Hierarchical Hybrid Neuromorphic Control System for Robotic Manipulators*

Takanori SHIBATA**, Toshio FUKUDA**, Shigetoshi SHIOTANI**, Toyokazu MITSUOKA*** and Masatoshi TOKITA***

In this paper, we present a hierarchical intelligent control system. We propose this system for generalization of the neural network-based controller using the higher-level control based on AI technology to acquire knowledge heuristically. Therefore, this system comprises two levels: a "learning" level and an "adaptation" level. The neural networks are employed for both the long-term learning of the control process and the short-term adaptation of the dynamic process. The learning level has a hierarchical structure for recognition and is used for the strategic planning of robotic manipulation in conjunction with the knowledge base in order to expand the adaptable range to the environment. New information from the adaptation level updates the learning level through the long-term learning process. On the other hand, the adaptation is used for the adjustment of the control law to the current status of the dynamic process. The motion controller at the adaptation level is particularly useful in nonlinear dynamical systems having uncertainty in the environment.

Key Words: Neural Network, Robotics, Artificial Intelligence, Adaptation, Learning, Knowledge Data Base, Knowledge Acquisition

1. Introduction

Research in neural networks has recently been growing as a new means of information processing[1]. The neural network is a mathematical model of biological brain neural networks. The brain has many excellent characteristics: for example, parallel processing of information, a learning function, and self-organizing capabilities. It can also provide associative memory[2] and is good for information processing such as pattern recognition. The neural network artificially connects many nonlinear neuron models and processes information in a parallel-distributed manner. In addition, the neural network has many interesting and attractive features such as a generalization capability. The neural network is proving to be useful as a tool in problem solving.

Incidentally, there are very many cases where automatic control theories and techniques play important roles. With the progress of control theories, there are now many applications for automatic control with increased performance. On the other hand, applicable systems have become increasingly complicated and highly composite; it is therefore expected that control theories and techniques will progress further. Like the model reference adaptive control (MRAC)[3] and the self tuning regulators (STR), adaptive control has become available to those systems containing much uncertainty. Nevertheless, conventional adaptive control had problems such as exponentially complicated calculation for the number of unknown parameters and fundamental limitations on the applicability to nonlinear systems.

The neural network makes use of nonlinearity, learning, parallel processing, and generalization capabilities for application to advanced intelligent control. For example, many attempts have been made to apply the neural network to control fields[4]-[17]. Particularly in recent years, studies have been carried out on control using the neural network for robotic manipulators. Kawato et al. performed feedforward control in such a way that the inverse system is built up by the

---

* Received 16th December, 1991. Paper No. 90-1499 A
** Dept. of Mechanical Engineering, Nagoya University, 1 Furo-cho, Chikusa-ku, Nagoya 464-01, Japan
*** National Kisarazu Technical College, 2-11-1 Higashi, Shimizudai, Kisarazu 292, Japan

Series C, Vol. 36, No. 1, 1993

JSME International Journal
neural network in trajectory control of the robotic manipulator, and have proposed a feedback error learning scheme. We showed that the neural network had some advantages over conventional control approaches, especially in nonlinear systems including uncertainty in the environment. In force control, one must consider not only system dynamics but also the characteristics of the environment as they affect the control loop. However, it is difficult to consider them. Therefore, we have proposed a neural network-based controller, the neural servo controller (NSC), to solve those problems. Then, we demonstrated its applicability to force control, stabling control, hybrid position/force control, and nonlinear impact control of robotic manipulators.

The problem in the previous studies is lack of generalizations. Since the neural network-based controllers have no “metaknowledge” or “data base” unlike human beings, they have only limited adaptable ranges. Especially in the case of force control, it is difficult for the neural network-based controllers to cover all characteristic ranges of the environment even though they have robustness. Therefore, the neural network-based controllers, as the neuromorphic control, require a global strategic planning level to generalize themselves and expand the task ranges.

