

# An Industrial-Robots Suited Input Shaping Control Scheme

Amine Kamel, Friedrich Lange, Gerd Hirzinger

**Abstract** Compliance in robot mounted force/torque sensors is useful for soft mating of parts in many assembly tasks. Nevertheless, it generates nearly undamped oscillations when moving a heavy end-effector in free space. In this paper, input shaping control is investigated to damp such unwanted flexible modes. However, the conventional method presents a major drawback: To eliminate the oscillatory dynamics, the desired motion profiles have to be shaped and thus modified. This means that although the unwanted vibrations are damped, the robot motion does not meet the desired one. In this paper we first review the conventional input shaping technique. Second we show how the mentioned problem may be fixed in the design phase by discretizing the filter and by using a predictive approach that compensates the shaped signals time delay and minimizes the resulting control error. Simulation results are presented.

## 1 Introduction

Compliant force /torque sensors are frequently used in robot assisted assembly tasks. They don't only perform measurements, but also inhibit high frequency-motions that commonly occur in the contact phase. This advantage drops into a drawback if the end-effector is moved in free space. In fact, due to the sensor compliance, poorly damped oscillations emerge which lead to unprecise motion of the tool. Such oscillatory behavior is critical in many applications with high speed and precision requirements. In this paper we address the problem of wheel assembly to a continuously moved car (Fig. 1). A camera is mounted at the tool to correct the motion profiles when needed and to detect the car hub. Undamped oscillations inhibit this process and have hence to be eliminated.

---

Amine Kamel e-mail: Amine.Kamel@dlr.de

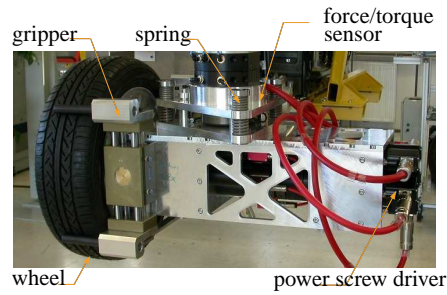
Friedrich Lange e-mail: Friedrich.Lange@dlr.de

Gerd Hirzinger e-mail: Gerd.Hirzinger@dlr.de

German Aerospace Center, Institute of Robotics and Mechatronics, D-82234 Weßling, Germany



**Fig. 1** (a) Setup for wheel assembly



(b) Compliant end-effector with force/torque sensor in the center of the springs

*Input shaping* also known as *command preshaping* is one of the easiest successfully applied feedforward control techniques that have been designed to suppress residual vibrations occurring within speedy maneuvers. Some pre-knowledge about the plant is used to generate commands which move the system without vibrations. The first form of input shaping, also called *posicast control*, was presented 1957 by Smith [1]. It consists in generating two transient oscillations that cancel each other and lead to a vibrationless response. The first paper of the conventional input shaping was presented by Singer and Seering [2]. Desired system inputs were convolved with an impulse train. The resulting commands move the system without residual vibration. Very good estimations of the plant parameters were essential to eliminate the oscillatory dynamics. Many researchers addressed this problem and developed robust input shapers by adding more impulses to the filter. Singhose [3] presented a design method to add any desired order of robustness to the filter.

To suite input shaping to industrial robots, three major matters have to be discussed:

- 1- Filtering the inputs induces some time delay which leads to system performance degradation. In the task presented above, positional ramps are commonly commanded to move the tool from a given Cartesian position to another. Hence the ramp response time delay has to be compensated.
- 2- The conventional theory of input shaping has been primarily developed in the continuous time domain. Hence the application to systems with long sampling period leads to implementation problems. This can be fixed by digitizing the shaper.
- 3- An input shaper modifies slightly the reference signals to damp the vibrations of the end-effector. This means, although the oscillations are eliminated, the tool will not move as desired since modified trajectories are commanded! Hence, an exact tracking of the desired path and a total vibrations damping seem to be two contradictory goals that cannot be simultaneously fully fulfilled. However a tradeoff can be reached by minimizing the sum of the sensor deflection and the deviation between the desired and commanded signals. A formulation and a computationally efficient solution of such optimization problems are presented in this paper.

