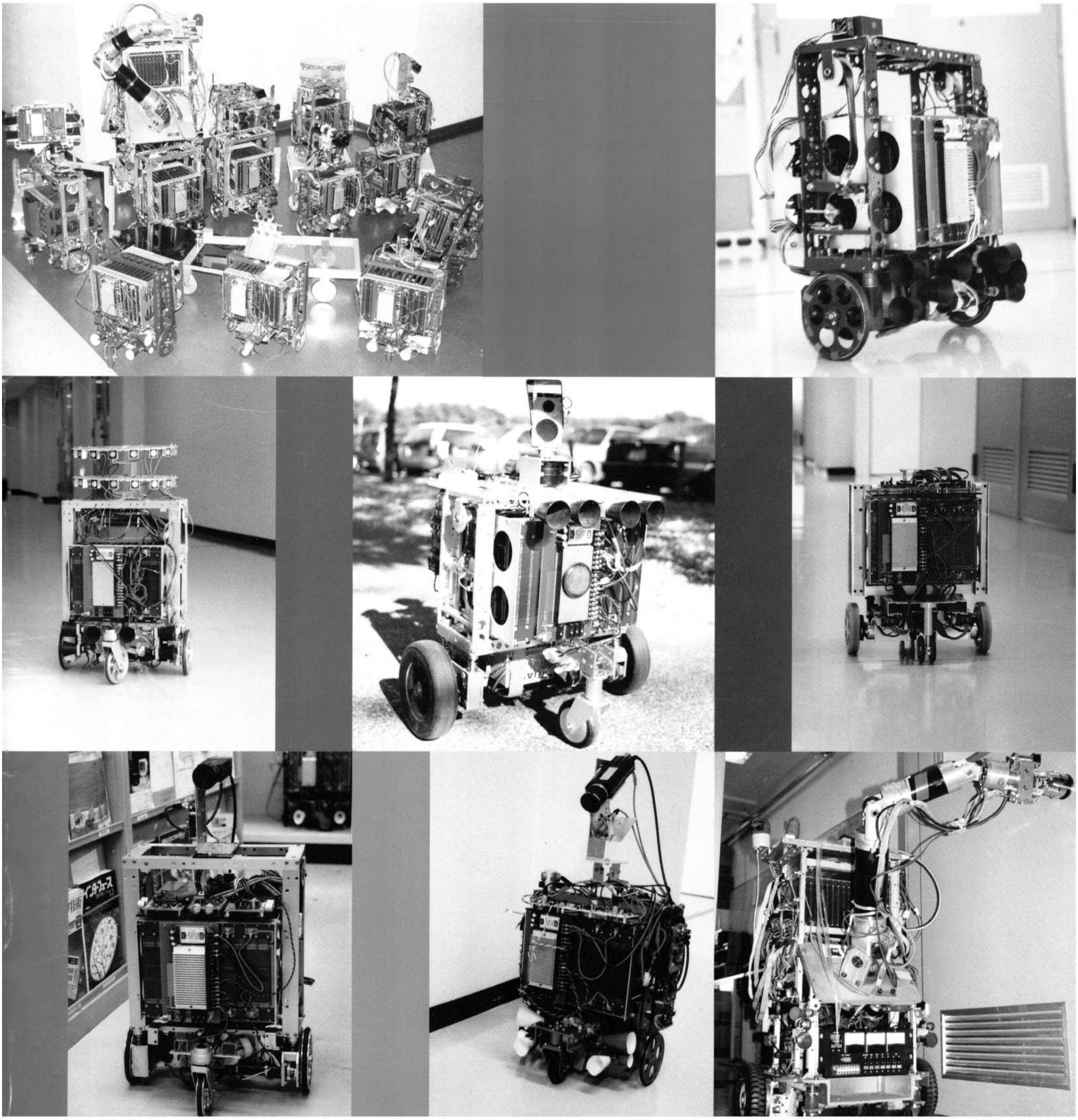
Journal of Robotics and Mechatronics

Vol.8 No.6 Dec. 1996

■ Special Issue on Integration of Intelligence for Robotics in VLSI Chips



Paper:

Intelligent Control of a Mobile Robot for Service Use in Office Buildings and Its Soft Computing Algorithms

Takayuki Tanaka[*], **Junji Ohwi**[*], **Ludmila V. Litvintseva**[**],
Kazuo Yamafuji[*] and **Sergei V. Ulyanov**[*]

*Department of Mechanical and Control Engineering, The University of Electro-Communications,
Chofu, Tokyo 182, Japan
**Artificial Intelligence Research Centre of Program System Institute, Russian Academy of Science,
Botik, Pereslavl-Zallesky, 152140, Russia

[Received January 31, 1996; April 30, 1996]

The arrangement principles and design methodology on soft computing for complex control framework of AI control system are introduced. The basis of this methodology is computer simulation of dynamics for mechanical robotic system with the help of qualitative physics and search for possible solutions by genetic algorithms (GA). On fuzzy neural network (FNN) optimal solutions for navigation with avoidance of obstacles and technological operations as opening of door with a manipulator are obtained and knowledge base (KB) for fuzzy controller is formed. Fuzzy qualitative simulation, GA and hierarchical node map (HN), and FNN have demonstrated their effectiveness for path planning of a mobile robot for service use. New approach for direct human-robot communication with natural language and cognitive graphics is introduced. The results of fuzzy robot control simulation, monitoring, and experimental investigations are presented.

Keywords: Mobile robot for service use, Soft computing, Intelligent control, Genetic algorithm, Fuzzy neural network, Direct human-robot communication, Cognitive graphics

1. Introduction

At present, two classes of mobile robot for service use are distinguished[1,2]: (1) Class A, robots to replace human beings at work in dirty, hazardous and /or repetitive operations; and (2) Class B, robots to operate on/or with human beings to alleviate inconvenience or to increase comfort/pleasure. Class A includes operations in hazardous or extreme environment (e.g., radioactive environments, high temperatures, underwater, vacuum, harmful gases), fire fighting, military applications and so on. Class B includes medicine, housework, entertainment and others.

Extensive research has been conducted in recent years in developed countries for the purpose of fabricating mobile robots for service use in class A which are capable of moving along horizontal, inclined or vertical surfaces. Such robots must be able to overcome or avoid obstacles encountered on their paths, or returning to the required initial (start) position, set up to execute srequired industrial operations, and to perform such execution. Such robots and robotic complexes are needed because of increasing demands on industrial operations, accident and emergency conditions, and conditions that are hazardous or difficult for human operators. Mobile robots for service use for Class A have been developed in papers.[3,4)]

The first mobile robot for service use for solving Class B tasks was investigated in papers[5,6)] and an intelligent control system with soft computing was developed in papers.[6,7,8,9)] In this paper, intelligent robots for service use (Class B) in buildings, especially in office buildings, are developed. **Figure 1** shows an experimental prototype of the mobile robot for service use. This robot has power wheeled steering mechanism achieved by two driving wheels and a caster with passive suspension for stable locomotion. Thirteen ultrasonic (US) sensors, nine infrared (IR) sensors, a five degree-of-freedom (DOF) manipulator with a three-finger hand, and a CCD camera are equipped on the robot for conducting tasks and work in buildings including human beings, opening doors, and getting on/off an elevator.

The service robot can be utilized as a "secretary or helper robot" by day and a "robot for security or maintenance, including floor cleaning " by night in office buildings.[5,6,7,8)]

Industrial robots have proved their usefulness in the manufacturing environment and become essentially tools in advanced production systems. Next generation robots must be have intelligent control systems and work outside of the manufacturing environment.

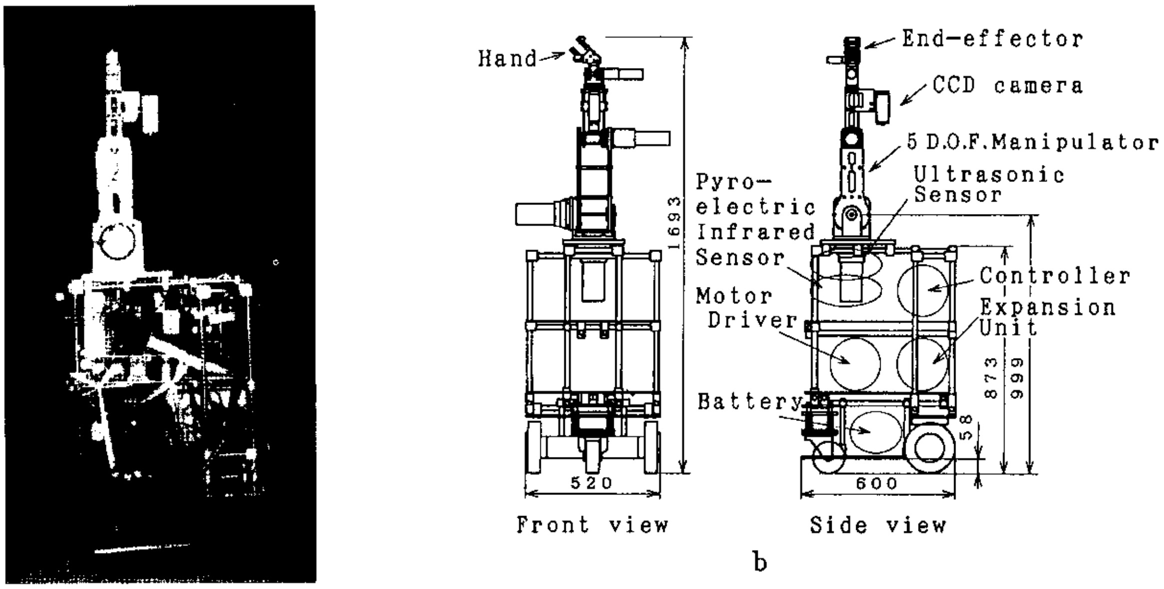
Mobile robots for service use are distinguished from industrial robots by the following faculties: (1) mobility, (2) maneuverability, (3) intelligence levels, (4) operating ease, (5) adaptability and (6) portability.

Created by trends toward higher education levels, shortages in the aging specialized labor market, created by the increasing age of highly skilled technical personnel, and trends toward replacing manual work by robot operation in hazardous environments are main factors favoring the market demand for robots for service use. Negative factors affecting the robots for service use market include insufficiently rapid development of requisite sensors and of their intelligent faculty, and high cost.[2)]

Further development in this new field in robotics engineering is needed, to compensate for negative factors and has required research aimed at improving robot mobility and maneuverability, as well as robot intelligence (sensing, learning, and judging functions), and expanding technological capabilities and effective fields of application.

In this paper, a new approach to intelligent control system design with two parts, as new forms of direct human-robot communications (including emotion, instinct and intuition) and an autonomous locomotion control system are developed. **Figure 2** shows the structure of this intelligent control system. We will consider, as the first step, one line in this scheme: direct human-robot communications based on gestures or natural language (NL), and construct the simulation system of spatial scenes and robot behavior in virtual reality (VR). The structure of this system is shown in **Fig.3.**

We also explain the intelligent autonomous locomotion control system of the service robot. The intelligent locomotion system is composed of four functions, i.e., locomotion



a: Picture of robot; b: View of robot.

