On a Primitive Skill-Based Supervisory Robot Control Architecture

G.Milighetti, H.-B.Kuntze, C.W.Frey, B. Diestel-Feddersen, J.Balzer

Systems for Measurement, Control, and Diagnosis (MRD)
Fraunhofer Institut IITB
Fraunhoferstraße 1, 76131 Karlsruhe, Germany
Milighetti@iitb.fraunhofer.de; Kuntze@iitb.fraunhofer.de

Abstract – Smart interaction of humanoid robots in a complex public, private or industrial environment requires the introduction of primitive skill-based discrete-continuous supervisory control concepts. The functionality of the proposed hierarchical robot supervisory control architecture captures both the hierarchy that is required for representing complex skills as well as the mechanisms for detecting failures during their execution. At first by means of several complementary (e.g. internal, optical, tactile or acoustic) sensors and by Neuro-Fuzzy based fusion of relevant sensor features, the actual motion phase or fault event is continuously diagnosed. Depending on the identified motion phase or random fault event, the most appropriate discrete-continuous control strategy coping optimally with the corresponding situation will be selected and executed. First experimental and simulation results are reported in this paper.

Index Terms – Multi-sensor perception, Neuro-Fuzzy diagnosis, primitive skills, hierarchical supervisory control architecture, discrete-continuous control strategy

I. INTRODUCTION

Increasing complexity of robot tasks in public, private and industrial environments, requires more and more the introduction of a new generation of „humanoid“ robots able to interact with humans in a smart manner. In regards to the control requests humanoid robots differ from industrial robots on the market, especially in the following characteristics:

- close man-robot partnership to solve complex tasks cooperatively
- smart multi-sensorics for communication and interaction of the robot with its environment (e.g. human)
- learnability of the robot for acquiring human skills
- extreme safety and availability by fault tolerant hardware and software components
- high flexibility by dynamic reconfiguration of hardware and software components.

In the past, very efficient concepts have been developed for the sensor-based monitoring and control of robots. However, these sensor-based regulation concepts have been realized only for individual subtasks and/or movement phases in the manufacturing environment. For managing more sophisticated human interactive tasks the robot has to depose of sensor-based skills dedicated to specific application areas. Typical skills may be e.g. “to fit an object into a hole” or “to pick and to place randomly dropped parts from the bottom”. Each skill of humanoid robots is composed of a complex set of primitive skills (PSs) which can not be substructured in smaller actions. Thus, each PS may be e.g. described by specific autonomous differential equations.

For solving complex skills a hybrid discrete-continuous supervisory control architecture which has to cope with the following two problems is required. On the one hand depending on the multi-sensoric perception of the robot state, it has to provide an optimal selection and activation of different PSs (discrete control). On the other hand it has to continuously to control each PS by means of appropriate sub-controllers (continuous control).

Marketable industrial robot control architectures can not comply with such advanced requests. Thus, for the perception, learning and control of autonomous robots different system architectures have been proposed in the past. The majority of concepts deal with the learning of PS-based movements by system architectures based on neural networks (NN). The first NN-based hierarchical system architecture has been proposed by Albus [1]. Later the theoretical background of PS-learning has been laid by Schaal et al. [2]. As regards the multi-sensor perception of autonomous robots different cognitive architectures have been proposed in the past. While the ADAPT concept [3] deals mainly with hierarchical perception and planning schemes the GRAVIS architecture developed by Steil et al. [4] provides a more comprehensive concept for grasping tasks by gesture instruction.

The first promising system concept for multi-sensor based discrete-continuous control applied to multi-fingered robot grasping have been developed by Buss et al [6], [5]. A generic software architecture suitable for different PS based perception and control strategies has been recently proposed by Finkemeyer [16].

The PS-based supervisory control architecture proposed within this paper captures both the hierarchy that is required for representing complex skills as well as the mechanisms for detecting failures in execution of skills. It is currently developed by the Fraunhofer Institute IITB within the framework of the Collaborative Research Centre SFB 588 “Humanoid Robots” of the Deutsche Forschungsgemeinschaft (DFG). In this paper about first experimental and simulation results is reported.
II. SUPERVISORY CONTROL CONCEPT

The proposed supervisory control concept (cf. Figure 1) is able to cope with manifold scenarios and has already been successfully applied by IITB for the flexible automation of a complex industrial batch process [10]. It relies on the decomposition of the complex control problem into a sequence of smaller, more transparent phase specific sub-control problems.

