrated only in this second part. The second type has many levels of difficulty of dialog and interaction and what was practically demonstrated in version 2 is only the first step. In every performance the two interactive types are freely intermixed.

2.5. Challenges to meet Hahoe tradition

This project brings totally new challenges: (1) What should be the voices of the robots? (2) How to animate emotions, including emotional speech patterns? (3) How to combine digitized speech with text-to-speech synthesized voice? (4) What is the role of interactive dialogs? (5) How to animate gestures for interactive dialogs? (6) How to use uniformly the machine learning technology -- that we developed earlier -- to the movements, emotions, voice, acting and dialogs? (7) How much of the script should be predefined and how much spontaneous and interactive? (8) Role of archetypes of Korean culture, role model, traditions, art, ethical and esthetic values specific to Korea. (9) Mechanical design to create a space of movements in which emotions typical for these characters can be programmed. (10) Development of a language of voice synthesis and emotion modeling that will be easy enough to be used by artists (directors) that will program future performances without our help.

The design has been done with the goal in mind that the whole performance should be no longer than 45 minutes, each run of it should be sufficiently different to make it not boring for a viewer of repeated performances. While in the dialogs the Narrator is programmed to be a modern highly cultural and educated Korean student who even knows quantum computing, other robots have the "personalities" of thirteen-century feudal Korean society whom they represent, with all the limitations of these people. Currently, the Narrator robot answers questions related to Korean history, tax system under Choson Dynasty, Yangbans, their concubines, Confucian scholars, Hahoe play and village, and quantum computing. She can also sing and tell jokes. The texts of the entire performance and interactive dialogs were censored by native Koreans to avoid cross-cultural insensitivities of the play's authors/directors.

3. Software development

3.1. Principles of software

The system is programmed in Visual Basic 6, Visual Basic.NET and other languages from Visual Studio of Microsoft. We use the most modern technology for speech recognition and speech synthesis (Fonix and Microsoft SAPI). Vision programming uses heavily OpenCV from Intel. We use ready tools in which the internals of speech processing or vision are accessed from Visual Basic or Visual C environments. These are high quality tools, the best available, but of course the expectation for the recognition quality must be realistic. What we achieved so far is speech recognition of about 300 words and sentences, which is enough to program many interesting behaviors, since speech generation and movement are more important for theatrical effect than speech recognition. The commercial speech tools are improving quickly in every few months, and they are definitely far ahead university software. In general, our advice to robot builders is: “use the available multi-media commercial technology as often as possible”.

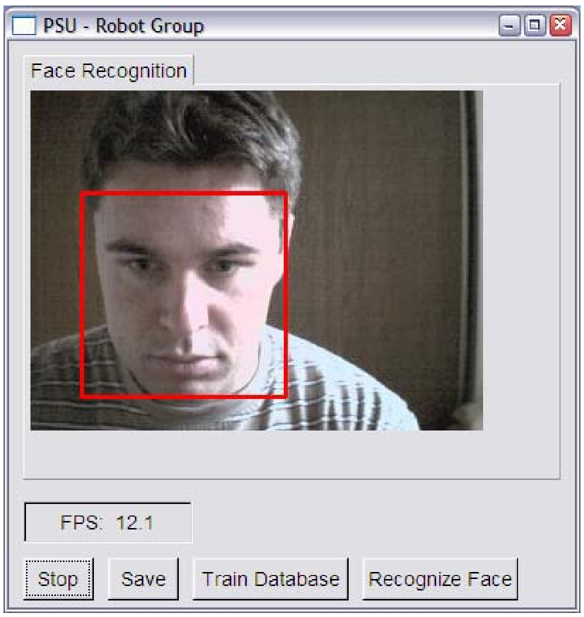

Figure 5. Face detection localizes the person(red rectangle around the face) and is the first step for feature recognition and face recognition.

Let us remember that words communicate only about 35 % of the information transmitted from a sender to a receiver in a human-to-human communication. The remaining information is included in: body movements, face mimics, gestures, posture, external view - so called para-language. Face detection (see Figure 5) can find where the person is located, thus aiding the tracking and movement of the robot. Figure 6 shows the human recognition software that learns about the human and what he/she communicates to the robot. Of course, here we are limited in animations of any of the above because of the robot mechanical construction and limited DOFs. It is therefore interesting that high quality artistic effects are achieved in Hahoe Village by actors playing in masks, thus also limited in their expressions – this is one of the reasons that we selected masks for our theatre rather than latex or more complicated robot faces. The CRL scripts link the verbal and non-verbal behavior of the robots. Movement animation is an art more than science.

