Inverse Dynamics of Flexible Manipulators

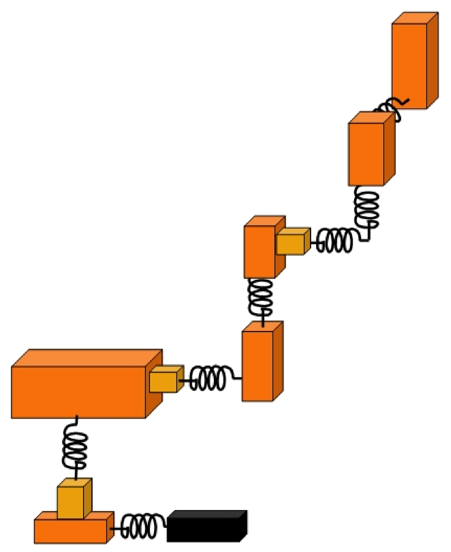
Stig Moberg[*,†] and Sven Hanssen[*,‡]

[*]ABB AB – Robotics, Sweden [†]Department of Electrical Engineering Linköping University, Sweden [‡]Department of Solid Mechanics, Royal Institute of Technology, Sweden

Background

Robot manipulators are traditionally described by the **flexible joint model** or the **flexible link model**. These models only consider elasticity in the rotational direction. A real manipulator has a distributed flexibility in all directions. This work investigates different methods for the **inverse dynamics** of a more general manipulator model, called the **extended flexible joint model**. The inverse dynamics solution is needed for **feedforward control**, which is often used for high-precision robot manipulator control.

Extended Flexible Joint Model

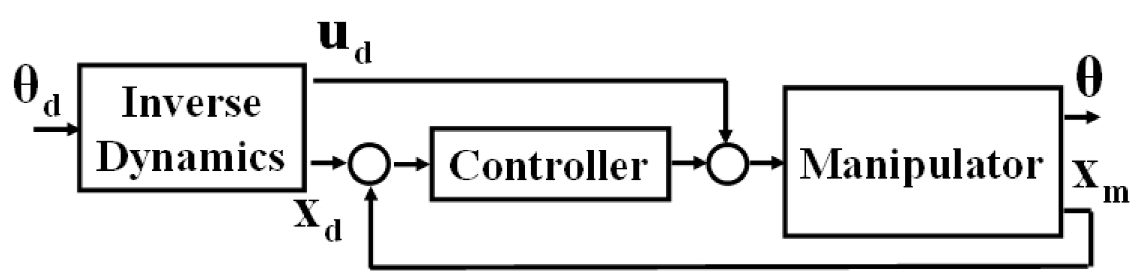


The model consists of a serial kinematic chain of rigid bodies. The rigid bodies are connected by multidimensional spring-damper pairs, representing the rotational deflection. If two rigid bodies are joined by a motor and a gearbox, we get an **actuated joint** represented by one spring-damper pair, modeling the rotational deflection of the gearbox. The other spring-damper pairs represent **non-actuated joints**, modeling the elasticity of bearings and links. A reasonable model of a 6-axes robot can have more than 20 degrees-of-freedom, i.e., more than 40 states. The model equations, including the forward kinematics, are

$$
\begin{aligned}
M(q_a)\ddot{q}_a + c(q_a,\dot{q}_a) + g(q_a) &= \begin{bmatrix}\tau_g\\ \tau_e\end{bmatrix},\\
\tau_g &= K_g(q_m - q_g) + D_g(\dot{q}_m - \dot{q}_g),\\
\tau_e &= -K_e q_e - D_e\dot{q}_e,\\
\tau_m - \tau_g &= M_m\ddot{q}_m + f(\dot{q}_m),\\
\theta &= \Gamma(q_a).
\end{aligned}
$$

