Model reference adaptive control of a modular robot

Model reference adaptive control of a modular robot

P C MULDERS, A P M A MARTENS, J JANSEN
Technical University Eindhoven, The Netherlands

SYNOPSIS. Flexible manipulators often have place - and time dependent parameters, varying during trajectory performance. Adaptive control is a process of modifying one or more parameters of the controller of a manipulator in action and so specially important for these robots.
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 - consisiting of a linear and a rotary actuator showing these parameter variations.

Necessary models - extended and reduced - of this RT 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.

1. INTRODUCTION

This work is a study of (optimal) adaptive control algorithms on systems with place- and time dependent parameters - varying during trajectory performance - with an implementation on mechanical manipulators of industrial scale.
These advanced control systems are tested on a modular robot system - for loads up to 50 kg, consisting of a linear and a rotary actuator - as shown in Fig. 1.
A 3D-force sensor is mounted on the linear arm to perform trajectory teach operations. After that the replay of the desired trajectory - eventually with varying parameters - is done, in which the known control signals are updated by the adaptive control algorithm.

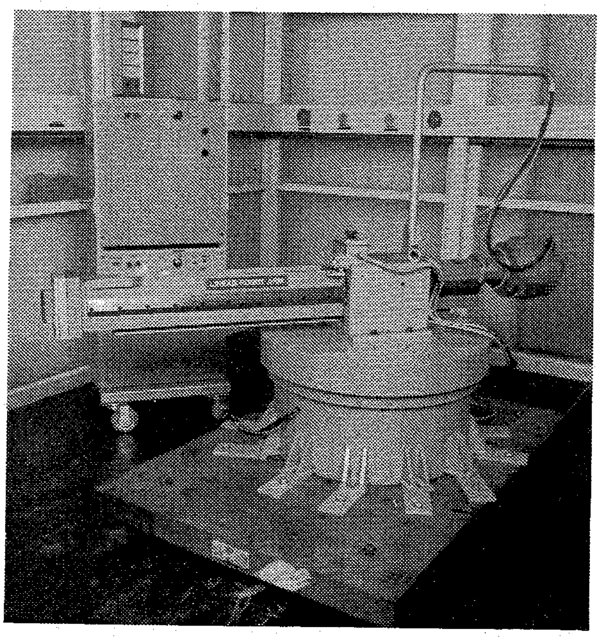

Fig 1 The modular robot system

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 to avoid deformations and vibrations.
For higher operating speeds robots should be light weight constructions to enable the robots to respond faster.
Hence, more accurate dynamic models should be taken into account to pursue better dynamic performance.

Now a number of (optimal adaptive) trajectory control strategies may be mentioned here:

- the PID method
- the optimal controller (Riccati equation) method
- the model reference adaptive control (MRAC)method

The PID controller uses the deviation from the desired trajectory to correct and is often used as a standard to compare with other controllers. However for coupled systems this type leads often to instability.
The linear optimal controller is based on the minimization of a performance criterion function with contributions of e.g. the deviations and the control signals with certain weighing matrices.
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 so the control law is designed explicitely on the basis of a model. If flexibilities play a role, it often results in an unstable system behaviour.

So the aim is to search for a control law achieving both trajectory tracking and a stabilization of acceptable vibrations.
Adaptive control is the 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 these adaptive methods the model reference adaptive control method (MRAC) is important.
So the adaptive controller described here is a combination of the computed torque method as feed forward control, and a PD feedback controller acting on the deviation, while this PD controller is updated by the MRAC algorithm.
This leads to relatively easy to implement systems, with a high speed of adaptation and may be used in a variety of

applications.
From the modular RT robot system an extended model - for simulation purposes - has been made, while a reduced model is applied to calculate the computed torque signals. The described method has been performed both in computer simulations and reality to draw conclusions with respect to convergency, stability and robustness.

