Cascaded Iterative Learning Control for Improved Task Execution of Optimal Control

Robertsson, Anders; Scalamogna, Domenico; Grundelius, Mattias; Johansson, Rolf

Published in:
Proceedings of IEEE International Conferance on Robotics and Automation (ICRA'02)

DOI:
10.1109/ROBOT.2002.1014720

2002

Proceedings of the 2002 IEEE
International Conference on Robotics & Automation
Washington, DC • May 2002

Cascaded Iterative Learning Control for Improved Task Execution of Optimal Control

A. Robertsson[+], D. Scalamogna[×][+], M. Grundelius[+], and R. Johansson[+]

[+]Department of Automatic Control, Lund Institute of Technology, SE-221 00 Lund, Sweden

[×]Department of Automatic Control, Università di Firenze, 50139 Firenze, Italy

E-mail: Anders.Robertsson@control.lth.se, fax +46-46-138118

Abstract

Iterative learning control (ILC) is used as a means for task execution of a time sub-optimal trajectory. The use of cascaded ILC procedures in a robot application is experimentally shown to drastically enlarge the region of convergence and efficiently compensate for unmodeled dynamics in the motion system. Objectives considered are the duration of motion, maximum liquid slosh during the motion, and the residual slosh after the motion.

Keywords Robotics, Iterative Learning Control (ILC), Motion Control

1 Introduction

Although time-optimal control lends itself to trajectory generation in mechatronic and robotic motion control, such theory provides few clues as to its implementation. One of the first aspects of motion control is whether to rely on open-loop control or solutions involving feedback control. Many industrial control solutions use open-loop solutions only and there are many obstacles to the use of sensor feedback. Among feedback control solutions one can distinguish among feedback used for dynamic control loops, adaptive control and iterative learning control. Iterative learning control (ILC) singles out itself as the most suitable method to improve on repetitive tasks without excessive requirements on sensor-feedback quality or control-loop bandwidth [1, 14]. Moreover, ILC permits the application to trajectories obtained from a variety of sources including open-loop control, time-optimal control, and model-predictive control (MPC).

We have chosen to study the usage of 'cascaded ILC' as a means to improve control performance of time sub-optimal/minimum energy based control by adapting the solution to unmodeled dynamics present in the mechatronic environment of a packaging machine/robot.

This paper deals with a control problem common in machines for packaging fluids. The operation of a packaging machine can be divided into three independent sub tasks: folding, filling, and sealing. These tasks are performed simultaneously on three different packages. The operation of the machine is as follows: The folded package is placed in a holder which carries the package through the machine. The movement of the package, which we will concentrate on, is stepwise and the same movement is repeatedly applied in every step and on every package in the machine (Fig. 1).

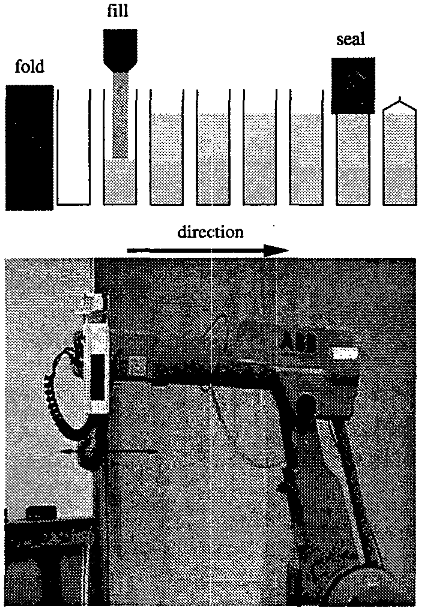


Figure 1: Scheme of the packaging machine (*upper*) and the setup for experimental evaluation of generated trajectories (*lower*). The arrow (⟷) in the lower picture indicates the horizontal motion of the container.

Recently, movement of open containers containing liquid has been investigated with the goal to move as fast as possible without slosh, see [18, 6, 5, 7].

The outline of the paper is as follows: In Section 2 we state the problem and relate to previously presented results. Section 3 outlines our proposed method and the experimental evaluation is described in Section 4 and concluded in Secs. 5–7.

