COMMUNICATION LANGUAGE FOR AUTONOMOUS MULTI-ROBOT SYSTEMS

Peggy Doerschuk[1]
peggy.doerschuk@lamar.edu

Ling Wang[2]
ling.wang@EnGlobal.com

Amit Keswani[1]
mt_keswani@yahoo.com

Narsing Mudhiraj Lingala[1]
narsing3@gmail.co

Tapan Singh Mokha[1]
tapan113@yahoo.com

David O. Doerschuk[3]
david@OrganResearch.com

[1]Lamar University
Computer Science Department
P. O. Box 10056, Beaumont, Texas 77710
USA

[2]EnGlobal Engineering, Inc.
3155 Executive Blvd 216, Beaumont, TX 77705
USA

[3]Organ Research LLC
239 Pinemont, Beaumont, Texas 77659
USA

ABSTRACT
This paper presents a model for constructing basic functional communication between autonomous robots. The Communication Language for Autonomous Multi-Robot Systems integrates concepts from Speech Act Theory, Agent Communication Language, and network protocols into a six-layer model. The model generates a syntactically simple, concise, yet extensible language structure for task-oriented autonomous robots. The layered approach makes it possible to add or change a layer without impacting the other layers. This provides flexibility to change the interaction protocol used for transmitting messages if desired. This model has been used successfully with both infrared-based and Bluetooth-based interaction protocols. The latter is described here.

KEY WORDS
autonomous robot, wireless communication, Bluetooth

1. Introduction

For humanoid robots, a natural language may be designed based on speech recognition. For human-centered industrial robots, a machine command set may be designed using a remote control device. A language for cooperative task-oriented autonomous robots lies between these two ends of the spectrum. The Communication Language for Autonomous Multi-Robot Systems (CLAMR) Model is a six-layer framework that serves as a prototype for constructing a communication language between task-oriented cooperative autonomous robots.

2. Background

In autonomous robot systems, there are five types of approaches to communication: none, implicit communication through the environment, simple semaphores or signals, simple message passing, and agent communication language [1]. The latter is the theoretical foundation of CLAMR.

2.1 Speech Act Theory, Agent Communication Language, and Language of Machine

According to Searle, "Speaking a language is performing speech acts, such as making statements, giving commands, asking questions, making promises, and so on [2]." The Speech Act Theory (SAT) of language states that the intentionality of language is to perform an act [3].

This theory can be used as a basis for teams of task-oriented robots that cooperate to perform their tasks. The tasks are the acts that are to be performed, and speech acts are used to communicate information related to these tasks. Communication reduces the perceptual burden of individual robots and facilitates the performance of tasks that require cooperation.

There are four basic categories of Speech Act [3]: utterances, propositional utterances, illocutionary utterances and perlocutionary utterances. At the simplest level, to utter is simply to say a word with no particular forethought or intention to communicate a meaning. In the act of making a propositional utterance, the speaker gains the opportunity to interact. Illocutionary utterances are usually sentences that contain propositional utterances; that is, they refer to things in the world -- but it is their intentional nature that is of the most importance. However, they don't necessarily require that the listener change his or her behavior. Perlocutionary utterances, on the other hand, do attempt to effect a change [4].

The Foundation for Intelligent Physical Agents ACL Agent Communication Specification [5] claims that an ACL is a language in which communicative acts can be expressed and hence messages constructed. It consists of three parts: (1) Interaction Protocols, which deal with pre-agreed message exchange protocols; (2)

Communicative Acts, which deal with different utterances; and (3) Content Languages, which deal with different representations of the content. FIPA ACL is a well-defined framework that allows agents to communicate with meaningful statements that convey information about their knowledge, based on SAT. However, the specification ignores the four categories of utterances in SAT described above.

The language of machines [6] is a task-based command set with data attributes, which should be simple, concise, and yet extensible. For most computing purposes, speech acts are classified into assertive (informing), directive (requesting), commissive (promising), permissive, prohibitive, declarative, and expressive. According to this classification, the basic types of Speech Act are defined: Inform, Request, Query, and Acknowledge (Accept/Refuse). This narrows down the ACL research into the simple language of machines, which bridges the gap between ACL and the language of robots.