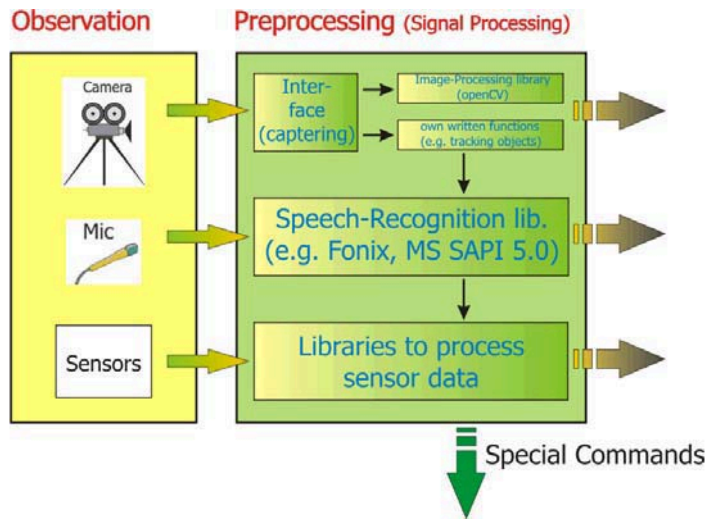

Figure 6. Acquiring information about the human: face detection and recognition, speech recognition and sensors.

In brief, the dialog/interaction programming has the following characteristics:

1. It includes Eliza -like dialogs based on pattern matching and limited parsing. Memoni, Dog.Com, Heart, Alice, and Doctor all use this technology, some quite successfully. For instance, Alice program won the 2001 Turing competition. This is a “conversational” part of the robot brain and a kind of supervising program for the entire robot, based on blackboard architecture principles. We use our modification of Alice.
2. Model of the robot, model of the user, scenario of the situation, history of the dialog are all used in conversational programs.
3. Use of both word spotting and continuous speech recognition. The detailed analysis of speech recognition requirements can be found in [14].
4. Avoiding *“I do not know”, “I do not understand”* answers from the robot. Our robot will have always something to say, in the worst case, nonsensical and random. Value “*zero*” of every variable in learning means “*no action*”. False positives lead to some strange robot behaviors with additional unexpected movements or words, while every false negative leads to an avoidance of the action corresponding to this variable. Thus, in contrast to standard learning from examples, we are not afraid of false positives, on the contrary, they often create fun patterns while observing the results of learning.
5. Random and intentional linking of spoken language, sound effects and facial gestures. The same techniques will

be applied for theatrical light and sound effects (thunderstorm, rain, night sounds).

6. We use parameters extracted from transformed text and speech as generators of gestures and controls of jaws (face muscles). This is in general a good idea, but the technology should be further improved since it leads sometimes to repeated or unnatural gestures.

7. Currently the robot tracks the human with its eyes and neck movements. This is an important feature and we plan to enhance it. To maintain eye contact with the human gives the illusion of robot's attention. Camera is installed on the head. In future there will be more than one camera for a robot. There will be also more "natural background" behaviors such as eye blinking, breathing, hand movements, etc. Simplified diagram of the entire software is shown in Figure 7.

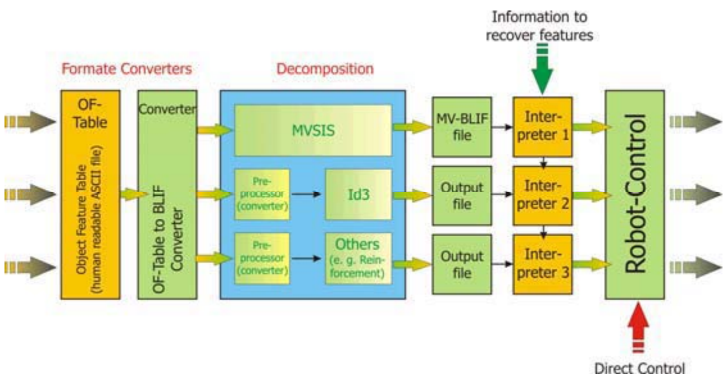

Figure 7. A simplified diagram of software explaining the principle of using machine learning to create new interaction modes of a human and a robot. ID3 is a decision tree based learning software, MVSIS is the general purpose multiple-valued tool used here for learning by decomposition and other MV logic synthesis methods. The input arrows are from sensors, the output arrows are to effectors.

3.2. Software modules

Here are the main software modules.

- **Motor/Servo**. Driver class with a large command set, relative and direct positioning as well as speed and acceleration control and positional feedback method.
- **Text To Speech.** Microsoft SAPI 5.0, Direct X DSS speech module because of its good viseme mapping and multiple text input format
- **Speech Recognition** Microsoft SAPI 5.0, using an OCX listbox extension the speech recognition can be easily maintained.
- **Alice**. One of the most widely used formats for Alice languages on the Internet uses *.aiml files, a compatible openSource version was found and modified.
- **Vision**. An openSource Package by Bob Mottram using a modified OpenCV dll that detects facial gestures was modified to allow tracking, and mood detection.
- **IRC Server**. To allow for scalability an OpenSource IRC server was included and modified so that direct robot commands could be sent from a distributed base.
- **IRC Client**. An IRC client program was created to link and send commands to the robot, this will allow for future expansion. Coded for in both .NET and VB 6.