On the other hand, in fields of intelligent information processing, such as symbolic or language processing, conventional AI techniques were used to manufacture knowledge-based systems as expert systems, which have proven to be useful. For motion control, there have been some examples of symbolic feedback control for higher-level feedback. However, it is difficult to classify the data in order to map the numerical sensed data into the symbolic data for recognition of the process state.

In this paper, we present a new system for intelligent control of robotic motion in order to solve previous problems. We call this system the hierarchical hybrid neuromorphic control system. This is a hybrid system of neuromorphic control based on the neural networks and the symbolic control based on the AI, which can acquire new knowledge self-augmentatively. Therefore, this system comprises two levels: a “learning” level and an “adaptation” level. The neural networks can be employed for both levels. The learning level has a hierarchical structure: recognition level and planning level for the control strategy of robotic manipulation. The neural networks transform the sensed signals from numeric quantities to symbolic qualities for recognition and planning in conjunction with the knowledge-based system. We call this knowledge data base system based on the neural network the neural knowledge data base (NKDB). On the other hand, the adaptation adjusts the control law to the current status of the dynamic process. Nonlinearities, their compensation, and uncertainty in the environment must be dealt with by the neural network. On the contrary, recent sensed information at the adaptation level updates the learning level through the long-term learning process. Eventually, the neural networks connect neuromorphic control at the adaptation level with the symbolic feedback at the learning level. In order to show the effectiveness of the proposed system, we deal with the grasping task in which the characteristics of the objects, representing the environment, are assumed unknown.

2. Concept of Hierarchical Hybrid Neuromorphic Control

The hierarchical hybrid neuromorphic control system is a hybrid system of the neural network and AI technologies for hierarchical intelligent control, and comprises two levels: a “learning” level for the long-term learning of the control process and an “adaptation” level for the short-term adaptation of the dynamic process (Fig. 1). The learning level has a hierarchical structure: recognition and planning for the control strategy of robotic manipulation. Plural neural networks are used at the learning level as nodes of a decision tree for reasoning. Those neural networks are trained using the training data by a
priori knowledge in order to transform various sensory data from numerical quantities to symbolic qualities, to perform "sensor fusion" and to produce "meta-knowledge." Vision, weight, force, touch, acoustic, and other sensors can be used. Then, the planning level performs reasoning for the strategic plans of robotic manipulation such as task, trajectory, force, and other plans in conjunction with the knowledge-based system. The system can include various kinds of "common sense" or "skill" for robotic manipulation. Thus, the learning level can reason unknown fact from a priori knowledge and detected information. Then, it can generate control strategies and approximate targets for the adaptation level depending on the state of the process, and it also evaluates the adaptation level using the knowledge. For these strategies, the database system maintains the control gains for the adaptation level, and particularly in the proposed system, it maintains the initial values of the interconnection weights of the neural network for the servo controller. The data are prepared through trial and error by the adaptation level and human expert in long-term learning. Moreover, the new detected information from the adaptation level updates the learning level through the long-term learning process.

On the other hand, the adaptation adjusts the control law to the current status of the dynamic process. In particular, the nonlinearities, their compensation, and the uncertainty of the environment must be dealt with by the neural network. Thus, the neural network in the "adaptation" process must work more rapidly than that in the learning process. We use the NSC at the adaptation level. As the NSC employs a dynamic neural network, the NSC can be applied in nonlinear dynamic control with uncertainty, as in the force control of a robotic manipulator. In this case, we do not need to sense acceleration. Eventually, the neural networks connect neuromorphic control with symbolic control.

In our proposed system, if the environment is unknown, the NKDB can recognize it. Moreover, if the NKDB which uses the strategy produced by the NKDB cannot adapt to the environment successfully, the NKDB evaluates the result and recognizes the recognition error which causes that failure at the adaptation level. In those cases, the NSC tries to adapt to the environment through trial and error so as to obtain force-sensory information, and feeds it back to the NKDB. In this way, the NKDB can acquire new knowledge about visual-sensory information by itself and force-sensory information from the NSC at the adaptation level. The neural networks in NKDB learn from the training sets of a priori knowledge, supplemented with new knowledge in the long term.