## 2 Review of the Conventional Input Shaping

The original method has been primarily developed for linear second order systems with the transfer function:

$$G(s) = \frac{y(s)}{u(s)} = K \frac{\omega_0^2}{s^2 + 2D\omega_0 s + \omega_0^2} \quad (1)$$

$$g(t) = \frac{K\omega_0}{\sqrt{1-D^2}} e^{-\omega_0 D t} \sin(\omega_d t) \quad (2)$$

With a static gain  $K$ , a positive damping ratio  $D$  smaller than 1, a natural frequency  $\omega_0$  and a damped natural frequency  $\omega_d = \omega_0 \sqrt{1-D^2}$ .  $u(s)$  and  $y(s)$  denote respectively the system input and output.

It's known that applying an impulse  $A_0 \delta(t - t_0)$  to such a plant will result in an oscillating response  $g_0(t)$ . However a well chosen second impulse  $A_1 \delta(t - t_1)$  can excite a second oscillation  $g_1(t)$  that totally cancels the first one for  $t \geq t_1$ . This idea can be extended to an impulse sequence with  $n$  impulses

$$f_\delta(t) = \sum_{i=0}^{n-1} A_i \delta(t - t_i); \quad t_i < t_{i+1}; \quad i \in \{0, 1, \dots, n-1\} \quad (3)$$

which compensates any oscillation immediately after applying the last impulse. By convolving this sequence with any desired command signal, new control inputs are generated which move the system without vibration. This command generation process is called input shaping. To eliminate the oscillations, the filter has to satisfy the following conditions (see [2])

$$\mathcal{C}(\omega_0, D) = \sum_{i=0}^{n-1} A_i e^{\omega_0 D t_i} \cos(\omega_d t_i) = 0 \quad (4)$$

$$\mathcal{S}(\omega_0, D) = \sum_{i=0}^{n-1} A_i e^{\omega_0 D t_i} \sin(\omega_d t_i) = 0 \quad (5)$$

The constraints (4) and (5) can be satisfied by setting all the amplitudes  $A_i$  to zero or by allowing them to have infinite values. Such trivial solutions are uninteresting for practical sakes and have to be eliminated. Therefore we require that:

$$\sum_{i=0}^{n-1} A_i = 1 \quad (6)$$

This condition makes sure that the filter has a unity static gain. Thus, the steady state values of the references and the commands will be the same. Since the range of the commands is dependent on the actuators, we can set constraints to the actuators' limitations and then solve for positive and negative amplitudes which satisfy them (Singhose [7]). A general form of these constraints is:

$$A_{i_{min}} \leq A_i \leq A_{i_{max}} \quad \text{and} \quad \Delta A_{i_{min}} \leq \Delta A_i \leq \Delta A_{i_{max}} \quad (7)$$

Thereby  $A_{i_{min/max}}$  and  $\Delta A_{i_{min/max}}$  are the respective minimal/maximal allowed amplitude and increments values.

For a first order robustness, also the derivatives of (4) and (5) with respect to  $\omega_0$  are constrained to zero (Singer and Seering [2]).

$$\sum_{i=0}^{n-1} A_i t_i e^{\omega_0 D t_i} \cos(\omega_d t_i) = 0 \quad (8)$$

$$\sum_{i=0}^{n-1} A_i t_i e^{\omega_0 D t_i} \sin(\omega_d t_i) = 0 \quad (9)$$

Equations (4), (5), (6), (8), (9) and the restrictions (7) define a constrained set of nonlinear equations (CSNE) that can be numerically solved for amplitudes  $A_i$  and time instants  $t_i$  to get a zero vibration robust input shaper.

### 3 Ramp Time Delay Compensation

Long sequences of impulses afford many design degrees of freedom and allow to accommodate the input shaper to complex and demanding constraints. However, the longer the sequence is, the bigger is the filtering time delay. In this section we focus on the ramp time delay compensation since positional ramps are common for the task presented above.

Let  $\tau$  be the ramp response time delay when applying input shaping. Kamel et al. described in [9] the dependency of  $\tau$  from the input shaper parameters and the plant parameters:

$$\tau = \frac{2D}{\omega_0} + \sum_{i=0}^{n-1} A_i t_i \quad (10)$$

$\frac{2D}{\omega_0}$  describes the delay caused by the plant (1) whereas  $\sum_{i=0}^{n-1} A_i t_i$  is the shaping delay. By setting  $\tau$  to zero, the dead time will be totally compensated:

$$\sum_{i=0}^{n-1} A_i t_i = -\frac{2D}{\omega_0} \quad (11)$$

Notice that the statement (11) compensates not only the input shaping time delay but the one of the plant too! This feature may now be included into the filter design by adding (11) to the CSNE as an additional constraint.