Fig. 1. Developed Mobile Robot for Service Use.

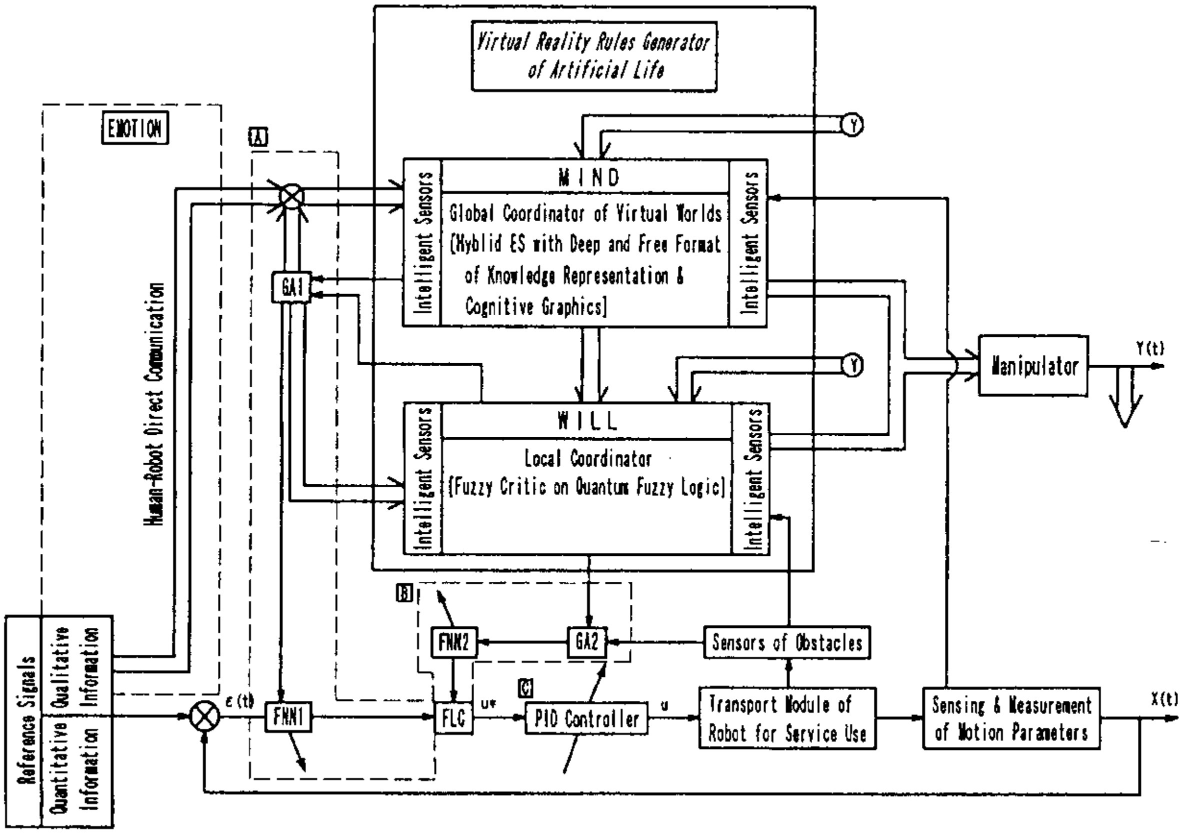


Fig. 2. Structure of AI control system with distributed knowledge representation (on control signal levels). A: Intelligent control "in large"; B: Intelligent control "in small;" C: Control on executive level.

control, planning for works, learning, and recognition, which are related each other. The robot system is organized by a locomotion system, a handling system for a mobile manipulator, and an image processing system as human vision system conducted together (**Fig.4**). Experiments of locomotion control on the developed robot show that the proposed methods are very useful for autonomous locomotion control of the robot.

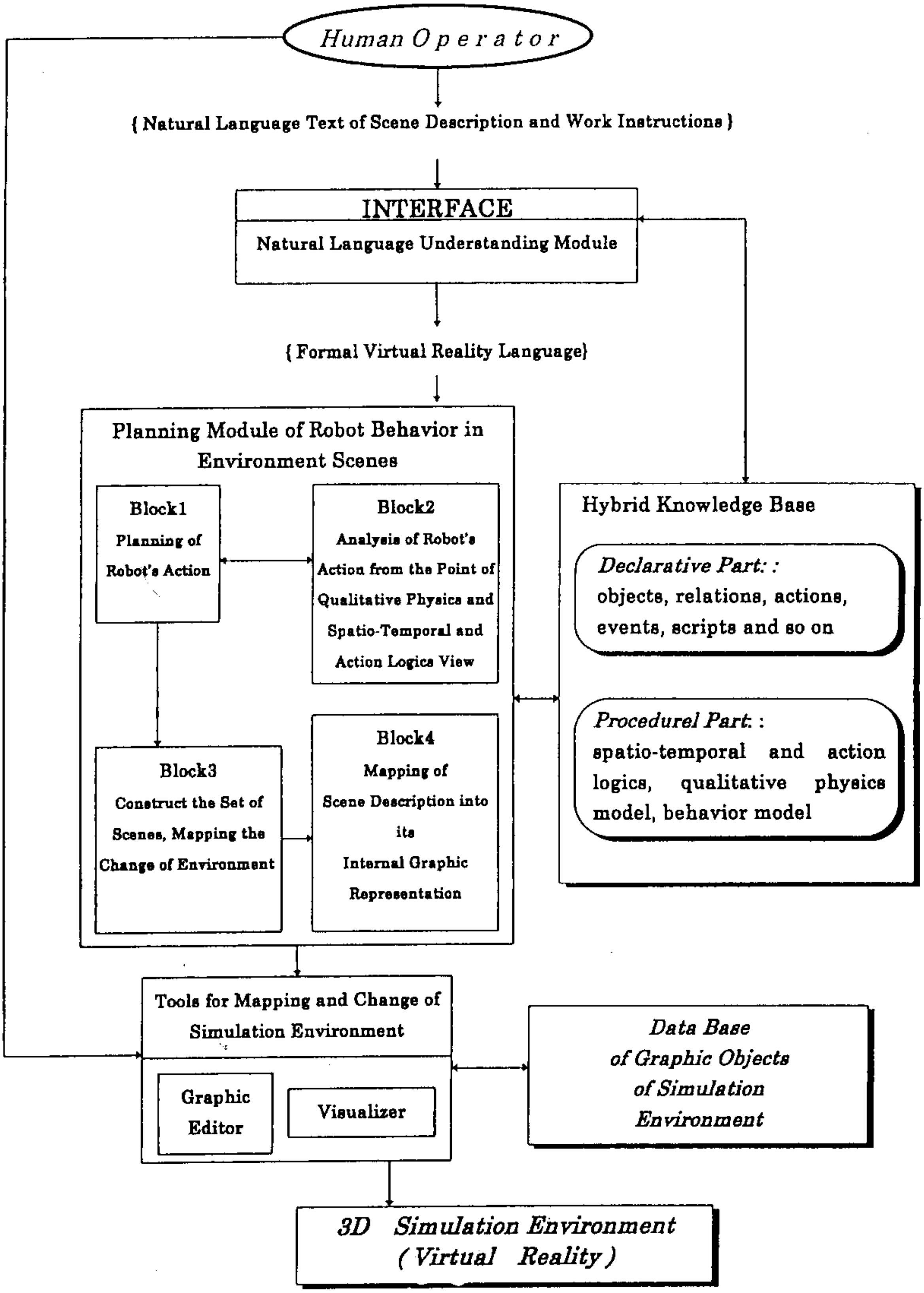


Fig. 3. Structure of robot behavior simulation system.

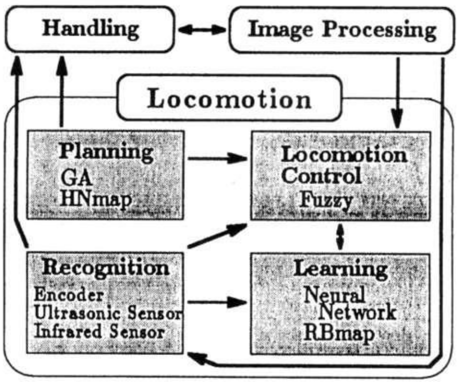


Fig. 4. Proposed autonomous locomotion system.

2. Direct Human-Robot Communication with Behavior Simulation System

In this part, we consider the use of NL and cognitive graphics for conditional descriptions of artificial robot life and direct human-robot communications for the mobile service robot shown in Fig.2. The mobile robot for service use works in buildings with different floors and rooms and moves in unstructured environments in the presence of people and unexpected obstacles. We propose to construct a simulation system for mobile service robot behavior based on cognitive graphics. This system is used for possible world simulation in artificial robot life. This allows us to evaluate the control algorithms of real-time robot behavior and to reduce difficulties connected with such problems as robot collisions with obstacles and robot hardware damage.

The mathematical background of the robot behavior simulation system is knowledge engineering based on spatiotemporal and action logics, default reasoning, cognitive graphs, and soft computing. Here, we discuss the main concepts, structure, and conceptual model of behavior simulation systems for description of artificial mobile robot life for service use in office buildings.

2.1. Task Definition

In accordance with the scheme shown in Fig.2 consider the line of direct human-robot communication based on NL. We will construct the simulation system of spatial scenes and robot behavior in the simulation environment (virtual reality). The output information of this simulation system used in the autonomous locomotion system (Fig.4) for global path planning and as commands from human beings.

The human operator represents the NL-description of artificial life conditions of the mobile robot for service use. This condition description include:

1) Environment scenes (for example, rooms in the building, objects in a room, the fuzzy spatial relations between them, and so on); 2) scripts of artificial robot life (for example, actions as "go to the room;" "image the room;" "show images of the room;" "open the door;" "grasp the book on the desk," and so on).

We consider the NL-text as human-robot communication input (Fig.2) that describes environment situations in space and in time.[10] The NL-text is transformed into internal representation (IR) of the simulation system. Based on IR and knowledge about spatiotemporal relations and actions, the system can generate and visualize 3D graphical images corresponding to the NL-input.

2.2. Main Concepts and Structure of Simulation System

The robot behavior simulation system structure is represented in Fig.3. The system consists of the following modules: 1) menu interface with an NL-language understanding module (linguistic processor); 2) robot behavior planning module in environment scenes with 4 blocks:

— Block 1: planning of robot action;

— Block 2: analysis of robot action from the stand point of qualitative physics, spatiotemporal, and action logics;

— Block 3: construction of the set of scenes mapping environment changes connected with actions;

3) tools for mapping and change of simulation environment scenes including:

— Graphic editor and visualizer;

4) Knowledge and graphic object bases.

