UNIVERSITY OF CALIFORNIA, SAN DIEGO

Systematic Dynamic Modeling Based on Step-Response Experiments Thesis

A Thesis submitted in partial satisfaction of the
requirements for the degree
Master of Science

in

Engineering Sciences (Mechanical Engineering)

by

Alicia Dautt-Silva

Committee in charge:

Professor Raymond A. De Callafon, Chair
Professor Thomas R. Bewley
Professor Robert R. Bitmead

2017

The Thesis of Alicia Dautt-Silva is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Chair

University of California, San Diego

2017

DEDICATION

In memory of my brother Charles, who taught me the love for music.

In memory of nana Natalia, for all her lessons.

In memory of professor Dagoberto Cruz Sibaja, whose teachings defined my path.

To my parents, Alicia and Alejandro, for their unconditional support.

To my little sister, Joan, my future doctor.

To my always smiling sister, Nathalie.

To my stress relief little monsters, Tita, Luna and Colin.

And to my best friend and beloved husband, Daniel, for his encouragement to pursue my dreams.

EPIGRAPH

Grown-ups like numbers. When you tell them about a new friend, they never ask questions about what really matters. They never ask: 'What does his voice sound like?' 'What games does he like best?' 'Does he collect butterflies?' They ask: 'How old is he?' 'How many brothers does he have?' 'How much does he weigh?' 'How much money does he have?' Only then do they think they know him.

—Antoine de Saint-Exupéry, The Little Prince

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

Thanks to my coworkers at CYMER, who inspired me to pursue a graduate degree. Thanks to Charles Kinney, the best TA I could have; to Bob Jacques, the most inquisitive person who helped me with my questions; to William Whitty, 'the MATLAB prodigy' for his mentoring and tricks; to Steve Chang, for his support and explanations of the experimental data. To Meindert Norg, Simon Mushi, Rob Receveur, Joost de Pee, Dai Feng, Ivo Houtzager, Alberto Villalta, Daniel Riggs, Wayne Dunstan, Grant Cavalier, Andrew Liu, Liane Mattes, Ronald Truong, Paul Frihauf, Dean Richert, and the whole CYMER team for their support. Thanks to Omez Mesina for his support since the beginning of this long journey.

Thanks to all my fellow graduate students who made the Master's program more enjoyable and fun: Aaron Ma, Matt Atlas, David Larson and Alex vanDine. Thanks to Amir Valibeygi, Thayjes Srivas, Abhishek Subramanian and Leobardo Camacho, for their support through the final stage working on my thesis. Thanks to Jeremy Orosco, my friend and study partner for the last three years, who helped me with his invaluable critical feedback and encouragement.

Thanks to CYMER for allowing the use of the data from Lasers Technology Development Testing. Thanks to General Atomics for allowing me the time and supporting me to complete the Master's degree program.

Thanks to my academic advisors in the Student Affairs team, Sandra de Sousa and Anne Campbell, for their support in the most unbelievable chores.

Thanks to the UCSD faculty staff for their great teaching. Thanks to Professor Bewley, for the many challenges he presented to me during the past 3 years. Thanks to Professor Bitmead, for his kind help on my defense and great feedback on my work.

Mostly, I would like to thank my faculty advisor Professor Raymond A. de Callafon, for all his time and patience. For showing me the world of system identification, and mentoring me on this thesis. For always guiding me in the right direction.

Thank you Professor de Callafon for making me feel as a truly valued member of the student community.

ABSTRACT OF THE THESIS

Systematic Dynamic Modeling Based on Step-Response Experiments Thesis

by

Alicia Dautt-Silva

Master of Science in Engineering Sciences (Mechanical Engineering)

University of California, San Diego, 2017

Professor Raymond A. De Callafon, Chair

The purpose of this thesis is the computation of optimal input signals for output tracking of a dynamic system. The starting point is a non-optimal input signal given in the form of a simple step input and the measurement of the resulting output. The data is used to formulate a (linear) dynamic model from which the optimal input signal is computed via "input shaping". This paper presents a summary of the method to obtain a dynamic model from the step response data, and compares various "input shaping" methods to compute (sub)optimal input signals to achieve a desired output signal. The method to estimate a dynamic model is based on the

realization algorithm; the optimal input shaping techniques compared in this paper include zero vibration (ZV), finite impulse response (FIR) filtering and a convex optimization formulation using linear programming (LP). It is shown that the linear programming solution for input shaping can also be generalized to find optimal input signals with a fixed resolution using a mixed integer linear programming formulation. The approach of dynamic modeling and input shaping is illustrated on a simulation example of a two-mass system as well as experimental data obtained from a class IV LASER system characterized by varying the pulse length of the low power used to seed it.

Chapter 1

Introduction

The evolving demands of modern technologies require the development and use of smaller, cheaper semiconductors which are both high speed and energy efficient. The modeling and analysis of novel techniques used in the manufacture of these new devices is a common, important practice for ensuring expected performance.

For such systems, obtaining accurate models that can be used for the corresponding analysis is often achieved by experimentation and the examination of the input-output relationships of the system via system identification [1].

The available analysis tools yield to models that capture the relevant dynamics of the systems even in the presence of uncertainty. Within this context, experiments are usually designed that utilize the process input as the experimental variable. A common input for the analysis of dynamics systems such as LASERs or servo systems is the step input, from which one obtains the system's step response [2].

By using a step input, we can perform everything in the time domain: average, isolate the particular step and then utilize the realization algorithm. Using a proper deterministic input signal, allows us to obtain a model with the realization algorithm

[3].

Different approaches to compute a desired response are explored; a desired response is defined for a system, consisting in the time to reach a steady state and the maximum allowed overshoot/undershoot. First, obtaining an inverse model.

Input shaping is defined as a feed-forward control technique first proposed in the 1980s. It is a technique for shaping the system input (commands) to eliminate the system vibration in the response.

From input shaping, first we explore zero vibration (ZV) [4]. Based on the knowledge that most of the systems will vibrate when an impulse is given, applying a second impulse to the system can cancel the vibration induced by the previous impulse. In ZV, is needed to derive the amplitude and locations of the impulse commands [5].

The use of input shaping techniques in control systems helps to reduce the overshoot/undershoot and the settling time of under-damped systems. Finite impulse response (FIR) filters are a different approach [6]. Unlike ZV input shapers, which are impulse-based, the FIR filters preshape an input command to reduce the vibration. [7]

The inverse model and input shaping techniques approaches enforce no limits on the computed input or constraints on the system response. Convex optimization will also be explored with a view to applying both constraints [8]. Finally mixed integer linear programming (MILP), in which the at least one of the input variables is restricted to integers, will be applied [9, 10].

In this paper, we review the full process of testing new equipment, analyzing a step-response experiment, doing the system identification for it and analyzing the

possible input shaping techniques to obtain an ideal response. Two examples are utilized to illustrate this process, the LASER experiment and a two-mass system experiment. An Educational Control Product (ECP) is used to verify the results obtained for the two-mass system.

Chapter 2 shows the experimental data and how it is obtained, as well as the system identification for each experiment. Chapter 3 shows the input shaping techniques and Chapter 4 shows the application of each. In Chapter 5 the experimental results are presented. Conclusions are Chapter 6.

1.1 Motivation

The main objectives of this work are as follows: to describe a systematic procedure for modeling consistency and to use the obtained model in conjunction with input shaping techniques in order to enhance control inputs such that a more desirable system response can be achieved.

To this end, two sets of data are utilized to exemplify the input shaping methods previously mentioned.

1.1.1 Two-Mass System Model

The two-mass system is used for verification and interpretation of the results we get with the processes we compare; since it's a well understood dynamic system, ZV, FIR and optimization will be illustrated. We consider a step input on the two-mass system and compute the inputs for each method based on the step-response. The motivation is to move both masses, m_1 and m_2, as fast as possible to a desired

position. The objective is to find the right input signal to obtain the ideal result for m_1.

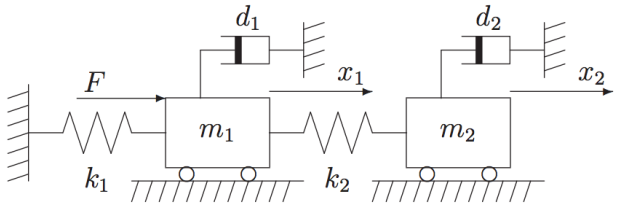

Figure 1.1: Two-mass system. Cascade spring/mass/damper system.

The two-degree-of-freedom (2DOF) mechanical system depicted in Figure 1.1 consists of 2 masses, m_1 and m_2, each having positioning freedoms x_1 and x_2, respectively. The masses are connected via spring elements having stiffness coefficients k_1 and k_2 (spring or flexible shaft). Additionally, to model the damping present in the system, a viscous damping, d_1 and d_2, are assumed to act on each of the masses in the mechanical system [11].

To obtain data, we work with the ECP model 210, rectilinear plant in Figure 1.2.