However requiring a total dead-time elimination leads often to huge amplitude values within short sequences of impulses. This can be avoided either by lengthening the sequence or by using predictive path scheduling within a known time delay (backward time shifting): When the desired trajectory is a priori known, then the

control inputs may be time advanced (Lange and Hirzinger [8]). In this case, (10) is used to enforce some known time delay  $\tau_0$  which can be compensated due to command shifting (see [9]).

## 4 Time-Discrete Input Shaping

The discretization of input shapers has been the emphasis of many publications ([4],[5],[6]). Singer described in [4] a digital shaper by fixing the time between the impulses and only changing the magnitudes  $A_i$ . Based on the ideas in [4], Kamel et al. described in [9] a systematic design to generate a time discrete input shaper for low sampled robotic systems. This will be briefly reviewed in this section.

In order to fit the time instants of the impulses to the sampling period  $T$  we can explicitly constrain all  $t_i$  and  $\tau_0$  to be a multiple of  $T$ . An intuitive choice may be:

$$t_i = iT \quad ; \quad \tau_0 = mT \quad (12)$$

where  $i \in \{0, 1, \dots, n-1\}$  and  $m \in \mathbb{N}$  are design parameters used to set the ramp time delay to a known value.

Adding (12) to the CSNE eliminates the time instants and replaces them by the known integers  $i$ . Note that the nonlinear statements are transformed to linear ones by fixing the time instants  $t_i$ . Thus the CSNE becomes a constrained set of linear equations (CSLE):

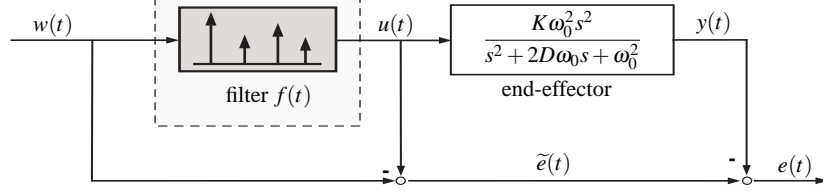
$$C\underline{A} = \underline{b} \quad (13)$$

with the  $C \in \mathbb{R}^{6 \times n}$ , the amplitude set  $\underline{A} \in \mathbb{R}^n$  and the right side  $\underline{b} \in \mathbb{R}^6$  (see [9]). The problem can now be stated as follows: Find a vector of amplitudes  $\underline{A}$  that satisfies the CSLE stated above. For  $n > 6$ , the statement (13) is under-determined. The problem has consequently for a given sequence length  $n$  an infinity of solutions from which we need to select one that satisfies (7) if it exists. This task can be solved by many numerical iterative tools. An iterative algorithm is presented in [9] to solve this problem and to keep the length of the impulse sequence to a minimum.

## 5 Minimization of the Quadratic Control Error

At this level, it's very important to realize that a good performance of the filter does not only depend on a good vibrations' damping, but also on other important criteria. In fact, the elimination of the oscillatory dynamics invokes implicitly a modification of the commands. This means concretely that the robot will effectuate vibrationless motions which unfortunately do not match exactly the desired motion. One way to deal with this problem is to minimize the positional control error due to the filtering

and to the sensor deflections. To do so we first of all consider the following signal flow diagram stated in Fig. 2:



**Fig. 2** Signal flow diagram of the total positional control error

$w(t)$  denotes the desired motion profile. For the following optimization task  $w(t)$  is supposed to be a unity gain step  $\sigma(t)$ . Using these commanded references, the input shaper generates the system inputs  $u(t)$ :

$$u(t) = f(t) * w(t) = \sum_{i=0}^{n-1} A_i \sigma(t - t_i) \stackrel{(12)}{=} \sum_{i=0}^{n-1} A_i \sigma(t - iT) \quad (14)$$