NL-text input describes 3D spatial scenes (for example, rooms) with objects and spatial relations between it and contains work instructions for the robot about what needs to be done in this scenes. The robot may act and change spatial scenes. We introduce the concepts of *static spatial scene* and *dynamic scene*. The static spatial scene description represent *the set of objects* connected with each other by *definite spatial relations*.

The dynamic spatial scene description consists of *initial spatial scene* description and *action* description.

Consider *the structure of the simulation system.*

The menu interface contains the pathways to access the primary modules of the system: "NL-text Input", "Tools for Mapping and Change Scenes," and so on.

The NL-text inputs information to a linguistic processor that transforms this to IR in formal virtual reality language (VRT) (see below). IR includes frame copies, such as objects, subjects, spatial relations, actions, scenes, and scripts. For NL-processing, the modification of Wood's augmented transition networks is used. Blocks 1 and 2 in Fig.3 are used for realization of the following functions:

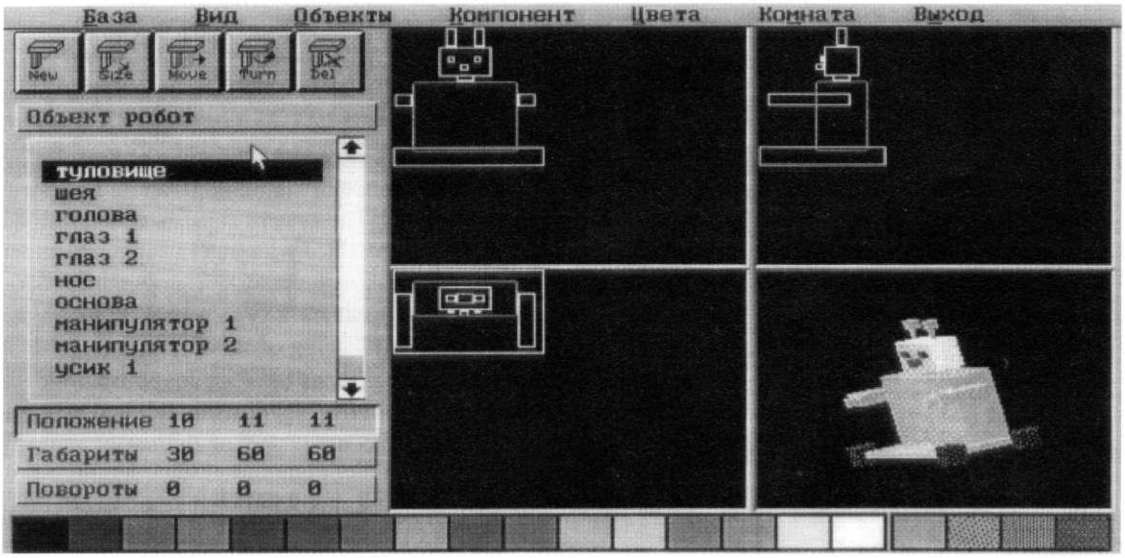
- checking whether it is possible to realize the action from a qualitative physics stand point;
- construction of an action realization plan based on script;
- casual actions or event outcome;
- trajectory construction connected with a give action;
- analysis of correctness from the stand point of spatial logic for the given scene.

Blocks 3 and 4 are used for:

- object location planning in the current spatial scene;
- construction of internal graphical representation of the scene description.

So, we have the set *<XYZ>* of object coordinates in the coordinate system connected with the spatial scene as the output from Block 4. The visual processor (Visualizer) converts these coordinates to an absolute coordinate to be displayed.

The Graphic Editor is used for creating and modifying 3D object images in environment scenes. An example of graphical object representation is shown in **Fig.5**. In this case, Fig.5 presents a Russian version of a graphical editor (developed by S. Nalitov). The English graphical menu version for graphical representation of mobile robots (at left



Положение 10 11 11
Габариты 30 60 60
Повороты 0 0 0

Fig. 5. Example of graphic representation of object obtained by tools of Graphic Editor (4 versions of visualization).

in Fig.5) is translated as: object-robot; body (of the robot); head; neck; eye 1; eye 2; nose; manipulator 1; manipulator 2; position; size; angle. Here we use the main principle of cognitive graphics for graphic representation of our simulation environment. This is as follows[11]: instead of complicated mathematical models of graphic representation of environment objects, we use more simple images as a result of knowledge-based mapping. For out task, it is important to have adequate mapping of spatial and dynamic situations to examine the algorithms of rational robot behavior.

The knowledge base (KB) consists of a few knowledge classes. A priori knowledge is acquired in the process of problem solving.

All *a priori knowledge* can be divided into three classes: 1) *Class A* contains syntactic knowledge about objects, actions, and relations; 2) *Class B* contains knowledge about pseudophysical spatial logic and action logic, one logical deductive system for a geometrical and physical description of space and action[12]. This logic is used for simulation of spatial scenes and dynamic situation; 3) *Class C* includes knowledge about semantic and pragmatic properties of actions, of objects, and of spatial relations between them.

Class D contains knowledge acquired in the process of problem solving.

A mixed frame and productions rules approach are used for knowledge representation. The frame part of the KB describes objects and their properties, relations, and action. The production part of the KB describes spatial logic axioms and qualitative physics productions.

2.3. Formal Language for Simulation of Robot Behavior in 3D Spatial Scenes

We will consider the following virtual environment for a mobile robot for service use: This robot works in a complex world, like a building with many rooms, floors, corridors, elevators, other robots, and so on. In each room, there are many objects with different spatial relations. The robot realizes actions connected with objects in the room and change the spatial relations between them. To simulate such an environment, we developed a formal language, Virtual Reality Language (VRL), for the description of environment scenes and change according to robot actions. **Table 1** and **Table 2** show the basic concepts, syntax, and semantics of VRL.

2.4. Problems of Spatiotemporal and Action Logic Design for Intelligent Decision Making and Reasoning of the Robot for Service Use

Reasoning about time, space, and action is an important aspect of the rational behavior of intelligent systems. Much work has been done to describe various paradigms for reasoning and attention is focused on logic-based theoretical frameworks.[11,12,13]

We developed an approximate reasoning system, based on so-called pseudophysical spatiotemporal and actions logic.[12,13] The name "pseudophysical logic" is explained by the fact that neither real physical and metrical properties of time, space, and action nor human perception properties are used in axioms or in inference rules. The important factor of this model are: 1) some reasoning is connected with time scales; 2) a few components are connected; 3) this is the logic of relations; 4) some components of logic dealing with dynamic situations are nonmonotonic logical systems.

For behavior simulation of the intelligent robot for service use, we modified the logic developed.[11,12] Examples of experiments showed the effectiveness and adequacy of the proposed logic to a given task. Examples of axioms of this logic are shown in Table 2.

2.5. Example of Soft Computing for Intelligent Position Control of Mobile Robot in Simulation Behavior System

Consider the rules of soft computing of object coordinates according to fuzzy spatial relations used in the algorithm of Block 4.

We describe the spatial relations between two objects as spatial relations between its basic points. The distance (L_{AB}) between two objects (A and B) connected with fuzzy spatial relation (R) is the function of the following parameters:

Table 1. Basic temporal, spatial relations, and actions in the behavior simulation system of the service use robot.

TEMPORAL LOGIC RELATIONS	SPATIAL LOGIC RELATIONS	ACTION LOGIC ACTIONS
1) For temporal points and instant events *Nonmetric order relations*: - to be earlier ; - to be later ; - to be simultaneously ; *Metric relations :* - to be earlier on N unit on the time scale L ; - to be later on N unit on the time scale L; *Fuzzy relations :* - to be approximately simultaneously; -to be substantially earlier; - to be substantially later; - to be not so substantially earlier; to be not so substantially later ; *2) For temporal interval and events :* All mentioned above relations plus 7 different types of interval intersection ; *3) For actions :* - at time t (or approximately) begin action d; - at time t (or approximately) finish action d; - after action d1 (or interval T1) begin action d2; - before action d1 (or interval T1) begin d2; - immediately after action d1 (or interval T1) begin action d2 ; - begin action d1 simultaneously with action d2; - while action (or event) d1 make action d2; - after each N unit on scale L during T interval make action d.	*1) Relative position relations* *Nonmetric*: - to be inside; - to be outside; - to be on; - to be under, over; - to be left, right; - to be in front, behind; - to be between; - to be left (right) and behind (in front) ; *Metric :* - to be by the angle A; *2) Spatial proximity relations* *Nonmetric fuzzy relations :* - to contact ; - to be closely ; - to be very closely; - to be near; - to be far; - to be very far; - to be not far and not closely; *Metric and fuzzy relations :* - to be in the distance of N unit on scale A from (to); - to be in the distance of approximately N unit on scale A from (to); *3) Fuzzy location relations :* - to be in the center of ; - to be in the left (right) back (front) angle of ; - to be in the top (bottom) of ; *4) Others relations :* - to have size (small, middle, big, not small, not big and so on); - to have the point of support.	*1) Types of moving :* - to move itself with different types of velocity (quickly, slowly); - to move itself into localization; - to go from point A to point B; - to go nearly to an object; - to go to the left, right, forward, back (according with the current direction of moving) on some (fuzzy) distance; - to change velocity of moving (including stop) ; - to move itself by the elevator ; *2) Actions with objects:* - to grasp an object (with different types of grasping); - to hold an object; - to put an object at point P; - to put one object on (into, under, from left, from right and so on) another object; - to take the object ; - to bring the object ; *3) Force actions :* - to exert force to an object; - to push on the object; - to throw an object; - to turn an object ; *4) Others actions :* - to open a door; - to close a door; - wait (event).

$$L_{AB} = F(R, L_A, L_B, L_{scene}),$$

where L_A, L_B, and L_{scene} are the sizes of object A, object B and scene.

For spatial proximity relations, we developed the following rules for computing L_{AB}:

$$L_{AB} = \begin{cases} K_{closeness}(L_{scene} - L_A - L_B), & L_A \simeq L_B \\ (K_{closeness} + L_A/L_B)\,(L_{scene} - L_A - L_B), & L_A \gg L_B. \end{cases}$$

Table 2. Virtual reality language for task of robot behavior simulation.

