2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007



# Indirect Adaptive learning of Acceleration feedback control for Chained Multiple Mass-Spring-Damper Systems

J.-Y. Dieulot, F. Colas, P.-J. Barre, P. Borne, *Fellow, IEEE*

***Abstract*—An indirect Iterative Learning algorithm has shown to be able to update the parameters of an acceleration feedback controller for flexible manipulators. The fine estimation of the masses of a chain of mass-spring-dampers units allows the simultaneous tuning of both the feedback controller and the trajectory generation. This algorithm has been validated on an industrial robot arm.**

***Index Terms - Iterative Learning Control, Flexible Arms, Path Planning for Manipulators, Calibration and Identification.***

## I. Introduction

CONTOUR following and the cancellation of end-point vibrations are two of the major objectives that motivate the control of lightweight manipulators. Accurate trajectory following is quite difficult to perform because of the complex dynamics (nonlinearities, coupling between axes) and of the lack, in practice, of the measurement of joint position and speed.

The introduction of extra joint acceleration sensors allows to derive simple and robust control algorithms which can decouple the joint dynamics from most nonlinearities [1-8]. In particular, the acceleration feedback controller of Luo and Saridis [1,2] only requires the inversion of the inertia matrix. In practice, fine tuning is not so easy since obtaining angular accelerations and torque measurements require further computations or approximations [3]. It has also been shown experimentally that an increase of the sampling frequency or the feedback gains leads to instability. Nevertheless, when flexible robots with prismatic joints are considered, the inertia matrix becomes diagonal, accelerations measurements are quite easy to perform, which results in a simplified algorithm which has proven to provide good experimental results [8]. A simplified acceleration feedback algorithm can be derived when the model consists of Chained multiple mass-spring-damper (CMMSD) units which may represent the dynamics of an Industrial Cartesian robot arm. This model is derived from finite element methods (FEM) or Assumed modes methods (AMM) (see e.g. [9]), for which the modal parameters (modal masses and stiffness) vary with the position of the end-effector within the working space.

The calibration of these position-dependent modal masses is essential to tune the acceleration feedback algorithm. As, in practical applications, the manipulators often perform repetitive tasks, the introduction of Iterative Learning Control is rational. Indeed ILC uses information from previous executions of the task in order to improve performances from trial to trial and to reduce sequentially the tracking error irrespective of model non-linearities (see e.g. [10-12] and references therein). Such direct learning algorithms bring a blind feedforward compensation which allows to track iteratively the trajectory.

There exists a few identification-based indirect ILC schemes are based on a least-square estimate of the model which allows the computation of the new controller parameters [13]. In [14,15,16], a general algorithm is given to derive the control of a robot with unknown parameters using ILC both for control and parameter update.

This paper presents an indirect ILC algorithm for CMMSD systems controlled by acceleration feedback, which main advantage is to capture, after a few iterations, the real value of the position-dependent modal masses (the inertia matrix ) – which are exactly the key parameters needed for the acceleration feedback controller, using only simple algebraic calculations. As it will be demonstrated, it has the advantage over learning-based identification coupled with ILC, to update only the model parameters which are needed to tune the controller (which is thus different from an approach such as in [16]). Moreover, when modal masses are varying more quickly than the global stiffness, it will be shown that the control algorithm allows both to update the servo controller, which aims at improving the tracking accuracy, and to modify the trajectory planning, which helps to reduce vibrations. This hybrid control scheme which combines ILC and acceleration feedback has been tackled [17], only for direct ILC algorithms.

In a first section, the CMMSD model will be presented. Secondly, an acceleration feedback of these systems will be derived which guaranties the exact tracking of the first mass irrespective of spring nonlinearities and of disturbances. The Iterative tuning of the acceleration feedback is presented next section. A general interpolation algorithm which helps to tune the servo parameters within the workspace and to update the value of the jerk is presented. Finally, the results are applied to an industrial pick-and-place robot.