In this paper, we relate neuromorphic control at the adaptation level in section 3. Then in section 4, we show the recognition method and the knowledge acquisition method for the learning level. Finally in sections 5 and 6, we explain the effectiveness of hierarchical hybrid neuromorphic control while showing examples.

3. Multilayered Neural Network Model and Neuromorphic Control

Each neuron model composing up a network simulates a biological neuron. Figure 3 shows the multilayered neural network. The neuron unit consists of multiple inputs and one output, and its internal state is given as the weighted sum of input signals. Here, we define the weighted sum of the output of the previous layer,

\[ \text{net}_p = \sum_i w_{ij} o_{pj}, \]

as the state of the unit. The \( o_{pj} \) is the \( j \)-th element of the actual output pattern produced by the presentation of input pattern \( p \). The output,

\[ o_{pi} = f(\text{net}_{pi}), \]

often uses the sigmoid function, which is a continuous and nonlinear function. In the input layer, we use the linear function. The nonlinear sigmoid function is represented by the following expression:

\[ f(x) = \frac{1 - e^{-ax}}{1 + e^{-ax}}, \]

In order to obtain the appropriate interconnected weights of the units, we use the back-propagation algorithm\(^2\). The back-propagation algorithm is based on the least mean squares algorithm. In this algorithm, an error function is defined, and is equal to the mean square difference between the desired output and the actual output of a multilayered feedforward neural network.

In neuromorphic control, when we use the neural
network as the servo controller, we define the error function which is based on the feedback error of the system since we cannot know the desired output of the neural network as the control signal. In order to minimize this error function, the back-propagation algorithm uses a gradient search technique. At each sampling time, the interconnection weights are changed to reduce the feedback error based on the back-propagation algorithm. Figure 4 shows one of the neuromorphic control systems as indirect inverse control. In section 5, we describe the neuromorphic control system precisely.

4. Neural Knowledge Data Base

The NKDB at the learning level generates the control strategy for adaptation-level reasoning with sensory data and knowledge, and then has feedforward control. In the recognition of the control process, it is difficult to transform the numerical sensory information to the symbolic quality. Moreover, it is difficult to transform the strategy of the symbolic quality to the control gains as a numeral quantity because there is vagueness in the control strategy at the servo level, which is dealt with symbolically.

In this section, we explain the proposed approaches to solve those problems. In this paper, we assume that the robotic manipulator grasps objects whose characteristics are unknown. We deal with objects such as fruits and vegetables. For simplicity, we use visual-sensory information about shapes and colors, and force-sensory information about the stiffness of unknown objects. Here, we discuss the sensing method, the recognition algorithm, the knowledge base for planning, the reasoning method, and the knowledge acquisition algorithm.

4.1 Sensing methods

The conventional AI method employs "if...then..." rules to infer facts from a priori knowledge. However, when the numerical sensory information is mapped into a state described by symbols, there are many difficulties in deciding the rules and the number of assortment. On the other hand, there is a possibility of inferring unknown facts from the knowledge and multiple sensory fusion in some periods.

In this paper, we use the neural networks as nodes of the decision tree to classify sensory information for recognition (Fig. 5). Those neural networks are trained using the input-target training data by a priori knowledge in order to transform various detected data from numerical quantities to symbolic qualities, since the neural networks have mapping and generalization capabilities. The neural networks can store the knowledge in synaptic weights of inter-connections by learning and also make inference rules within themselves.

In general, human beings roughly recognize or identify the objects from their shapes and then, precisely from colors. Therefore, we use processed image data from a CCD camera as visual-sensory information. Data on shapes comprise the following preprocessed parameters: A) area, B) perimeter, C) roundness, D) normalized variance of the radius, E) maximum radius, F) ruggedness, G) ratio between length and width, and lengths in the directions of H) x axis and I) y axis.