3.3. Layers

In order for the illusion to work each module must interact in what appears to be a seamless way. Rather than attempting to make one giant seamless entity, multiple abstractions of differing granularities where applied. The abstractions are either spatial or temporal, the robot's positional state is taken care of at collective and atomic levels, i.e. *Right arm gesture(x),* and *left elbow bend (x),* where the collective states are temporal and dynamic.

All functions are ultimately triggered by one of multiple timers. The timers can be classified as major and minor timers; major timers run processes to process video, sound, etc. and are usually static in frequency settings and always are enabled, minor timers are dynamic and based on situation they are enabled and disabled routinely throughout the operation of the robot. To help mask the mechanical behavior of the robot more, many of the major timers are intentionally set to non harmonic frequencies of one another to allow for different timing sequences and a less *"clock like"* nature.

Motions can be broken into reflex, planned / gesture related, and hybrid functions, with some body parts working in certain domains more than others. The mouth and eyes are mostly reflex, with arms being planned, and the neck slightly more hybrid. Each body part of course crosses boundaries but this allows for each function to be created and eventually easily prototyped. In fact the actual accessing of all functions becomes ultimately reflexive in nature due to the use of triggers that all ultimately result from reflexive and timed reflexive subroutines. For instance, a person says "hello" moments later a speech recognition timer triggers, Alice runs and generates a response, the response creates mouth movement, the mouth movement occasionally triggers gesture generation, all types of functions triggered by just one of the reflex timers.

3.4. Alice, TTS and SR

A standard Alice with good memory features was employed as the natural language parser. Microsoft SAPI 5.0 has a Direct Speech object which can read plan text, just like that provided from the Alice engine, it also has viseme information that easily is used to control and time mouth and body movement in a structured form.

MS SAPI 5.0 was also used for the speech recognition. With most speech recognition programs the library of words it searches are either based on Zipf's law or have to be loaded with a tagged language, meaning on the fly generation of new language recognition is troublesome. Fortunately SAPI has a listbox lookup program that requires no extra tagged information. The problem with using a finite list is that SAPI will attempt to identify things so hard it will make mistakes quite often. To combat this, short three- and two- letter garbage words for most phonemes were created. The program will ignore any word three letters or less when not accompanied by any other words.

Currently the mouth synchronization can lag due to video processing and speech recognition, to combat this on a single system computer the video stream had to be stopped and started during speech. The video processing and speech recognition are extremely taxing for one system and in order to get optimal responses the robot should be improved with the addition of a wireless 802.11 camera to enable other computers to do the video processing.

4. Using Constructive Induction in robot learning architecture

4.1. Constructive induction

While commercial dialog systems are boring with their repeating *"I do not understand. Please repeat the last sentence*" behaviors, our robots are rarely "*confused*". They always do something, and in most cases their action is slightly unexpected. This kind of robot control is impossible for mobile robots and robot arms but is an interesting possibility for stationary entertainment robots. It combines also some logic and probabilistic approaches to robot design that are not yet used in robotics. In addition to standard dialog technologies mentioned above, a general-purpose logic learning architecture is used that is based on Data Mining "Constructive Induction" methods that we developed in last 10 years at PSU [3-5,17-19,21,23-27], and just recently applied to robotics [2,21,14]. We assume that the reader has a general understanding of this approach and here we concentrate only on theatre application aspects.

The general learning architecture of our approach can be represented as a mapping from features to behaviors. The are two aspects of learning: (1) preparing input-output vectors, (2) generalizing from care input-output vectors (minterms) to don't care input-output vectors. While the second aspect has been much discussed in our previous papers, the first aspect is especially of interest to robot theatre. The feature values are extracted from three sources: camera, speech recognition and skin/body sensors. They are stored in a uniform language of input-output mapping tables with the rows corresponding to examples (input-output vectors, minterms of characteristic functions) and the columns corresponding to feature values of input variables (visual features, face detection, face recognition, recognized sentences, recognized information about the speaker, in current and previous moments) and output variables (names of preprogrammed behaviors or their parameters, such as servo movements and text-to-speech). This is a standard format in logic synthesis, Data Mining, Rough Set and Machine Learning. Such tables are created by encoding in the uniform way the data coming from all the feature-extracting subroutines. Thus the tables store examples for mappings to be constructed. If the teaching data is encountered again in table's evaluation, the same exactly output data from the mapping specified by the table is given as found by the teaching. But what if a new input data is given during evaluation, one that never appeared before? Here the system makes use of analogy and generalization based on <u>constructive induction</u> [24-27].