0-7803-7272-7/02/$17.00 © 2002 IEEE

2 Problem formulation

The package contains liquid when it is moved between the filling station and the sealing station. Acceleration of the package induces motion of the liquid in the package, this is referred to as slosh or liquid vibration. The amount of slosh depends on how the package is accelerated, the geometry of the package and the properties of the fluid. If there is too much slosh, the liquid might splash out of the package and contaminate the machine or onto the sealing surfaces. This can result in packages that are not properly sealed and possibly not airtight.

The problem is to move the container as fast as possible under the constraint of low slosh during the motion and rapid decay of the waves after each stop in order to increase the production rate and thereby lower the packaging cost.

Experiments show that the behavior of the slosh is very repeatable for a given acceleration profile, which makes it suitable for the application of Iterative Learning Control.

The movement of the packages is usually controlled by a conveyer servo system which controls the position of the packages along one degree of freedom. In our case we have used an industrial robot for the motion control, therefore we need to control at least three degrees of freedom. To mimic the motion control system of the conveyor belt, an acceleration reference is specified which is integrated to obtain a velocity and a position reference. For an industrial implementation the only measurement available to the servo system is the position of the container. The measurements of the liquid elevation by a laser sensor on the experimental platform is only used for off-line update of the acceleration profile. Therefore, the only way to control the slosh is open loop via the reference trajectory.

3 Methods

3.1 Iterative learning control

The method of iterative learning control (ILC) applies well to the repetitive nature of the task we consider. Early contribution to ILC applied in robotics were made in [16, 1, 3, 4]. During the last years ILC has re-gained a lot of attention, see e. g., [11, 14] and the references therein.

The general update law for a one-step ILC in discrete time can be written

$$u_{k+1}(t) = Q(q)[u_k(t) + L(q)e_k(t)] \tag{1}$$

where $Q(q)$ and $L(q)$ are linear filters and $e_k(t) = r(t) - y_k(t)$ is the output error with respect to the desired reference $r(t)$. As the update law applies to a whole time sequence of measurements (iteration k), there are no restrictions on causality of $Q(q)$ and $L(q)$.

3.2 Slosh

The amount of slosh depends on how the package is accelerated, the geometry of the package, and the properties of the fluid. Modeling of liquid slosh in open containers has been studied in several applications, e. g., ship dynamics, fuel slosh in aerospace application, movement of containers and has been used for earthquake analysis.

The nonlinear effects are evident for very rapid movements when the surface elevation is large and the nonlinear phenomena include a nonlinear amplitude dependent oscillation frequency and asymmetric oscillation. Various linear and nonlinear models for small surface oscillation amplitudes were presented in e.g., [17, 12]. A review of the slosh modeling problem can be found in [7].

A simple linear model which captures the behavior well for small amplitudes is given by

$$\dot{x} = \begin{bmatrix} -2\zeta\omega & -\omega & a\omega/(2g\tau) & 0 \\ \omega & 0 & 0 & 0 \\ 0 & 0 & -1/\tau & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} x + \begin{bmatrix} a\omega/(2g\tau) \\ 0 \\ 1/\tau \\ 0 \end{bmatrix} v^r_{lin}(t) \tag{2}$$

where x_2 is the surface elevation in meters, x_1 is the time derivative of the surface elevation divided by ω, x_3 is the container velocity, x_4 the container position, ω the natural frequency of the fluid oscillation, τ an equivalent motor constant for the horizontal motion and $v^r_{lin}(t)$ the desired linear reference velocity for the container.

For a rectangular container with liquid depth h and width a the oscillation frequency of the first harmonic is given by the following expression:

$$\omega = \sqrt{\frac{g\pi}{a}\tanh\left(\frac{b\pi}{a}\right)} \tag{3}$$

which gives the value $\omega \approx 21.0$ rad/s for the container used in the experiments of Section 4.

The following constraints should be considered:

C1 Acceleration: $|u(t)| \leq u_{max}$

C2 Slosh: $|x_2(t)| \leq s_{max}$

C3 Initial state: $x(0) = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T$

C4 Terminal state: $x(T) = \begin{bmatrix} 0 & 0 & 0 & d \end{bmatrix}^T$

With a model of the slosh available, the acceleration reference can be calculated using optimal control techniques [2, 15].