2. THE ROTATION - TRANSLATION ROBOT

2.1 The construction of the linear robotarm.

The linear arm consists of a hollow frame with a preloaded spindle. The rotation of the DC motor is converted into a translation of the arm with a ballscrew nut on the spindle. For the position - and velocity measurements there are a linear - and rotational encoder and a tachogenerator. The 3D-force sensor has strain gauges and is used in the teach-mode.

velocity	: 1 m/s
acceleration	: 1 m/s^2
load	: 50 kg
stroke	: 1 m
accuracy	: 0,01 mm
position measuring	: Heidenhain LS513
motor	: Axem MC19PR26, 1 kW
control system	: PID-or state controller.

Table 1. Linear robotarm specifications.

2.2 The construction of the rotational module.

The mechanical construction is based on a cylinder with side ribs to minimize the deformation. The transmission from the motor to the turntable consists of a four stage preloaded toothed wheel combination. Coupled to the DC motor there is a tachogenerator and a rotational encoder. Direct position measurement of the turntable occurs by a bended optical digital incremental encoder.

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

Table 2. Rotational module specifications.

2.3 The hierarchical controller.

The controller (Fig.2) consists of 4 Intel SBC's and 1 RAM board.

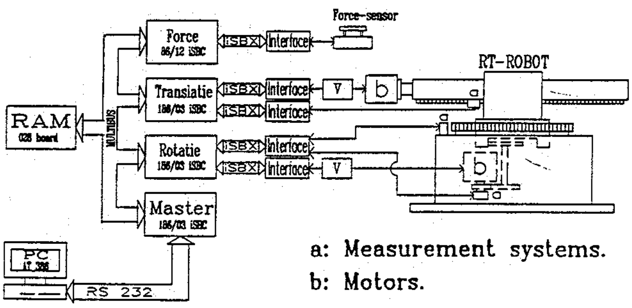

Fig 2 The hierarchical controller

The task of each slave SBC 186/03 is:
to calculate - according to the control algorithm - the motor voltages

- to read the position of each module
- to store these data - motor voltages and positions - into the RAM.

The task of the master SBC 186/03 is:

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

The PC 80386 may calculate the optimal control law and the nominal trajectory off-line and diagram the data.

2.4 Modelling of the RT-robot.

Although the robot is a system with divided parameters an attempt is made to realize a lumped mass model. This approach is based on previous studies [1] about drives of motor-tacho-spindle-carriage combinations. The extended model has 11 degrees of freedom (DOF):

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

By the coupling of the modules for rotation and translation also the eigenfrequencies may vary. So the lowest eigenfrequency of the rotational module is fo ≈ 18...20[Hz] and of the linear actuator is fo ≈ 110 ... 134 [Hz].

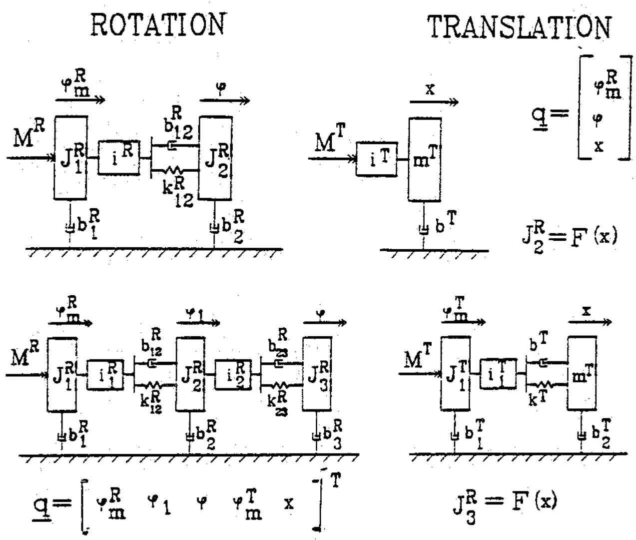

Fig 3 Reduced models of the RT-robot

The complexity of the simulation model may be reduced as a compromise between accuracy and duration of simulation calculations. The lowest eigenfrequency should be present in the reduced model. But there is another reason for model reduction i.e. the realization of a controller (e.g computed torque) via the control model.
With the combination of a simulation model (5 DOF) and a controller based on the control model (3 DOF) rather good results have been obtained. This controller has been implemented with good result in the RT-module.