4.2. Teaching by examples

The input-output vector serves for "teaching by example" of a simple behavior. For instance the vector can represent a directive "*If the human smiles and says "dance" then the robot dances*". Observe that this directive requires the following: (1) the camera should recognize that the person smiles, (2) this is encoded as a value of the input variable *"smile"* in set of variables "*facial features"* (Figure 9*),* (3) the word-spotting software should recognize the word "*dance*", (4) the word "*dance*" is encoded as a value of variable "*word_command*", (5) the output variable *"robot_action"* obtains value *"robot_dances",* (6) there must be a ready subroutine "*robot_dances*" with recorded movements of all servomotors and text-to-speech synthesis/recorded sound". This directive is stored in robot memory, but more importantly, it is used as a pattern in constructive induction, together with other input-output vectors given to the robot by the human in the learning phase. Observe in this example that there are two components to the input-output vector. The input part are symbolic variable values representing features that come from processing of the sensor information. They describe "*what currently happens*". The teacher can give for instance the command to the robot: "*if there is THIS situation, you have to smile and say hello*". "This situation" means what the robot's sensors currently perceive, including speech recognition. When the robot communicates in its environment with a human (we assume now that there is only one active human in the audience) the input variables of the vector continuously change, for instance when the person interrupts the smile, says another word or turns away from the camera. The output part of the vector is some action (behavior) of the robot. It can be very simple, such as frowning or telling "*nice to meet you*" to complex behaviors such as singing a song with full hands gesticulation.

Examples of feature detection are shown in Figures 8 and 9. Eye, nose and mouth parameters of a human are put to separate windows and the numerical parameters for each are calculated (Figure 8). The symbolic face in right demonstrates what has been recognized (in this example smile was correctly recognized, eyebrows were correctly recognized but eyes direction were not).

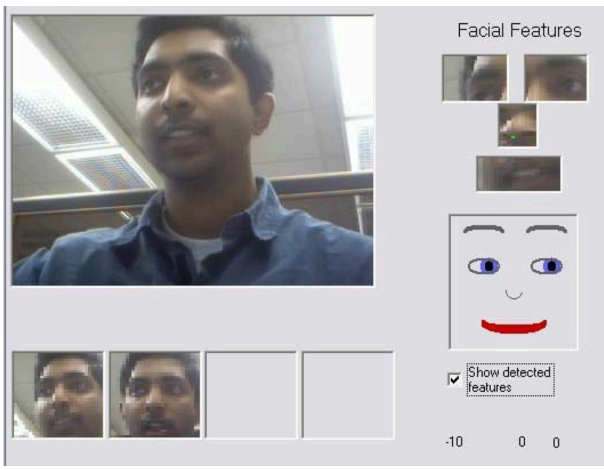

Figure 8. Face features recognition and visualization.

The teaching process is the process of associating perceived environmental situations and expected robot behaviors. The robot behaviors are of two types. One type are just symbolic values of output variables, for instance variable *left_hand* can have value "*wave friendly*" or "*wave hostile*", encoded as values 0 and 1.

The actions corresponding to these symbols have been previously recorded and are stored in the library of robot's behaviors. The second type of symbolic output values are certain abstractions of what currently happens with robot body and of which the robot's brain is aware. Suppose that the robot is doing some (partially) random movements or recorded movements with randomized parameters. The input-output directive may be *"if somebody says hello then do what you are actually doing"*. This means that the directive is not taking the output pattern from the memory as usually, but is extracting parameters from the current robot's behavior to create a new example rule. This rule can be used to teach robot in the same way as the rules discussed previously. Finally there are input-output vectors based on the idea of reversibility. There can be an input pattern which is the symbolic abstraction of a dancing human, as seen by the camera and analyzed by the speech recognition software. Human's behavior is abstracted as some input vector of multiple-valued values. Because of the principle of "symmetry of perception and action" in our system, this symbolic abstraction is uniquely transformed into an output symbolic vector that describes action of the robot. Thus, the robot executes the observed (but transformed) action of the human. Moreover, robot can generalize this pattern by applying the principles of constructive induction that are used consistently in our system. This is a form of combining the learning by example (or mimicking) and the generalization learning.