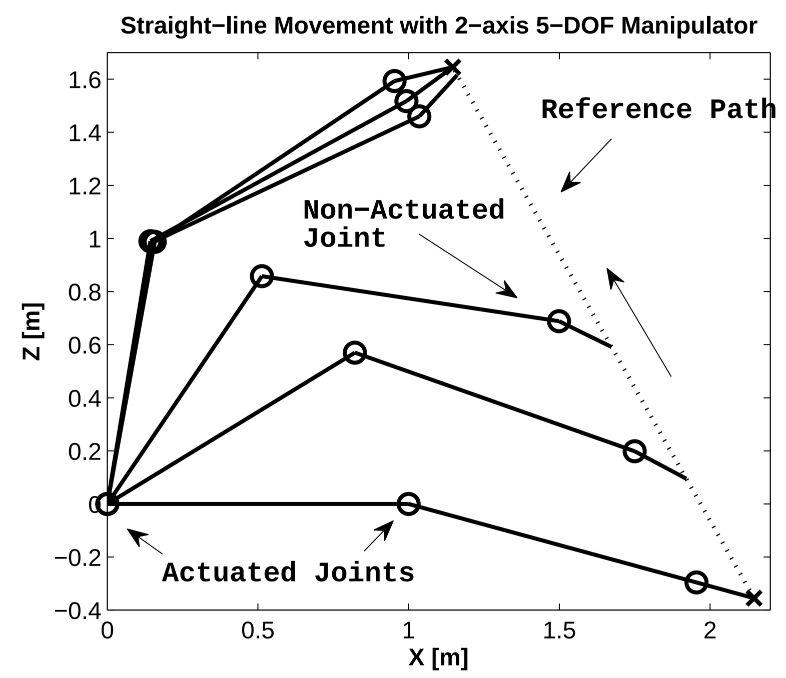
Inverse Dynamics Solution

The task of the inverse dynamics is to compute the required control torque u_d and state references x_d from the desired Cartesian trajectory $\theta_d(t)$. The kinematics is not invertible which makes the inverse dynamics very hard to solve. The inverse dynamics and kinematics must therefore be solved simultaneously.

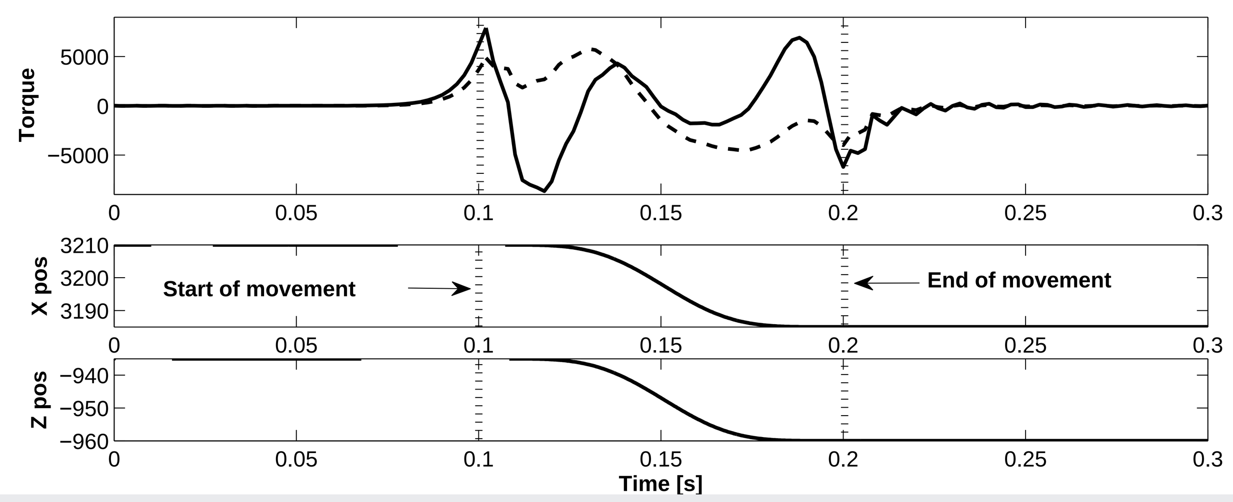


- Solution of high-index ($\geq$ 4) differential algebraic equation (DAE), $F(t,x,\dot{x},u) = 0$.
- An accurate solution requires a small stepsize, which makes the problem ill-conditioned.
- Solver based on constant-stepsize constant-order backwards differentiation formula (BDF) has been developed. No commercially available solver that can solve this problem has yet been found!
- The discretized DAE, $F(t_i, x_i, D_h x_i, u_i) = 0$, solved for each time-step.
- Index reduction by utilizing the structure of equations and the dummy derivatives method (minimal extension).
- Original high-index DAE can be solved directly if proper scaling is performed. This is the preferred method (to avoid differentiations).

For **stable zero dynamics**, i.e., the system from control signal to tool position is minimum-phase, the inverse dynamics can be solved as an **initial-value problem**.