Alphabet	Syntax	Semantics
Alphabet consists of the following sets: Basic notions - set Q including following sets: $\{O_k\}$ - objects ; $\{A_k\}$ - actors (robots); $\{T_k\}$ - times ; $\{P_k\}$ - 3D points; $\{L_k\}$ - spatial localizations; $\{SC_k\}$ - scripts; $\{S_k\}$ - 3D scenes; $\{EE_k\}$ - external influences and events; The set RR of relations including the following sets : $\{SR_k\}$- spatial relations; $\{CR_k\}$- causal relations; {TRk}- temporal relations; $\{ACT_k\}$ - actions; $\{ACTR_k\}$ - relations for action descriptions; R_{value} - special relation describing the noncorrectness of the scene from the point of given logics model view.	Following rules for well formed formulas (WFF) : 1) If $x \in Q$, $y \in Q$, $R \in \{RR\}$, then (xRy) is WFF. 2) If F - WFF, then $\forall t F, \forall s F, \exists t F, \exists s F$ - WFF; 3)If F - WFF , then $\exists(t_1 \ldots t_n) F, \exists (s_1 \ldots s_n) F, \forall (t_1 \ldots t_n) F, \forall (s_1 \ldots s_n) F$ - WFF; 4) $< s,t >$ - WFF, $< s,t > \Rightarrow < s_1,t_1 >$ -WFF, $< s,t > \Rightarrow < s_1,t_1 > \ldots < s_n,t_n >$ - WFF, where "$\Rightarrow$" is the operator of scene change; 5) If F - WFF, then $< s,t > F$ - WFF; 6) If F_1 , F_2 - WFF, then F_1 & F_2 - WFF; 7) There is no other rules.	The following axioms (examples): 1) $< s,t > d \rightarrow \exists(t_1 \ldots t_k)$ & $\exists (s_1 \ldots s_k) < s,t > \Rightarrow < s_1,t_1 > \Rightarrow \ldots \Rightarrow < s_k,t_k >$; 2) $< s,t > (d_1 R d_2) \rightarrow \exists s_k \exists t_k < s,t > d_1$ & $< s_k,t_k > d_2$, where R -" to be later", $d_1, d_2 \in \{ACT_k\}$; 3) $(O_1 R O_2) \rightarrow (O_2 R_1 O_3)$, where $O_1, O_2, O_3 \in \{O_k\}$, R - " to be on", R_1 - " to be the point of support" ; 4) $< s,t > (O_1 R_1 O_2)$&$(O_1 R L)$ &$(O_2 R S) \rightarrow (s R_{value} F)$, where R_1 - " to be in", R - " to have size", L,S - the linguistic value of size (L -" large",S -" small"); 5)$(d_1 R_1 a_1)$& $(d_1 R_2 O_1) \rightarrow \exists t, \exists s (a_1 R_{near} O_1)$, where $R_1, R_2 \in \{ACTR_k\}$ and R_1 - to be the actor of action, R_2 – to be the object of action; R_{near} – " to be near to the object".

Here, $K_{closeness}$ is the maximal value of the membership function on scale u: $K_{closeness} = \max_u \mu_{ri}(u)$, where u is the universal linguistic scale of distance.

The rules of computing the basic point coordinates for relative position and fuzzy location relations were also developed.[11)]

2.6. Example of Simulation

Let us examine the following task: By the direct human-robot communication line in NL, the robot receives the following NL-input : "Room number 117 is located on the second floor of the building. A desk is in the center of this room, the chair is left of the desk. The chair is near the desk. The lamp is on the desk. Go to the room, take this lamp, and put it on the wardrobe." Using the cognitive graphic block, the robot may make a graphic representation of the described room (this may be considered a global map of this room). NL-input is converted by the linguistic processor into IR on VRL. Two processes are connected with WFF formulas of VRL: the analysis and construction of set of VR-scenes corresponding to input; and the graphic representation of VR-scenes. These tasks are realized by the

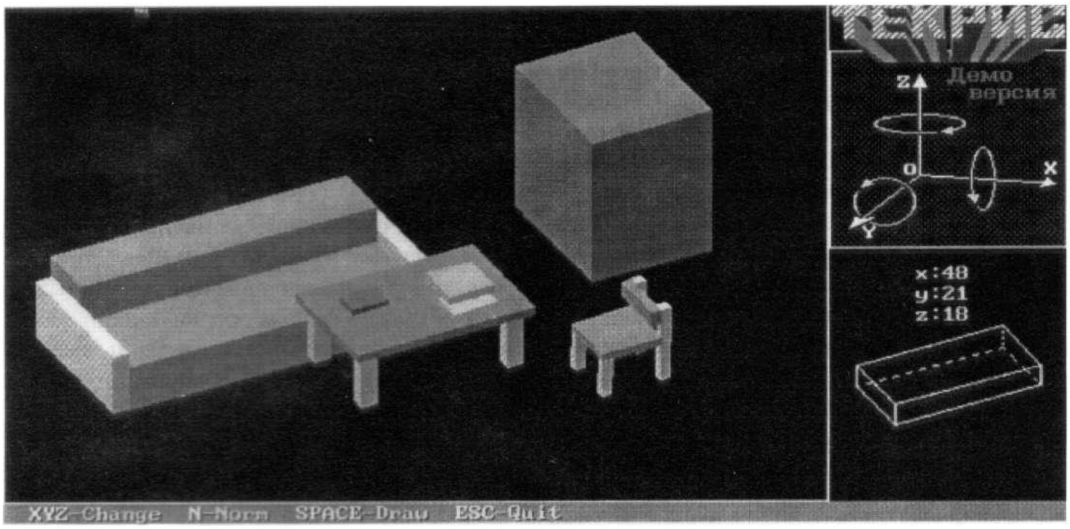


Fig. 6. Example of 3D visualization of static scene (described by text 1). Different points of view are obtained by change of (x,y,z)-axes.

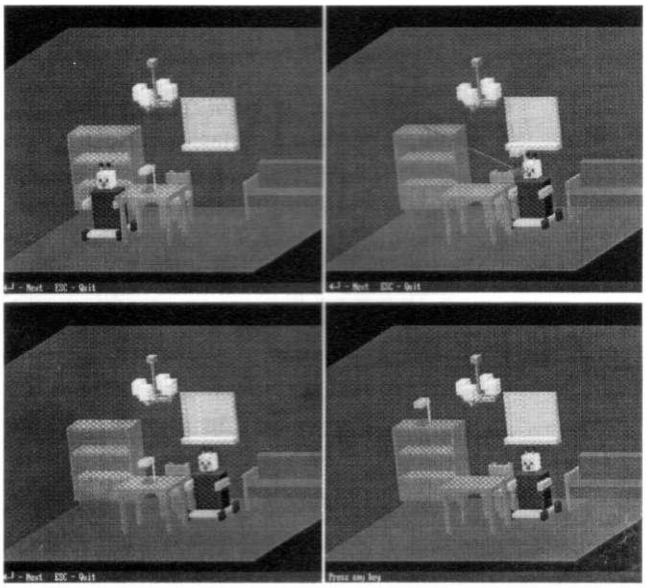


Fig. 7. Example of 3D visualization of dynamic scene (described by text 2).

above system modules. For actions, algorithms for simulation and visualization of this actions have been developed. We use the "step by step" method, in which a set of spatial points characterizing phases of realizing of current action are considered. For example, for the action "take the object," the following set of scenes is visualized: the initial scene; the navigation of the robot in this scene needed for the given action; the scene in which the robot is near the object; and the finally, the scene in which the robot has this object in manipulator.

The results of computer simulation of spatial room scenes and robotic actions in the room VR are shown in **Fig.6** and **Fig.7**.

Figure 6 shows the 3D graphical representation of the NL-description of the room as follows (*text 1*): "The desk is in the center of the room. The book and the lamp are on the desk. A sofa is beside the desk and near it. The chair is located right of the desk not far from it. The wardrobe is right of the chair."

Figure 7 shows the 3D graphical representation of the dynamic scene described by NL (*text 2*): "The desk is in the center of room. The lamp is on the desk. A sofa is beside the desk and not far from it. The chair is beside the desk and close it. The wardrobe is left of the chair. The robot stays left of the desk and near it. The robot takes the lamp and puts it on the wardrobe." In this case, a light bulb and a window in the room are described default logic.

Such a simulation system allows us to examine and correct the different algorithms of task-level planning and navigation of the robot for service use. If developed algorithms are adequate for our tasks and goals, the modules of robot action planning and direct human-robot communication can be realized in an actual environment.

Output information of the simulation system is input for managing control of the second-level locomotion system (Fig.2). The structure shown in Fig.4 presents main modules of the autonomous locomotion control system. Consider in detail the main modules in Fig.4.

3. Managing System for Cooperative Control

3.1. Main Concept of System Structure

It is necessary in a real environment to control each robot's subsystems cooperatively and smoothly, as locomotion, handling, and image processing system (Fig.4). The managing system is constructed upon subsystems. This system makes the robot acquire the following abilities:

(1) The robot can understand the command of the simulation system from a human being on NL; (2) it can control a subsystem according to given commands; (3) it can control one subsystem according to other subsystems; (4) it can declare the robot's will to human being or the other robot.

In this paper, an intention function which has preceding capacities (1) and (2) is constructed as a fundamental managing system. Under this function, the autonomous control of subsystems (robot mind in Fig.2) is planned according to given commands by using the man-robot-interface. In short, scheduling of each sub-systems is conducted by this system.

3.2. Scheduling

In previous papers,[6,7,8] the working-plan function aimed at giving efficient and correct path planning introduced by using Distributed GA (DGA) and the HN-map method. However, when the robot's tasks and work came to be more complicated, it is difficult to make efficient path planning in real time. Because, in this DGA, a gene was a node (small space) in genetic operation.

In this paper, we use a DGA in which a gene is a distributed area (wider space) in the HN-map and the planned path can be written in a one-dimension matrix of areas. Thus, this problem must be solved more efficiently so that we achieve real time control.

Meanwhile, in the robot's locomotion, there are one target point and one aimed task or work in every area. For example, in a case of moving from room to corridor, the robot must go to the front of a door and then open it, and for going to another floor, it has to go to the front of an elevator and push a button and then get on.

So, the HN-map[6,7,8] can be used in this scheduling system. This map has a hierarchy of nodes where handling or image processing system must be controlled. Using the hierarchical map in the control system, HN-map area information of can be used to decide a target point and a task or work on every planned area. The autonomous scheduling of the service robot can be conducted by this means.

3.3. Work File

Planned scheduling data by the proposed method must be changed to recognizable information as a *Work File.* This file is written as follows:

MOVE COMMAND : WORK COMMAND : AREA
⋮ : ⋮ : ⋮

- ○ MOVE COMMAND: A command to Locomotion system, a start and a target node;
- ○ WORK COMMAND: A command to Handling and Image-processing system;
- ○ AREA : Locomotion area.

In this paper, there are two commands. One is "two Move Command" and the other is "six Work Command".