3.3 Slosh ILC

The drawback with the minimum-time solution is that it is very sensitive to model uncertainties as shown in [8]. Grundelius and Bernhardsson instead suggested to use the solution of a Minimum Energy Problem, which generates a much smoother acceleration profile already for a marginal increase in time to the time optimal solution. The update law is similar to algorithms presented in e. g., [10, 8].

3.3.1 Initial acceleration profile, $u_0(t)$: The initial acceleration profile $u_0(t)$ for the iterative slosh update is taken as the sampled version of the continuous solution $u_c(t)$ to the Minimum Energy Problem

$$J(u_c) = \int_0^{T_{opt}+\Delta} u_c^2(t)\,dt \tag{4}$$

subject to the constraints $(C1-C4)$. T_{opt} is the time from the solution of the minimum-time problem and Δ can be viewed as a tuning parameter relaxing the solution by offering more time.

3.3.2 Update law for $u_k(t)$:

The update law for correction of the acceleration profile is based on discrete time algorithm presented in [9].

From the linear slosh model and the initial value $u_0(t)$ of the Minimum Energy Problem a reference $r^b(t)$ for the surface elevation on the backward side of the package and an initial acceleration reference $u_0(t)$ can be obtained.

The derivation of the update law considers a sampled discrete time version of the system where $u_k(t)$ denotes the acceleration reference of iteration k at time $t = n \cdot t_{samp}$, where t_{samp} is the sample time. The algorithm uses measurements of the slosh elevation on the backward and on the forward side of the package for each iteration k. These are denoted by $y_k^b(t)$ and $y_k^f(t)$

The optimal update laws is the correction to the previous reference $\delta u_k(t) = u_{k+1}(t) - u_k(t)$ which minimizes the prediction error $e_{k+1}(t) = r(t) - y_{k+1}(t)$. The error in iteration $k+1$ is approximately

$$\begin{aligned} e_{k+1}^b(t) &\approx e_k^b(t) - G(q)\delta u_k(t) \\ e_{k+1}^f(t) &\approx e_k^f(t) + G(q)\delta u_k(t) \end{aligned}$$

where $G(q)$ is the discrete time operator for the slosh model in Eq. (2). The update law for the ILC can be written in the form

$$U_{k+1} = QU_k + L_1E_k^1 + L_2E_k^2 \tag{5}$$

where U_k is the vector of references in iteration k; $\{u_k(n \cdot t_{samp})\}_{n=0}^{m-1}$, and the update of the acceleration reference δU_k is given as the solution to the following quadratic optimization problem

$$\begin{aligned} \mathcal{J}(\delta U_k) = \min_{\delta U_k}\{&(E_{k+1}^1)^T W_1 E_{k+1}^1 + (E_{k+1}^2)^T W_2 E_{k+1}^2 + \\ &(U_{k+1})^T W_u U_{k+1}\} \end{aligned} \tag{6}$$
subject to $A\,\delta U_k = 0$

where the constraint $A\,\delta U_k = 0$ is a sampled version of the constraints $C3$ and $C4$. The choice for the weighting matrices W_1, W_2, W_u gives a trade-off between the slosh error and the control signal. The solutions to Eq. 6 can be parameterized as $\delta U_k = ker(A)\theta$, where θ is an arbitrary vector. Inserting this expression gives

$$\begin{aligned} \delta U_k = K\left[K^T(G^T(W_1+W_2)G+W_u)K\right]^{-1} \times \\ K^T(G^T(W_1E_k^1 - W_2E_k^2) - W_uU_k) \end{aligned} \tag{7}$$

which can be written in the form of Eq. 5.

3.4 Robot system

The robot controller was implemented in the most simple way as distributed joint controllers without any feedforward terms to compensate for cross-coupling effects between the joints. Each joint controller consists of cascaded PI-controllers for velocity and position control to which the integrated acceleration references are fed.

3.5 Robot ILC

At iteration k the slosh ILC algorithm calculates the acceleration profile $u_{k+1}(\cdot)$, which should be applied to the package, mounted at the robot TCP.

