Self-tuning control for polynomial systems: an algorithmic perspective

Iven M. Y. Mareels
Department of Electrical and Electronic Engineering
The University of Melbourne
Melbourne, Vic 3010
Australia
e-mail: i.mareels@unimelb.edu.au

Abstract

A two stage adaptive or self-tuning control algorithm applicable to systems described by transition maps that are polynomial in state, input and parameter variables is discussed. The feedback is defined only on the basis of past input and output measurements. In a first finite time stage the system to be controlled is identified together with its state trajectory. In a second stage a local observer, is used in conjunction with a receding horizon control scheme to effectuate the control objective. We discuss briefly the computational complexity aspects of this approach to adaptive or self-tuning control.

1 Introduction

It is common in mass manufactured products that certain characteristics vary from product to product. For example, the resonance frequency of a piezo-electric actuated read/write head for use in a hard disk drive varies by as much as 20% over a whole batch. Other examples may be derived from automotive engine systems. Control algorithms used in such environments are either designed based on a robustness or a self-tuning philosophy. In the robust methodology, the control is designed in such a way as to allow for the variation in system properties by rendering the desired performance objective insensitive to the expected variation. In the self-tuning framework, the control algorithm is tuned to the particulars of the plant on the basis of a measurement period, prior to its actual deployment.

In this paper we consider a self-tuning control approach applicable to systems modelled (but perhaps not governed) by difference equations of the following form:

$$\begin{aligned} x(t+1) &= f(\theta, x(t), u(t)), \\ y(t) &= h(\theta, x(t), u(t)). \end{aligned} \tag{1.1}$$

Here x is the state, u the input, y the output variable. It is assumed that f and h are polynomial in their arguments. The parameter θ represents the possible variation in system behaviour. The input and output are measurable, but the state and the parameter are not directable measurable.

A self-tuning strategy would proceed as follows. In a first stage observe the output $y(1), \cdots, y(T)$ as produced by a 'probing' input $u(1), \cdots, u(T)$ over a window of observation of length T. For sufficiently large T, under conditions that are generically statisfied by polynomial maps, it is possible to recover the state trajectory $x(1), \cdots, x(T)$ and the parameter θ. In a second phase this information can then be exploited to tune a control law. Finally, once the control law is implemented, the behaviour could be continually monitored to verify if it achieves and maintains the desired performance.

In this context, let us observe that in practice the integration of diagnostics with control is typically one of the strongest selling points on which modern control technology can be introduced.

The ideas in the paper are a natural sequel to [1]. In that paper we essentially discussed the first phase only. The combination of finite time observers with control ideas is also discussed in [3]. The latter is developed in continuous time setting, for which no tight computational complexity bounds are available to date. In our setting the theory expounded in [2] enables a conservative, yet effective treatment of computational complexity. Nevertheless this paper is focused primarily on the algorithm itself, robustness and computational complexity (which are very much related through condition numbers [2]) are discussed elsewhere.

Our setting is such that the distinction between an 'adaptive' and a 'non-adaptive' problem is in a natural way artificial. Indeed, we may rewrite (1.1) as follows

$$\begin{aligned} x(t+1) &= f(\theta(t), x(t), u(t)), \\ \theta(t+1) &= \theta(t), \\ y(t) &= h(\theta(t), x(t), u(t)). \end{aligned} \tag{1.2}$$

which allows a 'parameter free' interpretation. Obviously the 'θ' partial state is not controllable, but this is hardly an issue in a control setting, as the control objective can be formulated on the basis of the controllable states alone.

The important distinction between the problems of this kind is thus to be seen in the nature of the functions f and h and the number of variables we are dealing with. This computational complexity distinction is pursued here.

The paper is organized as follows. In Section 2 we describe an algorithm for the tuning and control phase. In section 3 this is more precisely worked out based on a global Newton algorithm for locating zeroes of multidimensional polynomial equations. Next we consider the computational complexity cost associated with the algorithm, using the theory from [2]. In particular we focus our attention on the difference between a so-called adaptive and a non-adaptive problem. Then we illustrate the algorithm using an example based on controlling a system defined based on the Hénon map. This allows us to probe robustness issues. We conclude by indicating where further development is required.

2 An in principle algorithm

Denote by $x_{[i,j]}$ the sequence $x(i), x(i+1), \cdots, x(j)$. Consider a dynamical system, a model for the measurements, of the form

$$\begin{aligned} x(t+1) &= f(x(t), u(t)), \\ y(t) &= h(x(t), u(t). \end{aligned} \tag{2.3}$$

Assume that f, h are polynomials in all their arguments.