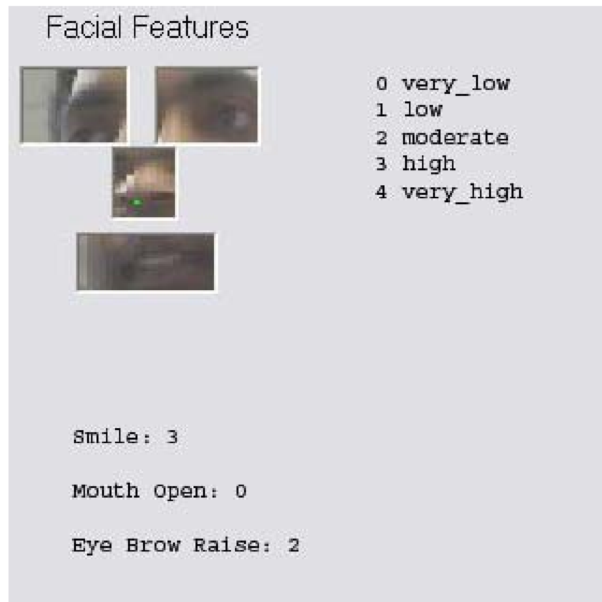

Figure 9. Use of Multiple-Valued (five-valued) variables Smile, Mouth_Open and Eye_Brow_Raise for facial feature and face recognition.

4.3. Using "Multiple-Valued Logic Synthesis" for Robot Learning

Data in tables are stored as binary, and in general multi-valued, logic values. The teaching examples that come from preprocessing are stored as (care) minterms (i.e. combinations of input/output variable values). We use logic synthesis algorithms such as ID3, Ashenhurst/Curtis hierarchical decomposition and Bi-Decomposition algorithms in our software [2,3,4,5] (See Figure 7). These methods create our subset of MVSIS system developed under Prof. Robert Brayton at University of California at Berkeley [2]. The entire MVSIS system can be also used. The bi-decomposer of relations and other useful software used in this project can be downloaded from http://www-cad.eecs.berkeley.edu/mvsis/. As explained above, the system generates robot's behaviors from examples given by users. This method is used for embedded system design, but we use it specifically for robot interaction. It uses a comprehensive Machine Learning/Data Mining methodology based on constructive induction and particularly on decomposing hierarchically decision tables of binary and multiple-valued functions and relations to simpler tables, until tables of trivial relations that have direct counterparts in behaviors are found. In contrast to Neural Nets or reinforcement learning, the constructive induction methods are based on logic, combinatorial optimization and highly efficient data structures such as Binary Decision Diagrams [3,4,5]. Our approach can be thought of as a generalization of Decision Trees known from DM, ML and Robotics (ID3). It is similar to methods used in design automation of digital logic and it satisfies Occam Razor principle. In terms of quality (not speed) of learning it compares very well to NNs and Decision Trees in experiments [5].

MVSIS and the BLIF format are used to realize the learning of the robot. BLIF (Berkeley Logic Interchange Format) is used to describe circuits in text format. It is used in minimization programs as input and output file format MVSIS used to minimize circuits and got an extension for multiple-valued variables. To provide the information that the robot processes we created a format close to the BLIF format but better readable.

We explain some of these principles on an extremely simplified example. Let's use the following fictional scale for the properties or features of each persons: a = smile degree, b = height of a person, c = color of the hair. (For simplification of tables, we use four values of variable "smile" instead of five values as in shown in Figure 9). These are the input variables. The output variable Age has four values: 0 for kids, 1 for teenagers, 2 for grownups and 3 for old people. The characteristics of feature space for people recognition is given in Figure 10. The input-output mapping table of learning examples is shown in Figure 11. All variables are here quaternary. The learned robot behavior is the association of the input variables (*Smile, Height, Hair Color*) with the action corresponding to the perceived age of the human (an output variable *Age*). Thus, the action for value Kid will be to smile and tell "Hello, Joan" (the name was learned earlier and associated with the face). If the value Teenager is the output Age of value propagation through learned network with inputs Smile, Height and Hair Color, then the action of the robot "Crazy Move" and the text "Hey, Man, you are cool, Mike" is executed, and so on for other people. The quaternary map (a generalization of Karnaugh Map called Marquand Chart, variables are in natural and not Gray code) in Figure 12 shows the cares (examples, objects) in presence of many "don't cares". The high percent of don't cares, called "don't knows" is typical for logic synthesis applied to Machine Learning. These don't knows are converted to cares as a result of learning the expression (the network). This is illustrated in Figure 13, where the solution [Age=3] = a^0 is found, which means – old person is a person with value low of variable smile. In other words, the robot learned here from examples that old people smile rarely. Similarly, it is found that [Age=0] = a^3 which means that children smile opening mouth broadly. Observe that the learning in this case found only one meaningful variable – Smile and the two other variables are vacuous. Observe also that with different result of learning (synthesizing the minimum logic network for the set of cares) the solution would be quite different, [Age=3] = c^3 which means, "old people have grey hair". The bias of a system is demonstrated by classifying all broadly smiling people to children or all albinos to old people. Obviously, the more examples given, the lower the learning error. Another interesting examples of using constructive induction in robotics are in [2,21].