The reference trajectories to the robot joints involved in the movement are then calculated via the inverse kinematics. As the motion is along a straight line in the plane as indicated in Fig. 1, we can restrict ourselves to use only three joints of the robot to generate the trajectory: Joint 2 and Joint 3 for the horizontal motion and Joint 6 to compensate for the rotation of the package.

As each joint is independently controlled we approximate the closed loop system from the angular position reference to the angular position response with a first order linear continuous time model. That is for the generic Joint i:

$$\widehat{T}_{c,i}(s) = \frac{1}{a_i s + 1} \tag{8}$$

The parameters a_i in Eq. (8) were identified from simple step response experiments. As will be shown in Section 5 neglected cross-couplings between the links and unmodeled resonant dynamics deteriorate the desired performance considerably before compensation (at iteration $k = 0$).

The joint model of Eq. (8) is used for the implementation of the model-based ILC algorithm as well as for the heuristic method described in [14]. Simulations with the two ILC methods gave rather similar results, wherefore the heuristic method was implemented in the experiments.

3.6 Task executive ILC algorithm

The general algorithm used in the experiment includes the use of the slosh ILC and the robot ILC according to the following scheme:

1. $k \leftarrow 0$
2. Calculate the initial acceleration u_0 by solving the Minimum Energy Problem and the slosh reference $r(t)$

3. Consider the acceleration reference $u_k(t)$
    - 3.1. Execute an iteration of robot ILC and measure the acceleration performed, $u_k^*(t)$
    - 3.2. If $u_k^*(t)$ does not approximate $u_k(t)$ well enough, go to step 3.1.
4. Execute an iteration of slosh ILC by reproducing on the robot the acceleration $u_k^*(t)$ performed in the last iteration of robot ILC.
5. $k \leftarrow k+1$
6. Calculate the new acceleration $u_k(t)$ using the slosh ILC algorithm.
7. If the slosh behavior needs to be improved go to step 3.

4 Experiments

4.1 Experimental setup

The experiments were implemented using the Open Robot Control System (ORC) at the Robotics Laboratory, Dept. of Automatic Control, Lund [13]. The manipulator is an industrial robot with six degrees of freedom, Irb2000, manufactured by ABB Robotics. The ORC system entitles full freedom to implement own low-level controller software by for instance down-loading and dynamically linking compiled Matlab/Simulink-code to the robot real-time system as well as it provides convenient methods to implement the high-level ILC-controller on a host computer which receives batch-data from the robot system and down-loads new reference trajectories for the next iteration.

The motion for moving the package horizontally is generated by providing position and velocity references for three of the distributed joint controllers of the robot (joint 2 and joint 3 for the horizontal motion and joint 6 for controlling the orientation of the package).

The container is attached to the tool center point of the robot and an infrared laser scanner is mounted above it to measure the slosh, see Fig. 2.

The sampling time for the laser sensor and for the update of references to the position control loop is $t_{samp} = 5$ ms.

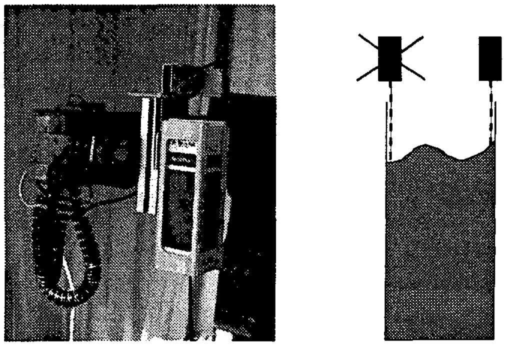


Figure 2: (Left) A The detailed picture how the package is fixed to the wrist of the robot. (Right) Above the container two slosh sensors are depicted. The sensor on the left side measures the slosh on the 'forward side' and the sensor on the right measures the slosh on the 'backward side' of the container. In the practical experiments only the sensor to the right was available.

To get a good description of the slosh in the package during motion, measurements of the liquid elevation on both sides of the container ('backward side' and 'forward side') are needed. However, in the experiments only one sensor is available (located on the 'backward side'), see Fig. 2. It is possible to overcome this problem by performing the same movement twice. The first time the package is moved forward by a spatial acceleration $u_k(t)$ the backward side surface elevation $y_k^b(t)$ is measured. After the distance $d = 0.2$ m has been covered the robot waits some seconds, necessary to let the residual slosh inside the container decrease towards zero. After that the robot moves the container backwards to the starting position with spatial acceleration $-u_k(t)$. The backward motion thus makes it possible to measure the same surface elevation $y_k^f(t)$ on the backward side which is present on the forward side when the robot moves forward.