## II. Acceleration Feedback control of lumped mass-spring models

Consider a model consisting of $N$ chained mass-spring-dampers presented in figure 1 (where the damping values are

Manuscript received September 12, 2006. J.-Y Dieulot, F. Colas, P.-J. Barre are with Technological Research Team, ENSAM, 8 bd Louis XIV, 59000 Lille, France, P. Borne is with LAGIS, UMR 8146, Ecole Centrale de Lille, BP 48, 59651 Villeneuve d'Ascq Cedex, France. (e-mail: barre@lille.ensam.fr).

zero), which can be put under the normalized form

$$M(Q)\ddot{x} = K(Q)x + A(Q)\dot{x} + Bu + \eta \, . \quad (1)$$

$x = (x_1 \cdots x_N)^T$ represents the position of the springs $M(Q) = diag(m_1 \cdots m_N)$ is the matrix of modal masses, $K(Q) = \begin{pmatrix} -k_1 & k_2 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & k_{N-1} & -k_N \end{pmatrix}$, where $k_i$ is the $i^{th}$ modal stiffness $A(Q) = \begin{pmatrix} -a_1 & a_2 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & a_{N-1} & -a_N \end{pmatrix}$, where $a_i$ is the $i^{th}$ modal damping, $B = (1 \quad 0 \quad \cdots \quad 0)^T$, $\eta = (\eta_1 \cdots \eta_N)^T$ is a vector representing random measurement noises, $Q(t, x_1 \cdots x_N)$ is a set of external variables.

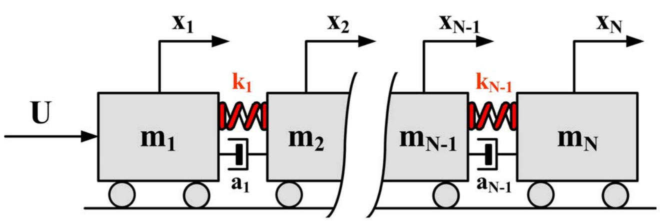


Fig. 1 Chained Mass-Springs-Dampers

When applied to a flexible axis, $x_1$ accounts for the modal mass relative to the motor position and $x_2 \cdots x_{N-1}$ the position of the successive modal masses. The control problem consists of tracking a prescribed trajectory for the motor position $x_1$ by monitoring the motor force $u$, without any knowledge on the position and speed of the modal masses $m_i$ but that of the motor, which is the pratical industrial configuration for Cartesian robots. The end-effector position $x_N$ should thus be controlled in open-loop.

An alternative to the design of an estimator of the positions $x_2 \cdots x_N$ consists of monitoring the motor position with the help of acceleration measurements $\ddot{x}_2 \cdots \ddot{x}_N$ (provided by accelerometers) of the modal masses. Note that, the acceleration signal being very noisy, it is unexpectable to extrapolate the velocities or position signals. In the remainder, $x_1^d$ is the reference trajectory for the position $x_1$ of the mass $m_1$, $\tilde{x}_1 = x_1^d - x_1$ the corresponding tracking error.

*Theorem 1*

Consider a CMMSD with $N$ units (1), $x_1^d$ a reference trajectory, $\tilde{x}_1 = x_1^d - x_1$ the tracking error.

The controller (2)

$$u = \sum_{i=2}^{N} m_i \ddot{x}_i + m_1 \ddot{x}_1^d + k_1 \dot{\tilde{x}}_1 + k_2 \tilde{x}_1 \, . \quad (2)$$

guaranties that $\tilde{x}_1$ converges exponentially to zero if the polynomial $m_1 s^2 + k_1 s + k_2$ is Hurwitz; $k_1, k_2$ are user-defined parameters.

*Proof*

Summing the lines of equation (1) yields $\sum_{i=1}^{N} m_i \ddot{x}_i = u + \sum_{i=1}^{N} \eta_i$. This equation, in this particular context, is independent of the spring stiffness and dampings. Replacing the control $u$ by the expression in (2) yields

$m_1 \ddot{\tilde{x}}_1 + k_1 \dot{\tilde{x}}_1 + k_2 \tilde{x}_1 = \sum_{i=1}^{N} \eta_i$ which proves the theorem.