Smile	Very High	High	Middle	Low
Values	3	2	1	0
Height	Very Tall	Tall	Middle	Short
Values	3	2	1	0
Color	Grey	Black	Brown	Blonde
Values	3	2	1	0

Figure 10. Space of features to recognize age of people.

Name (examples)	**Age (output)**	Smile	Height	Hair Color
Joan	Kid	a(3)	b(0)	c(0)
Mike	Teenager	a(2)	b(1)	c(1)
Peter	Mid-age	a(1)	b(2)	c(2)
Frank	Old	a(0)	b(3)	c(3)

Figure 11. Input-output mapping of examples for learning (cares).

A unified internal language is used to describe behaviors in which text generation and facial and body gestures are unified. This language is for learned behaviors. Expressions (programs) in this language are either created by humans or induced automatically from examples given by trainers. Our approach includes deterministic, induced and probabilistic grammar-based responses controlled by the language. Practical examples are presented in [2,21,28]. Observe that the language is naturally multiple-valued. Not only it has multiple-valued variables for describing humans and situations (like {*young, medium,* old}, {man, woman, child}, or face features / behavior nominal variables such as *smile, frown, angry, indifferent*) but has also multiple-valued operators to be applied on variables, such as minimum, maximum, truncated sum and others.

ab\c	0	1	2	3
00	-	-	-	-
01	-	-	-	3
02	-	-	-	-
03	-	-	-	-
10	-	-	-	-
11	-	-	-	-
12	-	-	2	-
13	-	-	-	-
20	-	-	-	-
21	-	1	-	-
22	-	-	-	-
23	-	-	-	-
30	0	-	-	-
31	-	-	-	-
32	-	-	-	-
33	-	-	-	-

Figure 12. The quaternary Marquand Chart to illustrate or cares (learning examples) for age recognition example.

ab\c	0	1	2	3
00	-	-	-	-
01	-	-	-	3
02	-	-	-	-
03	-	-	-	-
10	-	-	-	-
11	-	-	-	-
12	-	-	2	-
13	-	-	-	-
20	-	-	-	-
21	-	1	-	-
22	-	-	-	-
23	-	-	-	-
30	0	-	-	-
31	-	-	-	-
32	-	-	-	-
33	-	-	-	-

Figure 13. One result of learning. The shaded rectangle on top has value 3 for all cells with a=0. The shaded rectangle below has value 1 for all cells with a=2

The generalized functional decomposition method, that hierarchically and iteratively applies the transformation, sacrifices speed for a higher likelihood of minimizing the complexity of the final network as well as minimizing the learning error (as in the Computational Learning Theory). For instance, this method automatically generalizes spoken answers in case of insufficient information.

Figure 14 shows the appearance of the Robot control tool to edit actions. On the right there is the "Servo Control Panel". Each servo has its own slide bar. The slide bar is internal normalized from 0 to 1000. As one can see, the servo for the eyes reaches from left to right and the two number next to each slide bar is the position of the slider (e.g. half way slider is always 500) and the next number on the most right is the position of the servo. These values are different for every servo and are just for control, that the servo is in its range and works properly. In addition to that there is a checkbox on each slide to select the servo for a movement.

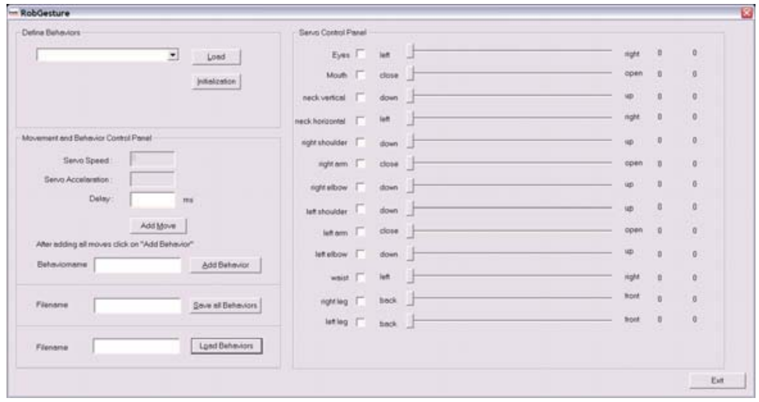

Figure 14. Robot control tool to edit actions.