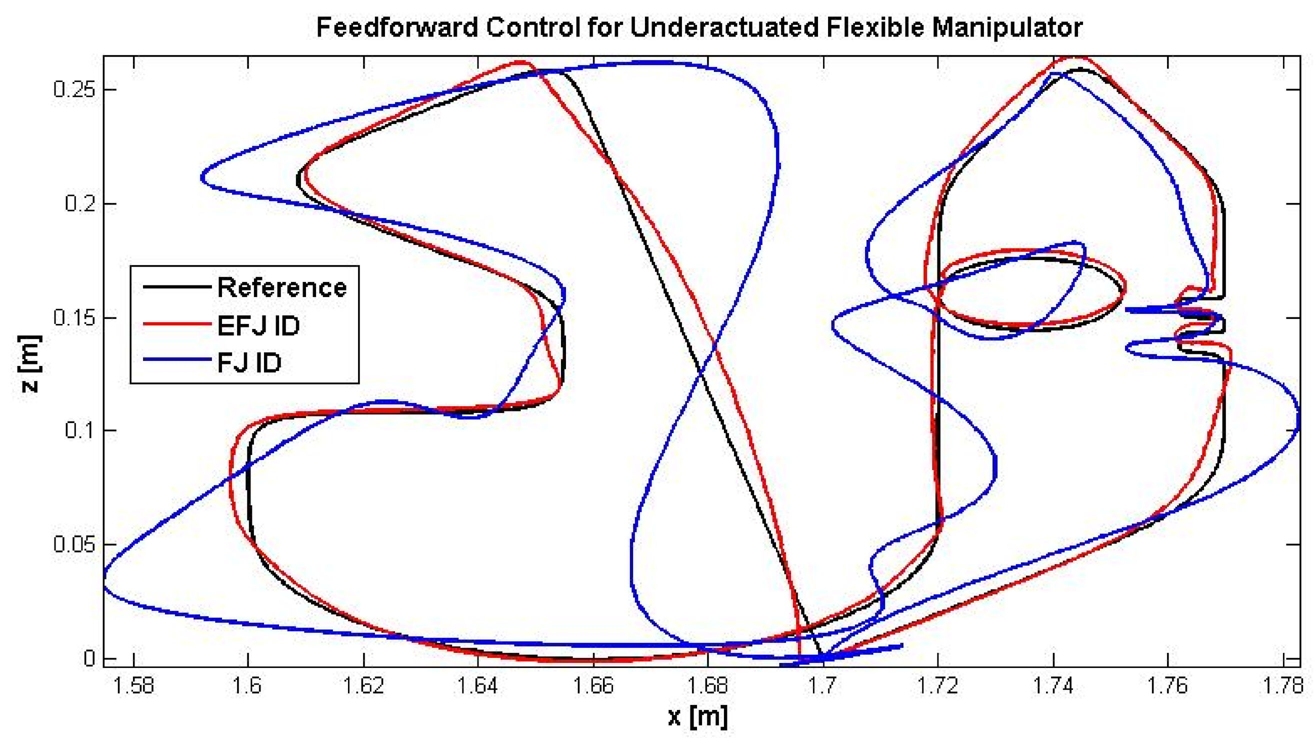
A non-causal solution of the inverse dynamics



For **unstable zero-dynamics** (non-minimum phase), the solution to the initial-value problem will be unstable. One solution to this problem is to solve the problem for the complete trajectory simultaneously, stacking the discretized DAEs for all time-steps, and allow the solution to be **non-causal**. An accurate solution requires a small step-size and for models with many DOFs and long trajectories, the optimization problem gets very large. One way to reduce the size of the problem is to express each DOF as a **basis function expansion**, computing the derivatives analytically and then solve for the unknown coefficients of the series expansion, i.e., to solve the DAE with a **collocation** method. Both problems can be solved as a **nonlinear least-squares** problem.

Experimental Evaluation

The method was evaluated by using 3 axes on a IRB6640 robot from ABB, using an experimental controller. Two axes were controlled by inverse dynamics (feedforward) control and one axis was made extremely elastic (eigenfrequency 1 Hz) by using a PD controller with zero reference as a simulated spring-damper pair, i.e. a non-actuated joint. A path is shown in the figure below (tool position is measured with a laser-tracker). The figure shows the result when using extended flexible joint inverse dynamics compared to flexible joint inverse dynamics.



Future Work

Some issues are a more efficient solver, more complex manipulator models, smooth trajectory generation with constrained control signals, handling of manipulator singularities and alternative configurations, and of course a real-time solution.