Data on colors comprise the following preprocessed parameters: averages of J) red, K) green and L) blue of the image's pixels, M) number of spots, and N(1) - N(2) ratio of 30 defined colors.

In this paper, since we deal with objects such as fruits and vegetables, the above preprocessed parameters are sufficient to express the features of the objects. We use a fixed camera.

Fig. 4  Neural Network (Indirect inverse control)

Fig. 5  Neural knowledge data base
We use the neural networks to classify sensory information.
4.2 Transformation method and recognition algorithm

In this paper, the learning level recognizes the objects while employing neural networks as nodes in a decision tree (Figs. 5 and 6). We train the neural networks using the training data acquired from a priori knowledge which comprises the set of numerical sensory data for inputs of the neural networks and symbols for output of the neural networks corresponding to the states of the environment. The learning level for recognition comprises $m$ hierarchical levels and each level contains some knowledge of the objects. In this paper, we define two specific levels for recognition of objects. The neural networks at the first level recognize ‘sorts’ from the visual data of the shape and average of RGB colors. Then, the neural networks at the second level recognize ‘states’ which imply the stiffness sensed by the adaptation level, from the visual data of detailed colors. We express the numerical input data for the neural networks at each level as follows.

First level:


Second level:

$$< \text{in}_{2,P} > : = < N_1, N_2, \ldots, N_m >.$$  (5)

The output of the neural network is used as a flag indicating the neural network at the next level (Fig. 5). If the number of patterns is enlarged, it is difficult for one neural network to classify each pattern. Therefore we use plural neural networks to deal with a number of objects at each level. We define output space $X_o$ at each level as follows:

$$X_o = \{ x_{i,j}, x_{j,m}, \ldots, x_{j,n} \},$$  (6)

$$x_{i,j} = [10 \ldots 0]^T,$$

$$x_{j,n} = [01 \ldots 0]^T,$$

$$x_{j,n} = [00 \ldots 1]^T,$$

where $n$ : the number of units in the output layer of the $j$-th
neural network at the $i$ level,

$x_{ij}$ : the symbolic output pattern for the $k$-th training datum or knowledge.

The training data for the neural network is described as follows:

$$< \text{object } P > : = < \text{in}_{1,P}, \text{in}_{2,P}, x_{ij,k}, x_{st} >,$$  (7)

where $P$ is the name of the object.

The output patterns at the last level can be connected to the symbolic reasoning level for the control strategy. In this paper, the output patterns at the second level as the last level are connected with data base directly as the control strategy which provides the initial data for the NSC in this paper. The neural networks for recognition learn such knowledge data in a long-term. For simplicity, we assume that objects are foods, such as apples, oranges and tomatoes, and we give the neural networks at the learning level knowledge about them from experimental results. In order to make mapping and reasoning rules in the neural networks, in this paper, we train the neural networks using 27 patterns of training data. In this case, the neural network at the first level has 3 output patterns for sorts; therefore we define "$n" as equal to 3. Each neural network at the second level has 5 output patterns for states of objects for reasoning of the control strategy at the next level. In this paper, these patterns are classified by the force-sensory data depending on the adaptable ranges of the NSC.

4.3 Strategic planning

After the recognition of the state, the learning level performs reasoning for the strategic plans with a knowledge-based system including “common sense” or “skill” for robotic manipulation. In this paper, we simply use knowledge about the objects and the several patterns of manipulation which are maintained at the data base. Therefore, the learning level can generate the appropriate strategy by using the data base. The data base maintains tasks, approximate targets, and initial values of the interconnection weights of the neural network for the NSC. These initial values are obtained through trial and error by the NSC at the adaptation level. By using these initial values, the NSC can adapt to an unknown object easily, if the characteristics of the object are similar to those of a trained object. However, the adaptable ranges of the NSC are not so wide that the NSC cannot cover all characteristic ranges of the object with only one trained neural network in itself. Therefore, we must prepare a finite number of data for the NSC to adapt to the object and accomplish the task. Consequently, the learning level determines the approximate target force and control law for the NSC while depending on

![Fig. 6 Recognition method using the neural networks. If $x$, the output of the neural network, is not included in the subspace $X$, the system judges the object to be unknown.](image-url)
the estimated characteristic of the object.