Furthermore, define the positive, scalar valued functions

$$\begin{aligned} R(x_{[1,T]}, u_{[1,T]}, y_{[1,T]}) = &\sum_{i=1}^{T-1} \|x(i+1) - f(x(i), u(i))\|_2^2 \\ &+ r \sum_{i=1}^{T} \|y(i) - h(x(i), u(i))\|_2^2, \end{aligned} \tag{2.4}$$

and

$$\begin{aligned} C(x_{[\ell,\ell+H]}, u_{[\ell,\ell+H-1]}) = &\sum_{i=\ell}^{\ell+H} \|Ax(i)\|_2^2 \\ &+ c \sum_{i=\ell}^{\ell+H-1} \|u(i)\|_2^2. \end{aligned} \tag{2.5}$$

Here r and c are positive scalars and A is a matrix selecting a linear combination of the state x of interest for control. The function R is used as a criterion indicating the quality of reconstruction of a state trajectory given the input sequence $u_{[1,T]}$ and measurements of an output sequence $y_{[1,T]}$ over an observation horizon T. Similarly the function C is used as a criterion indicating the cost of regulating the partial state Ax to zero over a control horizon of length H. The regulation task, i.e. to achieve $Ax(\ell + H + 1) = 0$ is the ultimate goal, C measures the cost of doing so.

The problem is to construct an input sequence $u(1), u(2), \cdots$ such as to regulate the partial state Ax to zero, using minimal effort. The input at time t, $u(t)$ may only depend on past inputs and outputs. Rather than trying to make the problem statement more precise, we present an in principle algorithm. The algorithm makes clear what the aim is.

Observer/control algorithm outline

Step 1 Select an integer $T > \lceil \dim(x)/\dim(y) \rceil$. Select an integer $H > \lceil \dim(x)/\dim(u) \rceil$.

Step 2 Initialise $t = 1$.

Step 3 Select an input sequence $u(t), \cdots, u(t+T-1)$ apply it to the system (1.1) and observe the output sequence $y(t), \cdots, y(t+T-1)$.

Step 4 For the given input and output sequence minimize $R(x_{[t,t+T-1]}, u_{[t,t+T-1]}, y_{[t,t+T-1]})$. The minimizer is denoted $\hat{x}_{[t,T+t-1]}$. It is the estimate for the state trajectory.

Step 5 Given $\hat{x}(t+T-1)$, consider the minimization of $C(x_{[t+T,T+H]}, u_{[t+T,T+H-1)})$ under the constraint $x(k+1) = f(x(k), u(k))$ for $k = t+T-1, \cdots, t+T+H-1$ with initial condition $x(t+T-1) = \hat{x}(T+t-1)$ and terminal constraint $Ax(t+1+H) = 0$. Denote any global minimizer as $u^*_{[t+T,t+T+H-1]}$. Apply the input $u(t+T) = u^*(t+T)$ (if multiple global minimiser co-exist, select the one with the smallest norm). Observe $y(t+T)$.

Step 6 Set $t = t+1$, go to Step 4.

Some remarks are in order:

- Each stage in the algorithm is in principle feasible under generic conditions. However, whether or not the in-principle algorithm will remain feasible indefinitely, or better will provide a good control response when indefinitely iterated is not at all clear. This is directly linked to the feasibility of the control task, which will require assumptions on the dynamics linked with the criteria. If the control objective is feasible, the algorithm will most likely provide an acceptable solution. To highlight but one condition linking dynamics with the criteria, observe that for example boundedness of the solutions may not be guaranteed, in view of the fact that only part of the state is penalized. Boundedness requires observability of the state from the Ax output.

- The definitions of the criteria R and C reflect particular choices. For example the reconstruction criterion views the model as approximate in both state transition as well as output equation. It does not impose a strict transition according to $x(t+1) = f(x(t), u(t))$, it only tries to find a sequence of $x(t)$ which approximates this transition as close as possible in a least square sense. A similar observation holds for the output equation contribution to the criterion. Alternatively, if we had more faith in our model, R could have been defined as $R^{'}(x_{[1,T]}, u_{[1,T]}, y_{[1,T]}) = \sum_{i=1}^{i=T} \|y(i) - h(x(i), u(i))\|_2^2$. The reconstruction would then be based on minimizing $R^{'}$ under the constraint that $x(t+1) = f(x(t), u(t))$ over the entire reconstruction horizon. This is not pursued here.