$k_1, k_2$ should thus be chosen such that the cut-off frequency of the filter $m_1 s^2 + k_1 s + k_2$ is smaller than the frequency of the measurement noise or that of neglected higher order modes. This choice of parameters $k_1, k_2$ has been discussed previously [1]. This control algorithm consists simply of a multiple acceleration feedback coupled to a PD algorithm, which is very easy to implement in an industrial numerical device and has shown to be quite robust when applied to Cartesian robots.

## III. Iterative Tuning of the Modal Masses

It has been shown in the previous paragraphs that the only important parameters of the acceleration feedback algorithm consists of the modal masses $m_i(Q)$. In practice, in a flexible structure, the modal masses and stiffness depend on time and on the position of the manipulator within the working space, and are not easy to obtain accurately from a physical model or identification experiments. When the trajectory is repeated, it is possible to update the controller parameters, for any chosen time $t$, in the following way:

*Theorem 2*

Consider the acceleration feedback controlled chained multiple mass-spring damper system (1-2), and let assume that, for repetitive trajectory tracking $x_1^d$:

- the measurement noise $\eta$ is a white Gaussian noise of covariance $\Omega(t)$,
- at each time $t$, the variation of the modal masses $(m_i^1(t) \cdots m_i^k(t) \cdots m_i^K(t))$ - where the subscript $k$ indicates the $k^{th}$ iteration and $K$ is the number of iterations - can be represented by a Gaussian noise with zero mean and covariance $R(t)$.

For any particular time *t*, the estimate $\tilde{M}^k(t) = (\tilde{m}_1^k(t) \cdots \tilde{m}_N^k(t))^T$ of the modal masses, defined by the Kalman-Filter-like algorithm:

$$\tilde{M}^{k+1}(t)=\tilde{M}^{k}(t)+\frac{P^{k}(t)\ddot{X}^{k+1}(t)}{R(t)+\ddot{X}^{k+1}(t)^{T}P^{k}(t)\ddot{X}^{k+1}(t)}\left(H^{k+1}(t)-\ddot{X}^{k+1}(t)^{T}\tilde{M}^{k}(t)\right) \quad (3)$$

$$P^{k+1}(t)=P^{k}(t)-\frac{P^{k}(t)\ddot{X}^{k+1}(t)\ddot{X}^{k+1}(t)P^{k}(t)}{R(t)+\ddot{X}^{k+1}(t)^{T}P^{k}(t)\ddot{X}^{k+1}(t)}+\Omega(t) \quad (4)$$

with $\ddot{X}^{k}(t)=\left(\ddot{x}_{1,k}\cdots\ddot{x}_{N,k}\right)^{T}$, ,

$$H^{k}(t)=\sum_{i=1}^{N}\hat{m}_{i,k}\ddot{x}_{i,k}+\hat{m}_{1,k}\ddot{\tilde{x}}_{1,k}+k_{1,k}\dot{\tilde{x}}_{1,k}+k_{2}\tilde{x}_{1,k}.$$

converges uniformly towards the average values of the modal masses $m_i(t)$.

Moreover, the tracking error $\tilde{x}_1^k(t)$ of the position of the mass $m_1$ converges asymptotically to zero as $k\rightarrow\infty$.