4.4 Knowledge acquisition and learning algorithm for neural knowledge data base

Figure 7 shows a flow chart of the knowledge acquisition strategy for the recognition level.
1) If the output \( x \) of the neural network at the \( i \) level is involved in \( X_i \), go ahead to the next \((i+1)\) level.
2) If the flag "unknown" occurs (Fig. 6), then go to "unknown i."
3) If the decision derived from visual data is suitable for the planning level and if the NSC can adapt to the object, execution is finished.
4) In the case of a mistake in the adaptation to the object, the recognition level recognizes the mistake and goes to "unknown \( m \)" which is the last level of "unknown," since we use the fixed data base for the planning level.
5) In the cases of 2) and 4), the NSC executes 'trial and error' using data of the planning level to obtain the force-sensory data on the stiffness of the object (such as Table 1).
6) The recognition level obtains the force-sensory data using visual-sensory data as new knowledge for recognition.
7) The recognition level learns knowledge supplemented with new information. If the amount of knowledge for the \( j \)-th neural network at the \( i \) level is more than \( n \), in order to add the \((n+1)\)-th knowledge to the \( i \) level, the recognition level uses the \((j+1)\)-th neural network at the \( i \) level. We prepare new neural networks which do not have knowledge at every level, which are used to store new augmented knowledge. In this way, the recognition level can acquire new knowledge self-augmentatively in long-term "learning" from visual-sensory data and force-sensory data from the NSC at the adaptation level.

5. Modelling and Neuromorphic Control Method at Adaptation Level

In order to show the ability of the proposed scheme, in this paper, we assumed that the manipulator grasped objects whose characteristics were unknown. For simplicity, we used a robotic manipulator with 2 degrees of freedom, a finger of a grasping hand (Fig. 8). Simulations were carried out to obtain force-sensory data in the case of hybrid position/force control\(^{11}\), where the neural network is used as a controller to compensate for nonlinearities and uncertainty in the environment. In order to be effective for nonlinear dynamic control, the neural network applied in the neural servome controller (NSC) must represent the dynamics. We use here an active time delay neural network (ATDNN) having time delay elements in the hidden layer of a multilayered neural network as a dynamic neural network configuration\(^{12}\). The following section describes the configuration of the NSC.

5.1 Configuration of neural network including time delay elements

For the neural network to represent the dynamics of the system, time delay units were provided in the first hidden layer in the four-layered neural network. While the time delay neural network had been proposed for the recognition problems\(^{23}\), we

![Fig. 7 Flow chart of learning for recognition level](image)

![Fig. 8 Mathematical model of the two-dimensional robotic manipulator. We modeled the object as the spring mass damper.](image)
proposed the ATDNN for dynamic control in order for the neural network to have dynamics without the acceleration information and to express strong non-linearities. In the control of the robotic manipulator, acceleration information is very noisy. We have shown the effectiveness of the dynamic ATDNN for the servo controller\(^{(12)-(16),(30)}\).

The following is a description of the internal state of the second hidden layer. With \(a_i\) and \(\beta_i\) (positive or negative constant) used as oblivion coefficients, we express the equation of the internal state as follows:

\[
\text{net}_i = \sum \{ w_{ji}(t) \cdot o_i(t) + a_i \cdot w_{ji}(t) \cdot o_i(t-1) \\
+ \beta_i \cdot w_{ji}(t) \cdot o_i(t-2) + \theta_{ji}(t) \}
\]

(8)

When \(a_i\) and \(\beta_i\) are 0, a static general hierarchical neural network is built.