3. MODEL REFERENCE ADAPTIVE CONTROL

For the RT-robot a non adaptive and an adaptive control have been designed. (Fig. 4). In the non-adaptive case the adaptation algorithm is out of operation, so these may be well compared.

Non-adaptive controllers require exact knowledge of the systemparameters and explicit use of the complex system dynamics. In practice one has to deal with uncertainties - so a number of parameters as moments of inertia, loads and armlength may vary, while non- linearities may be unknown-leading to a bad performance of the controller.
The application of feedback may reduce the sensitivity for parameter variations, but this leads to higher gain factors, bigger control efforts and a possible 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.

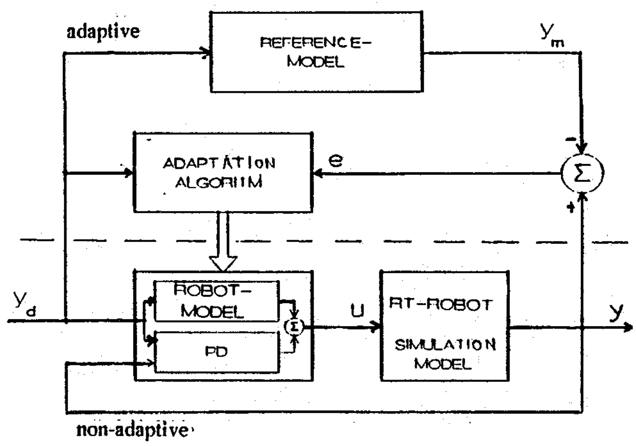

Fig 4 The non-adaptive and adaptive control concept

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.
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.
The complete controller is applied to a simulation model as well as the real RT-robot.

3.1 The non-adaptive controller.

The nominal control efforts - the torques - to perform a desired trajectory are computed by a control model. This has to be a representative reduced model and not too complex, otherwise the computation time of this part of the control signal becomes too big. The 3 DOF-model (R2T1) has been applied here.
Next the PD-controller acts on and compensates for the realized trajectory error.

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

The control model has three degrees of freedom:

Rotation of the rotation motor	:	ϕ_m^r
Rotation of the turntable	:	ϕ
Translation of the linear arm	:	X

The non-linear equations of movement are:

Rotation:

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

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

Translation:

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

The desired trajectory - with estimates of the other degrees of freedom - is substituted in equations (2) to (4) to calculate the nominal control torques. So this part of the input signal is:

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

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

$$\vec{u}_{PD} = -K_d\vec{e} - K_p\dot{\vec{e}} \quad (6)$$

K_p and K_d are of the following structure according to the assumption that deviations in the rotation or translation only lead to a control effort in that degree of freedom.

$$K_p = \begin{bmatrix} K_4^R & K_1^R & 0 \\ 0 & 0 & K_1^T \end{bmatrix} \quad K_d = \begin{bmatrix} K_5^R & K_2^R & 0 \\ 0 & 0 & K_2^T \end{bmatrix} \quad (7)$$

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 kind of feedback, in which the states of a process are divided in two categories, characterized by the difference in speed. It is assumed that the model parameters are slowly changing, while the degrees of freedom are quickly changing states. (Fig. 4).
The fast control loop is the computed torque part and the PD-controller. 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 dynamics of the process and to disturbances.
In the slow control loop there is a reference model which describes the desired trajectory in terms of the deviation.
The control parameters are determined such that the robot is forced to behave as the reference model. The adaption mechanism estimates on line the control model parameters and feedback gains by using the deviation and the reference model.

