Adaptive control of a modular robot system

Adaptive Control of a Modular Robot System

P. C. Mulders (2), R. J. Voorkamp, J. Jansen, A. C. H. van der Wolf (1); Technical University Eindhoven/Netherlands
Received on January 16, 1992

Adaptive control is a process of modifying one or more parameters of the controller and these adaptive control algorithms are specially important for flexible manipulators with place- and time dependent parameters, varying during trajectory performance.
Here an adaptive controller is described as a combination of the computed torque method and an adaptive PD controller based on the Model Reference Adaptive Control (MRAC) method.
It has been applied to a modular robot for loads up to 50 kg consisting of a linear and a rotary actuator showing these parameter variations.
Necessary models - extended and reduced - of this modular robot have been made and the proposed controller has been tested in simulations and in the real configuration also with respect to stability, convergency and robustness.

KEYWORDS: Robots, adaptive control.

1. INTRODUCTION

The aim of this work is a study of (optimal) adaptive control algorithms on systems with place- and time dependent parameters-varying during trajectory performance - and to implement this on mechanical manipulators of industrial scale. Experiments on this item already have been done with a linear robot arm.[1]
To test these advanced control systems, a modular robot system - for loads up to 50 kg, consisting of a linear and a rotary actuator, as shown in Fig. 1. - has been constructed.
On the robot system a 3D-force sensor is mounted to perform teach- and replay trajectory operations. After the teach-operation the desired trajectory is again performed eventually with varying parameters, in which the already known motor control signals are updated by the adaptive control algorithm.

Industrial robots are used today for various purposes and until now robotcontrol has been studied mostly under the assumption that actuators are stiff and that the links can be modelled as rigid bodies. Therefore most robots have a very stiff construction in order to avoid deformations and vibrations.
For higher operating speeds industrial robots should be light weight constructions to reduce the driving torque/force requirements and to enable the robotarm to respond faster.Hence, more accurate dynamic models should be taken into account to pursue better dynamic performance.

With respect to these developments a number of (optimal adaptive) trajectory control strategies may be mentioned here e.g.:

- the PID method
- the computed torque method
- the model reference adaptive control (MRAC) method.

All these methods should be considered with regard to convergency, stability and robustness.
With the P.I.D.-controller the deviation from the nominal trajectory is used in proportional, integral and differential form to correct and the P.I.D. gain factors are chosen with respect to the systemdynamics. For coupled systems with interaction this controller type leads often to instability.
By the robust and simple structure, the PID controller is often used as a standard to compare with other controllers.

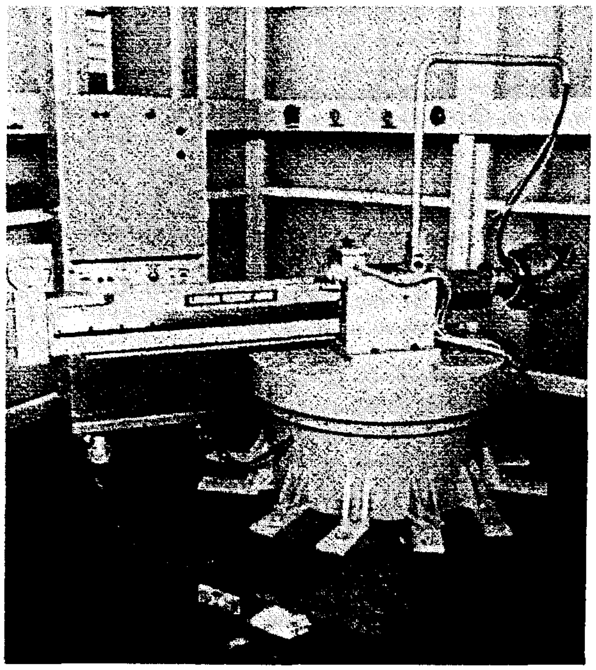

Fig 1 Photograph of the modular robot system.

The linear optimal controller is based on the minimization of a performance criterion-function, wich may contain e.g. contributions of the deviations in trajectory positions and velocities, but also the control efforts like the motor control signals. [1] Even the boundaries for the control signals may be taken into account. The feedback control signals are formed by an optimal linear combination of the state variables of the system, which means optimal pole-placement.
The updating of parameters in the algorithm for adaptive control depends on the time to solve the matrix-riccati equation - derived from the performance criterion function - and this is strongly dependent on the number of state-space dimensions of the model. So the optimal trajectory controlalgorithm is based on a good knowledge of the system, but on the other hand the model should not be too complex, because it might increase the computation time of the optimal control law, so that on-line control becomes impossible.
The two concepts mentioned above are not well applicable to flexible robots with elastic deformations and time varying parameters.