In order to accelerate the learning speed of the neural network at the adaptation level, we use the variable learning method, fuzzy turbo, based on a fuzzy set theory. Rules in fuzzy turbo are given by the NKDB. We have shown that fuzzy turbo can avoid stagnation and has insensitive characteristics at a stable extreme on learning\(^{(12)}\).

5.2 System configuration

Figure 8 shows the mathematical model of a two-dimensional manipulator. We modelled the handling object as a linear spring-damper mass. Figure 9 shows the neuromorphic control system of the two-dimensional manipulator, which is based on indirect inverse control (Fig. 5). We applied PID control for position control and PI control for force control. Thus, hybrid position and force control was applied using hybrid ratio \(s_1\). The ATDNN was applied to the NSC. The input layer had 12 units, each hidden layer had 8 units and the output layer had 2 units. Input data of the neural network consisted of targets and output of joint torques, joint angles, force at the tip, and joint velocity (except acceleration). The output of the neural network consisted of input torques and was added to each joint actuator. The neural network learned the system based on the back-propagation algorithm so as to reduce the feedback error between the desired output and the actual output of the force and the hybrid value of the torque and angle. Fukuda and Shibata\(^{(16)}\) showed that the NSC is effective for the force/impact control of the robotic manipulator, but in this paper, we did not take account of collision phenomena between the manipulator and its environment for the sake of simplicity.

6. Simulation Results

6.1 Recognition and planning for strategy by neural knowledge data base

The neural networks at each level were trained using 27 knowledge patterns in 1000 iterations in order to memorize and form inference rules. Input data were obtained experimentally from visual-sensory data, and output patterns depended on the force-sensory data obtained by the NSC at the adaptation level (Fig. 10, Tables 1 and 2).

After training, we fixed the interconnection weights of the neural networks at the recognition level. At the first "sort" level, the neural network recognized the sorts of 15 unknown patterns of

<table>
<thead>
<tr>
<th>Data base in the NKDB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data 1</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>Adaptive range [N/m]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Object</th>
<th>Numerical input data (vision)</th>
<th>Output vector (force sensed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orange 1</td>
<td>in_1_orange1</td>
<td>in_2_orange1</td>
</tr>
<tr>
<td>Orange P</td>
<td>in_1_orangeP</td>
<td>in_2_orangeP</td>
</tr>
<tr>
<td>Apple 1</td>
<td>in_1_apple1</td>
<td>in_2_apple1</td>
</tr>
<tr>
<td>Apple Q</td>
<td>in_1_appleQ</td>
<td>in_2_appleQ</td>
</tr>
<tr>
<td>Tomato 1</td>
<td>in_1_tomato1</td>
<td>in_2_tomato1</td>
</tr>
<tr>
<td>Tomato 2</td>
<td>in_1_tomato2</td>
<td>in_2_tomato2</td>
</tr>
<tr>
<td>Tomato R</td>
<td>in_1_tomatoR</td>
<td>in_2_tomatoR</td>
</tr>
</tbody>
</table>

Fig. 9 Hybrid position/force control system by the NSC at adaptation level. We applied PID control for position control and PI control for force control.

Fig. 10 Experimental image data
apples, oranges and tomatoes perfectly. We experimented with tomatoes to evaluate the recognition capability of the neural networks at the second level. We assumed that the stiffness of the tomato was classified into 3 patterns by the force-sensory data. At the second "states" level, the neural network trained using 27 patterns recognized the states of 15 other unknown patterns of tomatoes perfectly. It can be said that the learning level could recognize the objects. Then, the learning level performed reasoning for the strategic plans based on the knowledge. In this case, we assumed that the object was a "mellow tomato"; its stiffness $K^*$ was equal to 100 [N/m] and its damping coefficient $C^*$ was equal to 1 [Ns/m]. The knowledge base contained knowledge about the object; i.e., that "mellow tomato" was 'soft'. Using this knowledge in the strategy, the learning level generated the approximate target force, gains for the conventional feedback loop, and the initial values of the interconnection weights of the neural network for the NSC from the data base system.