On the left side there is the "Define Behaviors" box in which one can select a particular movement and then load it to the servos. The "Initialization" button must be pressed at the beginning to make sure that all servos are in their initial position. The initial positions are given in the servo.ini file. The file looks like this:

```
//Servo.ini
eyes:2600 1690 780
mouth:2000 2000 3200
neck_vertical:600 2000 3300
neck_horizontal:2500 1000 -700
right_shoulder:3700 2900 -400
right_arm:3400 1720 45
right_elbow:-700 -700 2800
left_shoulder:60 650 4000
left_arm:500 1300 3800
left_elbow:3200 3200 -500
waist:1100 2320 3400
right_leg:-300 1500 3200
left_leg:3200 1700 -100
```

This file provides all servo information that is necessary. The first line shows the details for servo #1. The first number is the most left value, the second is the initialization value and the last value is the most right value for the servo.

Below the "Define Behaviors" field we have the "Movement and Behavior Control Panel". In the first field the user enters the delay for the particular movement in milliseconds. Then there is the "Add Movement" Button to simply add a movement to a behavior. When he is done with defining all movement for a behavior he enters a name for the behavior and clicks the "Add Behavior" Button. With the edit fields "Load Behaviors" and "Save Behaviors" he can load and save the programmed behaviors. It is so easy that 10-years old children have programmed our robot in Intel's high-tech show.

5. Conclusions

Intelligent educational robots are quite different from other known types of robots. They integrate several soft computing methodologies and multimedia and control software modules. We created an innovative and captivating robot system to communicate with people and based on various machine learning methods.

Using multiple-valued logic for Data Mining and Machine Learning is not a new idea [3-5,30]. However, so far it has been not used for robotics except for [2,21,31]. This way, huge amount of real-life data can be accumulated for multiple-valued logic synthesis; creating benchmarks for multiple-valued logic minimizers has always been a difficult problem. Now we can create many benchmarks from practical examples. This will help comparing our methods, their variants and other machine learning approaches.

Our robots were demonstrated to several audiences and seem to be very liked, especially by children, who can teach simple robot behaviors by themselves. Much work is however still needed, especially in integration and to improve the robustness and reliability of system's hardware and software.

Acknowledgements

The authors are thankful to Korea Advanced Institute of Science and Technology and particularly to the deans of the Department of Electronics and Computer Science, Profs. Myung-Jin Chung and Yong Hoon Lee, for the sympathetic support of this project. We would like also to thank the following people who contributed to the success of Hahoe Theatre: Jake Biamonte (robot Narrator version 1), Kyle Reid (robot Paekchong), Normen Giesecke, Akashdeep Aulakh, Myron Machado and Atsumu Iseno (robot Sonbi, version 1), Pete Skaggs (head design), Kyung Jin Kim (robot character animation and music selection, version 3), Prof. Eunjeong Lee (costume design for version 3). David Ng designed the early variant of the Narrator robot head and the latex skins for versions 2 and 3. Version 2 of the theatre would not have been possible without Mr. Minbae Jeong's immense knowledge of Korean culture, traditions and history. He also proofread and edited the entire CRL/AIML script and interactive dialog files. We are very grateful for his altruistic and enthusiastic help to create the theatre. Ms. Grace Choi donated her high-quality hanbok for the Narrator robot. Several robots for version 3 theatre are still under design, started Fall 2003 by a group of high school students from Oregon, including Brandon Lei, David Rosenbaum, Mao Mao Zhang, Kevin Xue and Kevin Chen. We are very grateful for their help.

Literature

1. M. Perkowski, "Oregon Cyber Theatre," *Proc. 3rd Oregon Symposium on Logic, Design and Learning*, May 2000.

2. U. Wong and M. Perkowski, "A New Approach to Robot's Imitation of Behaviors by Decomposition of Multiple-Valued Relations," *Proc. 5th Intern. Workshop on Boolean Problems*, Freiberg, Germany, Sept. 19-20, 2002, pp. 265-270

3. A. Mishchenko, B. Steinbach and M. Perkowski, "An Algorithm for Bi-Decomposition of Logic Functions," *Proc. DAC* 2001, June 18-22, Las Vegas, pp. 103-108

4. A. Mishchenko, B. Steinbach and M. Perkowski, "Bi-Decomposition of Multi-Valued Relations," *Proc. 10th IWLS,* pp. 35-40, Granlibakken, CA, June 12-15, 2001. IEEE Computer Society and ACM SIGDA.

5. M. Perkowski, and S. Grygiel, "Decomposition of Relations: A New Approach to Constructive Induction in Machine Learning and Data Mining – An Overview," *Proc. Workshop of National Inst. Of Telecommunications and Polish Association for Logic and Philosophy of Science*, May 25, 2001, pp. 91-111, Warsaw, Poland.

6. Ch. Finch, Ch. Finch and J. Henson, "Jim Henson: The Works: The Art, The Magic, the Imagination," see also other books by and about Jim Henson on Amazon.com.