If the robot is supposed to be ideal (only rigid body interactions with no delay), then  $u(t)$  corresponds to the measured position profile of the robot flange. In this case, the difference  $\tilde{e}(t)$  between the references  $w(t)$  and the input  $u(t)$  is the position error caused by the input shaper.  $y(t)$  denotes the deflections measured in the compliant sensor. One can easily verify that:

$$y(t) = \frac{d}{dt} g(t) * f(t) = \sum_{i=0}^{n-1} A_i \frac{d}{dt} g(t - iT) \quad (15)$$

where  $g(t)$  is the impulse response of plant (1). The total position error reads:

$$e(t) = \tilde{e}(t) - y(t) = w(t) - u(t) - y(t) \quad (16)$$

### 5.1 Analytical formulation of the cost function

The equations (14), (15) and (16) provide an analytical characterization of the control error occurring when applying a positional step. We define the following minimization problem:

$$\min_{\underline{A} \in \mathbb{R}} \mathcal{J}_q \quad \text{subject to} \quad \underline{C}\underline{A} = \underline{b} \quad \text{where} \quad \mathcal{J}_q(\underline{A}) = \underline{A}^T \underline{Q}\underline{A} + \int_0^{\infty} e^2(t) dt \quad (17)$$

$Q$  is a positive definite weighting matrix to penalize high amplitude values. One can already feed the problem at this level to a numerical iterative solver. However, the computational effort will be extremely high. That's why, it's recommendable to formulate and solve the problem analytically. Due to (6) and to the fact that the input shaper totally eliminates any oscillations immediately after applying the last impulse, the control error  $e(t)$  exists only between  $t = 0$  and  $t = t_{n-1} = (n-1)T$ . For the derivation below we suppose that a prediction over  $m$  sampling steps is performed. This means that  $w(t)$  switches to 1 at  $t = mT$ .

$$\begin{aligned} \int_0^{\infty} e^2(t) dt &= \int_0^{(n-1)T} e^2(t) dt = \int_0^{(n-1)T} (w(t) - u(t) - y(t))^2 dt \\ &= (n-m-1)T - 2 \underbrace{\int_{mT}^{(n-1)T} u(t) + y(t) dt}_l + \underbrace{\int_0^{(n-1)T} (u(t) + y(t))^2 dt}_q \end{aligned} \quad (18)$$

$l$  and  $q$  denote the terms of the cost function that respectively lead to a linear and quadratic dependency on the magnitudes  $A_i$ . The linear term  $l$  can be easily computed:

$$\begin{aligned} l &= \sum_{i=0}^{n-1} A_i \int_{mT}^{(n-1)T} \sigma(t-iT) + \frac{d}{dt} g(t-iT) dt = \sum_{i=0}^{n-1} A_i \left( \int_{\max(m,i)T}^{(n-1)T} dt + \int_{\max(m-i,0)T}^{(n-i-1)T} \frac{d}{dt} g(t) dt \right) \\ &= \sum_{i=0}^{n-1} A_i \left[ (n - \max(m, i) - 1)T + g((n-i-1)T) - g(\max(m-i, 0)T) \right] = \sum_{i=0}^{n-1} A_i \theta_i \end{aligned} \quad (19)$$

To determine the quadratic term  $q$  we need first to compute some sub-integrals:

$$q_1 = \int_0^{(n-1)T} u^2(t) dt = \sum_{i,j=0}^{n-1} A_i A_j \int_{\max(i,j)T}^{(n-1)T} dt = \sum_{i,j=0}^{n-1} A_i A_j (n - \max(i, j) - 1)T \quad (20)$$

$$\begin{aligned} q_2 &= \int_0^{(n-1)T} u(t)y(t) dt = \sum_{i,j=0}^{n-1} A_i A_j \int_{\max(i-j,0)T}^{(n-j-1)T} \frac{d}{dt} g(t) dt \\ &= \sum_{i,j=0}^{n-1} A_i A_j (g[(n-j-1)T] - g[\max(i-j, 0)T]) \end{aligned} \quad (21)$$

$$\begin{aligned} q_3 &= \int_0^{(n-1)T} y^2(t) dt = \sum_{i,j=0}^{n-1} A_i A_j \int_0^{(n-1)T} \frac{d}{dt} g(t-iT) \frac{d}{dt} g(t-jT) dt \\ &= h_{i,j}((n-1)T) - h_{i,j}(0) \end{aligned} \quad (22)$$

where

$$h_{i,j}(t) = -\frac{K^2 \omega_0^3 e^{\omega_0 D T (i+j)}}{4(1-D^2)} e^{-2\omega_0 D t} \left[ \frac{\cos(\omega_d T (j-i))}{D} - \cos(2\omega_d t - \omega_d T (i+j) - \varphi) \right]$$