3.3 The adaptation algorithm.

The adaptation of the control parameters is done here by the model reference Adaptive Control approach given by Seraji [2].
The control effort consists again of a computed torque - part (control model) and a PD - part as given in (1).
The model feedforward part may be written as:

$$M^R = A^R\ddot{\phi}_m^R + B_1^R\dot{\phi}_m^R + B_2^R\dot{\phi} + +C_1^R\phi_m^R + C_2^R\phi + F^R \quad (8)$$

$$M^T = A^T\ddot{x} + B_1^T\dot{\phi} + B_2^T\dot{x} + C_2^T x + F^T \quad (9)$$

so: $\vec{u}_{model} = A(t)\ddot{\vec{q}}(t) + B(t)\dot{\vec{q}}(t) + C(t)\vec{q}(t) + \vec{F}$ (10)

with:

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

$$C(t) = \begin{bmatrix} C_1^R & C_2^R & 0 \\ 0 & 0 & C^T \end{bmatrix} \quad F(t) = \begin{bmatrix} F^R \\ F^T \end{bmatrix} \quad (11)$$

The PD-part is: $\vec{u}_{PD}(t) = -K_d(t)\vec{e}(t) - K_p(t)\dot{\vec{e}}(t)$ (12)

with:

$$K_p = \begin{bmatrix} K_1^R & K_2^R & 0 \\ 0 & 0 & K_1^T \end{bmatrix} \quad K_d = \begin{bmatrix} K_3^R & K_4^R & 0 \\ 0 & 0 & K_2^T \end{bmatrix} \quad (13)$$

This results in totally in 17 controlparameters i.e. 11 in the model-part (11) and 6 in the PD-part (13).
The total control effort, as the sum of (10) and (12) becomes:

$$\vec{u}(t) = -K_p(t)\vec{e}(t) - K_d(t)\dot{\vec{e}}(t) + \\ + A(t)\ddot{\vec{q}}(t) + B(t)\dot{\vec{q}}(t) + C(t)\vec{q}(t) + \vec{F} \qquad (14)$$

The behaviour of the robot is described by a non-linear equation with unknown parameters (A^*,B^*,C^*, and $\vec{F}^*$):

$$\vec{u}_{robot}(t) = A^*(\vec{q},\dot{\vec{q}})\ddot{\vec{q}}(t) + B^*(\vec{q},\dot{\vec{q}})\dot{\vec{q}}(t) + C^*(\vec{q},\dot{\vec{q}})\vec{q}(t) + \vec{F}^* \qquad (15)$$

Combining (14) and (15) results in:

$$A^*\ddot{\vec{e}}(t)+(B^*+K_d)\dot{\vec{e}}(t)+(C^*+K_p)\vec{e}(t) = \\ (\vec{F}^*-\vec{F})+(A^*-A)\ddot{\vec{q}}_d(t)+(B^*-B)\dot{\vec{q}}_d(t)+(C^*-C)\vec{q}_d(t) \qquad (16)$$

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 (16) becomes zero.

The feedback gains K_p and K_d are also adapted to get stability of the closed loop at the desired performance.

After defining the position - velocity error $\bar{z}(t)$ eq. (16) is transformed into an adaptive system.

In the reference model the desired trajectory is described in the error $\bar{\varepsilon}(t)$ and it is assumed that the error of each DOF is decoupled and described as a second order differential equation.

$$\vec{z}\,(t) = (\vec{\varepsilon}(t), \dot{\vec{\varepsilon}}\,(t))^T \qquad (17)$$

$$\ddot{\vec{e}}_i(t) + 2\xi_i\omega_i\dot{\vec{e}}_i(t) + \omega_i^2\vec{e}_i(t) = \vec{0}\,; \qquad i = 1,2,3 \quad (18)$$

With ξ_i the relative dampingsfactor and ω_i the undamped eigenfrequency and D_1 = diag (ω_i^2) and D_2 = diag $(2\xi\omega_i)$ as constant 3 x 3 diagonal matrices it follows:

$$\ddot{\vec{e}}_m(t) + D_2\dot{\vec{e}}_m(t) + D_1\vec{e}_m(t) = \vec{0} \qquad (19)$$

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

The reference model is stable, so there exist a symmetric positive definite 6 x 6 matrix P, wich obeys the Lyapunov equation in which D is a 6 x 6 system - and Q is a symmetric constant 6 x 6 matrix.

$$P = \begin{pmatrix} P_1 & P_2^T \\ P_2 & P_3 \end{pmatrix} \qquad PD + D^TP = -Q \qquad (20)$$

From this the adaptation algorithms are derived so that for a trajectory the state of the adaptive system converges t the reference model. The unknown robotparameters A* B*, C* and F* are slowly time dependent compared with the adaptation.