7. A. Edsinger, U-M O'Reilly, and C. Breazeal, "A Face for a Humanoid Robot," *MIT Memo*, 2000

8. F. Hara, H.Kobayashi, F.Iida, and M. Tabata, "Personality characterization of animate face robot through interactive communication with Human," *Proc. 1st Int'l Workshop on Humanoid and Human Friendly Robots,* pp. 1-10, 1998.

9. C. Breazeal, "Sociable Machines: Expressive Social Exchange Between Humans and Robots" *Ph.D. Thesis,* MIT, 2000.

10. Edsinger and U-M. O'Reilly, "Designing a Humanoid Robot Face to Fulfill a Social Contract," *MIT Memo,* 2002

11. C. Breazeal and B. Scassellati, "Infant-like social interactions between a robot and a human caretaker," In *Adaptive Behavior,* August 2000.

12. T. Lee, “Hahoe Robot Head,” Project report, *RIT Lab of KAIST,* Daejeon, December 2002.

13. D. Ng, and M. Perkowski, “Construction of a robot head,” *report,* Portland, 2003.

14. M. Perkowski, T. Sasao, “Professor Perky Talking Robot Head,” *report*, Iizuka. August 2002.

15. J. Allen, *report*, Daejeon, November 2004.

16. S. Gebauer, *report*, Portland, October 2004.

17. S. Grygiel, M. Zwick, M. Perkowski, “Multi-level decomposition of probabilistic relations,” *Kybernetes: The International Journal of Systems & Cybernetics,* Vol. 33, Number 5/6, 2004, pp. 948 - 961.

18. A. N. Al-Rabadi, M. Perkowski, M. Zwick, “A comparison of modified reconstructability analysis and Ashenhurst-Curtis decomposition of Boolean functions, “ *Kybernetes: The International Journal of Systems & Cybernetics,* Vol. 33, Number 5/6, 2004, pp. 933-947.

19. P. Burkey and M. Perkowski, ``Efficient Decomposition of Large Fuzzy Functions and Relations," *Proceedings of International Conference on Fuzzy Information Processing. Theories and Applications,* March 1 -- 4, Beijing, China, Tsinghua University Press and Springer, pp. 145-154.

20. M. Folgheraiter, G. Gini, M. Perkowski, and M. Pivtoraiko, ``Blackfingers: a Sophisticated Hand Prosthesis," *Proceedings of ICORR 2003 (the 8th International Conference on Rehabilitation Robotics),* April 22-25, 2003, KAIST, Korea, pp. 238 -- 241.

21. M. Perkowski, T. Sasao, A. Iseno, U. Wong, M. Pivtoraiko, M. Folgheraiter, M. Lukac, D. Ng, M. Fix and K. Kuchs, ``Use of Machine Learning based on Constructive Induction in Dialogs with Robotic Heads," *Proceedings of ICORR 2003 (the 8th International Conference on Rehabilitation Robotics),* April 22-25, 2003, KAIST, Korea. pp. 326 -- 329.

22. M. Folgheraiter, G. Gini, M. Perkowski, and M. Pivtoraiko, ``Adaptive Reflex Control for an Artificial Hand," *Proceedings of 7th IFAC Symposium on Robot Control, SYROCO 2003,* Holliday Inn, Wroclaw, Poland, 1-3 September, 2003.

23. B. Steinbach, M. Perkowski, and Ch. Lang, ``Bi-Decomposition of Multi-Valued Functions for Circuit Design and Data Mining Applications," *Proc. ISMVL'99*, pp. 50 - 58, May, 1999.

24. M. Perkowski, S. Grygiel, Q. Chen, and D. Mattson, ``Constructive Induction Machines for Data Mining," *Proc. Conference on Intelligent Electronics,* Sendai, Japan, 14-19 March, 1999.

25. M. Perkowski, L. Jozwiak, and S. Mohamed, ``New Approach to Learning Noisy Boolean Functions," *Proc. ICCIMA'98 Conference,* February 1998, Australia, published by World Scientific, pp. 693 - 706.

26. C. Files, and M. Perkowski, ``Multi-Valued Functional Decomposition as a Machine Learning Method," *Proc. ISMVL'98,* May 1998.

27. C. Files, and M. Perkowski, ``An Error Reducing Approach to Machine Learning Using Multi-Valued Functional Decomposition," *Proc. ISMVL'98,* May 1998.

28. M. Lukac, several working reports on Common Robotics Language and scripts in CRL, 2004.

29. Ch. Brawn, *report*, Portland, June 2004.

30. A. Ngom, D. Simovici, I. Stojmenovic, “Evolutionary Strategy for Learning Multiple-Valued Logic Functions”, *Proc. ISMVL* 2004, pp. 154-160.

31. A. Saffiotti, K. Konolige, E.H. Ruspini, “.A Multivalued Logic Approach to Integrating Planning and Control,” *Artificial Intelligence*, 1995.