In the upper control level the actual process phase is identified with the help of both model based and multi-sensor diagnosis. From a designed sequence of PSs constituting the task the optimal PS is activated in order to reach the planned goal.

The lower level of the control structure consists of different sub-controllers which are optimized with respect to the specific PS. Principally the various low level sub-controllers may have any structure (Fuzzy control, model-based or hybrid).

Thus, depending on the identified process phase a mode selector chooses the most appropriate control strategy, which can comprise both a redesigning of the task and a reconfiguration of the control routine including both the adaptation of control parameter sets without structural changes and the switching of different situation-specific low level controller structures. Of course, structural switching requires the compatibility of the alternative controller structures. Moreover, shock disturbances have been avoided by introducing a Fuzzy-based soft switching strategy (see [11] for details).

III. MULTI-SENSOR FUSION AND DIAGNOSIS

In order to automatically diagnose malfunctions and special phases or states of a technical process the successive steps of diagnosis signal (residual) generation and evaluation have to be performed [12]. This general abstract model allows a broad variety of different realizations and implementations.

A. Residual Generation

Within the residual generation step the available sensor signals of the process are preprocessed for the purpose of extracting relevant signal characteristics. The generated residuals can be regarded as a condensed signal representation, ideally containing all important signal information. Basically residual generation in technical processes is performed by the following methods (cf. Figure 2):

- signal based methods
- model based methods

At IITB the model based approach is used in order to monitor the robot axes. Following the method proposed in [7] founded on the idea of the generalized momenta, a residual vector with \( n \) decoupled components (with \( n \) the number of the axes) is generated to supervise the dynamic behavior of each joint. An adaptation scheme is implemented in order to cope with the unavoidable inaccuracies of the model (see [8] for details).

At the same time the robot interaction with the environment is monitored through the information coming from the available external sensors (e.g. force sensor, microphone). In this way a hybrid residual vector in the
can be designed.

B. Residual Evaluation

Once the diagnostic signals have been generated they must be evaluated. The purpose of this second step is to classify the actual motion phase or fault event according to the available signals.

Desirable is a residual evaluation module which stores the decision knowledge in an interpretable and modifiable form. In this respect Fuzzy Logic provides an ideal tool for realizing residual evaluation modules. Fuzzy Logic gives the possibility to describe knowledge by linguistic rules like e.g.:

\[
\text{if Residual } 1 \text{ is } \ldots \text{ and Residual } 2 \text{ is } \ldots \text{ and Residual } n \text{ is } \ldots \text{ then Process Phase...} \]

The implementation of a Fuzzy module for residual evaluation can be very difficult with an increasing number of residuals to be taken into account. The problem of finding appropriate membership functions and rules is often a tiring process of trial and error. Just like linear classifiers, Fuzzy systems require in contrast to Artificial Neural Networks (ANNs) manual tuning to obtain good classification results. In order to automate the design phase of the Fuzzy system in the proposed diagnosis scheme Neuro-Fuzzy (NF) approaches are used for designing Fuzzy residual evaluation modules.

Therefore, the NEFCLASS (NEuro Fuzzy CLASSification) approach proposed by Nauck [9] has been followed. The NF model is characterized by a three layer topology (cf. Figure 8). The input nodes in the input layer are connected by Fuzzy sets \( \mu \) with the rule nodes \( R \) in the hidden layer. For semantical reasons each rule unit is assigned to a single output node. In the output layer in order to avoid weighted rules the weights are fixed for convenience to 1.

To obtain an optimal classification result the learning algorithm creates the rules and adjusts the Fuzzy sets from training examples. After training the system the classification knowledge can be easily accessed and extended by the user [9].

IV. HEURISTIC CONTROL MODE SELECTION

The heuristic control mode selection has to consider both the discrete and the continuous state of the robot in its environment. The discrete state is identified by the NF based process phase and fault event detection, the continuous one by the variables identifying the robot dynamics and the sensor measurements. Thus, depending on the hybrid process state a discrete-continuous control law has to be designed which is able to manage the manipulation sequence in the appropriate way without instability in the transition phases. Unfortunately, no comprehensive modeling paradigm exists so far to solve the hybrid control design problem analytically [6].

Therefore, the hybrid control decision concept proposed in this paper relies on a heuristic control mode selection approach.