For the control - and systemparameters follows e.g.

with $R(t) = P_2\,\bar{e}(t) + P_3\,\dot{\bar{e}}(t)$ (21)

$$\begin{aligned} \dot{\vec{F}}(t) &= \delta\vec{R}(t) \\ \dot{K}_p(t) &= \alpha\vec{R}(t)\vec{e}^T(t) \\ \dot{K}_v(t) &= \alpha_1\vec{R}(t)\dot{\vec{e}}^T(t) \\ \dot{C}(t) &= \beta\vec{R}(t)\vec{q}_d^T(t) \\ \dot{B}(t) &= \gamma\vec{R}(t)\dot{\vec{q}}_d^T(t) \\ \dot{A}(t) &= \lambda\vec{R}(t)\ddot{\vec{q}}_d^T(t) \end{aligned}$$

So $\quad \vec{F}(t) = \vec{F}(0) + \delta\int_0^t \vec{R}(t)dt \quad$ etc.

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 control parameters are adapted on-line.
3. Feedback takes place from the performance of the fast loop.

4. RESULTS

4.1 Simulations.

The simulations have been performed with the package PC-Matlab.The simulation-model of the robot has 5 DOF (R_3T_2), while the control model for the computed torque part has 3 DOF(R_2T_1). The desired trajectory is a skew sine wave in both φ and x

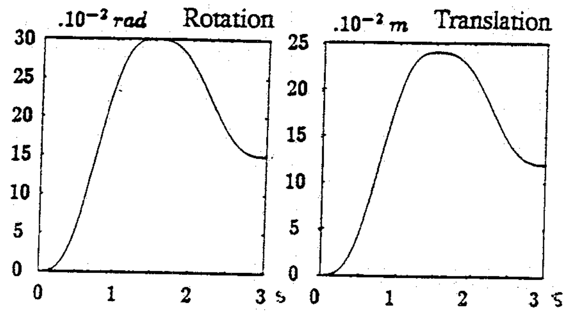

Fig 5 The desired trajectory

The minimal sample time is 7 ms, applied in the simulations and the implementation.

In Fig. 6 the results of the non-adaptive controller are shown with and without the computed torque part (feed forward).

Feed forward control improves the control performance considerable.

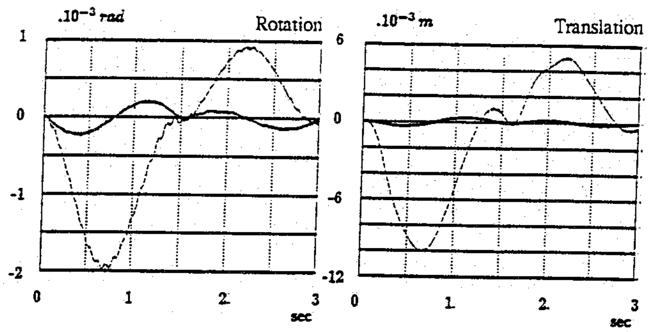

Fig 6 Position errors with the non-adaptive controller (____ with feedforward)

A comparison between the performances of the adaptive - and the non-adaptive controller (with a load of 0 kg) is

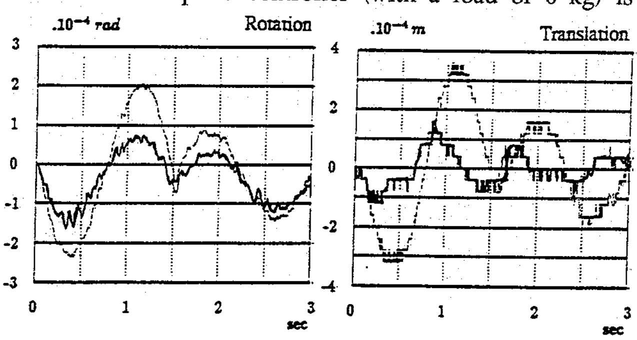

Fig 7 Performance adaptive vs non adaptive controller (........ non adaptive)

shown in Fig. 7. The initial conditions of the control parameters are at the start of the trajectory the same for both the adaptive and the non-adaptive controller. The adaptative controller performs better than the non-adaptive controller because of the adaptation mechanism.