So another approach to improve the behaviour of robots is the computed torque control method, sometimes called the inverse dynamics control. The necessary torques are calculated from the prescribed trajectory and here the control law is designed explicitely on the basis of a model in order to compensate for robot non-linearities. If flexibilities play an important role, it often results in an unstable system behaviour.
So the aim is to search for a control law achieving both reasonable trajectory tracking and a certain stabilization of acceptable vibrations.
Adaptive control is a process of modifying one or more parameters of the structure of the control system to force the response of the closed-loop system towards a desired trajectory. Among the various types of adaptive robot control systems the model reference adaptive control (MRAC) systems are important, since they lead to relatively easy to implement systems, with a high speed of adaptation and may be used in a variety of applications. However it is still difficult to derive convergence, stability and robustness conditions.
The applied MRAC - system is described more detailed in Ch.3 .
Study of the various control methods have been performed both in reality and computer simulations. So from this modular RT robot system an extended model - simulation model - has been made, which is verified by modal analysis techniques. Next this extended model has been reduced to a control model on which the various mentioned control methods are based.

2. DESIGN AND MODELLING OF THE MODULAR ROBOT

2.1 The construction of the linear robotarm.

maximum velocity	1 m/s
maximum acceleration	5 m/s^2
maximum load	50 kg
stroke	1 m
position accuracy	0.01 mm
position measuring system	Heidenhain LS 513
power source	DC-motor: Axem MC 19 PR 26, 1 kW
control system	μC PID - or state controller

Table 1. Design specifications of the linear robotarm.

The mechanical construction is fairly stiff due to the hollow frame construction. The rotation of the motor into a translation of the actuator is converted by a spindle with a ballscrew nut. An advantage of this combination is that the back-lash can be eliminated by preloading the nut.
The DC motor is of the disc-armature type. Coupled to the motorshaft is also a tachogenerator and a rotational encoder.

For direct position measurement along the arm an optical linear digital incremental encoder has been mounted.type Heidenhain LS513 with a length of 1020 mm and an accuracy of 0.01 mm. The necessary frequency range of the encoder is determined by the speed of the arm and the accuracy of the lineal.
The free end of the linear robotarm is extended with a 3D- force sensor, based on the bending principle and measured by strain gauges. The force sensor is used in the TEACH mode.

2.2. The construction of the rotational module.

The mechanical construction is based on a cylinder with side ribs - to minimize the deformation - and is fixed to a groundplate.
The transmission from the motor to the turntable consists of a 4 stage toothed wheel combination with divided and preloaded wheels - realized with torsion springs - to eliminate backlash.

maximum angular velocity	$\pi/2$ rad/s
maximum angular acceleration	$\pi/2$ rad/s^2
maximum angular range	$\pi \rightarrow 2\pi$ rad
position accuracy	10^{-5} rad
position measuring system	Heidenhain LIDA 360
power source	DC-motor: BBC-MC 19P, 1 kW
control system	μC PID - or state controller

Table 2. Design specifications of the rotational module.

Coupled to the motor shaft is also a tachogenerator and a rotational encoder. For direct position measurement of the turntable an optical digital incremental encoder as a lineal has been mounted along the circumference of the turntable type Heidenhain LIDA 360 with 20200 lines and a pulse shaper EXE-702, and an interpolation factor of 1; 5 and 25 and a subdivider of 1; 2 and 4. By this the accuracy may be multiplied by 25 x 4 = 100.

2.3 The hierarchical control structure.

The on-line computer-capacity of one controller is often too small or not fast enough to implement an advanced control algorithm in real time.

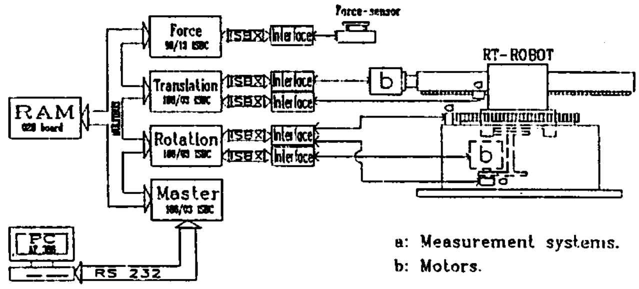

Fig. 2. The hierarchical controller system.

The controller system (Fig. 2) consists of 4 Intel single board computers - working in parallel - and 1 RAM board.
The boards can communicate directly - via the multibus system. Data however are transported to each other via the common RAM-board 028.
The rotation as well as the translation is controlled by its own board Intel SBC 186/03, coupled to its own input (the measuring system) and its own output (the motor amplifier) by the Intel SBX-interfaces.
The task of each SBC 186/03 is:

. to calculate - according to a control algorithm - the motor voltages
. to read the position of each module
. to store these data - motor voltages and positions - into the RAM board.