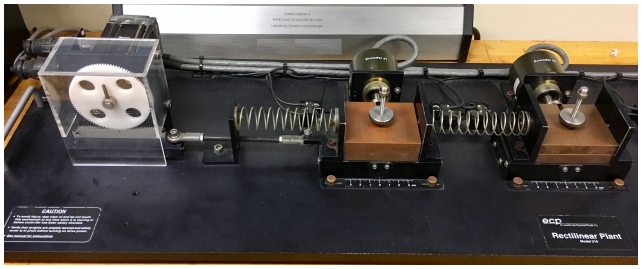

Figure 1.2: ECP Model 210.

1.1.2 LASER Pulse Length Experiment

In the semiconductor industry, to expose good wafers it is necessary to control the amount of extreme ultraviolet (EUV) generated as a function of time. The exper-

iment aimed to identify a means to control the amount of EUV, using the LASER as an actuator. The amount of EUV generated is proportional to the energy delivered by the LASER. Consequently, it is required to adjust the LASER power in order to compensate for other system variations as a means to keep the EUV constant.

The step-response experiment is used in order to identify a model and its parameters. The goal is to characterize the optical properties of the LASER pulse at the output of a LASER amplification chain as a function of the input pulse length used to seed the chain.

Figure 1.3 shows the raw data from the experiment. The experiment has 16 cycles in total. Figure 1.4 shows the data of one of the sixteen steps.

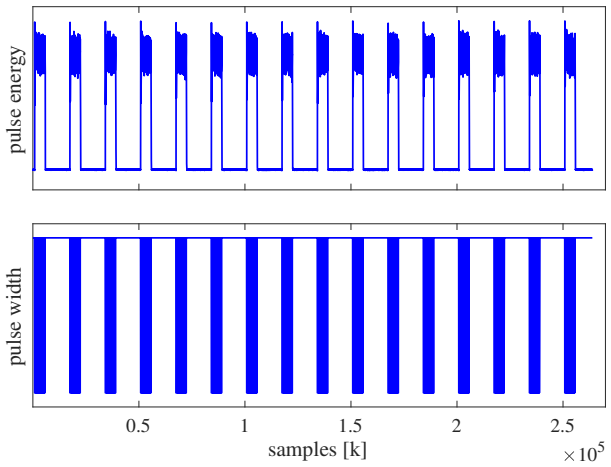

Figure 1.3: Raw data from experiment consisting of sixteen steps. The pulse width and energy are in arb. units.

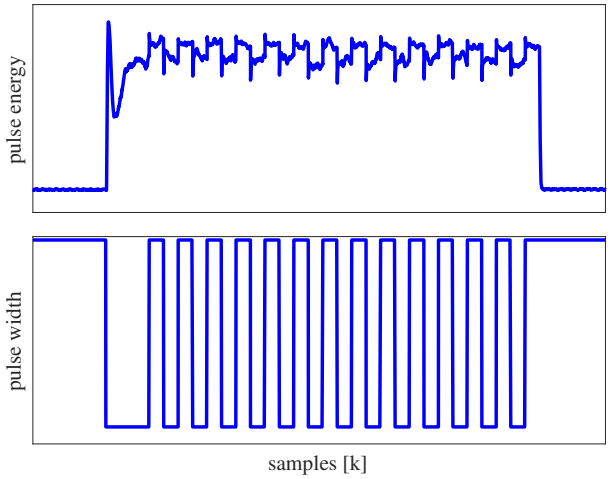

Figure 1.4: Raw data of one of the sixteen steps.

Chapter 2

Experimental Data and Step Response Model Estimation

2.1 Experimental data

2.1.1 Data for the Two-Mass system Experiment

The mechanical system can be moved and positioned by a control force, denoted by F. The control force is a rectilinear force applied to the first mass m_1.

To obtain the data for the two-mass system, we apply a step input to the ECP and log the results. The data obtained for m_1 is shown in Figure 2.1. There is an undesirable oscillation as noted in the response and a settling in the later samples. Notice the stiction/friction effects in the ECP data. In the following section we will discuss this compared to the obtained model.

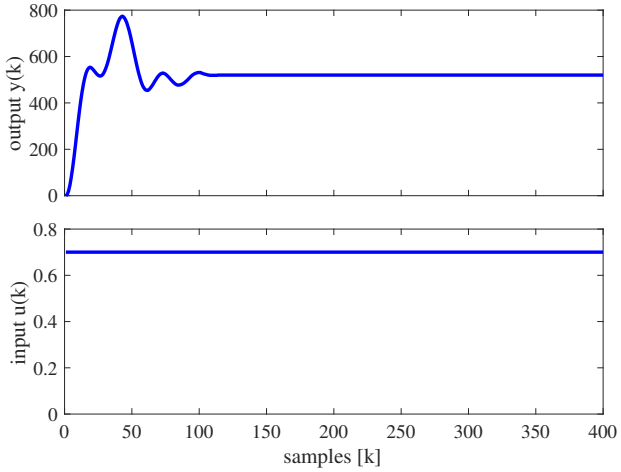

Figure 2.1: Step-response for the ECP two-mass system.

2.1.2 Data for the LASER Pulse Length Experiment

The repeated steps in the LASER experiment are averaged to produce one set of data summarizing the results and in order to reduce the noise. We can see in Figure 2.2 that the Laser Energy signal is noisy when the step goes up and down, and we can predict there are different dynamics during the step up and step down transitions.

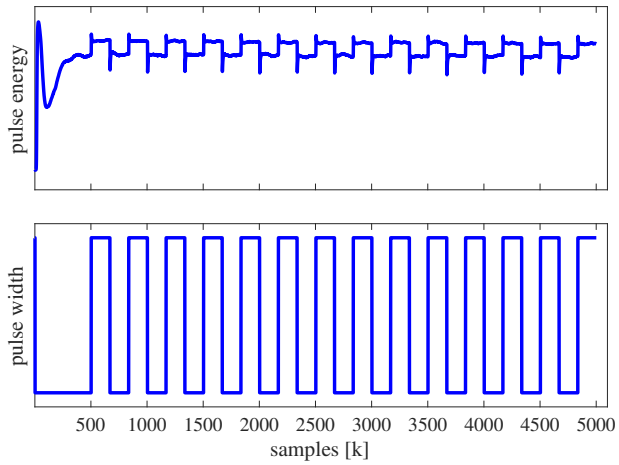

Figure 2.2: Average of the sixteen steps.

The data for step up and step down is split and from now on, we refer to two

single-input single-output (SISO) models; step up and step down. The signal split is made by identifying the transitions from low to high and high to low in the pulse width input. The purpose is the system identification of both systems [1].

Figure 2.3 shows our starting point for the data from the LASER pulse experiment; a non-optimal input signal in the form of a step input, and the measurement of the resulting output.

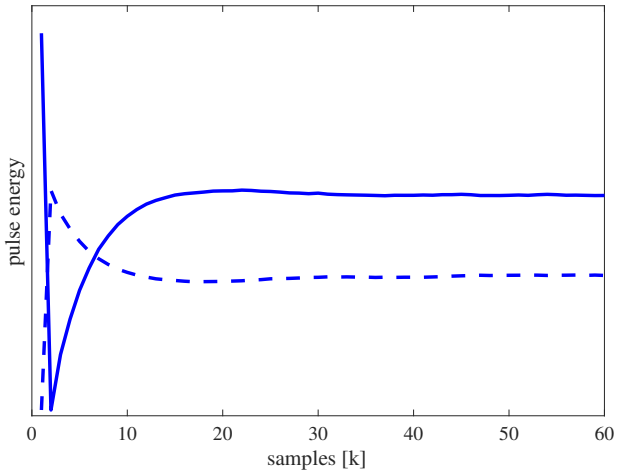

Figure 2.3: Data split for each of the SISO models. Step up data (dashed line), and step down data (solid line).

The overshoot and undershoot are undesirable, since it is required to control the EUV. If the LASER can't be controlled, then there is not a control knob. This is why the system characterization takes place in order to determine if there is a 'fixed' response.

2.2 Step Response Model Estimation

For the Two-Mass system and LASER pulse length experiments, we estimate a discrete-time model using the realization algorithm for step input experiments. The

same process is applied to obtain the model for the step up, step down and two-mass system data.

Defining R = H_Y - $\underline{T}_G H_U$, where H_Y is an N_1 x N_2 Hankel matrix with the outputs *y(t)*; $\underline{T}_G$ is an N_1 x N_1 + 1 Toeplitz matrix with Markov parameters *g(t)*; and H_U is an N_1 + 1 x N_2 Hankel matrix containing the input data *u(t)*, where *u(t)* represents the step input, thus H_U is a unity matrix. The product $\underline{T}_G H_U$ can be computed without knowledge of *g(t)* in $\underline{T}_G$. The singular value decomposition (SVD) of R can be computed to obtain

$$R = U\Sigma V^T = \begin{bmatrix} U_n & U_s \end{bmatrix} \begin{bmatrix} \Sigma_n & 0 \\ 0 & \Sigma_s \end{bmatrix} \begin{bmatrix} V_n^T & V_s^T \end{bmatrix}. \tag{2.1}$$

The SVD can be used to approximate the matrix R; R allows a decomposition $R = R_1 R_2$, where $R_1 = U_n {\Sigma_n}^{1/2}$ and $R_2 = {\Sigma_n}^{1/2} V_n^T$. The diagonal values of Σ can be used to determine the order of the model.