*Proof*

From equation (1-2), one obtains:

$\sum_{i=1}^{N}m_i^k\ddot{x}_i^k=\sum_{i=2}^{N}\hat{m}_i^k\ddot{x}_i^k+\hat{m}_1^k\ddot{x}_{1,k}^d+k_1\dot{\tilde{x}}_1^k+k_2\tilde{x}_1^k+\eta$ and thus for every time t,

$$\sum_{i=1}^{N}\left(\hat{m}_i^k(t)-m_i^k(t)\right)\ddot{x}_i^k+\hat{m}_1^k(t)\ddot{\tilde{x}}_1^k+k_1\dot{\tilde{x}}_1^k+k_2\tilde{x}_1^k+\eta=0.$$

At time *t*, the algorithm behaves as if the system could be put under the form $\ddot{X}(t)^{T}\left(M(t)+\omega(t)\right)=H(t)+\eta(t)$, where $\omega,\eta$ are white noises, and the vectors $\ddot{X},M,H$ vary at every iteration. Under the assumptions above, one can consider that, for each time $t$, one can estimate the modal masses with the help of a Kalman filter given in equations (10-11) for which the "time" is the iteration number $k$. It is well known [18] that, in the case where the regression is linear, this Kalman Filter algorithm gives the optimal trade-off between tracking ability and noise sensitivity, in terms of a minimal a posteriori parameter error covariance matrix. Moreover, since

$\sum_{i=1}^{N}\left(\hat{m}_i(t)-m_i(t)\right)\ddot{x}_i+\hat{m}_1(t)\ddot{e}_1+k_1\dot{e}_1+k_2e_1+\eta=0$, if $\hat{m}_i^k$ converges to $m_i$ when $k\rightarrow\infty$, $\hat{m}_1(t)\ddot{e}_1+k_1\dot{e}_1+k_2e_1$ also converges to zero as $k$ tends to infinity.

## IV. Open-loop control of the end effector

Theorem 2 provides an indirect learning algorithm, which has the main advantage over blind direct iterative learning control to determine the key (in the sense of control purposes) model parameters which will be embedded into the controller. It is interesting to note that the algorithm only estimates those model parameters which are of interest for the control strategy.

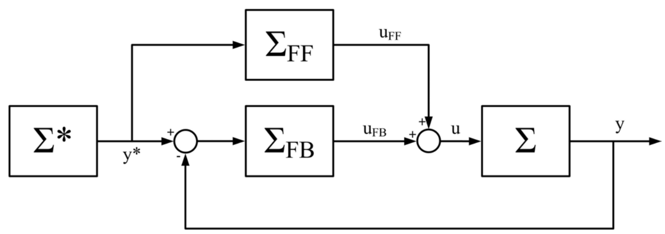


Fig. 2 Functional representation of a machining process.

The generic control structure of an industrial positioning system, shown in figure 2, consists of a control generator, which calculates the reference position of each axis by projection of the contour to be realized, and a control structure, which is linear and uncoupled axis by axis [19]. The feedback controller $\Sigma_{FB}$ drives the system $\Sigma$ to the reference trajectory $\Sigma^*$, whereas, in many traditional controllers, an additional feedforward term $\Sigma_{FF}$ helps to improve the tracking performances.

The acceleration feedback algorithm, which main purpose is to track a predefinite trajectory, is said to be a servo (feedback control) algorithm. However, in high-speed positioning systems, the mechanical or structural vibration is one of the most critical factors involved in deteriorating the machine's contouring performances, particularly in the stop stage or in presence of contour discontinuities [20]. A first way of reducing the vibrations consists of a fine tuning of the servo parameters in order to reach the required precision with minimum sacrifice of control bandwidth, which has been shown to be achieved with an acceleration feedback algorithm [1,8]. The second way of reducing the vibrations, which also has a dominant action on the contouring accuracy consists of an adequate path planning. In most industrial drives, the speed scheduling is carried out, classically to the maximum reachable values of speed and acceleration for each axis. However, the use of bang-bang profiles in speed or acceleration always deteriorates the vibratory behaviour when controlling a flexible structure. A method to reduce the vibration in the start or stop stages for a one-axis machine consists of smoothing the trajectory according to the flexibility of the structure, i.e. by cancelling the effects of its dominant frequency. This can be achieved by tuning the value of the jerk [21] or using input shapers (see [22] and related references).