First of all a transparent modular software architecture which permits to build easily a flexible discrete structure of every robot task has been implemented. Thanks to such an architecture every task can be seen as a sequence of simpler sub-tasks; the goal of each sub-task can thus be achieved by performing simple Skills (cf. Figure 3). Moreover each skill can be divided into a succession of still easier Primitive Skills (PSs) which realize elementary functionalities of the robot (see [16] for details). Just to illustrate a simple example, let’s consider a task during which the robot has to prepare a shake. This task can be decomposed in the four skills “look for ingredients”, “pick them and put them into the shaker”, “plug in the shaker” and “switch it on”. These skills are then realized by PSs such as “move down”, “open gripper”, etc.

Parallel to the ideal sequence of PSs needed in order to achieve the task, other sub-optimal solutions can be designed, thus obtaining alternative PS-paths which realize a PS-tree (cf. Figure 4). During the execution of the task, once a PS has been successfully (leave condition fulfilled)
or abruptly (some failures or resources no more available) ended, depending on the identified actual hybrid state of the robot and of its environment, the optimal path and the most suitable control strategy with the most appropriate sub-controller is activated by the mode selector. Hitchless activation of new control routines as well as de-activation of still running ones is managed by fuzzy based switching algorithms.

Let us assume for example that the PS2a in Figure 4 has been ended, two possibilities arise in order to reach the goal: both the PS4 and the PS4a can be activated and a choice between these two options has to be made. According to the sensor information and the diagnosis signals the paths of the tree are weighted and the optimal path is chosen using a pathfinding algorithm; thus, also the next PS that has to be activated can be consequently selected.

The head is equipped with a 3D stereo camera and a stereo acoustic sensor (microphone array). Moreover an optical 3D sensor (laser stripes sensor) is integrated in the gripper and a force–torque sensor is mounted on the wrist.

The implementation of the control program is developed in C++ under Windows.

For the first investigations a simulation platform is also available. It consists of a robot model realized with ADAMS® and controlled in the MATLAB/Simulink® environment.

Various skills of robots which require multi-sensor interaction with its environment have been first investigated on the simulation platform and then validated on the experimental one. Three of these will be presented in the following, the first one stressing more the residual generation and evaluation part, the second one emphasizing a task architecture based on primitive skills, the last one stressing the sensor fusion.

**B. Constraint Motion with Collision**

One important basic skill of a robot is a hybrid motion of its arms within a complex constraint environment with different obstacles. The motion is composed of two phases: “free position control motion” and “constraint force control motion” along the obstacle surfaces. The inevitable collision must not damage either objects or robot mechanics.

Therefore this action has been thoroughly investigated on the experimental robot platform taking as nominal motion a circular planar motion (cf. Figure 6). In order to verify the effectiveness of the diagnosis method, an actuator fault of the 4th actuator has also been introduced in the interval \( t \in [2, 3] \) sec.

In Figure 7 some calculated residuals are shown; the residual vector in this case includes the seven residuals coming from the seven axes of the robot (generated by the

![Figure 4. Example of a Primitive Skill Tree](image)

![Figure 5. Experimental robot platform](image)
measurement of positions, velocities and commanded torques) and by one more residual coming from the microphone signal filtered by the band pass \( H(z) \).

Figure 6. Circular motion constraint by a wall

Figure 7. Example of residual generation

From the time responses shown, it can be seen that it is very difficult to detect clearly with the naked eye (not to mention to classify) the malfunction in the joint. Moreover the signal disturbances due to collision can be clearly identified but without additional information it might be difficult to distinguish them from other fault classes. Thus, after the generation phase an evaluation phase must anyway be carried out.

The results coming out of the NF evaluation of the residual vector are shown in Figure 8.

The output of the neural network is a probability for each possible phase considered. By choosing the phase with the highest probability at each time step, a discrete classification is obtained. E.g. the transition between the fault free phase (FF) and the fault in the 4th joint (FJ4) at the time \( t = 2 \) seconds can be clearly seen.

Figure 8. Example of residual evaluation

Depending on this result the most appropriate control strategy is consequently chosen. That means a reconfiguration of the robot kinematics for the FJ4 and a switching between position and force control for the collision. In order to avoid shock disturbances and to have a transition between the different controllers as soft as possible, a fuzzy switching has been implemented.

C. Fitting an Object into a Hole

A further basic problem which has frequently to be solved by robots is the so called “peg in hole” problem. Therefore an intelligent concept for performing part mating based on the supervisory control concept has been developed.