The performance of the adaptive controller on different loads is shown in Fig. 8. The position errors are reduced by a factor 2 for rotation and 6 for translation compared with the non-adaptive controller.

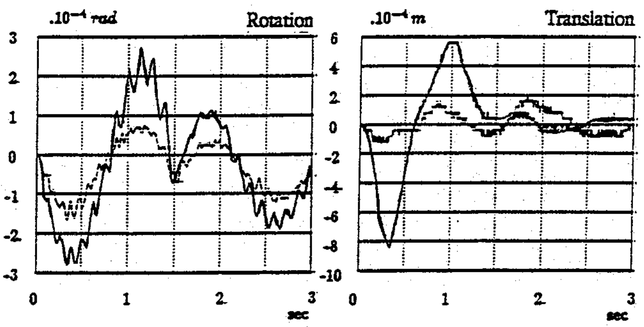

Fig 8 Position errors-adaptive controller-different loads (........ 0 kg; ____ 50 kg)

4.2 Implementation on the RT-robot.

The same experiments as in the simulations have been performed with the real RT-robot. The adaptive controller needs a sufficient long trajectory to estimate the control parameters well, so here the trajectory consists of four skew sine waves.
In Fig. 9. a comparison is made between the adaptive and the non-adaptive controller.
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 control model is chosen such that the parameter values are 30% lower than the real RT-configuration, then the adaptation mechanism updates the control parameters such that the control performance rather quickly becomes better, as shown in Fig. 9, which means a good robustness.

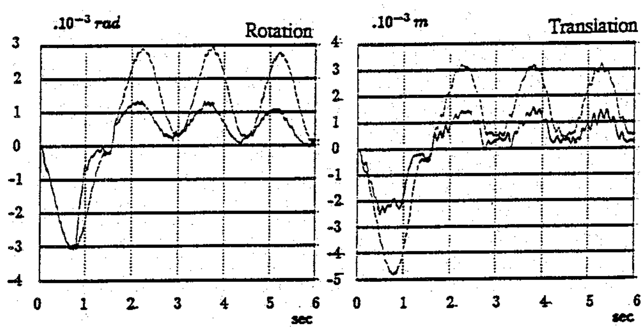

Fig 9 Position errors-adaptive and non-adaptive controller RT robot (........ non adaptive)

4.3 Robustness and Adaptation Speed.

In the case of different loads the control performance becomes less but no instabilities occur. Also 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.
The adaptation mechanism is able to stabilize an initial unstable controller. It also restricts the feedback gains to become negative.
If the controlparameters are updated only every 20 samples then the occuring errors are hardly different from the fast updating situation.

CONCLUSIONS

The application of feedforward (computed torque) control derived via a control model from the desired trajectory improves the control performance considerably.
The experiments have been done with maximum adaptation speed. Speed reduction by a factor twenty, gives no significant difference in the position error.
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 the 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 robust. An initial deviation of parameter values 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. Oosterling J.A.J. Wolf A.C. H. van der. A model study of a feeddrive for a numerical controlled lathe.
CIRP-Annals 1982, Vol 31/1, pp 293-298

(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,
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) 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 pp 359-364.

(6) Kreffer G.J. Design of a rotational module and a study of adaptive trajectory control of the RT-robot.
WPA 0575 MS thesis
Technical University Eindhoven. 1988.

(7) Bax W.H.M Study of the dynamic model of a RT-robot WPA 0787 MS thesis
Technical University Eindhoven. 1989.

(8) Martens A.P.M.A Comparative study and implementation of a trajectory control of a RT-robot.
WPA 0955 MS thesis
Technical University Eindhoven. 1990.