2.2 Network Protocols

A basic description of the OSI seven-layer network architecture [7] is:

- Physical layer: conveys the bit stream through the network at the electrical and mechanical level. It provides the hardware means of sending and receiving data.
- Data Link layer: encodes and decodes data packets into bits. This layer is divided into two sublayers: the Media Access Control (MAC) and Logical Link Control (LLC). MAC focuses on multi-access and collision problems, while LLC handles error detection and recovery, flow control and frame synchronization.
- Network layer: provides switching and routing as well as addressing, internetworking, congestion control and packet sequencing.
- Transport layer: splits messages into smaller packets, assigns sequence and dispatches them.
- Session layer: sets up, coordinates and terminates conversations, information exchanges and dialogues between the applications at each end.
- Presentation layer: translates application requests into network operations. It acts as an application interface so that syntax, formats and codes are consistent between two network machines.
- Application layer: supports application and end-user processes.

Media contention occurs when communications operate on a single channel. Two local area networks, Ethernet and Token Ring, are designed to handle collision [8]. Ethernet is based on Carrier Sense Multiple Access Collision Detect (CSMA/CD) technology. When a device has data to send, it first listens to see if any other device is currently using the network. If not, it starts sending its data. While transmitting it listens to detect if a collision occurs. A collision occurs when two devices send data simultaneously. When a collision happens, each device waits a random length of time before resending its data. A token passing mechanism applied in a Token Ring network is designed to let the device wait until it has the token and then send its data, which is deterministic.

2.3 Bluetooth Communication for robot systems

McCain [9] used the Bluetooth protocol on iPAQ 3970s for communication in a small multi-robot system. The iPAQ 3970s do not provide support for a network wherein multiple robots can communicate simultaneously but instead allow for a direct serial communication between two devices. McCain's team used a staggered message passing scheme wherein a robot wishing to communicate would open a serial communication channel, pass its message and then close the channel. The authors concluded that Bluetooth was a viable option for multi-robot communication schemes. This work focused on strategies for multi-robot communication, not the communication language itself.

3. The Multi-Robot System

Prototypes of the CLAMR model have been implemented on two multi-robot systems. The first used robots with HandyBoard controllers and IR as the physical medium [10]. A more recent implementation uses IntelliBrain-Bot robots and the Bluetooth protocol.

The IntelliBrain-Bot [11] is an educational robot that is Java programmable. RoboJDE [12] is used to program the IntelliBrain. It supports inheritance, interfaces, multi-threading, thread synchronization, floating point arithmetic and other essential features of Java. It also provides an API that lets user programs interface with the IntelliBrain hardware, sensors and actuators [13].

The IntelliBrain has more memory and a faster processor than the HandyBoard. It is controlled by an Amtel ATmega 128 microcontroller. It can connect to a host computer or interface with other serial devices through a high speed RS232 serial connection on its COM1 port. We use this port to connect the robot to an Aircable Industrial mote [14], a Bluetooth compatible device that can communicate with any Bluetooth device and also has a built-in temperature sensor. The AirCable is programmed in BASIC. It uses RF signals for communication. Its 3.3dBi antenna gives the capability of communicating in a range of 150 feet. Other antennas can provide a range of 35m to 700m.

4. The CLAMR Model

In conformance with the ACL Communication Specification, CLAMR consists of three parts: Interaction Protocol, Speech Act, and Content Language. The six-layer CLAMR framework, which is shown in Figure 1, is designed based on these three portions.

The Interaction Protocol part of CLAMR is critical for making reliable contact between communicating robots. The physical layer ensures physical transmission of information bits; the data link layer is responsible for valid message delivery; the network layer provides for routing of the messages; the session layer provides the basic rules for conversation and dialogues; and the presentation layer acts as the interface between the Interaction Protocol layers and the application layer.

Together, the Content Language and Communicative Act portions comprise the application layer. The Speech Act portion of CLAMR contains the Communicative Acts layer, which is the core and the basic unit of robot communication language. Speech Act refers to illocutionary and perlocutionary utterances. Robots express their communication intention and their attempt to change others' behaviors through the Speech Act. The simplest and propositional utterances of robots can only be clearly understandable as communicative acts when they are contained within a pre-agreed upon conversation that is defined in the Session Layer of CLAMR. The Language part of CLAMR is defined to complement the meaning of the Communicative Acts and express complex communicative acts (sequences, conditional acts, etc.)

4.1 Application, Presentation, and Session Layers

The Language and Speech Act portions together form the Application Layer. The Application Layer processes the messages embodied by the communicative acts and content language portions, effecting the communicated speech act. These portions are implemented on the robot using the RoboJDE programming environment.