The main issue that has to be addressed is that of high contact forces during the insertion process. These can result from uncertainties of the location of the peg relative to the hole caused by positioning errors of the robot or grasping errors of the gripper. Passive and active compliance techniques are used successfully in automated assembly operations. In case of a robot acting in a time variant environment there is no precise a priori knowledge about the location of the hole, in contrast to industrial assembly tasks. The proposed concept solves the “peg in hole” problem in five different phases (every phase intended as combination of more primitive skills) by combining visual sensor with force-torque sensor data. A constant monitoring of different process parameters is required in order to classify correctly the current state of the insertion process and to choose consequently a suitable strategy to follow and the appropriate sub-controllers to activate (see [13] for details).

The presented concept is basically derived from human behavior and it is represented by the flow chart in Figure 9 with the sequence of the PSs and their leave conditions.

First the hole center coordinates are approximately estimated by the stereo camera. An impedance controller or position controller is activated for the first motion phase (phase I) during the approach to the hole’s environment. A detection of contact with the surface initializes the second phase (phase II).

In this case a hybrid force/position control is used to scan the surface in order to locate the exact position of the hole.
Once the peg partially dips into the hole during the scan, the processed torque signal peaks. This peak occurs when the peg lies nearly in the center of the hole. This estimation can thus be used to plan a new trajectory (see [15] for details).

Phase III, still in hybrid control mode, consists of a tilted insertion of the peg until the characteristic three points contact configuration described in [14] has been established.

Phase III, still in hybrid control mode, consists of a tilted insertion of the peg until the characteristic three points contact configuration described in [14] has been established.

This phase is followed by the execution of a model based motion with predefined velocity profiles, which ensures the stability of the contact with the hole and results in a good pre-insertion alignment (phase IV). Finally in phase V the peg can easily be inserted without wedging and jamming at the entrance. For the last two phases a switching back to position control takes place.

Figure 10 shows the simulated different motion phases and the corresponding diagnosis signals monitored in order to classify them. Force/torque data as well as alignment error are used. It can be clearly seen that the switching to the next phase can be easily detected (see [13] for details).

In Figure 11 some pictures taken from the experimental process are shown.

Figure 9. Flow chart of the “peg in hole” skill

This phase is followed by the execution of a model based motion with predefined velocity profiles, which ensures the stability of the contact with the hole and results in a good pre-insertion alignment (phase IV). Finally in phase V the peg can easily be inserted without wedging and jamming at the entrance. For the last two phases a switching back to position control takes place.

Figure 10 shows the simulated different motion phases and the corresponding diagnosis signals monitored in order to classify them. Force/torque data as well as alignment error are used. It can be clearly seen that the switching to the next phase can be easily detected (see [13] for details).

In Figure 11 some pictures taken from the experimental process are shown.

D. Pick and Place of a Dropped Part

A very frequent skill which has commonly to be managed by robots is a visual based pick and place operation concerning objects which randomly drop from the gripper during a carrying process. The random position and orientation of the fallen off part have to be detected, identified and measured by visual and acoustic sensorics.

First of all the noise due to the dropped part is identified by the audio array and is used to adjust the head orientation towards the object location. The falling event is also detected by the information coming from the residuals extracted from current and position of the gripper.

Then the stereo camera of the robot head is used to identify and to measure the coarse object position.

Finally the 3D camera attached to the arm wrist provides a more precise measurement of position and orientation of the object by visual servoing.

By approaching to the object the localization will become more and more accurate. At the end the part can be picked up precisely by the gripper.
The proposed visual servoing concept is based on the fusion of the measurements of the two visual sensors according to their different working range, precision and transmission time. Two different phases can so be distinguished (cf. Figure 12): a first phase (wide range) where only the information from the camera is used and a second one (close range) where both sensors are used simultaneously (see [17] for details).

![Figure 12. Two phases solution of the visual servoing problem](image)

V. CONCLUSIONS

In this paper a new primitive skill-based supervisory control concept for robots has been presented. It relies on a Neuro-Fuzzy based diagnosis of the current motion phase or fault event. For each identified situation an appropriate dynamic reconfiguration either of sub-tasks through the activation/deactivation of primitive skills or of optimal sub-controllers is realized. The efficiency of the proposed concept has been investigated by various experiments and simulations. Different human-like basic skills such as constraint motion with collision and fitting an object into a hole and pick have been considered.

ACKNOWLEDGEMENT

The authors gratefully acknowledge that the research project presented in this paper has been supported by the Deutsche Forschungsgemeinschaft (DFG).

REFERENCES

[16] Bernd Finkeneyer: “Robotsteuerungsarchitektur auf der Basis von Aktionprimitiven", Fortschritte in der Robotik, Band 8, Shaker Verlag