The shifted elementary data matrix (EDM) equation allows us to write $\overrightarrow{R}$ = $\overrightarrow{H}_Y$ - $\underline{\overrightarrow{T}}_G \overrightarrow{H}_U$. $\overrightarrow{R}$ helps us find A; $\overrightarrow{R} = R_1 A R_2$. We can also find matrices B = $R_2(:,1)$, C = $R_1(1,:)$ and D = $y(0)$ [3].

From the state-space representations, we obtain the transfer functions.

2.2.1 Realization Algorithm for Two-Mass System Experiment

The two-mass model is defined as 4^{th} order model, Figure 2.4.

The transfer function for it is:

$$G_{m1} = \frac{13.4458z^3 - 25.8688z^2 + 12.4532z + 0.2059}{z^4 - 3.8772z^3 + 5.6868z^2 - 3.7395z + 0.9301}. \tag{2.2}$$

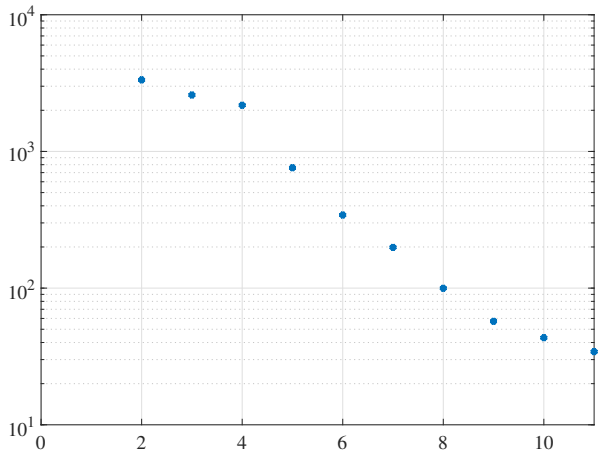

Figure 2.4: Hankel singular values for the two-mass system.

The model is stably invertible since the numerator roots are within the unit circle. In this case, we do not work this advantage in future sections.

The model response is simulated with a step input to compare its response with the measured step response data from the experiment. The comparison in Figure 2.5 verifies that the identified model is a reasonably good estimate.

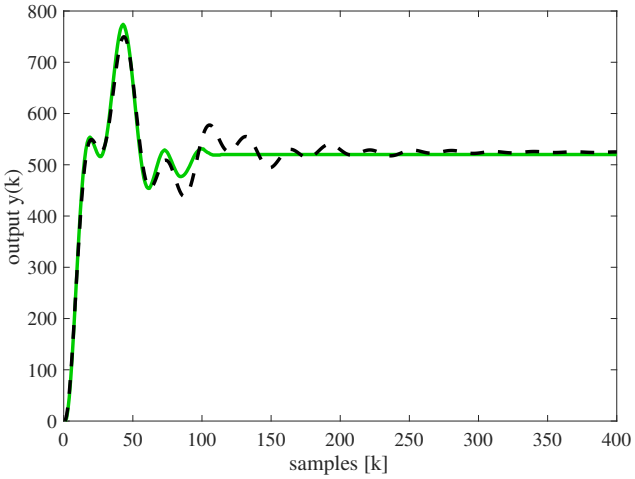

Figure 2.5: Measured response vs simulated response. Data from the two-mass system experiment (green solid line), and simulated response from the GRA model obtained (black dashed line).

Notice the stiction/friction effects in experimental data, but good coherence in oscil-

lation and damping modeling. The difference in oscillatory response at longer times is likely due to extraneous damping sources in the ECP apparatus.

2.2.2 Realization Algorithm for LASER Pulse Length Experiment

Both the step up and step down models are defined as 3rd order models, Figure 2.6.

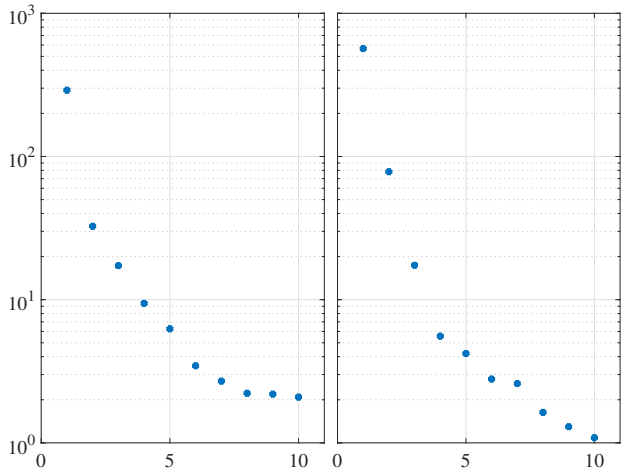

Figure 2.6: Hankel singular values for the LASER system. Left: Step up model. Right: Step down model.

The transfer function for the two models are:

$$G_{up} = \frac{0.3158z^2 - 0.5801z + 0.2665}{z^3 - 1.7313z^2 + 0.7293z + 0.0130}, \tag{2.3}$$

$$G_{down} = \frac{-6.5196z^2 + 12.2565z - 5.7570}{z^3 - 1.7374z^2 + 0.7348z + 0.0096}. \tag{2.4}$$

The models are stably invertible since the numerators roots are within the unit circle.

The obtained models of the systems are simulated with a step input to compare their response with the measured step response data from the experiment. The

comparison in Figure 2.7 verifies that the identified models are a good estimate.

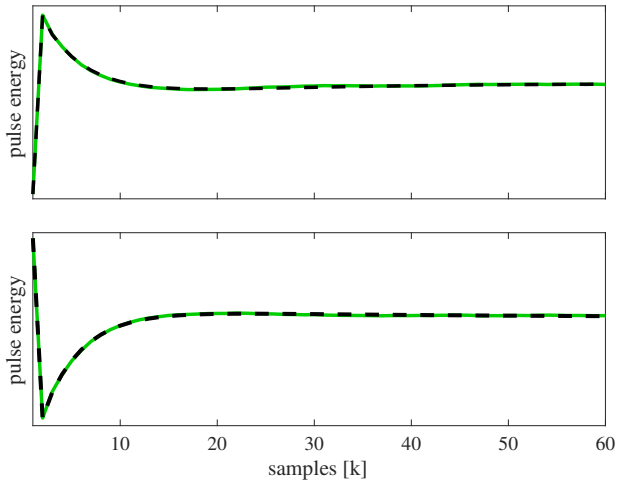

Figure 2.7: Measured response vs simulated response. Top: Step up model. Bottom: Step down model. In each plot: data from the LASER experiment (green solid line), and simulated response from the GRA models obtained (black dashed line).

Chapter 3

Input shaping Techniques

Figure 2.3 reveals that the response of the system has a positive overshoot when the step goes up, while the step down data has an even more pronounced undershoot. The ideal output would not have any overshoot or undershoot. The data suggests using a gradual step input which would control the overshoot and undershoot in the signals. In the following sections we review the possible input shaping techniques to create a desirable system response.

In input shaping, the original input (unmodified) signal is passed through an input shaper and this new shaped signal is fed to the system. The purpose of this new signal is to remove oscillation that can be caused by the unmodified input signal [12]. Input shapers used in the design of control systems helps to decrease the overshoot or undershoot and the time for a steady response in an oscillating systems [13], such as the LASER pulse experiment.

The feed-forward technique consists of a convolution of the input command with a series of impulses. The result is a modified input command that reduces the residual vibration in the response. To apply input shaping the amplitudes and timing

of such impulses have to be determined to reduce the residual vibration [4, 14].

3.1 Inverse Model Approach

For a given model, the most straightforward approach is check if the inverse is stable. The models for the step up and step down systems are stably invertible. We apply a step input signal the size of our output (once stable) to the inverse of the model, G^{-1}; this gives us the ideal input. Let's consider this step signal our first approach. By making a simulation of the model, G, with the ideal input, we are able to get the ideal output as shown in Figure 3.1.

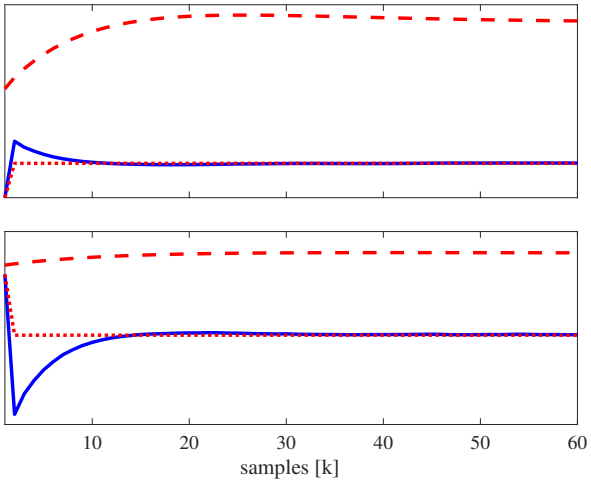

Figure 3.1: Inverse model approach. Top: Step up model. Bottom: Step down model. In each plot: data from the LASER experiment (blue solid line), ideal response (red dotted line), and computed input with G^{-1} (red dashed line).