The Presentation Layer provides the interface between the robot that is requesting communication and the lower layers of the Interaction Protocol. It handles recasting of data types, formatting and padding of the message structure. It likewise is programmed on the robot in Java.

This SES Layer provides a fixed dialogue for each communicative act. The basic principle of this layer is to give the proper response to an incoming message. Because the design for the SES strongly depends on the next layer – Communicative Acts (CA), more detail will be included in the next section. The session layer is implemented by a BASIC program on the Aircable Bluetooth device.

4.2 Communicative Acts and Content Language Design

CA is the core layer of the whole CLAMR model. The basic skeleton for communicative acts is:

< Speech Act Type> < Action / Performative > < Parameters>

As shown in Table I, there are six SATs for the CLAMR: Inform, Request, Query, Acknowledge, Offer and RequestWhen. The first four types are the basic SATs. However, Acknowledge provides extra meaning in the CLAMR. It can be used to express the robot's willingness to accept a proposal; to admit a truth; or to notify others of its task performance status (whether the job is done or not). The message that implies Acknowledge as the SAT without a performative type is considered the simplest CA, an acknowledgement called OK, and it is used as a basic response which does not require the protocol support from the SES.

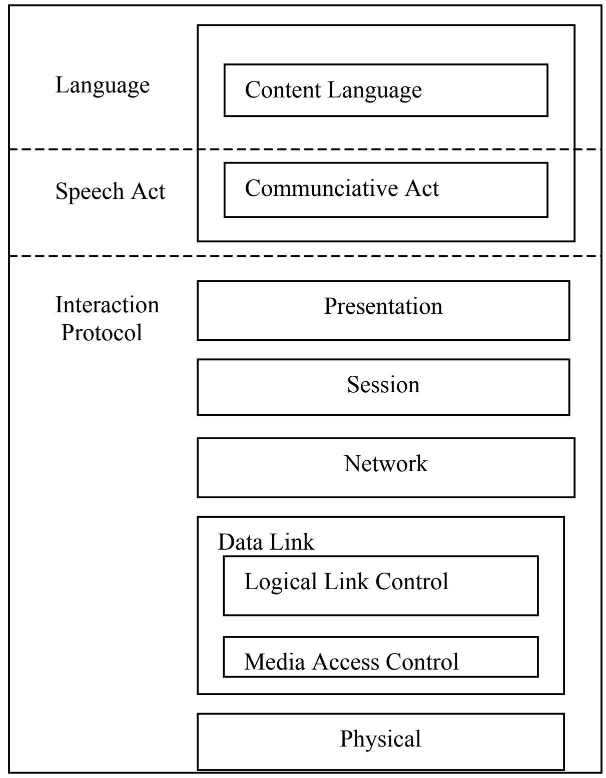

Figure. 1 Hierarchy of the Layered CLAMR Model

Table 1. Speech Act Types for CLAMR

Speech Act Type (SAT)	Description
Inform	Tell something
Request	Request a performance
Query	Ask for information
Acknowledge	Accept a proposal, answer a question, or notify of a status
Offer	Offer to perform in the future
RequestWhen	Request performance when signaled

RequestWhen allows synchronous cooperation between the sender and receiver, since there is no common clock for robots and they are using asynchronous communication. When a robot receives a CA in which the SAT is RequestWhen, the robot understands that it will also receive a synchronous signal at any moment triggering the beginning of that performance.

The SES protocols for Acknowledge, Inform, Request and RequestWhen are illustrated in Figures 2 and 3. The only difference between the latter two is that in a RequestWhen, the receiver will not execute the request until a start signal is received. Session protocols for Query and Offer are also specified in [10].

For the sake of brevity, only a sample of possible actions is listed in Table II. Additional actions can be added to support the functionality of the target robots. Three types of performatives, Accept, Admit and Report, are designed for the particular SAT – Acknowledge.

Table III gives a general view of possible parameters for selected actions. The parameters must be fine-tuned to support the functionality of the robots.