It has been shown, that, when a flexible structure which can be represented by an oscillator with a pulsation $\omega_m$, the maximum vibration $\varepsilon_{vib}$ can be expressed as:

$$\left|\varepsilon_{vib}\right|=\frac{A}{\omega_m^2}\left|\mathrm{sinc}\left(\pi\left(A/J\right)/T_m\right)\right|, \quad (5)$$

where $A$ is the maximum acceleration and $J$ is the constant jerk value. The natural frequency of such a two-mass spring oscillator, is given by:

$$\omega_m=\sqrt{K/M_{load}}\sqrt{1+M_{load}/M_m},$$

where $K$ is the spring stiffness and $M_m, M_{load}$ are the motor and load masses.

From equation (5), it was straightforward to conclude that the value of the maximum jerk value, which would minimize the vibratory amplitudes of the system with the minimum sacrifice of rapidity, would be:

$$J \approx A / T_m . \quad (6)$$

Note that the damping ratio value has little influence provided that its value is sufficiently small (> 0.2) [21].

Once the maximum jerk value is correctly tuned, motion error on axial movements is considerably reduced. Since the stiffness $K$ of a two-mass spring damper which was able to represent the vibratory behavior of a Cartesian robot, was shown to vary less than the value of the corresponding modal masses, and is assumed to be known, the following algorithm can be proposed

*Jerk Tuning Algorithm*

*Step1 Perform acceleration feedback control (2) of the mass-spring-damper system (1)*
*Step2 Update modal masses with equations (3-4) and determine their mean value at stop stage*
*Step 3 Update the jerk value using (5-6)*
*Step 4 Go to Step 1 (new trajectory)*

In the neighborhood of the stop point, for which residual vibrations remain, the system can be considered as linear, and thus jerk or input shaping can be designed. Hence, the proposed method consists of embedding modal parameters which are updated by learning from the servo algorithm, into the trajectory planning algorithm. In practice, ILC requires that the reference trajectory remains the same during the iterations, whereas the new algorithm updates this trajectory every run, which will deteriorate the learning performances. However, for a small local modification of the trajectory, the model parameters are simply changing locally and slowly and results from the literature show that the gain in vibrations, given by equation (5), is very significant and that a good tuning of the jerk value can outperform the aforementioned drawbacks due to trajectory modification. The use of indirect learning can thus achieve the simultaneous tuning of path planning and servo parameters.

## V. Experimental Validation

The experimental validations are carried out on a 3-axes robot (figure 3). It has been equipped with a real-time “dSPACE 1103” control card. The available measurements on the motor part come from the actuator axis encoders, an accelerometer located on the end-effector gives the load acceleration. A laser sensor (measuring distance: 50 mm / measuring range: 20 mm) directly gives the load position and is only used for experimental verification (note that the cost of this sensor is prohibitive for industrial use).

Fig. 3 Overview of the first test-setup prototype (stroke [mm]: X-1000 Y-400 Z-800, maximum feedrate: 120 m.min$^{-1}$, maximum acceleration: 4 m.s$^{2}$).

When the horizontal axis is only moving, it will be considered that the axis stiffness remains almost constant $K = 0.95\ 10^5$ N.m . The validation was undertaken for a displacement on the X axis, with $x_2$ varying from $x_{2_0} = 0$ to $x_2 = 900$ mm , and $y_2 = 0$ mm with a height $z = 315$ mm .

The feedrate profile is a classical jerk-limited bang-bang, i.e. the acceleration exhibits a trapezoidal profile, as seen in figure 4 (when the jerk equals zero, the trajectory reduces to a bang-bang in acceleration). For comparison purposes, the axis was controlled with a PI-cascaded-loop for which the parameters were $k_p = 14$ s$^{-1}$, $k_v = 2$ A.rd.s$^{-1}$, where $k_p$, $k_v$ stand for the position and speed loop gains.