The task of the master SBC 186/03 is to:

. synchronize the software in both the other SBC's
. transfer data over the RS 232 bus.

A PC 80386 is coupled to the master. In this PC software may be developed and tested. By a data switch it can be used as a terminal for each SBC.
The optimal control law and the nominal trajectory is calculated off-line by this computer and the results are transmitted to the SBC's.
The PC also serves to diagram the measurement data.

2.4 Modelling of the modular robot.

Although the modular robot has been constructed with many distinct components, each one with its own properties like mass, stiffness etc, the combination leads to a system with divided parameters.
An attempt is made to realize a lumped mass model which describes the behaviour of the robot as good as possible.This approach is a.o. based on previous studies [5] about drives of motor-tacho-spindle-carriage combinations.
This model has 11 degrees of freedom:

$$q = [\ \varphi_0\ \varphi_1\ \varphi_3\ \varphi_5\ \varphi_7\ \gamma\ \psi_0\ \psi_1\ \psi_2\ s_1\ x]^T.$$

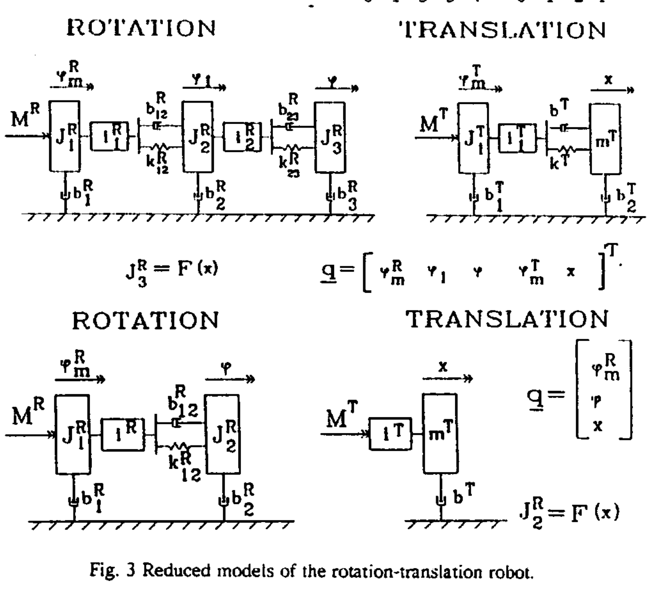

Fig. 3 Reduced models of the rotation-translation robot.

The rotation and translation are coupled to each other, so also the eigenfrequencies. These are also dependent on the position x(t) of the linear actuator (varying from - 0.270 to + 0.365 m) and an eventual load m_L [kg].
The lowest eigenfrequency f_0 of the linear actuator : f_0 = 110 ... 134 [Hz]
The lowest eigenfrequency f_0 of the rotational module : f_0 = 18..20 [Hz]
The necessity to make an extended model with 11 degrees of freedom (rotation 6 and translation 5) is based on the idea to describe the reality as good as possible. The behaviour of the robot may so be predicted over the complete frequency - or time domain.
By reducing the complexity (less D.O.F.) of the simulation model a compromis is made between the accuracy and the duration. But there is another reason for model reduction, i.e. the realization of a controller via a control model.
By this reason the on-line controller is based on a simpler control model and so a balance is made between accuracy and time. In spite of the model reduction the derived controller should be robust enough with respect to higher order control phenomena like stability.
Model reduction is arbitrary and some methods are known e.g. Guyan. In general the lowest eigenfrequency has most influence on the system dynamics and this one should be present in the reduced model.
With the combination of the 5 D.O.F. as a simulation model and a controller based on the 3 D.O.F. control model rather good results have been obtained. This controller has been implemented also with good result in the R-T module.

3. MODEL REFERENCE ADAPTIVE CONTROL OF THE ROBOT.

For the R-T robot a non-adaptive and an adaptive control have been designed. In the non-adaptive case the adaptation algorithm is out of operation, so these two cases may be well compared.

Non-adaptive controllers require exact knowledge of the systemparameters and explicit use of the complex system dynamics. Uncertainties lead to a bad performance of the controller. In practice one has to deal with uncertainty in robotdynamics. So a number of parameters as moments of intertia, loads and armlength may vary, while non-linearities in the actuators may be unknown.
By applying feedback one may reduce the sensitivity for parameter variations, but this leads to higher gain factors, bigger control efforts and increases the possibility of instability.
In adaptive control the model parameters of the system are estimated on-line. Based on this estimation the control effort is determined. So adaptive control is very suitable for manipulators, with a complex system description with unknown and varying parameters.