- ○ MOVE COMMAND
  - ◇ GOTO-P1-P2: Go from node P1 to node P2 with fuzzy locomotion;
  - ◇ WAIT-P1-P2: Stay at node P1.
- ○ WORK COMMAND
  - ◇ MOVE-TRGT: Go to the target node;
  - ◇ PULL-DOOR: Detect a door knob and pull it;
  - ◇ PUSH-DOOR: Detect a door knob and push it;
  - ◇ GETON-ELV: Detect and push an elevator button, and get on;
  - ◇ OPERA-ELV: Operate an elevator and recognize achieving to another floor to get down on;
  - ◇ GETOF-ELV: Get off an elevator;

3.4. Experimental Results

In order to test the effectiveness of the proposed managing system, an experiment was conducted using the HN-map made by measuring the real environment as shown in **Fig.8.** In this experiment, the robot position is decided from the scheme based on the simulation system and the first node is set. It is given the following command by using the man-robot-interface: **"Go to node 34 (elevator)".** The planned area is shown in [13 – 1 – 23] and the following Work File is obtained by using scheduling function:

GOTO-0026-0021 : PULL-DOOR : AREA-0013
GOTO-0008-0010 : GETON-ELV : AREA-0001
GOTO-0034-0034 : MOVE-TRGT : AREA-0023

The robot can move continuously from a room (node 26) to an elevator (node 34) with this Work File. Intelligent control for avoidance of obstacles and execution of technology operations such as opening a room door is described in section 4. In experimental results, it is shown that the proposed managing system is useful for the intelligent robot system.

4. Intelligent Control and Soft Computing for Avoidance of Obstacles and Execution of Technology Operations

In the robot locomotion, from point 26 to point 34 in Fig.8, the mobile robot must avoid obstacles in Room13 (on the path from point 26 to point 21), and achieve the starting position (point 21) for successful opening a door (as technology operation from point 21 to point 8) and go out of Room 13 to Elevator 23 (point 34) in the presence of obstacles in a corridor. We discuss in detail this process as recognition of position and obstacles together with intelligent control in the navigation system.

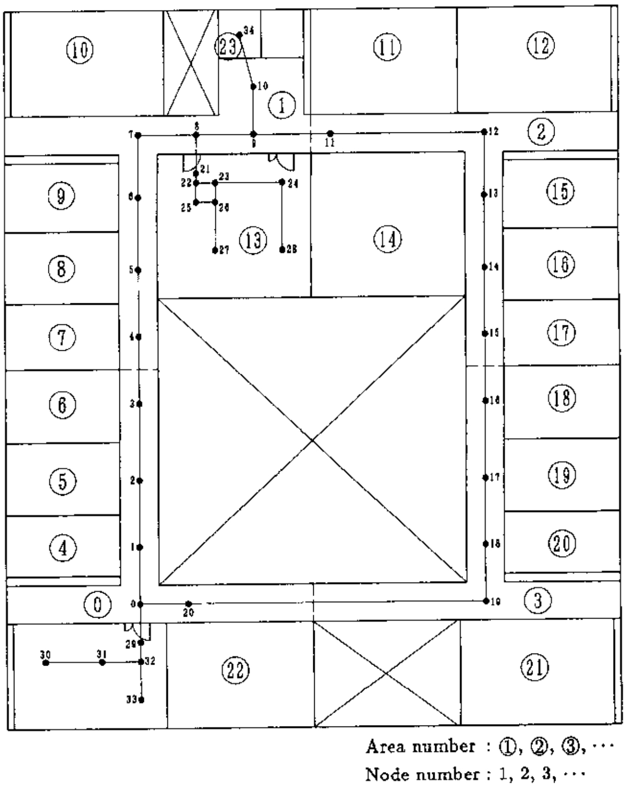


Fig. 8. Hierarchical node map on floor of building.

4.1. Recognition

A. *Recognition of Absolute Position* The robot's position related to the global reference coordinate is computed by the measured value from two encoders installed on the wheel motor, which is called the dead-reckoning method. The absolute position of the robot can be computed approximately by means of the following three equations:

$$X_{i+1} = X_i + \frac{W(M_r + M_l)}{2(M_r - M_l)}(\sin\theta_{i+1} - \sin\theta_i), \quad \ldots (1)$$

$$Y_{i+1} = Y_i + \frac{W(M_r + M_l)}{2(M_r - M_l)}(\cos\theta_i - \cos\theta_{i+1}), \quad \ldots (2)$$

$$\theta_{i+1} = \theta_i + \frac{M_\tau - M_l}{W}, \quad \ldots\ldots\ldots\ldots (3)$$

where X_i and Y_i are the robot's absolute position at the ith step, θ_i is robot's absolute direction at the ith step, M_l and M_r are the wheel's rotation angle, and W is the distance between the left and right wheelss.

B. *Recognition of Obstacles* In order to avoid obstacles in buildings, it is necessary to measure the distance between the robot and obstacles. In locomotion control in real time, detection, recognition, and judgment on information are required, so it is better for the robot to use sensors, which are minimal but sufficient to search for obstacles in front of and beside the robot, and handling time information should be short. Thus, 13 ultrasonic (US) sensors **(Fig.9)** are used for the detection of the robot's position related to obstacles.

C. *Recognition of Human Beings* The service robot moves in buildings in the presence of human beings, and should never hurt human beings. In the previous method using only US sensors, it is impossible for the robot to discriminate between human beings and other obstacles. In this paper, by joint use of infrared (IR) sensors and US sensors, the robot can detect human being as well as obstacles in buildings. Nine IR sensors are equipped on the robot at the same directions of FL1-SL0, FR1-SR0, and FC0 in Fig.9. IR sensors are coupled with US sensors using the following two equations:

IR sensor does not react to: $US = US_0, \ldots\ldots\ldots\ldots$ (4)

IR sensor react to: $US = US_0 - r_{robot}, \ldots\ldots\ldots\ldots$ (5)

where US_0 is US sensor information, r_{robot} is an active radius of the robot, and US is a parameter for locomotion control. In this method, the robot can avoid human beings more carefully than usual obstacles.

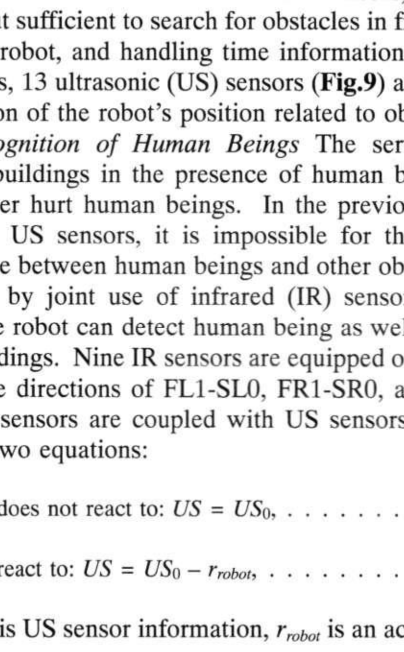


Fig. 9. Disposition of ultrasonic sensors on robot.

4.2. Fuzzy Control

The robot can get information from sensors. For autonomous locomotion control, it is necessary to change the detected information to recognizable information. In this paper, we propose to take simple Mamdani's fuzzy inference method for this problem. The inference is conducted by the product of a fuzzy set in the max-min form. Output control is obtained by the defuzzification of control rule with a weight method. Fuzzy inference is also adopted for the choice of path tracking control or obstacle avoidance control.

A. *Path tracking* Locomotion control for tracking path determined by the working plan part as a connected line is conducted by fuzzy inference.

The following 3 kinds of parameters are taken as the input of fuzzy inference for path tracking control: D_P: Distance between planned path and robot; D_T: Distance between target and robot; A_T: Angle between target and robot (4 fuzzy levels are set for D_P, and 7 fuzzy levels are set for D_T and A_T).

The following 2 parameters are taken as output of fuzzy inference: S_T: Steering angle; V_T: Movement speed (7 fuzzy levels for S_T, and 4 fuzzy levels for V_T).

The following 2 control rules are taken as a rule base: For steering: input is D_P and A_T, output is S_T, number of rules is 49; For movement speed: input is D_T and A_T, output is V_T, number of rules is 28.

B. *Obstacle Avoidance* The robot avoids obstacles according to information about the distance between the robot and human beings or other obstacles from the environment recognition part.

The input parameter is distance information calculated by Eqs (4) and (5). In right (SR0,SR1) and left (SL0,SL1) side parameters, the shorter is used as SR and SL, so that the number of input parameter is 11, and there are 4 fuzzy levels for each parameter.

The following 3 parameters are used as output of fuzzy inference: S_A: Steering angle; V_A: Movement speed; *Ratio*: Control ratio decision coefficient (7 fuzzy levels for S_A and 4 fuzzy levels for V_A and *Ratio*).

In this paper, by using layer fuzzy inference for the steering angle in obstacle avoidance control, it is easy to construct a rule base. In this method, the infrared output is used as input the next time in the fuzzy inference, and the whole rule number is decremented.

The control rule is set below. In the first layer, FC0, FL0, and FR0 are taken as the input of fuzzy inference for path tracking control. The following 2 parameters are used as outputs of fuzzy inference: S_A1: Steering Angle 1; *Safety*: Degree of Safety (7 fuzzy levels for S_A1 and 4 fuzzy levels for *Safety*). The following 2 control rules are taken as a rule base: For steering: inputs are FC0, FL0, and FR0, output is S_A1, and the number of rules is 64; For degree of safety: input is FC0, FL0, and FR0, output is *Safety*, and the number of rules is 64.

In the second layer, FL1-FL3, FR1-FR3, and *Safety* are taken as the input of fuzzy inference for path tracking control. Following parameter is used as output of fuzzy inference: S_A2 : Steering angle 2 (7 fuzzy levels). In the

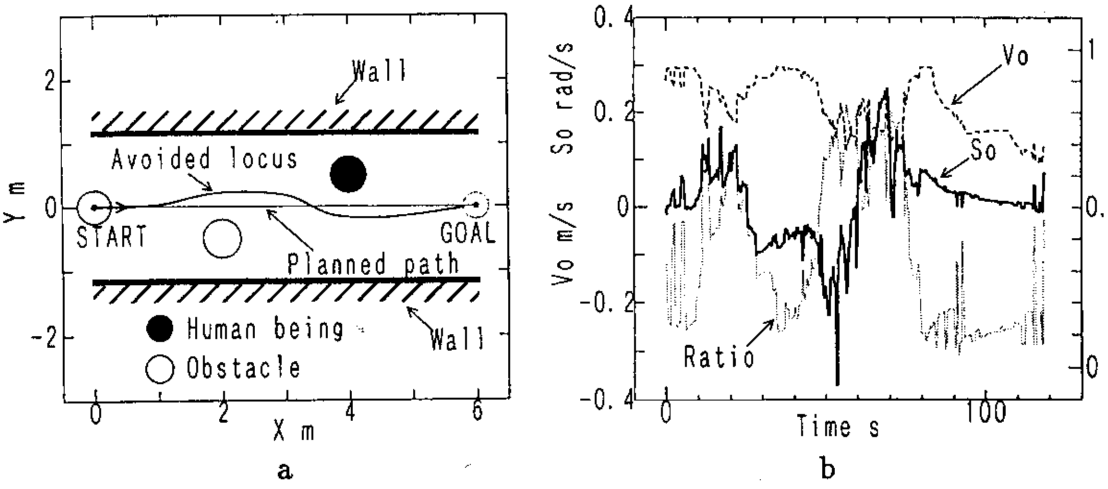


Fig. 10. Experimental results by locomotion control.
a: Locus of locomotion; b: Control signals of V_O, S_O and *Ratio*.