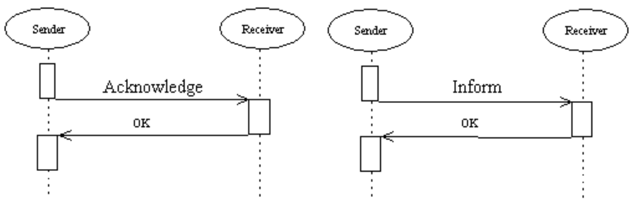

Figure. 2 Session Protocol for Acknowledge and Inform

Content Language is designed to represent complex communicative acts, including sequences of acts, conditional acts, compound acts, selective acts, etc. Eight types of logical relationships between CAs are supported in CLAMR, as shown in Table IV. These are similar to those found in Knowledge Interchange Formalism (KIF), which includes both a specification of syntax for the language as well as a specification for the semantics [15].

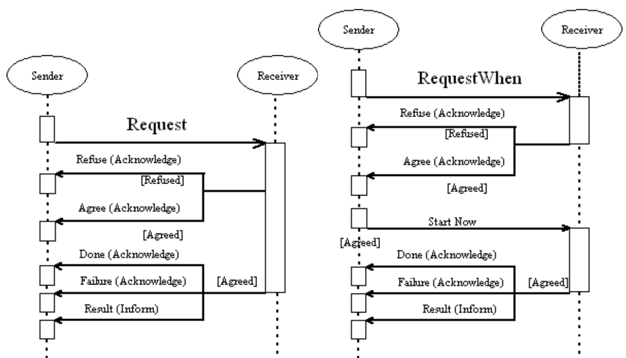

Figure. 3 Session Protocol for Request and Request When

4.3 Network Layer Design

The Aircable mote can establish wireless virtual serial cable connections to only two other devices. In one connection, it is the master, and in the other it is the slave. This limits the topology of the multi-robot connections via virtual cables to a ring, wherein each device in the ring is connected to its master in one direction and its slave in the other direction.

Another factor that influences the choice of a Ring network is the lack of buffering capability possessed by the Aircable Industrial device. If an Aircable device is performing another operation, say sensing surrounding temperature, when a message is sent, the message is not buffered for the device, which results in the loss of the transmitted message. No mechanism exists through which the lost message can be retrieved, thereby reducing reliability. Also collisions are possible when more than one robot tries to transmit information simultaneously, which again results in loss of unrecoverable messages.

These problems are overcome by choosing a Ring network that saves a considerable period of connection time, since we do not need to establish a connection before sending every message. With the use of a Ring network, all the connections need to be established just once during the initialization. Only the device that currently holds the token is capable of transmitting data. The others are in a listening mode, thereby preventing collisions or other possibilities of losing transmitted data. Once the robot decides that it has no more information to transmit, it passes the token to the immediate neighbor and waits on its slave port for the reception of any message from its Master. If none of the robots wish to communicate with others, the token keeps revolving around the Ring until one of the robots decides to talk. When a robot has information to transmit, it grabs the token from the Ring and uses it for an entire session of communication. After the termination of this communicative session, the token is released by the robot for use by the others.

Table 2. CLAMR Action / Performative List

Action Performative	**Meaning / Purpose**
Move	General movement
Stop	Stop ongoing performance
Pickup	Pickup object of certain description
Drop/Deposit	Drop/Deposit specified object
Sense	Note a sensor reading
Accept	Accept/Refuse request/query/ proposal
Admit	Admit truth, answer yes/no question
Report	Report current status of performance

Table 3. CLAMR Sample Parameter List

Action	**Possible Parameters**			
Move	Dir	Dis	Per	Spd
Pickup	Col	RGB	Shp	Siz
Sense	Temp	Press	Light	

Table 4. Complex CAs Supported in CLAMR

Type	**Description (by data)**	**Description (by meaning)**
Sequence	NA	Perform CAs in order
And	two values in the same data attribute	Perform both CAs
Or	either of the values in the data attribute	Perform either CA
Greater	all the values greater than the one as data	NA
Smaller	all the values smaller than the one as data	NA
Range	all within the range of two data values	NA
OutRange	all outside the range of two data values	NA
If	NA	Perform the next CA if this CA is true

However, the use of a Ring network also comes with its share of restrictions. The network does not allow direct communication between robots since each message needs to follow the ring until the destination is reached. The absence of direct communication consumes more time than usual but is a far better option in terms of time and efficiency than one in which a connection needs to be established for every message sent or received.

Another drawback of a Ring network is that failure of even a single device breaks the Ring thereby making such networks highly sensitive to failures. In order to revive the Ring, the Aircable devices must have an idea about the Ring topology. In the current implementation each Airmote has knowledge about its next two immediate neighboring motes. In this way, if a device detects the failure of its immediate neighbor, it has the address of the device that will be its new immediate neighbor, so the Ring can be formed again without human intervention.