In this case, the zeros are stable and we can perform a simple inversion on the obtained model. We can create the inverse of a model to compute the ideal input, but we do not have any assurance that the computed ideal input is feasible, since we are working without any input constraints. The hardware may not have enough

freedom to generate such an idealized input.

3.2 ZV and FIR filters

Two impulse responses can be superimposed such that the system moves forward without vibration after the input has ended. This is the two-impulse ZV input shaper. The amplitudes, A_i; and times, t_i; of the impulses are given by [15, 16]:

$$\begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} \frac{1}{1+K} & \frac{K}{1+K} \\ 0 & 0.5T_d \end{bmatrix}, i = 1, 2, \tag{3.1}$$

where

$$K = e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}}, \tag{3.2}$$

T_d is the damped period of vibration, and ζ is the damping ratio. The convolution is performed between the original input and the amplitude of the first and second impulse, shifted by one-half of the damp vibration period. A constraint for the amplitude A_i of the impulse has to be met, $A_i \geqslant 0$, i = 1,..,m, and $\sum_{i=0}^{m} A_i = 1$.

The impulse convolution with a step input signal might not be the ideal approach for some systems' behaviors. It creates smaller size steps until it reaches the full size step: a staircase command. Each impulse added will result in an output delay. Figure 3.2 shows a basic example of step command and impulse convolution as command shaping [17].

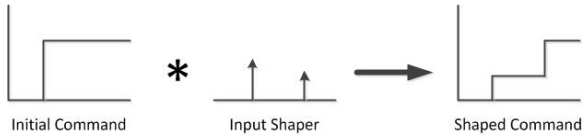

Figure 3.2: Input shaping a step to produce a staircase command.

In a different feed-forward approach, the input signal and FIR filters are convoluted; this type of input shaper adds zeros to the system [18].

FIR filters are withing the most common. FIR filter consist of nonzero impulse responses. The poles in the z-plane are only at z=0 and are represented by [6]:

$$H(z) = \sum_{n=0}^{M} b_n \cdot z^{-n}, \tag{3.3}$$

where b_n are parameters of the filter. Similar to the ZV conditions, the FIR filter gain shall be one.

3.3 Convex Optimization

A different class of optimization problems is linear programming, where the objective and all constraint functions are linear. The general formulation for this is [8]:

$$\begin{aligned} &\text{minimize} \quad c^T x \\ &\text{subject to} \quad a_i^T x \leq b_i, \quad i = 1, ..., m, \end{aligned} \tag{3.4}$$

where x is the vector of optimization variables and a_i, b_i and c are parameters and constraint functions.

We look to minimize the maximum error of our desired trajectory, y_{des},

$$E = max|y(t) - y_{des}(t)|. \tag{3.5}$$

The constraint we work with in this case is defining a range for the input, $U_{low} \leq u(t) \leq U_{high}$. The optimization problem is now expressed as [19]:

$$\begin{aligned} &\text{minimize} \quad E = max_{T=1,...,M}|y(t) - y_{des}(t)| \\ &\text{subject to} \quad U_{low} \leq u(t) \leq U_{high}, \quad t = 1, ..., N, \end{aligned} \tag{3.6}$$

where M is the output we have defined up to and *u(t)* is the optimization variable.

Having the z-Domain transfer functions in equations 2.3, 2.4 and 2.2, we can define the following equations for each system,

$$\begin{aligned} &G(q) = q^{-1}\frac{b_0 + b_1q^{-1} + ... + b_nq^{-n}}{1 + a_1q^{-1} + ... + a_nq^{-n}}, \\ &y(k) = G(q)u(k), \\ &y(k) + a_1y(k-1) + ... + a_ny(k-n) = \\ &b_0u(k-1) + b_1u(k-2) + ... + b_nu(k-n-1). \end{aligned} \tag{3.7}$$

The output *y(k)* can be written as,

$$\begin{aligned} y(k) &= b_0u(k-1) + b_1u(k-2) + ... + b_nu(k-n-1) \\ &- a_1y(k-1) - ... - a_ny(k-n), \end{aligned} \tag{3.8}$$

from which we can define y = Φ u, where Φ is a Toeplitz matrix using the elements of *y(k)*, impulse response [19].

$$A_{ij} = \begin{cases} h_{i-j} & if 0 \leq i - j \leq k, \\ 0 & otherwise \end{cases}$$

Developing the previous statement,

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ . \\ . \\ . \\ y(n-1) \\ y(n) \end{bmatrix} = \begin{bmatrix} h_0 & 0 & 0 & 0 & . & . & . & 0 \\ h_1 & h_0 & 0 & 0 & . & . & . & 0 \\ h_2 & h_1 & h_0 & 0 & . & . & . & 0 \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ 0 & ... & 0 & h_k & ... & h_2 & h_1 & h_0 \end{bmatrix} \begin{bmatrix} u(0) \\ u(1) \\ u(2) \\ . \\ . \\ . \\ u(n-1) \\ u(n) \end{bmatrix}.$$

We can now solve the optimal control problem using standard LP software, such as MATLAB's function 'linprog' [20].

Figure 3.3 schematically illustrates the two different amplitude constraints [21]. The first constraint during the initial response is a large one, due to the response initial oscillation. Once the response is settled, a second tight constraint is created. The maximum and minimum constraints are defined as $\overline{y}$ and $\underline{y}$, respectively.

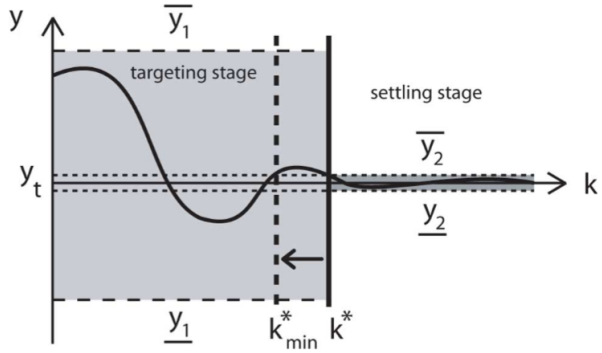

Figure 3.3: Definition of the output constraints.

k^* represents the samples to reach the steady state; if k^* is longer than the minimum necessary, the response will oscillate due to the freedom on time to reach the target.

In order to find the minimum sample number k^*_{min}, the bisection method is used.

The bisection method consists of defining the problem as feasible, and start with the range [k_{low}, k_{high}] which contains k^*. The convex feasibility problem it's solved at its midpoint t = (k_{low} + k_{high})/2, to determine whether the optimal value is in the lower or upper half of the range, and update the range based on the result. This produces a new range, which contains the optimal value k^*, but has half the width of the initial one. This is repeated until the interval is within a valid maximum value, [8].

3.4 Mixed Integer Linear Programming

Since we have emphasized on the restrictions the input may be subject to based on the hardware, we explore further techniques such as MILP. Integer linear programming investigates linear programming problems in which at least one of the variables is restricted to integers [9].

In linear programming the values obtained from a solution procedure may be real or an integer. The linear programming models shown in the previous systems have been continuous, meaning the variables are allowed to be real [10].

Solutions so far use full design freedom on the input, but in many practical situations only input signals where a particular constraint on the level/size can be applied.

In the LASER experiment, the lower bound is a restriction on the hardware used to actuate the pulse length. It can't go lower than the value used in the experiment. For the upper bound, the main reason is not because of what the hardware is physically capable of, but rather that the response curve flattens off. However, the

continuous values solution is not valid, since only integers are allowed as the pulse width. A valid approach could be rounding the values to the nearest integer, but this might affect the optimal solution. In those cases the MILP solution would be best [22].

$$\begin{aligned} &\text{minimize} \quad c^T x \\ &\text{subject to} \quad a_i^T x \leq b_i, \quad i = 1, ..., m, \\ &\text{where} \quad x \in \mathbb{Z}. \end{aligned} \tag{3.9}$$

Chapter 4

Application of Input Shaping Techniques

4.1 Application of ZV and FIR filters

4.1.1 Two-mass System experiment

Following the mentioned ZV and FIR filters input shaping techniques, we can see in Figure 4.1 an improvement by utilizing the ZV method in regards of the overshoot and oscillation, and does not add a delay.

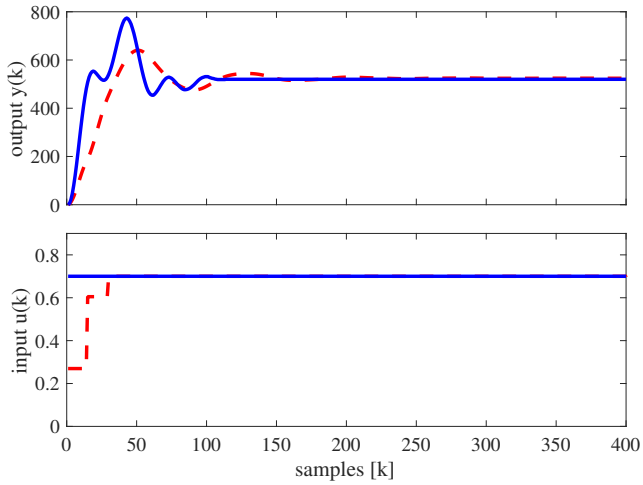

Figure 4.1: ZV for Two-mass system. Top: Optimization response. Bottom: Optimization input. In each plot: data from Two-mass experiment (solid blue line), and data from ZV (red dotted line).