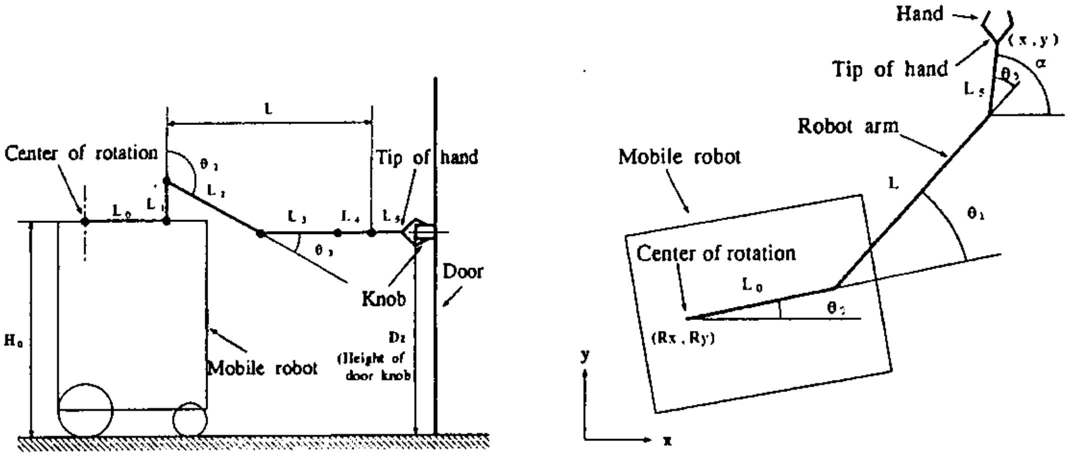


Fig. 11. Geometric model of mobile robot in touch with a door knob.

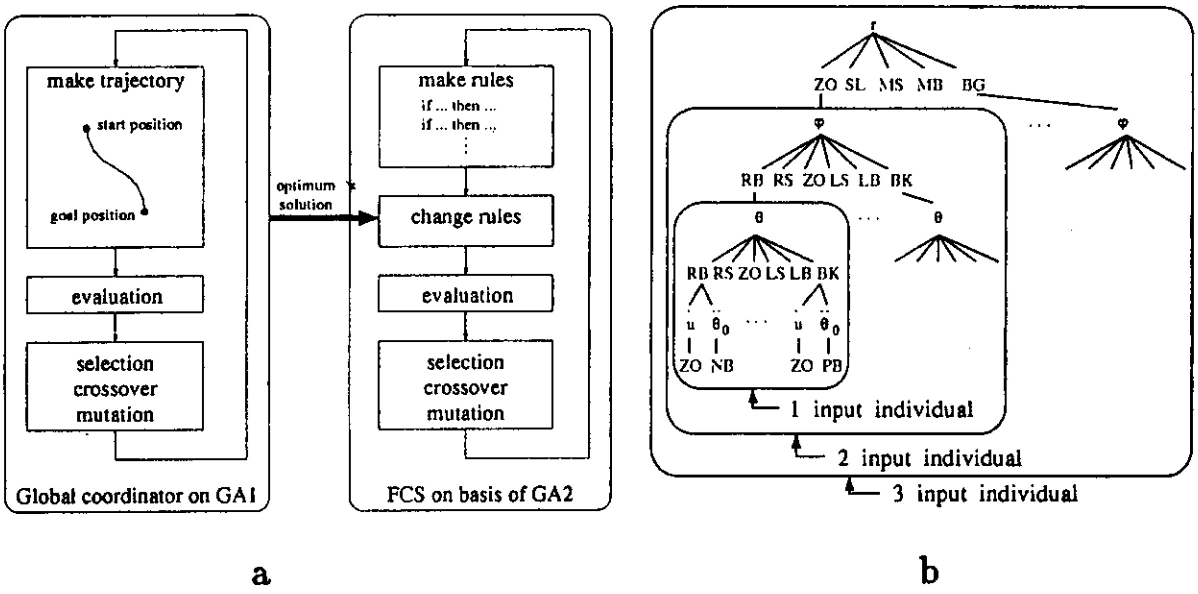


Fig. 12. Presentation of FCS algorithm.
a: Algorithm of FCS; b: Fuzzy Classifier System (FCS).

inference for S_A2, only one rule base, which has 64 rules, is used in four groups as Input 1 (*Safety*, FL1, FR1), Input 2 (*Safety*, FL2, FR2), Input 3 (*Safety*, FL3, FR3), and Input 4 (*Safety*, SL, SR). S_A is calculated by the defuzzification of S_A1 and S_A2.

In the inference for V_A and *Ratio*, it is possible for fuzzy level of V_A and *Ratio* to accord with one of inputs, so no fuzzy rule base is constructed. Output values are obtained by the defuzzification of all inputs.

Control ratio decision coefficient *Ratio* is used for the determination of the ratio in path tracking fuzzy output and obstacle avoidance fuzzy output in the calculation of practical control output. Steering angular speed S_O and the robot's speed V_O are computed in the equation (6) and (7) as

$$S_O = k_{STR}((1.0 - Ratio)S_T + Ratio S_A), \quad \ldots\ldots\ldots\ldots (6)$$

$$V_O = (1.0 - Ratio)V_T + Ratio V_A, \quad \ldots\ldots\ldots\ldots\ldots (7)$$

where k_{STR}- angular speed translate coefficient; *Ratio* is a real value changed in [0.0,1.0].

When the degree of safety (*Safety*) is too small, fuzzy locomotion control is stopped momentarily and the robot turned at a fixed point to confirm safety in the environment.

C. *Experimental Results* The experiment on the developed robot was carried out in a real situation using the proposed method. **Figures 10a and 10b** show experiment results of path tracking control with obstacle avoidance on the corridor of building. In the experiment, a man and an obstacle are placed. Figure **10a** shows that the robot can arrive at the target by avoiding obstacles in real surroundings. Figure 10b, which indicates the control signals of S_O, V_O, and *Ratio*, reveals that the robot is able to avoid a man more safely than any another obstacle, as the values of S_O and V_O for avoiding human beings are smaller than those for avoidance of a nonhuman obstacle.

4.3. Soft Computing and Intelligent Control for Opening Door with Manipulator

An intelligent mobile robot for service use, which is mainly utilized in office buildings, can locomote autonomously from one room to others on different floors and buildings using elevators. The robot equips a 5 DOF manipulator and must often open and close doors. The complex processes of opening a door seem to be a mechanical analogy of evolutionary computing (a new mechanical benchmark was proposed in paper[9]) and requires new intelligent strategic operation for the mobile robot manipulator. The successful fulfillment of this evolutionary operation depends on the selection of the initial position of the mobile robot and path planning of manipulator trajectory motion. For flexible realization of this evolutionary operation, it is necessary to develop efficient and intelligent computing algorithms for the next two items: (1) determination of the initial position of the mobile robot in form of a door-knob like parking a car and (2) path planning of manipulator trajectory motion of opening a door. We present new methods for making these algorithms. For the *first case*, we propose an algorithm on the basis of Fuzzy Classifier System (FCS) applying classifier system to fuzzy production rules. This FCS is a kind of GA in which a chromosome expresses a fuzzy rule like a classifier system. For the *second case*, we developed a method of making a trajectory for the mobile robot manipulator by means of GA for opening a door, and using Genetic Algorithms in Continuous Space (GACS) instead of GA. By these algorithms, the mobile robot has succeeded in opening a door. A comparison of GA and GACS on the basis of simulation results is presented. Characteristics of genetic operators (crossover and mutation rates) and convergence of GACS are also discussed.

The developed mobile robot is a 2 DOF autonomous vehicle and has a 5 DOF manipulator on the locomotive mechanisms. **Figure 11** shows a geometrical model of the mobile robot for service use. The mobile robot constitutes seven DOF systems in total. However, six of the seven DOF are fixed at a position in front of a door knob in order to open the door. On the other hand, each angle of an arm and wheel are not fixed. Therefore we must decide that only one DOF (direction of the mobile robot motion), which is controlled by GA. As depicted in Fig.11, angles θ_2 and θ_3 are fixed by the height of the door knob. For simplification of computation, we assume that the robot arm has three links.

A. *Positioning for the Mobile Robot with FCS* Before conducting an operation (for example, opening a door), the mobile robot must move to approaching a position to execute it. If the position of the robot is inaccurate (right or left of the present position), it must approach the objects while it is moving front or backwards just like parking a car. This operation can be described by using fuzzy production rules and the control system of the mobile robot can correct the position using ultrasonic or image sensors. However, the number of fuzzy rules becomes very large as the number of input is large. Thus, it is difficult even for human experts to make a lookup table of fuzzy production rules. Therefore, we propose an algorithm (soft computing), Fuzzy Classifier System (FCS) applying classifier system to produce of fuzzy rules. Algorithm of FCS on the basis of GA is shown in **Fig. 12a**.

B. *Fuzzy Classifier System* In computer simulation, FCS is a kind of GA in that a chromosome expresses a fuzzy production rule as "if... then" like classifier system. Fuzzy rules are expressed according to Fig.12b. The algorithm of FCS for soft computing is described as follows: (1) evolution in parallel from individual that has a smalll number of input (low individual) to individual that has sa large number of inputs (high individual); (2) high individuals are expressed as colonies of low individuals; (3) the chromosome expresses connection of each individual; (4) each individual is evaluated as a rule set; (5) if evaluation of a high individual is better than that of a low individual connected with it, the low one gets new evaluation of the high one.

C. *Results of Computing Simulation* Part of fuzzy rules in a lookup table obtained by FCS is shown in **Table 3**, and mobile robot motion simulation using these rules is shown in **Fig. 13.** In Table 3, FCS was made of unbalanced rules for θ. An expert may make the same rules, but there is no guarantee whether they are best. With soft computing on FCS, flexible learning process without stereotype examples was presented. In Fig. 13, the robot moves like switchback motion. From simulation results, we obtain the following: (1) fuzzy production rules to let the mobile robot move to the goal were made automatically by FCS; and (2) the robot behaves intelligent motion just like a switchback motion. However, the mobile robot could not reach a goal exactly because FCS cannot adapt itself to slight differences of position. For realization of an evolutionary process such as opening a door with fine inaccurate position of the mobile robot, we utilized a new approach for intelligent computation based on GACS.

D. *Evolutionary Process of Opening a Door using GA* We develop an evolutionary intelligent computing method to make the trajectory of manipulator motion for opening a door by GA and discuss effects of GA's parameter change using GACS instead of GA.