Therefore:

$$\begin{aligned} q &= q_1 + 2q_2 + q_3 = \sum_{i,j=0}^{n-1} A_i A_j \left[ (n - \max(i, j) - 1) T + h_{i,j}((n-1)T) - h_{i,j}(0) \right. \\ &\quad \left. + 2(g[(n-j-1)T] - g[\max(i-j, 0)T]) \right] \\ &= \sum_{i,j=0}^{n-1} A_i A_j \psi_{i,j} \end{aligned} \quad (23)$$

The equations (18), (19) and (23) give an analytical formulation of the cost function:

$$\begin{aligned} \mathcal{J}_q(\underline{A}) &= \underline{A}^T \underline{Q} \underline{A} + \sum_{i,j=0}^{n-1} A_i A_j \psi_{i,j} - 2 \sum_{i=0}^{n-1} A_i \theta_i + (n-m-1)T \\ &= \underline{A}^T \underbrace{(\underline{\Psi} + \underline{Q})}_{\tilde{\Psi}} \underline{A} - 2 \underline{A}^T \underline{\theta} + (n-m-1)T \end{aligned} \quad (24)$$

Since analytical expressions for  $g$  and  $h$  are available, the computation of the matrix  $\underline{\Psi}$  and the vector  $\underline{\theta}$  does not need any numerical integration. Once computed,  $\underline{\Psi}$  and  $\underline{\theta}$  can be used to evaluate the costs for any given amplitude set  $\underline{A}$ . Hence the determination of the optimal solution  $\underline{A}_{opt}$  does not need huge computational effort. At this level, an iterative solver can be used to compute the optimum. However an analytical solution can be derived to figure out the dependency of  $\underline{A}_{opt}$  on  $\underline{\Psi}$ ,  $C$  and  $\underline{\theta}$  and hence on the plant parameters.

## 5.2 Analytical solution of the minimization problem

Using an appropriate Lagrangian function, the constraints (13) may be coupled to the cost function to compute a general solution for the problem formulated above:

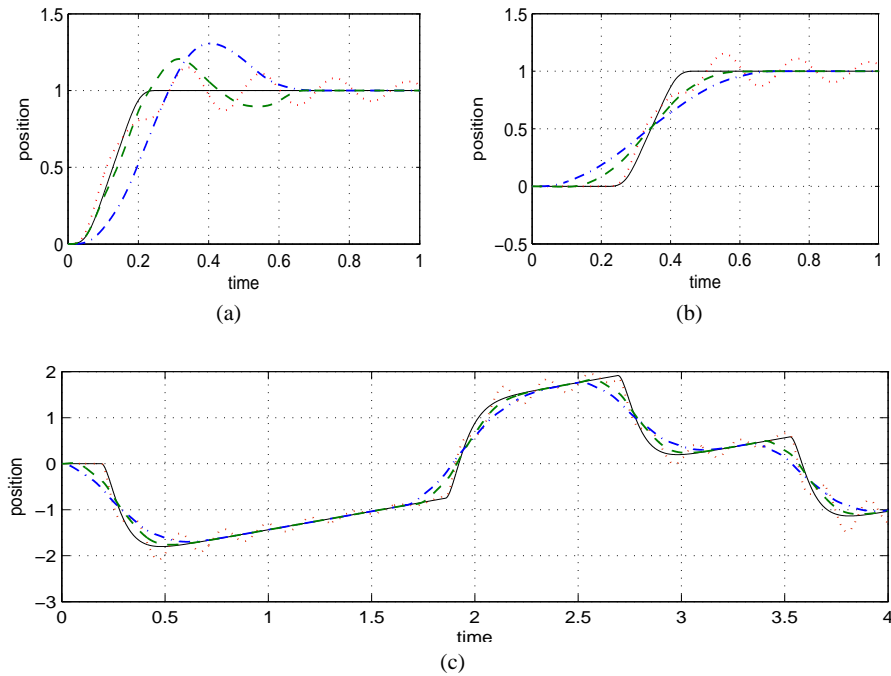
$$\underline{A}_{opt} = P \left[ 2\underline{\theta} - C^T \left( (CPC^T)^{-1} (2CP\underline{\theta} - \underline{b}) \right) \right] \quad (25)$$

with  $P = \left( \tilde{\Psi} + \tilde{\Psi}^T \right)^{-1}$ . Note that this solution is only valid for regular matrixes  $P$  and  $C$ . Note also that the stated solution is a minimum if and only if  $P$  is positive definite. In fact, this restriction is not that dramatic, since we can always influence  $P$  by the choice of the elements of the matrix  $Q$ .