In this chapter an adaptive controller is proposed, which is a combination of the computed torque method for the main control input and an adaptive PD controller acting on the deviation of the desired trajectory (Fig. 5).
The computed torque signal is derived directly from the equations of a control model.
An adaptation algorithm based on the Model Reference Adaptive Control (MRAC) method adapts the PD gain-factors on-line for the additional loop.
The complete controller is applied to a simulation model as well as the real RT-robot.

3.1 The non-adaptive controller.

As stated before this controller consists of two parts, a model dependent feedforward controller - also called the computed torque part - and a PD controller.

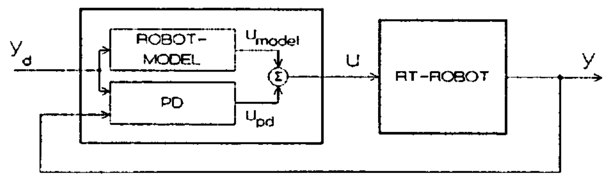

Fig. 4 The non adaptive controller

The computed torque controller computes by a control model the nominal control efforts - the torques - to perform a desired trajectory. Next the PD-controller acts on and compensates for the occuring trajectory error. The control model has to be a representative reduced model of the complete system. It may not be too complex, because otherwise, the computation time of this part of the control signal becomes too big. From the RT-configuration the 3 DOF - model (R2 T1) has been applied her.

So the control signal consists of: $\vec{u} = \vec{u}_{model} + \vec{u}_{PD}$ (1)

The control model has three degrees of freedom:

Rotation of the rotationmodule motor:	ϕ^r_m
Rotation of the turntable:	ϕ
Translation of the linear arm:	x

Then the non-linear differential equations of movement are obtained.

Rotation : $$M^R = J_1^R \ddot{\phi}_m^R + \left[\frac{b^R}{(i^R)^2} + b_1^R\right]\dot{\phi}_m^R - \frac{b^R}{i^R}\dot{\phi} + \frac{k^R}{(i^R)^2}\phi_m^R - \frac{k^R}{i^R}\phi \qquad (2)$$

$$0 = J_2^R \ddot{\phi} + \frac{\partial J_2^R}{\partial t}\dot{\phi} + \left[b^R + b_2^R\right]\dot{\phi} - \frac{b^R}{i^R}\dot{\phi}_m^R + k^R\phi - \frac{k^R}{i^R}\phi_m^R \qquad (3)$$

Translation: $$M^T = \frac{1}{i^T}\left[m_T\ddot{x} - \frac{1}{2}\frac{\partial J_2^R}{\partial x}\dot{\phi}^2 + b^T\dot{x}\right] \qquad (4)$$

Substituting the desired trajectory in equatation (2) to (4) delivers the nominal control torques, from which the nominal control voltages for the motors may be derived:

$$\vec{u}_{model} = \left[M^R(q_d, \dot{q}_d, \ddot{q}_d) \quad M^T(q_d, \dot{q}_d, \ddot{q}_d)\right]^T \tag{5}$$

The real trajectory is compared with the desired trajectory and the deviation and PD control effort is obtained:

$$\vec{e} = \vec{q}_d - \vec{q} \tag{6}$$

$$\vec{u}_{PD} = -K_d\vec{e} - K_p\dot{\vec{e}} \tag{7}$$

The assumption is made that deviations in the rotation or translation only lead to a control effort in that degree of freedom. This means that K_p en K_d are of the following structure:

$$K_p = \begin{bmatrix} K_4^R & K_1^R & 0 \\ 0 & 0 & K_1^T \end{bmatrix} \quad K_d = \begin{bmatrix} K_5^R & K_2^R & 0 \\ 0 & 0 & K_2^T \end{bmatrix} \tag{8}$$

The feedback gains are determined such that the total system is stable with poles in the left half of the s- plane.

3.2 The adaptive controller.

Adaptive control is a special kind of feedback, in which the states of a process are divided in two categories, characterized by the difference in speed. The signals related to the degrees of freedom are quickly changing states, while the modelparameters are slowly changing.
The fast control loop is the PD-controller with the modeldependent feedforward (computed torque) part.The systemparameters and subsequently the control parameters (modelparameters and feedback gains) are not constant, but they are updated in a slower control loop as an answer to the change in the dynamics of the process and to disturbances.