E. *Evolutionary intelligent computing method* Because of the difficulty of describing the relation between an unknown quantity and evaluation function in explicit equations, we apply a GA scheme. Encoding methods based on GA and GACS are shown in **Fig. 14**. A trajectory can be decoded by expressing a curve in order to pass two points (t_1, θ_{01}) and (t_2, θ_{02}) using a spline function (where, t - time, and θ_0 - angle direction of mobile robot motion). Each angle of an arm and velocity of the mobile robot are fixed values at the position of a door knob. As illustrated in Fig. 14, the chromosome of GACS is made of a real number. Genetic operations of GACS are the same as GA except for the operator of mutation. Mutation of GACS is due to Gaussian additive mutation. We define evaluation fitness function F as

$$\begin{aligned} F = {} & A\,(D_y - D_w/2 - y_1\,|_{x_1 = D_x})^2 \\ & + B\,\theta_0\,|^2_{x_1 = D_x} + C\,\alpha_{total} + D\,\theta_{over}\,, \end{aligned} \quad \cdots\cdots \quad (8)$$

where x_1,y_1 - coordinates of the first shaft; D_x, D_y - coordinates of a door's rotational center; D_w - width of a door; α_{total} - sum of absolute of wheel angle acceleration; θ_{over} - maximum angle of limit; A,B,C,D - coefficients.

Table 3. Lookup table obtained by FCS.

input					output	
$\dot{\theta}_0$	u	r	ϕ	θ	$\dot{u}$	$\ddot{\theta}_0$
ZO	ZO	ZO	ZO	RB	PS	PM
ZO	ZO	ZO	ZO	RS	PS	-
ZO	ZO	ZO	ZO	ZO	PS	PS
ZO	ZO	ZO	ZO	LS	-	PB
ZO	ZO	ZO	ZO	LB	NS	NB
ZO	ZO	ZO	ZO	BK	NB	PB

r : distance up to final position
ϕ : difference from final angle
θ : direction of final position
u : velocity of mobile robot
θ_0 : direction angle of mobile robot
ZO, RB etc. : fuzzy-set values

This function means, physically, that the robot goes straight to the center of the door advancing across the door, acceleration is assumed small, and angles of the arm will not overrun their limits. If F is small, fitness is large. In this case, we solve the following optimization problem:

$$\arg\min_{x} F\,(x) \quad \cdots\cdots\cdots\cdots\cdots\cdots \quad (9)$$

and, for GA, it is necessary that F has only many finite global maxima, $0 < F(x) < \infty$, $\forall x \in \Gamma \subseteq R^m$, Γ is the feasible region of real parameter vector; $F(x)$ has many finite discontinuous points. For function F, these conditions are fulfilled from Eq.(8).

In our case, genetic operations are carried out as follows: (1) operators - one point crossover and mutation; (2) parent selection techniques - roulette wheel selection; (3) fitness techniques - linear normalization; (4) reproduction method - generational replacement with elite strategy. In case of GACS, we use Gaussian additive mutation.

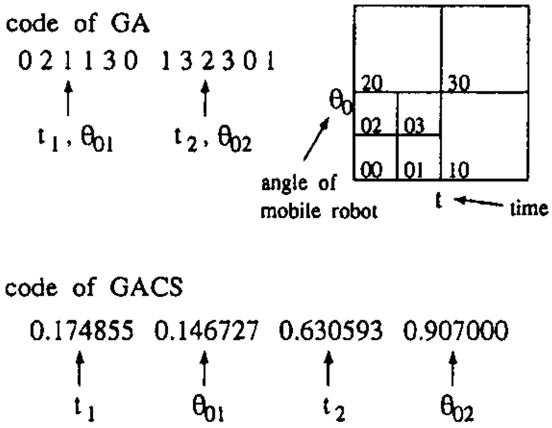


Fig. 14. Encoding process in GA and GACS schemes.

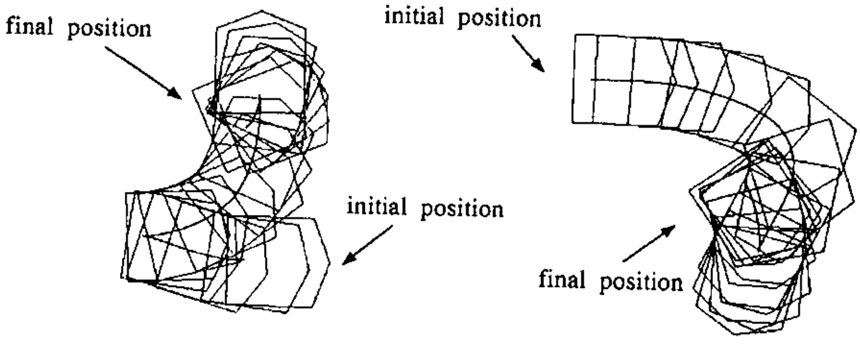


Fig. 13. Examples of turning motion of mobile robot.

F. *Soft computing with GACS* After a sufficiently long time, the probability density function (PDF) of the population will become narrow and concentrate around global minimum of the fitness function. In this case $x^{i}_{k+1} = x^{*i}_{k} + w^{i}_{k}$, $i = 1, 2, ..., N$, where w^{i}_{k}, $i = 1, 2, \ldots, N$ are independent and identically distributed m-dimensional random vectors with a zero mean, $E(w^{i}_{k}) = 0$, and a common density $f_{w_k}(\cdot\,|\,\cdot)$, $N \to \infty$; x^{*i}_{k} - intermediate population after selection at time k (but before mutation with conditional PDF $f_{w_k}(\cdot\,|\,\cdot)$ that characterizes the mutation operator at time k). Then

$$f_{x_{k+1}}(x) = \int_{\Gamma} f_{x_k}(y)F(y)f_{w_k}(x\,|\,y)dy \Big/ \int_{\Gamma} f_{x_k}(y)F(y)dy \quad \ldots\ldots\ldots\ldots\ldots (10)$$

or

$$f_{x_{k+1}}(x) = f_k'(x) * f_{w_k}(x), \quad f_k'(x) = f_k(x)F(x)/E[F(x_k)] \quad \ldots\ldots\ldots\ldots\ldots (11)$$

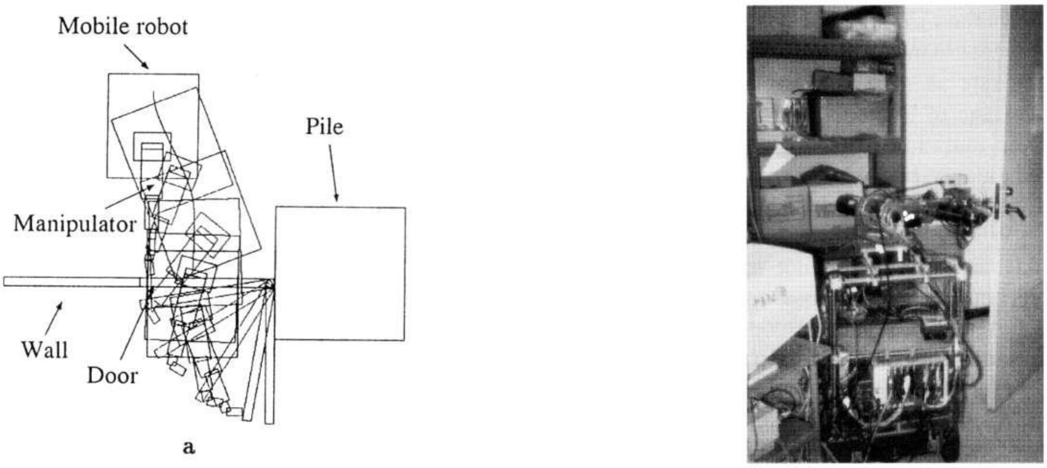
Operator * - m-dimensional linear convolution. From Eqs. (10) and (11), it is shown how evolution process consists of an alternation of implication with the fitness function (selection) and convolution with the mutation density (mutation): the former tends to "squeeze" the density of x around a global minimum of the fitness function, whereas the latter "spreads" the resulting distribution. The average radius of mutation is defined as [9,14],

$$\bar{r}(k,x) = \int_{\Gamma} \|\, y - x \,\| f_{w_k}(y - x)dy \quad \ldots\ldots\ldots (12)$$

and

$$\int_{\Gamma} \bar{r}(k,x)f_k(x)dx \le L^{-1}Var[F(x_k)], \quad \ldots\ldots (13)$$

where L - Lipschitz number. It is a sufficient condition for monotonous increase of average fitness. A large current average fitness requires small noise to guarantee a monoto-



a: Simulation result b: Experimental scene

Fig. 15. An example of a trajectory through a door.

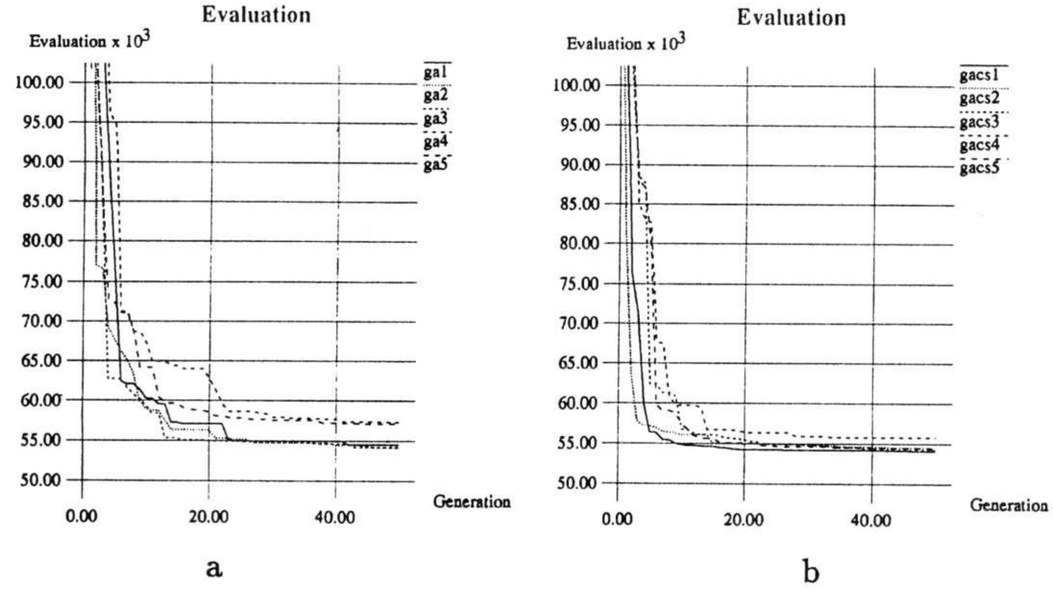


Fig. 16. Effect of parameter change.

Table 4. Parameters used for GA and GACS schemes.