## 6 Results

The robot motion and the end-effector oscillation are fairly decoupled when using the approach of [8] to control the setup of Fig. 1. Thus the desired motion of the end-effector almost coincides with the actual motion and thus deserves as input for the end-effector control. In particular, there is no interdependence with the robot joint states. Therefore input shaping can be directly applied to the desired positions/orientations of the Cartesian components of the robot motion.

The individual components can be modeled by independent second order transfer functions (1), considering only the dominant oscillation each. If several modes would be significant in each case, input shaping could be applied to each of them, thus yielding a sequence of input filters. In both cases, damping of the respective oscillations reduces as well the cross-couplings between the individual degrees of freedom.



**Fig. 3** Simulation results of the system response to finite rate step responses(a&b) and to a typical robot motion profile (c) with  $w = 30 \frac{rad}{s}$ ,  $D = 0.02$ ,  $K = 8 \cdot 10^{-4}$  and  $T = 12ms$ . (a): no prediction is applied i.e  $m = 0$ , (b&c): prediction over the first 18 sampling steps is applied i.e  $m = 17$ . Dotted: step response without input shaping. Dashed-dotted: step response using an unoptimized input shaper ( $n = 36$ ). Dashed: step response using an optimized input shaper ( $n = 36$ ) with  $Q = 0.1 \cdot I$  ( $I$  is the unity matrix).

Fig.3 shows samples of our simulation results. Both optimized and unoptimized filters could compensate the shaping time delay using a prediction over  $m$  sampling steps. One can clearly see, that an optimized input shaper does not only filter the oscillatory dynamics of the plant's output, but also tracks the references better than other shapers. For large impulse sequences, we could reduce the costs  $\mathcal{J}_q$  up to 35%.

Notice that compared with the shapers presented in [9] which only minimize the sensor deflection, the current shaper minimizes also the deviation between the references and the commands. Thus the deflected end-effector pose is controlled to track the reference. Measurements of the sensor deflection are required only for the identification of the system. They are not more used for control.

## 7 Conclusion

The paper demonstrates that the well known method of input shaping can be modified to fit some principle features of today's industrial robots. A systematical and extendable computational framework is provided to generate such modified shapers. Since fixed robot paths can be commanded in advance, the resulting time delay is not unfavorable and could be compensated. Besides, control errors due to the shaping process and to the oscillation of a compliant tool are minimized. Future work will address the matter of the optimum sensitivity with respect to the plant parameters.

## References

1. Smith, O. J. M.: Postcast Control of Damped Oscillatory Systems. Proc. of the IRE, pp. 1249-1255, 1957.
2. Singer, N. C. and Seering, W.P.: Preshaping Command Inputs to Reduce System Vibration, ASME, J. of Dynamic Systems, Measurement and Control, Vol. 112, pp. 76-82, 1990.
3. Singhose, W. E., Seering, W. P. and Singer, N. C.: Input Shaping for Vibration Reduction with Specified Insensitivity to Modeling Errors, Japan-USA Sym. on Flexible Automation, Boston, MA, 1996.
4. Singer, N. C.: Residual vibration reduction in computer controlled machines, Ph.D. dissertation, AI-TR 1030, Artificial Intell. Lab., MIT, Cambridge, MA, January 1989.
5. Murphy, B. R. and Watanabe, I.: Digital Shaping Filters for Reducing Machine Vibration IEEE Transactions on Robotics and Automation, Vol. 8, pp. 285-289, 1992.
6. Tuttle, T. and Seering, W.: A Zero-placement Technique for Designing Shaped Inputs to Suppress Multiple-mode Vibration, American Control Conference, Baltimore, MD, 1994, pp. 2533-2537.
7. Singhose, W. E., Seering, W. P. and Singer N. C.: Time-optimal negative input shapers, ASME, J. of Dynamic Systems, Measurement, and Control, Vol. 119, pp. 198-205, 1997
8. Lange F. and Hirzinger G.: Spatial Vision-Based Control of High-Speed Robot Arms. In: Industrial Robotics: Theory, Modeling and Control. Advanced Robotic Systems, Vienna, 2007.
9. Kamel, A., Lange, F. and Hirzinger G.: New Aspects of Input Shaping Control to Damp Oscillations of a Compliant Force Sensor, IEEE International Conference on Robotics and Automation (ICRA), Pasadena, California, May 2008.