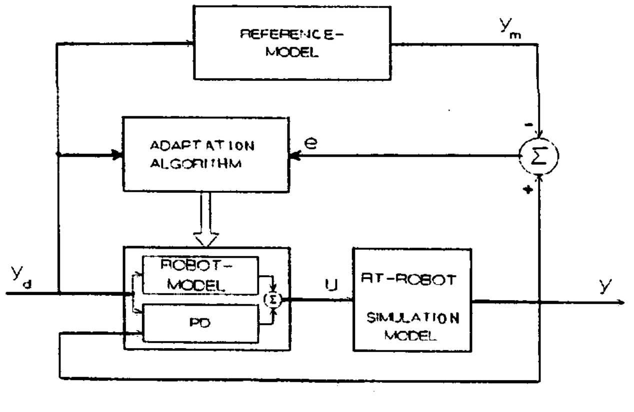

Fig. 5 The model reference adaptive control concept.

The adaptation of the control parameters is done here by the Model Reference Adaptive Control (MRAC) approach, given by Seraji [3].

In the slow control loop there is a reference model, which describes the desired trajectory in terms of the deviation. The controlparameters are determined such that the robot is forced to behave as the reference model. The adaptation mechanism estimates on line the controlmodel parameters and feedback gains by using the deviation and the reference model.

3.3 The adaptation algorithm.

Like in the previous case the control effort is divided in a computed torque-(control model) part and a PD-part: $\vec{u} = \vec{u}_{model} + \vec{u}_{PD}$
The modeldependent part may be written as:

$$M^R = A^R\ddot{\phi}_m^R + B_1^R\dot{\phi}_m^R + B_2^R\dot{\phi} + +C_1^R\phi_m^R + C_2^R\phi + F^R \tag{10}$$

$$M^T = A^T\ddot{z} + B_1^T\dot{\phi} + B_2^T\dot{z} + C_2^T z + F^T \tag{11}$$

and so the control effort is: $\vec{u}_{model} = A(t)\ddot{\vec{q}}(t) + B(t)\dot{\vec{q}}(t) + C(t)\vec{q}(t) + \vec{F}$ (12)

with:

$$A(t) = \begin{bmatrix} A^R & 0 & 0 \\ 0 & 0 & A^T \end{bmatrix} \quad B(t) = \begin{bmatrix} B_1^R & B_2^R & 0 \\ 0 & B_1^T & B_2^T \end{bmatrix} \tag{13}$$

$$C(t) = \begin{bmatrix} C_1^R & C_2^R & 0 \\ 0 & 0 & C^T \end{bmatrix} \quad F(t) = \begin{bmatrix} F^R \\ F^T \end{bmatrix}$$

The PD-part of the control effort is time dependent and may be written as:

$$\vec{u}_{PD}(t) = -K_d(t)\vec{e}(t) - K_p(t)\dot{\vec{e}}(t) \tag{14}$$

with:

$$K_p = \begin{bmatrix} K_1^R & K_2^R & 0 \\ 0 & 0 & K_1^T \end{bmatrix} \quad K_d = \begin{bmatrix} K_3^R & K_4^R & 0 \\ 0 & 0 & K_2^T \end{bmatrix} \tag{15}$$

Refering to (13) and (15) there are 17 controlparameters:

the control model: $A^R, A^T, B_1^R, B_2^R, B_1^T, B_2^T, C_1^R, C_2^R, C_2^T, F^R, F^T$
the feedback gains: $K_1^R, K_2^R, K_3^R, K_4^R, K_1^T, K_2^T$

A description of the real behaviour of the robot is done by a model.This is a non-linear robot equation with unknown parameters:

$$\vec{u}_{robot}(t) = A^*(\vec{q},\dot{\vec{q}})\ddot{\vec{q}}(t) + B^*(\vec{q},\dot{\vec{q}})\dot{\vec{q}}(t) + C^*(\vec{q},\dot{\vec{q}})\vec{q}(t) + \vec{F}^* \tag{16}$$

Here A^*, B^*, C^* and $\vec{F}^*$ are matrices, in which the elements are non-linear and unknown functions of the degrees of freedom.
The control effort is formed by the computed torque (model) part and the PD-feedback:

$$\vec{u}(t) = -K_p(t)\dot{\vec{e}}(t) - K_d(t)\vec{e}(t) + A(t)\ddot{\vec{q}}(t) + B(t)\dot{\vec{q}}(t) + C(t)\vec{q}(t) + \vec{F} \tag{17}$$

in which the deviation $\varepsilon(t)$ is defined in (6).

Combining (16) and (17) results in a differential equation for the deviation.

$$A^*\ddot{\vec{e}}(t)+(B^*+K_d)\dot{\vec{e}}(t)+(C^*+K_p)\vec{e}(t) =$$
$$= (\vec{F}^*-\vec{F})+(A^*-A)\ddot{\vec{q}}_d(t)+(B^*-B)\dot{\vec{q}}_d(t)+(C^*-C)\vec{q}_d(t) \tag{18}$$

The desired trajectory q_d and (F* - F) are at the right hand side of the equation (17). If the control parameters are constant, then the deviation will asymptotically not become zero, but depend on q_d and F. Therefore A,B,C and F have to be adapted such that the right hand side of (18) becomes zero.
The feedback gains K_p and K_d are also adapted to get stability of the closed loop at the desired performance.