Figure 4.2 shows there is no improvement for this system utilizing FIR filters. It is clear that ZV method has the best and fastest response in this case.

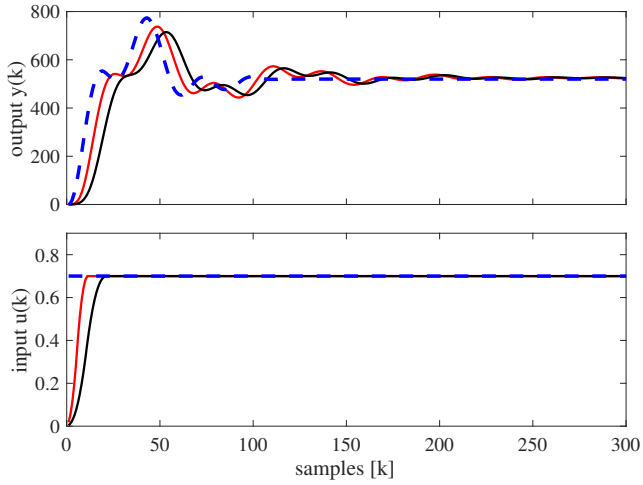

Figure 4.2: FIR filters for Two-mass system. Top: Optimization response. Bottom: Optimization input. In each plot: data from Two-mass experiment (solid blue line), 10^{th} order FIR filter (red line), and 20^{th} order FIR filter (black line).

Figure 4.3 shows a comparison between ZV ad the FIR filters.

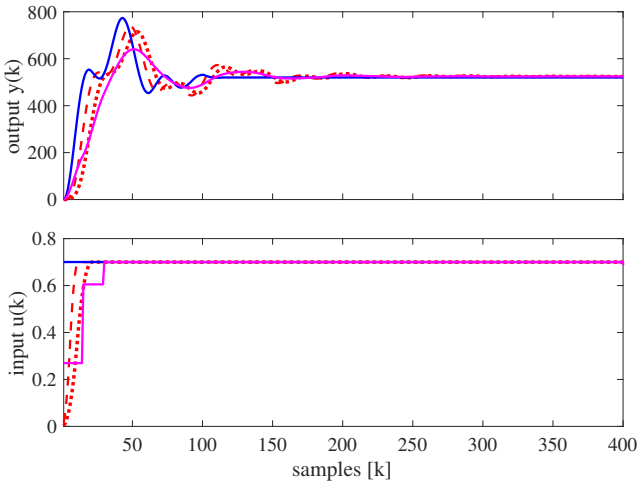

Figure 4.3: Two Mass System. In each plot: system simulation (solid blue line), data from ZV (magenta solid line), 10^{th} order FIR filter (red dashed line), and 20^{th} order FIR filter (red dotted line), optimization (green solid line).

4.1.2 LASER Pulse Length Experiment

ZV input shaping approach does not match the models G_{up} and G_{down} in an optimal way, since we are not looking to reduce system vibration. The input shaping based on impulse convolution is aimed to reduce the vibration on flexible systems. However, by applying a ZV input shaper to the previously mentioned systems, they show some improvement in smoothing the overshoot and undershoot in the response, Figure 4.4.

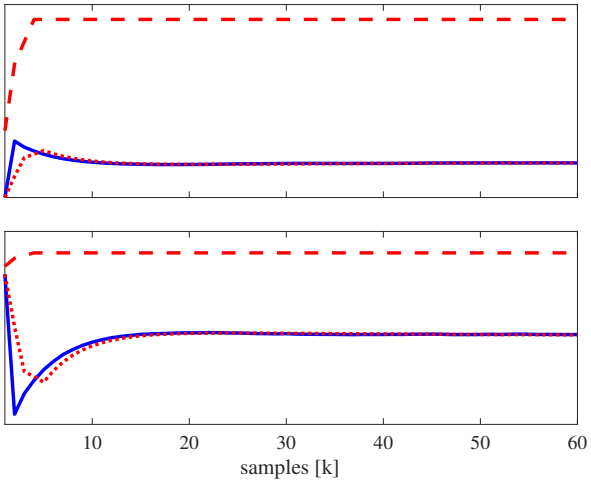

Figure 4.4: Computed ZV input and simulated output. Top: Step up model. Bottom: Step down model. In each plot: data from the LASER experiment (solid line), simulated response of the system to ZV input (dotted line), and ZV shaped input (dashed line).

The overshoot and undershoot in the response of the systems is reduced and the delay is negligible. The response matches the original behavior.

As a second approach, we apply triangular FIR filters of order of 10 and 20 to illustrate their behavior. Results are shown in Figure 4.5.

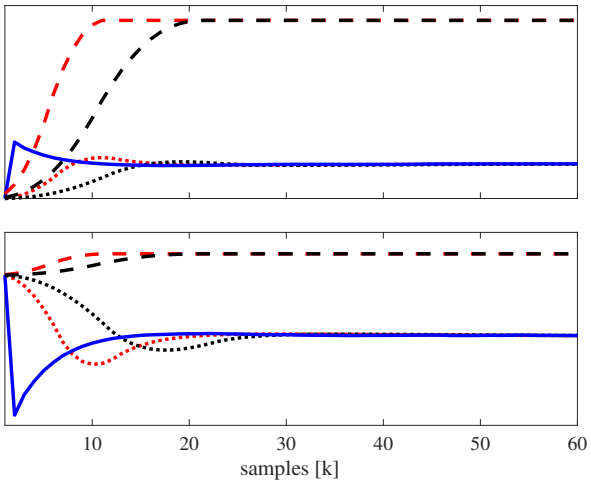

Figure 4.5: FIR filters. Top: Step up model. Bottom: Step down model. In each plot: data from the LASER experiment (solid line), response and input for the 10^{th} order (red dotted and dashed line), response and input for the 20^{th} order (black dotted and dashed line).

The overshoot and undershoot in the response, for both triangular FIRs, is clearly smoother, but the delay is proportional to the order of the filter.

Figure 4.6 and Figure 4.7 show a comparison of the system input and response, with the ZV input shaper and FIR filters.

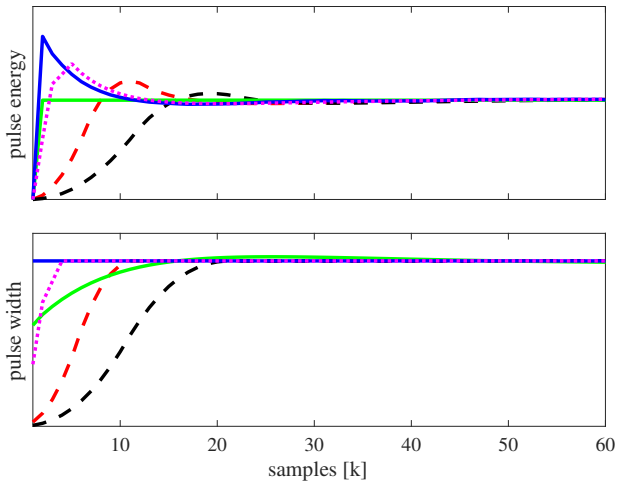

Figure 4.6: Step up model. In each plot: data from LASER experiment (solid blue line), data from G^{-1} (solid green line), data from ZV (magenta dotted line), 10^{th} order FIR filter (red dashed line), and 20^{th} order FIR filter (black dashed line).

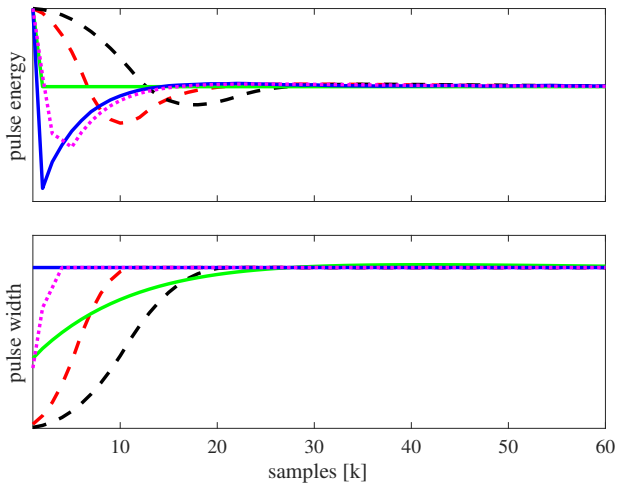

Figure 4.7: Step down model. In each plot: data from LASER experiment (solid blue line), data from G^{-1} (solid green line), data from ZV (magenta dotted line), 10^{th} order FIR filter (red dashed line), and 20^{th} order FIR filter (black dashed line).

4.2 Application of Convex Optimization

4.2.1 Two-Mass system Experiment

In this example, we are not considering having a desired output trajectory vector (e.g. y_des). We are only exercising an approach constraint on the output and a limit in the input. Figure 4.8 shows the optimization for the Two-mass system, compared with the data from the experiment. The overshoot is completely removed and the response is steady at around the half of the samples than the original experiment.