4.4 Data Link and Physical Layers

The Aircable Industrial mote is a Bluetooth enabled device, so the implementation of the physical layer and data link layer is realized by the built-in Bluetooth technology. Bluetooth has a built-in, 128 bit encryption and PIN code authentication. Whenever any Bluetooth devices connect with each other, they exchange a PIN code for authentication [16]. The 128 bit encryption also ensures secured connection.

5. Bluetooth Aircable-based Interaction Protocol Structure

The basic message structure is shown in Figure 4. Its size is fixed at 30 bytes, the maximum size needed to accommodate a CLAMR message. The figure shows the first eleven bytes of the packet. The remaining higher order bytes can contain data, if required, to represent parameters of the actions specified by the ACT bits.

The first two bytes of the packet contain the sender address and the receiver address, respectively. The next two bytes specify the type of the token to be distributed. The combination "10" represents a general token, and the combination "01" represents a session token.

The next 3 bytes of the packet constitute the meaning part of the packet. The 5th byte from the top represents the CL values. The first four bits in this byte are used to represent the types of complex communicative acts specified in Table IV. The remaining four bits are used to specify constraints, such as time and location. The next two bytes are used to represent the CA values. The CA values are determined by the combination of SAT and ACT bits. The first three bits specify the speech act type from Table I, and the remaining five bits specify the action or performative from Table II to be executed by the receiver. The next byte in the packet is used to specify the parameter bits. These bits and their significance depend on the action to be performed, as illustrated in Table III.

The next bytes can be used to represent the data. The number of bytes to represent the data bytes differs from packet to packet. The exact number of bytes for data is determined by the required parameters to be represented for performing the intended task. For example, a float can take 4 bytes of data using C. The same float takes 8 bytes if Java is used to represent the data. The representation of data depends upon the robot being used and its microcontroller. The IntelliBrain robot is programmed using RoboJDE. The integers are represented in 4 bytes; the float takes 8 bytes.

Sender ID		0	0	0	0	0	0	1	1
Receiver ID		0	0	0	0	0	0	1	0
Token Bytes		0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	1
Meaning	CL	0	0	0	0	0	0	0	0
	CA	1	1	0	0	0	0	1	1
	Para	0	0	0	0	0	0	0	1
Data		0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0

Figure 4. The Bluetooth Aircable Interaction Protocol message structure.

6. Example: A request to sense temperature

This section illustrates communication using CLAMR of a request from Robot 2 to Robot 3 to sense the temperature. Figure 5 shows the frame format for content language for the Request message sent from mote2 to mote3. This is a simple request to "SENSE" with no location and time constraints, so the CL value is 0.

The byte format for the CA is given in Figure 6. The first byte of CA specifies the Speech Act Type and Action. The second byte specifies the parameters for the action that is being requested. The bits for SAT specify that the sent packet is a request. The ACT bits specify that

KIF						TIME	LOC
0	0	0	0	0	0	0	0

Figure 5. CL byte format for "Temperature Request" session

SAT2	SAT1	SAT0	ACT4	ACT3	ACT2	ACT1	ACT0
0	1	0	0	1	0	0	0
0	0	0	0	0	0	1	0
						Temperatue	

Figure 6. CA and Parameter byte format to "SENSE" the temperature

the requested action is SENSE. The temperature bit in the parameter byte is set to 1 to specify that the robot is requested to SENSE the temperature. The token bytes specify the type of the token. The combination "01" shows that it is a session token. The possessor of this token can only talk to the sender of the session token. The complete packet format is as shown in the Figure 7.

Mote3 receives the token and gives it to its own Robot3. Robot3 decodes the packet. It sends an acknowledgement to Robot2 and performs the operation. To perform the operation, Robot3 sends a request to its own mote to sense the temperature and return it. After receiving the temperature from its mote, Robot3 forms a packet with SAT "acknowledge" and the "done" message. The packet format for the acknowledgment and done message is shown in Figure 8. The content language byte does not put any restriction on the packet. The SAT bytes of the Communicative Act byte specify that it is an acknowledgement. As the SAT = 6, the ACT bits specify that the acknowledgement type is "Report." The last bit in the parameter byte is set for Report, showing that the request or task is done successfully. The resultant temperature data is specified in the last 4 bytes of data.