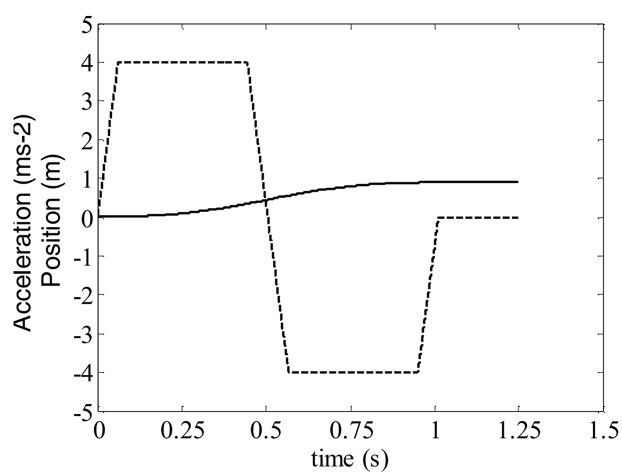


Fig. 4 Reference Trajectory Dotted: acceleration, Bold: Position

In a first time, a trajectory with a bang-bang in acceleration is considered, and the initial parameters of the acceleration feedback algorithm are set to $m_1 = 100\ Kg$ , $m_2 = 10\ Kg$ . A study of the root locus has shown that an underestimation of the modal masses leads to stable acceleration feedback controllers. The estimation of local masses from equation (4) is given in figure 5, and one can see that the iterative algorithm converges very fast.

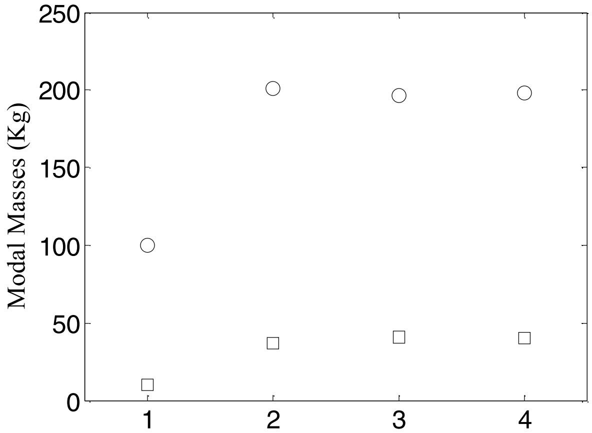


Fig. 5 Iterative evolution of the stop-point estimated modal masses, $m_1 \succ \circ$ , $m_2 \succ \square$

One can observe from Figure 6 that the acceleration feedback with inappropriate initial parameters is faster than the conventional controller, but, however, exhibits a higher overshoot. Note that, the industrial working requires that this overshoot should not exceed 0.2 mm, and that the rise time is defined as the time for which the motor position stays within 0.2 mm around its reference. The iteratively tuned algorithm converges very quickly and ensures a better vibratory behavior and improves the rise time by 20 %! Note that the vibrations do not completely vanish, due to unmodelled nonlinear dynamics such as dry friction.

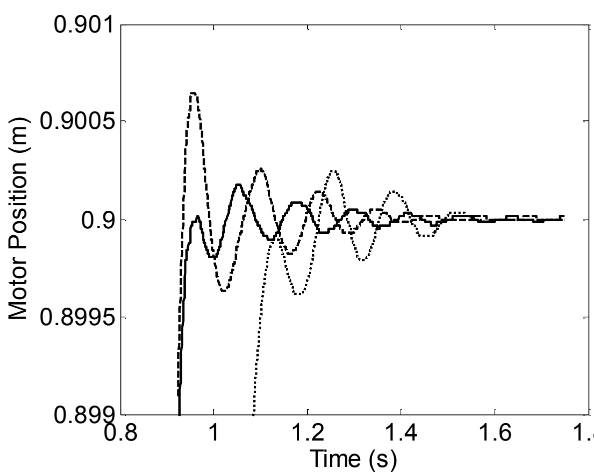


Fig. 6 Motor position
Conventional Control (…)
Acceleration feedback with initial parameters (- -)
Acceleration feedback tuned after 3 iterations (-)

Figure 6 represents the position of the load for cascaded-PI-loops with and without an adequate jerk value (which was found from frequency analysis and using equation 6), which emphasize the importance of an adequate path planning. Indeed, the manipulator is slower, but this is widely compensated by the improved level of vibrations, as the cycle time is given when the trajectory remains within 2 mm around the reference.