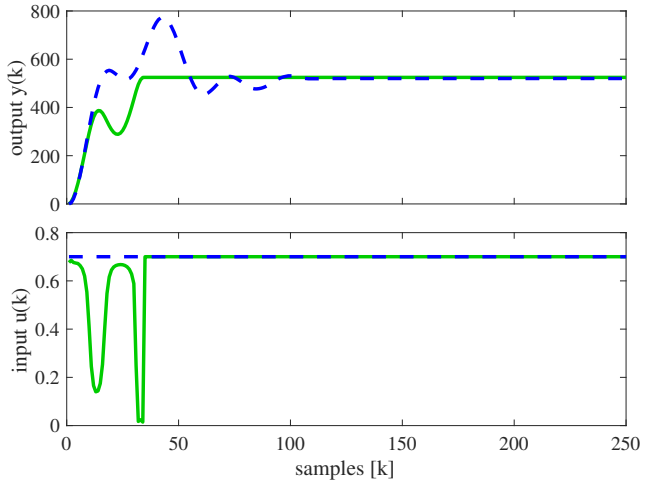

Figure 4.8: Optimization for Two-Mass system. Top: System responses. Bottom: System inputs. In each plot: data from Two-Mass experiment (blue dashed line), and optimization (green solid line).

4.2.2 LASER Pulse Length Experiment

From the inverse model, we have a desired output trajectory (in a vector), shown in Figure 3.1 as the ideal system response. We want to minimize the peak on the output behavior and have an output that matches the defined ideal system response.

Figure 4.9 and Figure 4.10 show the optimization for each model. We can clearly identify the peak on the input computed from the inverse model and the peak on the computed optimal input. The values are close, but with the optimization we can see the input constraint. Due to this constraint on the input value, we see a delay on the output reaching the ideal system response, but the difference is negligible.

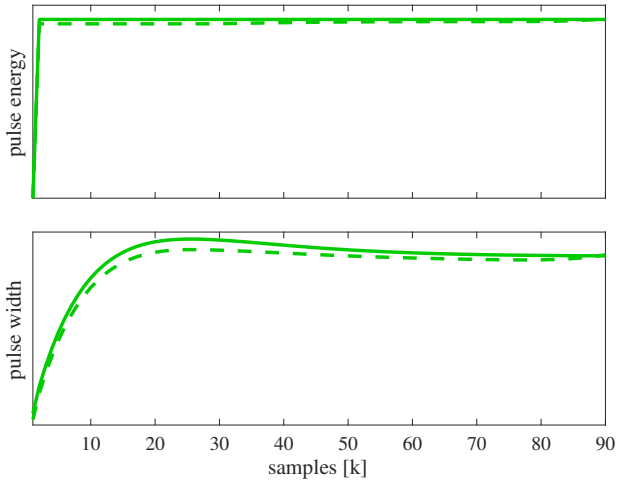

Figure 4.9: Optimization for step up model. Top: System responses. Bottom: System inputs. In each plot: desired data from inverse model (solid line), and optimization (dashed line).

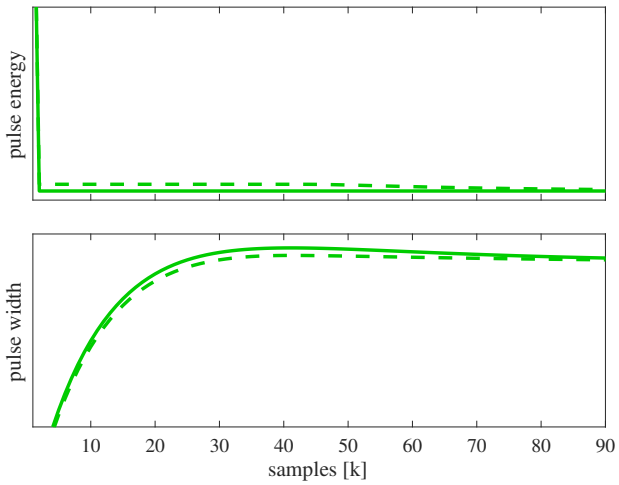

Figure 4.10: Optimization for step down model. Top: System responses. Bottom: System inputs. In each plot: desired data from inverse model (solid line), and optimization (dashed line).

Doing an approach where we don't have a desired output trajectory vector, we exercise a set of constraints on the output and limits in the input.

There are two important parameters to consider when defining output constraints: $\overline{y_1}$ and k^*.

First, if $\overline{y_1}$ is larger than necessary and k^* is too short, the amplitude of the response will increase in order to reach the target response within the time frame. Second, if k^* is too long, the response signal will not settle to the target value, but will oscillate about the target response until the k^* point is reached, this as a response to the oscillation of the input signal that will oscillate until k^* is reached. This is clearly reflected in Figure 4.11, exemplifying the step up model with an ideal k^*, and a k^* longer than required.

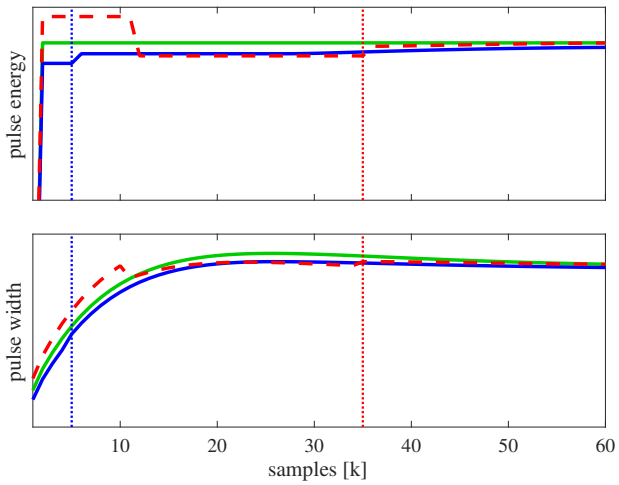

Figure 4.11: Optimization for system up model varying k^*. Top: System responses. Bottom: System inputs. In each plot: desired data from inverse model (green solid line), optimization with optimal k^* (blue solid line), and optimization with long k^* (red dashed line).

4.3 A different "ideal input" for the LASER pulse Experiment

As mentioned in Chapter 3, Figure 3.1, we computed an ideal input with the use of the model inverse, based on the ideal response. A new smoother approach as ideal response will be defined for each system. The new ideal inputs/outputs are shown in Figure 4.12.

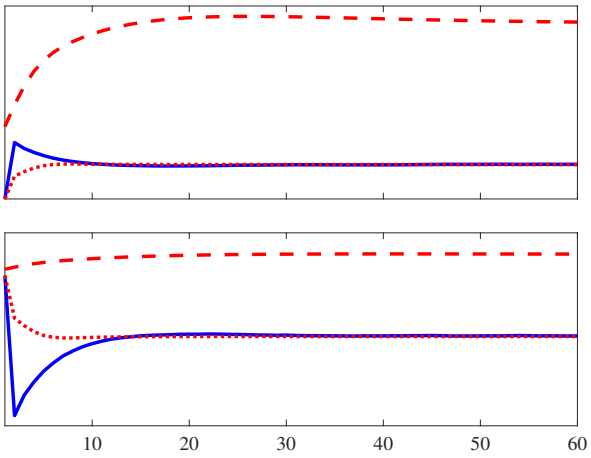

Figure 4.12: Computed Input from G^{-1} based on a new ideal system response.

Going again trough the optimization process with these new signals, Figure 4.13 and Figure 4.14 show the results. Since the changes on the ideal inputs were not significant, only smoother, the optimization results are similar. However, this new input might be a more feasible approach for the existing firmware.

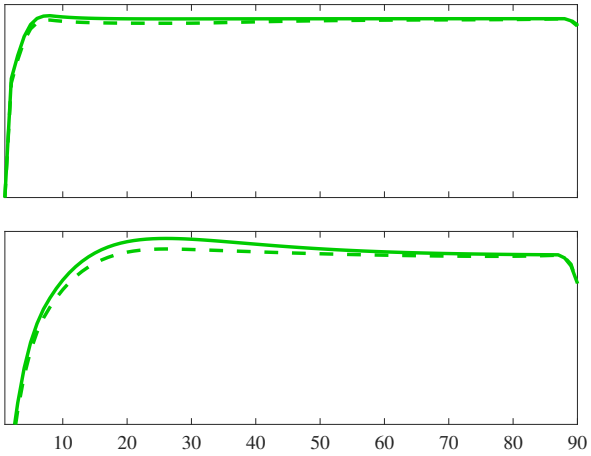

Figure 4.13: Optimization for system up model.