6.2 Adaptation at adaptation level

Figures 11 and 13 show the simulation results of hybrid position/force control of the robotic manipulator by the NSC. In the case of Fig. 11, the NSC did not use the data base system (Table 1) as the control strategy produced by the learning level. Figure 11(a) shows the transition of the force at the tip of the manipulator. Figure 11(b) shows the sum of the square feedback error at one trial. Without the learning level, the NSC was able to adapt to the unknown object, however, it required many attempts and inclusion of much error, since the NSC did not know the characteristics of the object. Moreover, there was the possibility of destruction of the object, since the NSC did not know the desirable manner of manipulation.

Using the strategy by the learning level, the conventional controller is given the gains from the data base system. However, if the estimated characteristics of the object are different from the actual characteristics, the conventional controller without the NSC cannot control the manipulator sufficiently since it cannot adapt to the object (Fig. 12). In the case of Fig. 13, the NSC was applied to the system at the adaptation level and was given the initial values of the interconnection weights of the neural network by the data base system at the learning level (Table 1). The NSC adapted to the object in a few iterations and the manipulator held it appropriately.

6.3 Knowledge acquisition of neural knowledge data base

In order to examine the ability of knowledge acquisition and learning, we experimented and simulated a case where bananas were the unknown objects. When we input 6 patterns of visual data of banana, the neural network output $x_{10}=(-1, -1, 0) x_{11}$ at the first level, and recognized "unknown" in all cases (4.4-2). To obtain force-sensory information,

![Fig. 12 Simulation results: Hybrid position/force control of robotic manipulator by only conventional feedback control with the learning level](image1)

![Fig. 13 Simulation results: Hybrid position/force control of robotic manipulator by the NSC with the learning level](image2)
Table 3: Acquired new knowledge
It comprises visual-sensory information and force-sensory information

<table>
<thead>
<tr>
<th>New Object</th>
<th>Numerical input data (vision)</th>
<th>Output vector (force sensed)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Level 1</td>
<td>Level 2</td>
</tr>
<tr>
<td>Banana 1</td>
<td>in_1_banana1</td>
<td>in_2_banana1</td>
</tr>
<tr>
<td>Banana S</td>
<td>in_1_bananaS</td>
<td>in_2_bananaS</td>
</tr>
</tbody>
</table>

In this way, the recognition level acquired new knowledge, inclusive of the existing knowledge and formed new reasoning rules for recognition. Presently, we are able to demonstrate the knowledge acquisition algorithm for the recognition level. In the future, we must attempt to solve the problem of forming super-class in order to establish more intelligent control.

7. Conclusions

In this paper, we proposed a new system for intelligent control of robotic manipulators. Through simulations, we showed that the NKDB at the learning level can reason out a desirable strategy for the adaptation level by using the neural networks and can acquire new knowledge from the visual-sensory and force-sensory information obtained by the NSC at the adaptation level. Although the simulations are simple, they contribute a clear illustration of the potential in the proposed concept. Our next aims are to develop a more complex reasoning algorithm to solve arrangement problems which avoid the explosion of knowledge, and to demonstrate the applicability to more complicated tasks and situations.

Acknowledgement

The authors gratefully acknowledge the constructive comments and technical assistance of Prof. K. Kosuge and Prof. F. Arai.

References

8. Psaltis, D., Sideris, D. and Yamamura, A., A Multi-

Series C, Vol. 36, No. 1, 1993

JSME International Journal


(17) Kuperstein, M. and Wang, J., Neural Controller for Adaptive Movements with Unforeseen Pay-