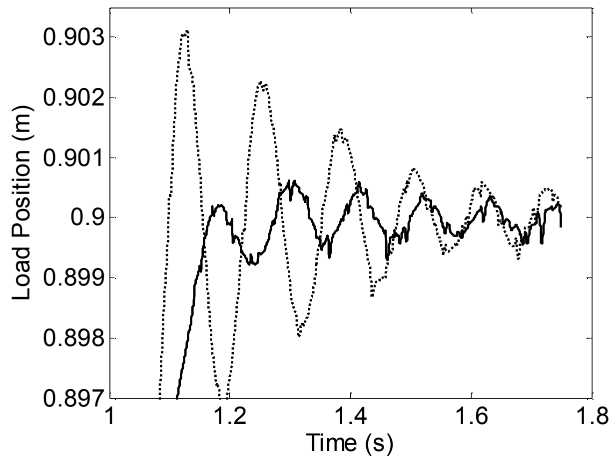


Fig. 7 Position of the load with cascaded loop with (bold) and without (dotted) jerk

The combination of iterative acceleration feedback and jerk tuning was tackled considering a constant stiffness $K = 0.95\ 10^5$ N.m and setting an initial jerk of $J = 65\ ms^{-3}$ derived from equation (5-6), when the modal masses are set to their initial value $m_1 = 100\ Kg, m_2 = 10\ Kg$ . Figure 8 shows how a wrong initial jerk value can deteriorate the vibratory performances. When, after 3 iterations, the modal masses are correctly estimated, the final jerk value will be correctly tuned, i.e. $J = 34\ ms^{-3}$ . The vibratory behavior is very good and respects the desired industrial performances. Moreover, the accurate tracking ensures an improvement of the cycle time by more than 10 % with respect to the industrial jerk-powered algorithm.

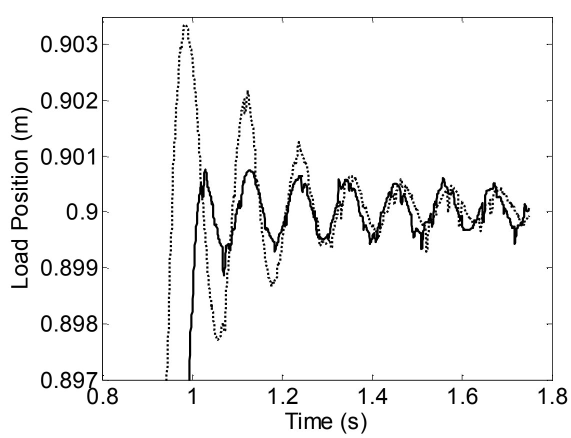


Fig. 8 Position of the load with acceleration feedback after iterative jerk and servo (bold) and with initial guess (dotted)

## VI. CONCLUSION

The control of industrial Cartesian robots arms, which can be represented by a chain of mass-spring-dampers units, is quite difficult because of the lack of end-point position and speed sensors. An acceleration feedback algorithm allows an exact tracking of the first modal mass which represents the motor part, when the values of the modal masses are known. It has been shown that, when the trajectory is repeated, an indirect Iterative Learning Control scheme allows the tuning of the acceleration feedback algorithm while estimating the value of its key parameters. The estimates of the modal masses can be used to modify the shape of the reference trajectory, by updating the value of the jerk or input shapers parameters, which results into a good improvement of the vibratory behavior. The algorithm has been validated on an industrial pick-and-place robot.