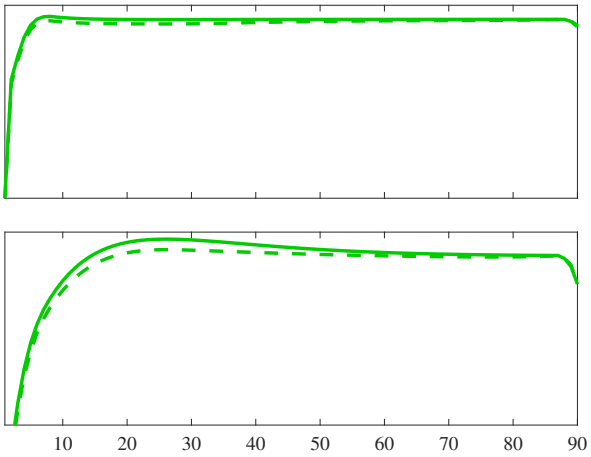

Figure 4.14: Optimization for system down model.

4.4 MILP and Fractional MILP

It might not be physically possible for a particular plant to generate the input from convex optimization. Following the same approach as in convex optimization, we exercise a MILP approach.

We can see the results for each are very similar to those in the convex optimization shown in Figure 4.8 and Figure 4.9. The inputs have a very similar path to the ones in convex optimization, but as in staircase due to the fractional/integer

limits.

4.4.1 Two-mass System experiment

The values are based in a MILP approach, but restricted to integers. For the two-mass system, we use the fractional MILP, since the maximum value is approx. 0.7072 and it is not possible to create an integer restricted input, as there is no range of integers in the input values.

To solve this, we set the input *u(k)* multiplied by a factor, defined as x_z, to create a range of integers. At the end, a filter $\frac{1}{x_z}$ is applied before the plant G_{m1}. The factor x_z is 40, giving an input range of *u(k)* = {0, 1, 2, ..., 28}. The obtained inputs passes through a $\frac{1}{x_z}$ filter, thus the real input is *u(k)* = {0, 1/40, 2/40, ..., 28/40}.

Figure 4.15 shows the results of the fractional MILP technique applied to the two-mass system.

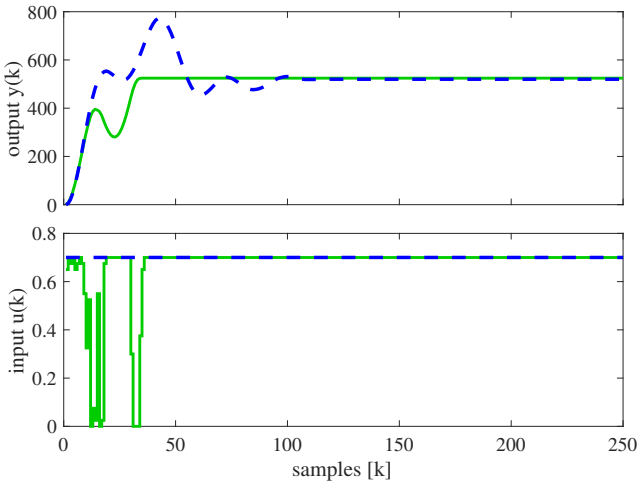

Figure 4.15: Computed fractional MILP and simulated response for two-mass system. Top: System responses. Bottom: System inputs. In each plot: data from two-mass experiment (blue dashed line), and MILP (solid green line)

4.4.2 LASER pulse length experiment

For the LASER system, we use the MILP approach directly, since input *u(k)* allows a wide range of integers. Figure 4.16 shows the results of MILP technique applied to the LASER experiment.

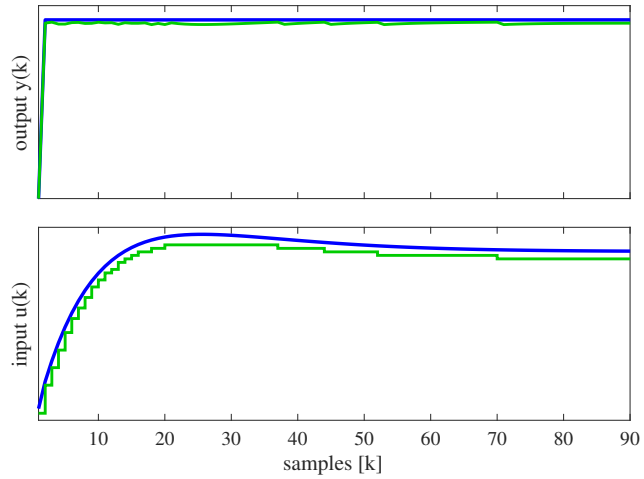

Figure 4.16: Computed MILP input and simulated response for step up model. Top: System responses. Bottom: System inputs. In each plot: desired data from inverse model (blue solid line), and MILP (green solid line)

This method would the ideal for the LASER experiment since input, *u(k)*, is restricted to integers.

4.5 The Two-mass system equations derived from the dynamic system

The derivation of the equations for the two-mass system is a more clear way to show some results. Though the model is only ideal and does not reflect the real behavior, it is a way to make a clean and clear example of convex optimization, MILP

and fractional MILP. Back to the initial system on Figure 1.1 we derive the equations [11].

The derivation of the equations of motion along with a Laplace transform, results in a transfer function model *G(s)* with

$$y(s) = G(s)u(s), \tag{4.1}$$

that relates the Laplace transformed input force *u(s)* and output signal $y(s) = x_1(s)$ or $x_2(s)$. In the case where the output is chosen as $y(s) = x_1(s)$, the transfer function $G_1(s)$ is given by

$$G_1(s) = \frac{b_2 s^2 + b_1 s + b_0}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}, \tag{4.2}$$

whereas for the output $y(s) = x_2(s)$, the transfer function $G_2(s)$ is given by

$$G_2(s) = \frac{b_0}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}, \tag{4.3}$$

where the coefficients b_i, $i = 0, 1, 2$ and a_j are determined by

$$\begin{aligned} b_2 &= m_2, \\ b_1 &= d_2, \\ b_0 &= k_2, \\ a_4 &= m_1 m_2, \\ a_3 &= m_1 d_2 + m_2 d_1, \\ a_2 &= k_2 m_1 + (k_1 + k_2) m_2 + d_1 d_2, \\ a_1 &= (k_1 + k_2) d_2 + k_2 d_1, \\ a_0 &= k_1 k_2. \end{aligned} \tag{4.4}$$

The step-response for $G_1(s)$ is shown in Figure 4.17. We can see the oscillation in the response and then a settling in the later samples.

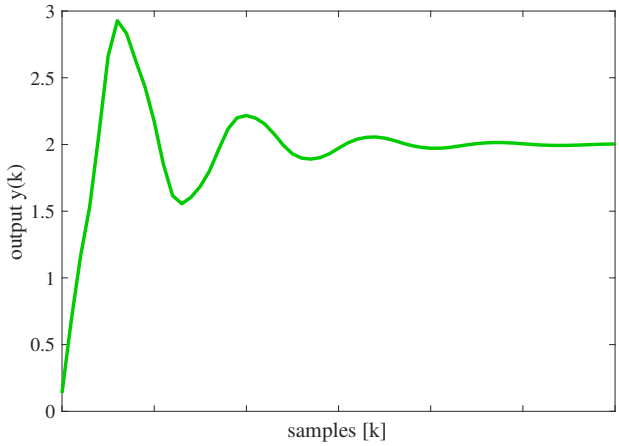

Figure 4.17: Step-response for the Two-Mass System.

The convex optimization for the two-mass system shows a reduction in the oscillation from the response in a short time, Figure 4.18.

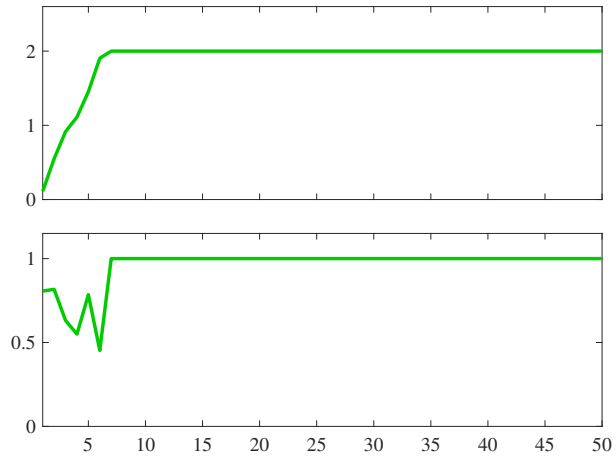

Figure 4.18: MILP for Two-mass system. Top: Optimization response. Bottom: Optimization input.

As a first test, we limit the input values to integers. Figure 4.19 shows the MILP response for the Two-Mass System. In this case the example is of binary type, since it is simulating a step input and the allowed values are 0 and 1.

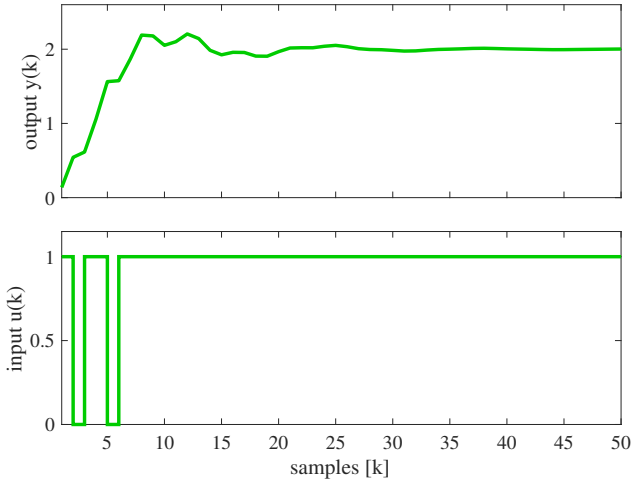

Figure 4.19: MILP for Two-mass system. Top: Optimization response. Bottom: Optimization input.