Sender ID	0	0	0	0	0	0	1	0
Receiver ID	0	0	0	0	0	0	1	1
Token Bytes	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	1
Content Language	0	0	0	0	0	0	0	0
Communicative Act	0	1	0	0	1	0	0	0
Parameters	0	0	0	0	0	0	1	0
Data	0	0	0	0	0	0	0	0

Figure 7. The first 8 bytes of the packet representing the request of Robot2 to Robot3 to SENSE the temperature

Sender ID	0	0	0	0	0	0	1	1
Receiver ID	0	0	0	0	0	0	1	0
Token Bytes	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	1
Content Language	0	0	0	0	0	0	0	0
Communicative Act	1	1	0	0	0	0	1	1
Parameters	0	0	0	0	0	0	0	1
Data	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	1	0	0	0
	0	0	0	0	0	0	1	0

Figure 8. The figure shows the first 11 bytes of the acknowledgement, with DONE message, of Robot3.

7. Conclusion

CLAMR successfully integrates concepts from Speech Act Theory, Agent Communication Language, and network protocols to develop a model for constructing a syntactically simple, concise, yet extensible language for task-oriented cooperative autonomous multi-robot systems. Because the language is concise there is no need to split messages into multiple packets, which reduces the overhead of communication. The framework presented here can be fine-tuned to fit the functionality of the specific robots. CLAMR's layered model makes it easy to add or change a layer without impacting the other layers.

The main limitation of the current implementation is the token ring network that was dictated by the limited connection capability of the Aircable motes. The passing of the token consumes time and energy. However, for small networks of robots the speed of communication afforded by the virtual serial cable connections still makes communication feasible even though messages must go around the ring. Also, the Bluetooth communication range is limited.

References

[1] B. R. Duffy, Towards social intelligence in autonomous robotics: a review, *Robotics, Distance Learning and Intelligent Communication Systems (RODLICS)*, Malta, September 1-6, 2001.

[2] J. R. Searle, *Speech acts: An essay in the philosophy of language* (London: Cambridge Univ. Press, 1969).

[3] K. Allan, Speech act theory -- An overview, *Encyclopedia of language and linguistics*, Vol. No. 8, 1994, pp. 4127-4138.

[4] J. Moeschler, Speech act theory and the analysis of conversations, Vanderveken D. & Susumu Kubo (eds), *Essays in Speech Act Theory*, (Philadelphia, PA: J. Benjamins Pub. Co. 1996) pp. 239-61.

[5] Foundation for Intelligent Physical Agents website www.fipa.org 2006.

[6] E. Jacobus and B. Duffy, The language of machines, *Proc. the 1st International Symposium on Information and Communication Technologies*, Dublin, Ireland, 2003, pp. 409-414.

[7] N. Briscoe, Understanding the OSI 7 layer model, http://www.pcsupportadvisor.com/nasample/t04124.pdf (accessed January 10, 2004), 2000.

[8] P. J. Bruckmeyer, E. M. Schwartz, and A. A. Arroyo. 2000. Hybrid communication network for autonomous agents. *Florida Conference on Recent Advances in Robotics (FCRAR)*, Florida Atlantic University, Boca Raton, Florida, May 4-5, 2000.

[9] J. T. McClain, B. J. Wimpey, D. H. Barnhard, and W. D. Potter, Distributed Robotic Target Acquisition using Bluetooth Communication, *Proceedings of the ACMSE04*, 2004, pp. 291-296.

[10] L. Wang, A Prototype IR Communication Language for Autonomous Robots, Lamar Univ. Master's Thesis, Aug, 2004.

[11] RidgeSoft.Intellibrain-Bot Educational Robot, http://www.ridgesoft.com/intellibrainbot/intellibrainbot.htm

[12] RidgeSoft.RoboJDE – Java-enabled robotics software development environment, http://ridgesoft.com/robojde/robojde.htm

[13] RidgeSoft.RoboJDEGuide, http://www.ridgesoft.com/robojde/1.3/docs/RoboJDEGuide.pdf

[14] Aircable. AIRcable.net – Products – AIRcable Industrial Wireless Cable, http://www.aircable.net/industrial-male.html

[15] B. Chaib-draa and F. Dignum, Trends in agent communication language, *Computational Intelligence*, Vol. 18, No. 2, 2002, pp. 89-101.

[16] How Bluetooth Technology Works, http://www.bluetooth.com/Bluetooth/Learn/Works/