If the position - velocity error is defined as $\vec{z}(t) = (\vec{e}(t), \dot{\vec{e}}(t))^T$
This transforms (18) into the adaptive system:

$$\dot{\vec{z}}(t) = \begin{pmatrix} 0 & I \\ -(A^*)^{-1}[C^* + K_p] & -(A^*)^{-1}[B^* + K_d] \end{pmatrix} \vec{z}(t) \tag{19}$$
$$+ \begin{pmatrix} 0 \\ (A^*)^{-1}[\vec{F}^* - \vec{F}] \end{pmatrix} + \begin{pmatrix} 0 \\ (A^*)^{-1}[C^* - C] \end{pmatrix} \vec{q}_d(t)$$
$$+ \begin{pmatrix} 0 \\ (A^*)^{-1}[B^* - B] \end{pmatrix} \dot{\vec{q}}_d(t) + \begin{pmatrix} 0 \\ (A^*)^{-1}[A^* - A] \end{pmatrix} \ddot{\vec{q}}_d(t)$$

Now the reference model has to be defined. It describes the desired trajectory in the error $\vec{e}(t)$ and it is assumed that the error of each DOF is decoupled and may be described as a second order differential equation.

$$\ddot{e}_i(t) + 2\xi_i\omega_i\dot{e}_i(t) + \omega_i^2 e_i(t) = \vec{0}\,; \qquad i = 1,2,3 \tag{20}$$

With ξ_i and ω_i the relative dampingsfactor and the undamped eigenfrequency.
This results into: $\ddot{\vec{e}}_m(t) + D_2\dot{\vec{e}}_m(t) + D_1\vec{e}_m(t) = \vec{0}$ (21)

With D_1 = diag(ω_i^2) and D_2 = diag $(2\xi_i\omega_i)$ as constant 3 x 3 diagonal matrices.

With $\vec{z}_m(t) = (\vec{e}_m(t), \dot{\vec{e}}_m(t))^T$ - the vector of desired model position - velocity errors - (21) may be written as:

$$\dot{\vec{z}}_m(t) = \begin{pmatrix} 0 & I \\ -D_1 & -D_2 \end{pmatrix} \vec{z}_m(t) \tag{22}$$

The reference model is stable, so there exist a symmetric positive definite 6 x 6 matrix P, which obeys the Lyapunov equation (23)

$$P = \begin{pmatrix} P_1 & P_2^T \\ P_2 & P_3 \end{pmatrix} \qquad PD + D^TP = -Q \tag{23}$$

in which D is the 6 x 6 system- and Q is a symmetric constant 6 x 6 matrix.
From this the adaptation algorithms are derived so that for a trajectory the state of the adaptive system converges to the referencemodel. The unknown robotparameters A^*, B^*, C^* and F^* are slowly time dependent compared with the adaptation.
For $\vec{R}(t) = P_2\vec{e}(t) + P_3\dot{\vec{e}}(t)$ and with the matrices $\delta,\alpha,\alpha_1,\beta,\gamma,\lambda$ representing the positive adaptationgains - defined by the designer - the controlparameters may be calculated from:

$$\vec{F}(t) = \vec{F}(0) + \delta\int_0^t \vec{R}(t)dt \qquad C(t) = C(0) + \beta\int_0^t \vec{R}(t)\vec{q}_d^T(t)dt$$

$$K_p(t) = K_p(0) + \alpha\int_0^t \vec{R}(t)\vec{e}^T(t)dt \qquad B(t) = B(0) + \gamma\int_0^t \vec{R}(t)\dot{\vec{q}}_d^T(t)dt$$

$$K_v(t) = K_v(0) + \alpha_1\int_0^t \vec{R}(t)\dot{\vec{e}}^T(t)dt \qquad A(t) = A(0) + \lambda\int_0^t \vec{R}(t)\ddot{\vec{q}}_d^T(t)dt \tag{24}$$

So summarizing the main properties of the adaptive control concept are:

1. Two control loops, a fast loop for the degrees of freedom and a slow loop to adapt the control parameters.
2. The controlparameters are adapted on-line.
3. Feedback takes place from the performance of the fast loop.

4. MEASUREMENT RESULTS

4.1 Simulations.

The simulations have been performed with the package PC-Matlab. It has a number of standard routines a.o. the Runge-Kutta difference routine to calculate the response of the robot via a simulation model. For this the model with 5 DOF (R_3T_2) has been chosen. With the control model 3 DOF (R2T1) the computed torque part of the input signal on the desired trajectory is calculated. The desired trajectory is a skew sine wave in both directions shown in Fig.6.