5 Results

Initial experiments without 'the outer reference iterations' (robot ILC) were tested in case 'the slosh ILC' would be able to compensate also for the uncertain dynamics in the motion control system. However, this approach showed very poor convergence performance for fast motions and the behavior of the slosh was very sensitive to the effects of the resonant modes excited by cross-couplings between the joints.

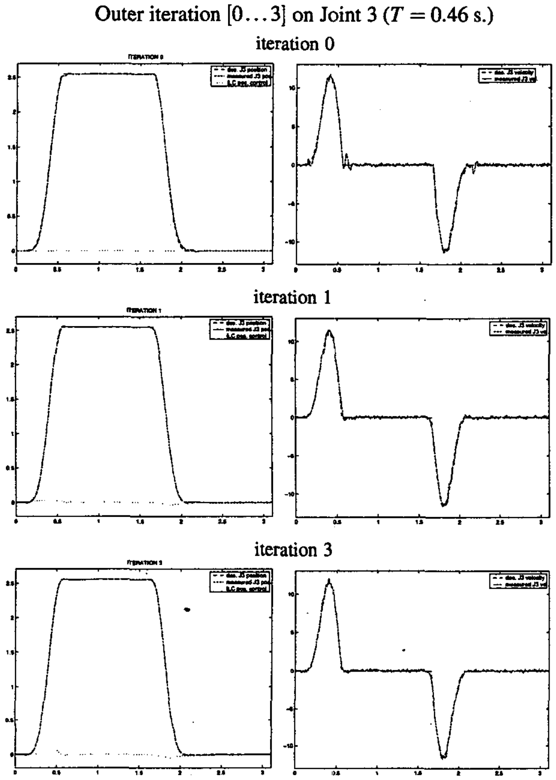
The convergence of the trajectories for Joint 3 using the 'robot ILC' is shown in Fig. 3. The effect of the cross-couplings between the joints is clearly seen in the overshoot of the velocity signal. The controller for Joint 3 is not perfectly tuned as there is a small overshoot also when we are 'locking' joint 2, but these effects are all taken care of by the reference update.

Using the combination of 'robot ILC' and 'slosh ILC' as outlined in Section 3 reduces the slosh considerably, see Fig. 4 for $T = 0.46$ s . For slower motion the residual will be further reduced.

Fig. 5 shows a comparison between the initial acceleration profile and the updated acceleration reference after for $T = 0.46$ s. The initial trajectory is generated by solving the Minimum Energy Problem. It was experimentally verified that for slow transition times the linear slosh model gave a good description of the true system, while the deviations grew larger with faster motion. However, the ILC procedure manages to capture and compensate for the model uncertainties.

In Fig. 6 the residuals for the first three iterations of the slosh ILC, $T = 0.46$ s, are shown together with a measure of the remaining energy in the residual, which clearly decreases with the iterations. However, after 3 or 4 iterations in general no improvements were made for the slosh ILC.

In Fig. 7 we see that the results improve for a longer transition time (slower motion), $T = 0.7$ s, as could be expected.


Outer iteration [0...3] on Joint 3 ($T = 0.46$ s.)