	No. of GA and GACS				
genetic operator	1	2	3	4	5
crossover rate	0.0	0.5	1.0	1.0	1.0
mutation rate	1.0	1.0	1.0	0.001	0.5

nous characteristic; the population has already concentrated itself on regions with large fitness. A large fitness variance corresponds to a population still rather spread out and able to tolerate large mutation effects. At convergence, $V_{ar}[F(x_k)] \rightarrow 0$, the mutation can be reduced accordingly. From our simulation results of a benchmark problem such as opening a door, we have observed that conditions similar to the aforementioned lead to satisfactory solutions in many case for common classes of radically symmetric mutation densities. In this case, it is reported that a very effective integration of GA and fuzzy neural network computation were carried out.

G. *Results of computer simulation* A typical trajectory of opening door motion made by GA is shown in **Fig. 15**. Thus, the mobile robot succeeded in opening a door.

Figure 16 shows the difference of convergence between GA and GACS in accordance with change of the crossover rate and mutation rate as shown in **Table 4**. This data is obtained as an average of five data points in changing random speed. If the mutation rate was large, both GA and GACS converged quickly, because we used an *elite* strategy, which simply always retained the best chromosome out of the population. When the crossover rate was large, GA converged quickly, but GACS did not so. It shows that mutation is more important than crossover in GACS. We have compared GA and GACS, and confirmed that each algorithm had the following merits as well as demerits: In particular, **GA:** (1) strong in local optimum, (2) weak in local search, and (3) crossover is important to accelerate convergence. **GACS:** (1) strong in local search, (2) mutation is more important than crossover in order to accelerate convergence, and (3) weaker in local optimum than GA.

5. Conclusion

We discussed the main ideas, structure, simulation method, and examples of a behavior simulation system of a mobile robot for service use in VR based on NL and cognitive graphics. We propose to use an active adaptation block, which helps for mobile robot to learn new actions and scripts based on soft computing as fuzzy neural networks, fuzzy control, and genetic algorithms. Mobile robot for service use acquires in this case intelligent behaviors and flexibility in the execution of technological operations and the avoidance of obstacles in office buildings in coexistence with human beings.

Acknowledgments
Part of the work (connected with direct human-robot communications) developed by Litvintseva L.V. was supported partially by the Russian Fund of Fundamental Research under Grant No. 95-01-007729a. This support is gratefully acknowledged.

References:

1) C. Pellerin, "Service robots in the 1990's," Industrial Robotics, **20-3**, 34-35, 1993.
2) S. Asami, "Robots in Japan: Present and Future (The New Generation of Service Robot)," IEEE Robotics & Automation Magazine, 1-2, 22-26, 1994.
3) S.V. Ulyanov, K. Yamafuji, V.G. Gradetsky and A. Pagni, "Expert Fuzzy-Neuro Controller Design for Wall Climbing Robot for Decontamination of Nuclear-Power Station," J. Robotics & Mechatronics, 7-1, 75-85, 1995.
4) S.V. Ulyanov, K. Yamafuji, V.G. Gradetsky and T. Fukuda, "Development of Intelligent Mobile Robot for Service Use and Mobile Automation Systems Including Wall Climbing Robot: Pt. 1. Fundamental Design Principles and Motion Models," Intern. J. Intelligent Mechatronics, **1**-3, 111-143, 1995.
5) K. Yamafuji, Y. Yamazaki, T. Watanabe, K. Ogino, M. Hamuro and K. Saeki, "Development of Intelligent Mobile Robot for Service Use," Japan Soc. Mech. Engineers Ann. Conf. on Robotics and Mechatronics (ROBOMEC'92), Kawasaki, **B**, 269-270, 1992.
6) A. Ishikawa, K. Miyagawa, T. Tanaka and K. Yamafuji, "Development of Intelligent Mobile Robot for Service Use (2nd Report, Autonomous Locomotion Control)," Japan Soc. Mech. Engineers Ann. Conf. on Robotics and Mechatronics (ROBOMEC'94), Kobe, **A**, 353-358, 1994.
7) S.V. Ulyanov, K. Yamafuji, K. Miyagawa, T. Tanaka and T. Fukuda, "Intelligent Fuzzy Motion Control of Mobile Robot for Service Use," IEEE/RSJ Intern. Conf. Intelligent Robots & Systems (IROS'95), Pittsburgh, **3**, 486-491, 1995.
8) T. Tanaka, K. Yamafuji, K. Miyagawa, H. Watabane, H. Takahashi and S.V. Ulyanov, "Intelligent Locomotion Control System of the Mobile Robot for Service Use," 2nd Intern. Conf. Mechatronics & Machine Vision (M^2VIP '95), Hong Kong, 107-112, 1995.
9) J. Ohwi, S.V. Ulyanov and K. Yamafuji, "GA in Continuous Space and Fuzzy Classifier System for Opening of Door with Manipulator of Mobile Robot: New Benchmark of Evolutionary Intelligent Computing," IEEE Intern. Conf. Evolutional Computing (IEEE/ICEC'95), Perth, **1**, 257-261, 1995.
10) S. Ulyanov, L. Litvintseva, S. Takanashi and K. Yamafuji, "Cognitive graphics and Virtual Reality for Direct Human-robot Communication in Mobile Robot for Service Use," 6th Intern. Symp. Micro Machine & Human Science, Nagoya, 241-246, 1995.
11) L.V. Litvintseva, "Visualization of Three-Dimensional Scenes Based on Textual Descriptions for Intelligent Systems," Computer & Systems Sciences Intern., **31**-1, 98-105, 1993.
12) E.Ju. Kandrashina, L.V. Litvintseva and D.A. Pospelov, "The Spatio-Temporal Knowledge Representation in the Intelligent Systems," (in Russian), Nauka Publ.Moscow, 1989.
13) S.V. Ulyanov and L.V. Litvintseva, "Pseudophysical Logics for Intelligent Systems," (in Russian), Data Processing in Intelligent Robotic Systems, Moscow, 92-118, 1993.
14) X. Qi and F. Palnieri, "Theoretical analysis of Evolutionary Algorithms with an Infinite Population Size in Continuous Space: Pts. 1,2," IEEE Trans. Neural Networks, **5-1**, 102-129, 1994.

Name:
Takayuki Tanaka

Affiliation:
Research assistant, Department of Mechanical and control Engineering, University of Electro-Communications

Address:
1-5-1 Chofugaoka, Chofu, Tokyo, 182 Japan

Brief Biographical History

1996 - Graduated from Graduate School of University of Electro-Communications
1996 - Research assistant, University of Electro-Communications

Main Works:

- Development of Intelligent Mobile Robot for Service Use (1st Report, Adjusting-to-Environment Type Autonomous Locomotion system), Transactions of the Japan Society of Mechanical Engineers, (1996 Accepted)
- Development of Intelligent Mobile Robot for Service Use (10th Report, Autonomous Navigation control Based on Hierarchical Node Map), JSME Annual Conference on Robotics and Mechatronics (ROBOMEC '96), B, 957-960, (1996).

Membership in Learned societies:

- Japan Society of Mechanical Engineering
- Robotics Society of Japan

Name:
Ulyanov Sergei V.

Affiliation:
Professor, Department of Mechanical and control Engineering, University of Electro-Communications

Address:
1-5-1 Chofugaoka, Chofu, Tokyo, 182 Japan

Brief Biographical History

1975 - Central Institute of Building Constructions
1983 - Central Institute of Biomedical Engineering
1992 - Institute of Physicotechnical Problems
1994 - Institute for Problems in Mechanics
University of Electro-Communications

Main Works:

- " Fuzzy Models of Intelligent Industrial Controllers and Control Systems, Pts 1,2,3 " , Journal of Computer and Systems Sciences International, 33-1, 123-144 (1994), 33-2, 94-108; 117-136 (1995)
- " Expert Fuzzy-Neuro Controller Design for Wall Climbing Robot for Decontamination of Nuclear-Power Station ", J. of Robotics and Mechatronics, 7-1, 75-85 (1995)
- " Development of Intelligent Mobile Robots for Service Use and Mobile Automation Systems Including Wall Climbing Robots", International Journal of Intelligent Mechatronics, 1-3, 111-143 (1995)

Membership in Learned societies:

- Russian Society to Fuzzy Systems (RSFS)

Name:
Litvintseva Ludmila V.

Affiliation:
Associate professor, Dr., The Chief of Laboratory, The Artificial Intelligence Research Center of Program System Institute, Russian Academy of Science

Address:
Botik, Pereslavl-Zalessky, 152140, Russia

Brief Biographical History

1973 - Graduated from Moscow State Physico-Technical University, Department of Automatic Control Systems and Applied Mathematics
1976 - Computer Center, Academy of Science, USSR
1991 - Artificial Research Center, Program Institute of Russian Academy of Science
1996 - Visiting Schoolar, University of Electro-Communications, Tokyo

Main Works:

- "The Spatio-Temporal Knowledge Representation in an Intelligent Systems" (in Russian), Nauka Publ., Moscow, 1989.
- "The Question-Answer System Based on Natural Language" (in Russian), Izv. Russian Academy of Science, Technical Cybernetics, 2, pp.39-46, (1977).
- "The Spatial Scene Visualization of 3D Dynamic Scenes Represented on Natural Language" (in Russian), Izv. Russian Academy of Science, Technical Cybernetics, 5, pp. 75-83, (1991).
- "Virtual Reality - New Step in Man-Machine Communication Technology" (in Russian), Izv. Russian Academy of Science, Theory and Control Systems, 5, pp. 173-183, (1995).

Membership in Learned societies:

- Russian Association for Artificial Intelligence
- Member of International Academy of Informatization

Name:
Junji Ohwi

Affiliation:
Graduate Course Student, Department of Mechanical and Control Engineering, Electro-Communications, Graduate School of Electro-Communications

Address:
1-5-1 Chofugaoka, Chofu, Tokyo, 182 Japan

Brief Biographical History:
1995 - Graduated University of Electro-Communications and entered Graduate School of University of Electro-Communications

Main Works:
- "Development of Intelligent Mobile Robot for Service Use (8th Report, Intelligent path planning for mobile robot)", JSME Annual Conference on Robotics and Mechatronics (ROBOMEC '95), A, 501-502 (1995).

Membership in Learned Societies:
- The Japan Society of Mechanical Engineers (JSME)

Name:
Kazuo Yamafuji

Affiliation:
Professor, Dr., Department of Mechanical and Control Engineering, Electro-Communications, Graduate School of Electro-Communications

Address:
1-5-1 Chofugaoka, Chofu, Tokyo, 182 Japan

Brief Biographical History:
1973 - Graduated from Graduate School of Engineering, University of Tokyo Dr. of Engineering
Lecturer, Faculty of Engineering, Yamanashi University
1974 - Associate Professor
1988 - Professor, University of Electro-Communications

Main Works:
- Chemical Engineering & Micrometrics
- Fluid Mechanics & Turbo-machinery
- Hydraulics & Pneumatics
- Manufacturing Automation
- Robotics & Mechatronics
- Intelligent Robotics & Systems

Membership in Learned Societies:
- Japan Society of Mechanical Engineering
- Robotics Society of Japan et al.