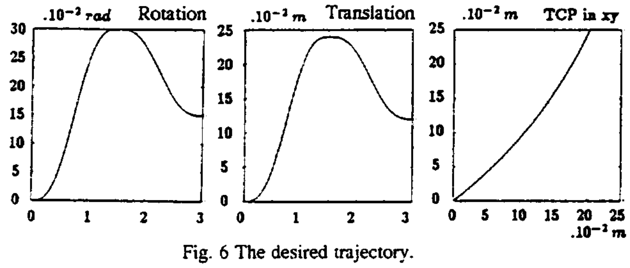

Fig. 6 The desired trajectory.

The minimal sample-time is 7 ms, derived from the implementation and also applied in the simulations.
In fig. 7 the results of the non-adaptive controller are shown with and without the computed torque part (feed forward control of the desired trajectory).

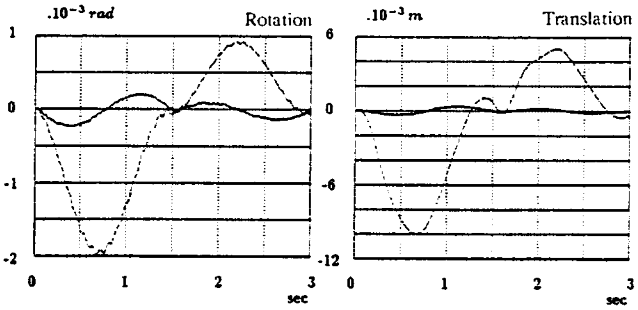

Fig.7 Position errors with the non-adaptive controller
(——— : with feedforward)

It is clear that the use of the feedforward control improves the control performance considerably.
If an extra load is applied to the robot an additional uncertainty is introduced to the control law. In fig. 8 the position-errors are shown with a load of 50 kg and a load of 0 kg for the non-adaptive controller.

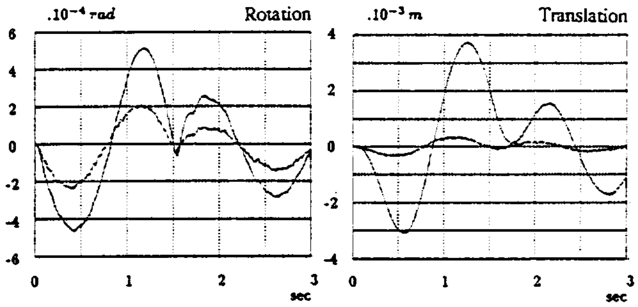

Fig.8 Positions errors-non adaptive controller-different loads.
(- - - - : 0 kg , —— : 50 kg)

The control performance (Fig.8) becomes less but no instabilities occur. So the controller is rather robust.

A comparison between the performances of the adaptive and the non-adaptive controllers (with a load of 0 kg) is shown in Fig. 9.

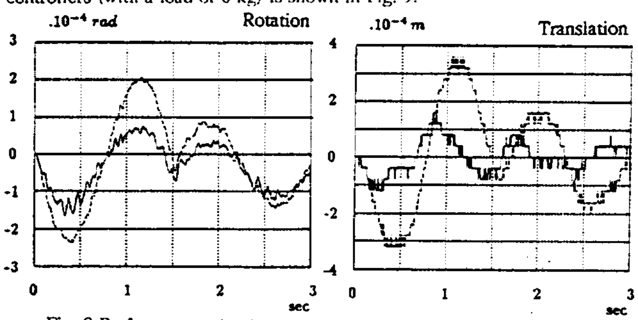

Fig. 9 Performance adaptive controller - vs - non adaptive controller
(- - - - - - - : non-adaptive)

The initial conditions of the control parameters are at the start of the trajectory the same for both the adaptive as the non-adaptive controller. The adaptation mechanism adapts the control parameters during the trajectory. The adaptive controller performs better than the non-adaptive controller.
The performance of the adaptive controller on different loads is shown in Fig. 10.

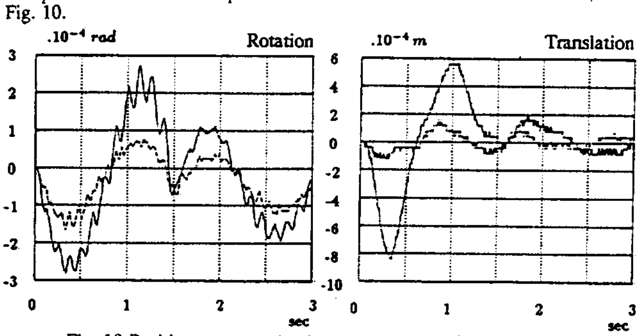

Fig. 10 Position errors - adaptive controller - different loads.
(- - - - : 0 kg, —— : 50 kg)

The position - errors (see Fig. 10 and Fig.8) are reduced by a factor 2 for rotation and by a factor 6 for translation compared with the non-adaptive controller.