## REFERENCES


[1] Luo,G., Saridis,G., L-Q design of PID controllers for robot arms, IEEE Journal of Robotics and Automation, 1,1985, 152 – 159

[2] Luo G., Saridis, G., Robust compensation for a robotic manipulator, IEEE Transactions on Automatic Control, 29 , 1984, 564 – 567

[3] McInroy JE, Saridis GN., Acceleration and torque feedback for robotic control: experimental results. J. Robotic Systems, 7, 1990, 813 – 832.

[4] De Jager B., Acceleration assisted tracking control, IEEE Control Systems Magazine, 14 , 1994 , 20 – 27

[5] Readman, M., Belanger, P., Acceleration feedback for flexible joint robots, Proceedings of the 30th IEEE Conference on Decision and Control, 1991, 11-13 Dec. 1991, .2, 1385 – 1390

[6] Xu W.L., Han J.D., Joint acceleration feedback control for robots: analysis, sensing and experiments Robotics and Computer-Integrated Manufacturing, 16, 2000, 307 – 320

[7] García-Benitez E., Watkins J., Yurkovich S., Nonlinear control with acceleration feedback for a two-link flexible robot Control Engineering Practice, 1, 1993, 989 – 997

[8] Dumetz E., Dieulot J.Y., Barre P.J., Colas F., Control of an Industrial Robot using Acceleration Feedback, J. Intell Robot. Systems, 46, 111 – 128, 2006

[9] Meirovich L., Principles and Techniques of Vibrations, Prentice Hall, 1994.

[10] Xu J.X., Lee T.H., Zhang H.W., Analysis and comparison of iterative learning control schemes, Engineering Applications of Artificial Intelligence, 17, 2004, 675 – 686

[11] Moore K.L., Iterative Learning Control: An Expository Overview, Applied and Computational Controls, Signal Processing, and Circuits, 1, 1998

[12] Visioli, A., Ziliani, G., Legnani, G., An iterative learning control algorithm for contour tracking of unknown objects, Proceedings IEEE Int. Conference May 15-19, 2006, 3318 – 3323

[13] Jiang P., Unbehauen, R., Woo P.Y., Singularity-free indirect iterative learning control, Proceedings 40th IEEE Conference on Decision and Control , 5 , 2001, 4903 – 4908

[14] Kuc T., Lee, J.S., An adaptive learning control of uncertain robotic systems, Proc. 30th IEEE Conference on Decision and Control, 1991, .2, 1206 – 1211

[15] Sun D., Mills J.K., High-accuracy trajectory tracking of industrial robot manipulator using adaptive-learning scheme, Proc. American Control Conference, 3, 1999, 1935 – 1939

[16] Bukkems, B., Kostic, D., de Jager, B., Steinbuch, M., Learning-based identification and Iterative Learning Control of Direct-Drive robots, IEEE Trans. Contr. Syst. Technol., 13, 2005, 537 – 549

[17] Md Zain M.Z., Tokhi M.O., Mohamed Z., Hybrid learning control schemes with input shaping of a flexible manipulator system Mechatronics, 16, , 2006, 209 – 219

[18] Ljung L., System Identification: Theory for the User, Prentice Hall, 2nd edition, 1999.

[19] Papiernik W., "Architecture and design of modern CNC/drive systems". Intelligent Motion, May 1996, 271 – 280

[20] Lee K., Ibaraki S., Matsubara A., Kakino Y., Suzuki Y., Arai S., Braasch J., “A servo parameter tuning method for high-speed NC machine tools based on contouring error measurement”. Laser metrology and machine performance VI (proc. Of LAMDAMAP 2003), Huddersfield, England, July 1-4, 2003.

[21] Bearee, R., Barre P.J., Bloch S., Influence of high feed rate machine tool control parameters on the contouring accuracy, J. of Intell. Robot. Systems, 40, 2004, 321 – 342

[22] Singer N., Seering W., Preshaping command inputs to reduce system vibration, J. Dyn Systs., Measurements and Control, 112, 76 – 82