iteration 0

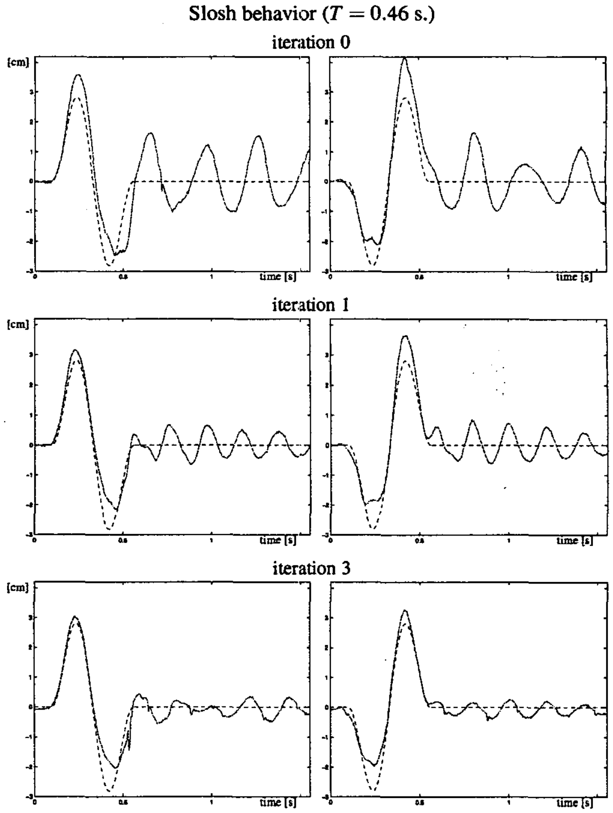
iteration 1

iteration 3

Figure 3: Tracking on the motor side of Joint 3 applying robot ILC to the position reference. The figure refers to iteration zero of Slosh ILC and to $T = 0.46$ s. Dashed line is used for reference signals, solid line for measured output signals, and dotted line for added ILC corrections. Every row represents one ILC iteration with the position signals (*left*) and the the velocity signals (*right*).

6 Discussion

The experimental setup was implemented on an open robot control system using an industrial manipulator to mimic the horizontal motion of a packaging machine. Even though the "conveyor belt-type-of-motion" described above may be the best motivated one from an industrial point of view for this application, other trajectories could be considered for further reduction of time and slosh. A method that utilize tilting of the container was derived in [7], which enables faster movements with less slosh. The methods simultaneously calculate the horizontal and rotational acceleration references by solving a minimum energy optimal control problem.


Slosh behavior ($T = 0.46$ s.)

iteration 0

iteration 1

iteration 3

Figure 4: Liquid surface elevation on the backward side (left column) and on the forward side (right column) using slosh ILC, when $T = 0.46$ s.

As the angle measurements of the robots are co-located with the actuators excessive number of 'outer reference iterations' may very well reduce the position and velocity errors on the motor-side, but may cause undesired motion on the arm-side, see also [14].

7 Conclusions

An iterative procedure to optimal motion control of liquid containers has been suggested and evaluated experimentally. The procedure combines 'outer iterations' for reference following with 'inner iteration' for slosh reduction. The cascaded use of the ILC procedures enlarges the region of convergence significantly and efficiently compensates for unmodeled dynamics in the motion system.

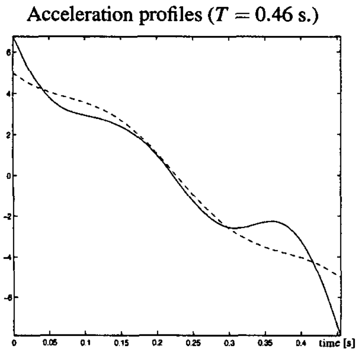


Figure 5: Initial acceleration profile (dashed line) and acceleration profile at iteration 3 of slosh ILC (solid line), $T = 0.46$ s.

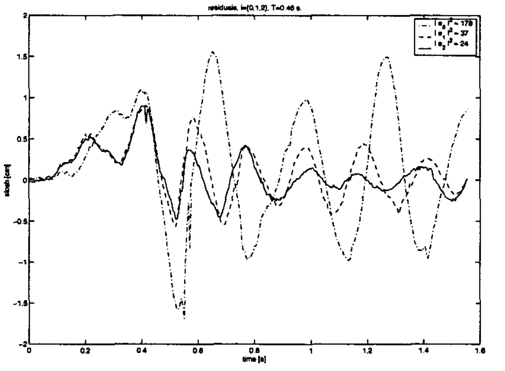


Figure 6: Residuals of slosh elevation measured on the 'backward side' using slosh ILC, iterations i=0,1,2, for $T = 0.46$ s.

Acknowledgments

This work was conducted when the second author visited the Dept. of Automatic Control, Lund, under the Erasmus program. The work has been supported by the National Board for Industrial and Technical Development(NUTEK), the Swedish Agency for Innovation Systems (VINNOVA), and Tetra Pak Research & Development AB.