4.2 Robustness and Adaptation speed.

If the controlmodel does not fit well to the robot behaviour or if the feedback gains are not chosen properly, then the response of the real robot may become unstable.

The adaptive controller however will try to stabilize this effect.
This is called the robustness of the adaptive controller, where the adaptation mechanism is able to stabilize an initial unstable controller.
Also the adaptation mechanism restricts the feedback gains to become negative.
In the case that the controlparameters are updated only every 20 samples a difference in the realised errors could hardly be noticed.

4.3 Implementation on the RT robot.

With the real R-T robot the same experiments have been performed as described in the simulations. It may be concluded that the use of the computed torque part improves the performance considerably.

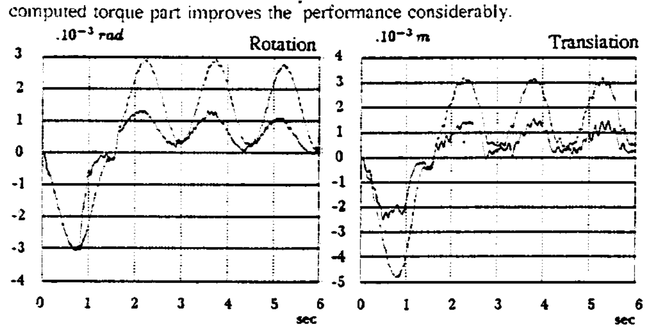

Fig. 11 Position errors - adaptive and non-adaptive controller - RT robot.
(- - - - - non-adaptive)

In Fig. 11 a comparison is made between the adaptive and the non-adaptive controller. The adaptive controller needs a sufficient long trajectory to estimate the controlparameters well. So the nominal trajectory here consists of four skew sine waves. The RT-robot is rather stiff, so small variations in the load are easily compensated by the PD-controller. With a load of 20 kg the adaptive controller tends to perform better.
If the controlmodel is chosen such that the parameter values are 30% lower than the real RT-configuration, then the adaptation mechnism updates the controlparameters such that the controlperformance rather quickly becomes better as shown in Fig. 11, which means a good robustness.
The experiments described above have been done with maximum adaptation speed. If this speed is reduced by a factor twenty, there is nearly no difference in the position error.

CONCLUSIONS

The application of feedforward (computed torque) control derived via a control model, calculated from the desired trajectory as a nominal control effort improves the control performance considerably.
The non-adaptive controller is sensitive to load variations, so a load of 50 kg makes the control performance worse.

The adaptive controller is preferable if the robotdynamics are poorly known. In that case a non-adaptive controller will give a bad control performance and possibly lead to instability.
The adaptation mechanism estimates the best control parameters and is an improvement compared to the non-adaptive controller.
The adaptive controller is also rather robuust.An initial deviation of the parametervalues of the control model with 30% causes the adaptation mechanism to update the controlparameters quickly and results again in a good control performance.

LITERATURE

[1] Mulders P.C., Jansen J., Pijls J.M.L.,
Optimal trajectory control of a linear robotarm by a state space method
CIRP-Annals 1989, vol 38/1/1989 pp. 359-364

[2] Seraji H.
Design of adaptive joint controllers for robots.
Recent Trends in Robotics pp. 251-260, Elsevier,Amsterdam,1986

[3] Seraji H.
Adaptive Control of robot manipulators.
Proc.IEEE Conf. on Robotics and Automation 565, San Francisco, 1986

[4] Voorkamp R.J.
Study of the adaptive trajectory control of a RT-robot
WPA-1076, MS-thesis . Technical University Eindhoven, 1991

[5] Kreffer G.J.
Design of a rotational module and a study of adaptive trajectory control of a RT-robot.
WPA 0575, Ms thesis, Technical University Eindhoven, 1988.

[6] Asada H., Slotine J.J.E.
Robot analysis and control, ISBN 0-471-83029-1
John Wiley and Sons, New York, 1985

[7] Craig J. Hsu P., Sastry S.,
Adaptive control of mechanical manipulators, pp 190-195
IEEE Conference on Robotics and Automation, San Francisco, 1986.

[8] Landau I.D.
Adaptive control techniques for robot manipulators
I.F.A.C. Symposium Robot Control, Barcelona, 1985

[9] Slotine J.J.E., Li W.,
On the adaptive control of robot manipulators.
ASME Winter Annual Meeting, Anaheim, 1986.