To smooth the MILP response, we will use a 'fractional' MILP approach. The values will be based in a MILP approach, but restricted to $u(k) = \{0, 0.2, 0.4, 0.6, 0.8, 1\}$. This is achieved by setting an input from $u(k) = \{0, 1, 2, 3, 4, 5\}$ and having

a $\frac{1}{5}$ filter before the system G.

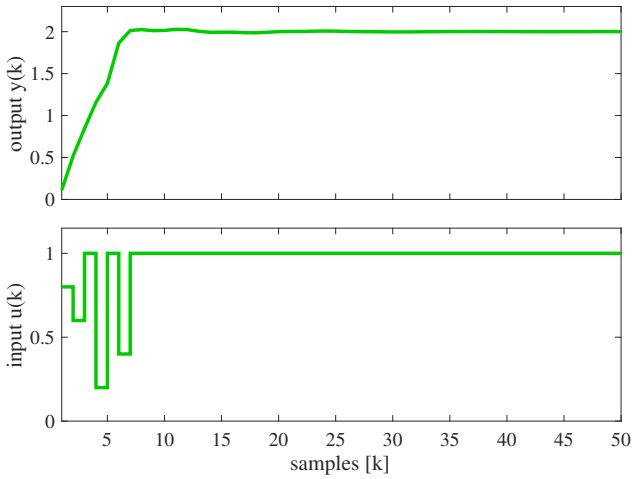

Figure 4.20: Fractional MILP for Two-mass system. Top: Optimization response. Bottom: Optimization input.

The response is better since the input is not of binary type, Figure 4.20. There is a smoother response from the initial shown in Figure 4.19. The response from fractional MILP is similar to the convex optimization response, with the advantage that the input values are restricted.

Chapter 5

Experimental Verification

We applied the diverse input shaping techniques, such as inverse model, ZV, FIR filters and convex optimization, for the two-mass and LASER systems. In general the conclusion is that the inverse model is the best approach, if you can perform it. When it is not possible, convex optimization is the best technique.

In order to test the convex optimization technique, we utilized the ECP 210 from which the data from the original experiment was obtained. The MILP settling time is fast and the control signals are fairly aggressive, so we decided not to test MILP. In order to find a good compromise, we test the convex optimization input.

The obtained input in Figure 4.8 is loaded as a trajectory (trj) file in the ECP. The results are shown in Figure 5.1. The response accurately tracks the prediction.

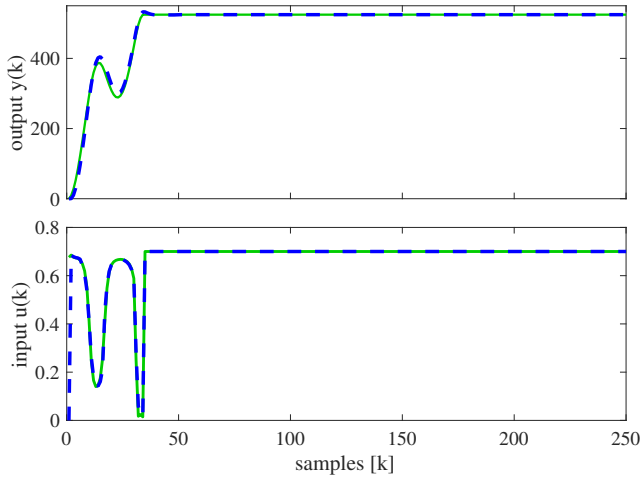

Figure 5.1: Computed optimization input and simulated response for two-mass system. Top: System responses. Bottom: System inputs. In each plot: calculated optimization data (green solid line), and ECP test obtained data (blue dashed line).

Chapter 6

Conclusions

We have shown different possibilities available when analyzing the input-output data result of a step experiment.

1. Step input experiment and data collection. For the two experiments shown in this thesis, the two-mass system and the LASER pulse length, we have a step input and its response.

2. System identification process. In this case, we are working with data from step input experiments, thus the use of the realization algorithm for step input experiments.

3. Identify the best response based on different input approaches (ZV, filter, optimization, etc.). We are analyzing the ZV option, which gives a great improvement. The FIR filters provide a good response for the LASER system, at the cost of delay in the response. This might be negligible for some systems, but in most cases time is a factor to consider. Inverse model and convex optimization give a combination of an ideal result in the best time possible.

4. Evaluate if the system can perform the required input for the given best response. We have to consider hardware restrictions, since it might not be capable of setting the input values, e.g. the MILP option is a solution for the hardware restrictions.

The real definitions for each experiment are: a) the freedom the system allows in an input rate change, and b) the constraints in the response (e.g. overshoot) and input limits. If the system cannot perform the required input, Steps 3 and 4 iterate until the best response is obtained.

This work has presented a method for modeling a system as well as available options for achieving a desirable response. The initial delay and response definitions are the constraints: the overshoot/undershoot allowed as an initial response and the time to have a steady response.

We can see convex optimization as the best option for any system, since we can have a restricted input and response within the optimal time.

Bibliography

[1] L. Ljung, "System identification: Theory for the user," Prentice-Hall, 1999.

[2] R. A. de Callafon, D. N. Miller, J. Zeng and M.J. Brenner, "Step-Based Experiment Design and System Identification for Aeroelastic Dynamic Modeling," AIAA, Chicago, Illinois, August 2009.

[3] R. A. de Callafon and D. N. Miller, "Identification of Linear Time-Invariant Systems via Constrained Step-Based Realization," The International Federation of Automatic Control, Brussels, Belgium, July 2012.

[4] N.C. Singer and W. P. Seering, "Preshaping Command Inputs to Reduce System Vibration," J. Dyn. Sys., Meas and Ctlr., vol. 112, no. 1: 76-82, March 1990.

[5] T. Singh and W. Singhose, "Tutorial on Input Shaping/Time Delay Control of Maneuvering Flexible Structures," American Control Conference, Anchorage, Alaska, May 2002.

[6] M. K. Jouaneh and E. Anderson, "Input Shaping Using Finite Impulse Response Filters," Proceedings of the 45$^{\text{th}}$ IEEE Conference on Decision and Control, December 2016.

[7] M.O.T. Cole and T. Wongratanaphisan, "Optimal FIR Input Shaper Designs for Motion Control With Zero Residual Vibration," Journal of Dynamic Systems, Measurement, and Control, Vol. 133, Issue 2, March 2011.

[8] S. Boyd and L Vandenberghe, "Convex Optimization," Cambridge University Press, 2004.

[9] A. Schrijver, "Theory of Linear and Integer Programming," John Wiley & Sons, 1986.

[10] P. V. Ubale, "An approach for the optimal solution of MILP problems," JBulletin of Pure and Applied Sciences, Volume 31 E, Issue No. 2: 169-173, 2012.

[11] T. H. Fay and S. D. Graham, "Coupled spring equations," International Journal of Mathematical Education in Science and Technology, vol. 34, no. 1, 6579, 2003.

[12] N. Singer, W. Singhose and E. Kriikku, "An input shaping controller enabling cranes to move without sway," American Nuclear Society 7th Topical Meeting on Robotics and Remote Systems, volume 1, 1997.

[13] M. A. Lau and Lucy Y. Pao, "Input shaping and time-optimal control of flexible structures," Automatica 39, pp 893 900, 2003.

[14] L. Y. Pao, "Strategies for Shaping Commands in the Control of Flexible Structures," Proc. Japan-USA-Vietnam Workshop on Research and Education in Systems, Computation, and Ctrl. Eng., June 2000.

[15] W. Singhose, R. Eloundou and J. Lawrence, "Command Generation for Flexible Systems by Input Shaping and Command Smoothing," Journal of Guidance, Control and Dynamics, Vol. 33, November 2010.

[16] M. Gniadek and S. Brock, "Basic algorithms of input shaping autotuning," MM Science Journal: 627-630, October 2015.

[17] W.E. Singose, "Command Generation for Flexible System," MIT, May 1997.

[18] M. D. Baumgart and L. Y. Pao, "Discrete time-optimal command shaping," Automatica, vol. 43, no.8: 1403-1409, June 2007.

[19] S. Boyd, C. Crusius and A. Hansson, "Control Applications of Nonlinear Convex Programming," Journal of Process Control, 1997.

[20] W. Durham, K. A. Bordignon and R. Beck, "Aircraft Control Allocation," Wiley, 2017, Appendix A: Linear Programming.

[21] U. Boettcher, R. A. De Callafon and F. E. Talke, "Close-Loop Input Shaping in Discrete-Time LTI System," ASME Information Storage and Processing Systems Conference, June 2010.

[22] G. Cornejulos, "Valid inequalities for mixed integer linear programs," Mathematical Programming B, Vol. 112, pp. 344, 2008.