References

[1] S. Arimoto, S. Kawamura, and F. Miyazaki. Iterative learning control for robot systems. In *Proceedings of IECON*, Tokyo, Japan, 1984.

[2] A. E. Bryson and Yu-Chi Ho. *Applied Optimal Control: Optimization, Estimation and Control.* Hemisphere Publishing Corp, New York, 1975.

[3] G. Casalino and G. Bartolini. A learning procedure for the control of movements of robotic manipulators. In *IASTED Symp on Robotics and Automation*, 1984.

[4] J.J. Craig. Adaptive control of manipulators through repeated trial. In *Proc. of ACC*, San Diego, CA, July 1984.

[5] F. Dubois, N. Petit, and P. Rouchon. Motion planning and nonlinear simulations for a tank containing a fluid. In *Proc. 1999 European Control Conf.*, Karlsruhe, Germany, September 1999.

[6] J.T. Feddema, C.R. Dohrmann, G.G. Parker, R.D. Robinett, V.J. Romero, and D.J. Schmitt. Control for slosh-free motion of an open container. *IEEE Control Systems*, 17(1):29–36, February 1997.

[7] M. Grundelius. *Methods for Control of Liquid Slosh.* PhD thesis, Lund Institute of Technology, Sweden, October 2001.

[8] M. Grundelius and B. Bernhardsson. Motion control of open containers with slosh constraints. In *Preprints 14th IFAC World Congress*, volume L, pages 487–492, Beijing, P.R. China, July 1999.

[9] M. Grundelius and B. Bernhardsson. Constrained iterative learning control of liquid slosh in an industrial packaging machine. In *Proc 39th IEEE Conf Decision and Control*, Sydney, Australia, Dec 2000.

[10] S. Gunnarsson and M. Norrlöf. Some aspects of an optimization approach to iterative learning control. In *Proceeedings of the 38th IEEE Conference on Decision and Control*, Phoenix, Arizona USA, December 1999.

[11] S. Gunnarsson, O. Rousseaux, and V. Collignon. Iterative feedback tuning applied to robot joint controllers. In *Proc. of the 14th IFAC World Congress*, pages 451–456, Beijing, P.R. China, July 1999.

[12] J. W. Miles. Nonlinear surface waves in closed basins. *Fluid Mechanics*, 75:419–448, 1976.

[13] K. Nilsson and R. Johansson. Integrated architecture for industrial robot programming and control. *J. Robotics and Autonomous Systems*, 29:205–226, 1999.

[14] M. Norrlöf. *Iterative Learning Control: Analysis, Design and Experiments.* PhD thesis, Department of Electrical Engineering, Linköping University, 2000. Sweden.

[15] A. L. Schwartz and E. Polak. *RIOTS: A Matlab Toolbox for Solving Optimal Control Problems Version 1.0.* University of California Berkeley, Berkeley, California, April 1996.

[16] M. Uchiyama. Formulation of high-speed notion patterns of a mechanical arm by trial. *Trans on Soc. Instrum. Control and Eng.*, 14(6):706–712, 1978. (in Japanese).

[17] R. Venugopal and D. S. Bernstein. State space modeling and active control of slosh. In *Proc 1996 IEEE Int Conf Control Appl*, pages 1072–1077, Dearborn, Michigan, September 1996.

[18] K. Yano, T. Yoshida, M. Hamaguchi, and K. Terashima. Liquid container transfer considering the suppression of sloshing for the change of liquid level. In *Proc 13th IFAC World Congress*, San Francisco, California, July 1996.

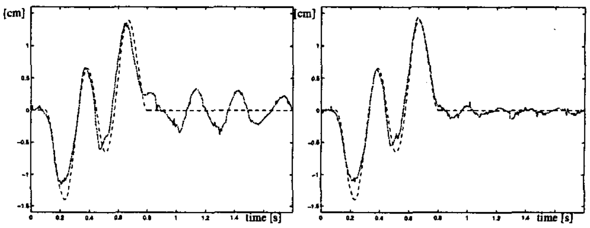


Figure 7: Liquid surface elevation on the 'forward side' using slosh ILC, when $T = 0.7$ s. Iterations 0 and 2 are shown.