- Similarly the control criterion with terminal constraint makes minimal use of prior knowledge. In particular, it does not make use of knowledge about possible equilibria. Such information could be used to advantage to ensure that regulation was exactly achieved.

- The control input that minimises C, even with the terminal constraint, will in general not lead to an input that will achieve regulation exactly, but only approximately. Typically, the smaller $c > 0$, i.e. cheap control, the better the regulation will be approximated.

- The selection of the observation horizon and the control horizon is affected by a number of considerations. The suggestions in the above algorithm are indicative of what is (under generic conditions) minimally required to be able to reconstruct the state trajectory or to be able to achieve (near dead beat) regulation (for the model). In the presence of disturbances, it is clear that a longer observation horizon provides the ability to average out disturbances. Equally a longer control horizon typically allows us to achieve regulation with less effort, but this must be tempered by the fact that longer prediction horizon lead to less certainty and thus may require more conservative control. Also, it is natural to expect that the computational complexity of finding the state estimate and control input grows with increasing T and H.

- The initial input sequence $u(1), \cdots, u(T)$ should be selected to assist the unique reconstruction of the state trajectory from the output measurements. In the context of polynomial systems this reconstruction task is generically feasible [5]. Given f, h it is conceivable to optimize the input sequence as to ensure that the minimization of $R(x_{[1,T]}, u_{[1,T]}, y_{[1,T]})$ yields a unique minimum, or failing this a well defined global minimum, well separated from local minima. However, this input selection task is not well posed, as it depends on the unknown initial condition $x(1)$. This is the inevitable difficulty in any identification problem. In order to proceed, some assumption on the distribution of the initial condition may be imposed, allowing one to formulate an optimization task like the expected effort to reconstruct, where the expectation is over the initial condition's distribution. This is not pursued here. Furthermore this initial input sequence should not unduly affect the control task. A larger state may be beneficial for identification, but will be penalised by the control criterion. This difficult, dual control problem is not addressed here.

- In the case of no measurement errors or any disturbances in the state transition map, the minimiser to be obtained in Step 4 is of course the ideal state trajectory, which zeroes R. In the presence of disturbances, the best achievable situation is to find a global minimum of R, which should be a good approximation of the ideal trajectory in the presence of small disturbances, for the algorithm to make sense.

- As in the identification task, the minimization of the control criterion C need not yield a unique (global) minimum. If regulation is to be achieved, the system model needs to satisfy the following condition $0 = f(0, u)$ has at least one real solution. Given that this is a polynomial in u multiple solutions are possible.

- The control problem in Step 5 is well posed, as the dynamics are entirely defined, as well as the initial condition. In principle it can be solved using a dynamic programming approach through the cost-to-go functional (a different method is suggested in the sequel). The answer that will be obtained would be the ideal control input provided the initial condition estimate $\hat{x}(T+t-1)$ were accurate. Any discrepancy in the initial condition results in a suboptimal input sequence. Given that initial condition errors may be amplified along the state trajectory, it makes little sense to select H much larger than what is required to steer the state to the origin. The trade-off is that shorter H may require more control effort than longer horizons. This can be investigated by analyzing the value function (or cost-to-go from the dynamic programming approach) as a function of the horizon H.

- In the above control algorithm there is a distinct difference between finding the minimisers for the first time, and for subsequent times. The first time either the reconstruction or the control law is computed, we are effectively without a reasonable initial guess. The search for a global minimum is thus truly a global search. However, every subsequent time significant computational savings can be realized as the previous minimizers provide very good initial conditions for the new minimization task. This will be exploited.

- The in principle algorithm distinguishes two phases. In the first stage, the initial state trajectory is recovered from some input and output measurements. In the subsequent stage, this state trajectory is used as a seed for control implementation. The process is repeated, with the state reconstruction stage and the control stage repeated consecutively shifted by one sample period, maintaining the same reconstruction and control horizons. As in the first phase the plant is effectively in open loop, it must be assumed that the open loop behaviour (for the particular input sequence) is acceptable over an horizon of length T. Such an assumption is inevitable in the context of self-tuning control.

- Alternative algorithms based on for example a parameterized control law of the form $u(t) = g(x(1), y(t-1), \cdots, y(t-m))$ (remember x can play the role of θ) are not pursued here.

3 Adaptive control algorithm

The minimization stages from the in principle algorithm of the previous section are now worked out in some more detail. A purely numerical approach is pursued.

We are guided by the following two observations:

- Observe that the in principle algorithm requires the minimization of two polynomial criteria. This minimization can be reduced to solving a set of polynomial equations, to identify the critical points, followed by verification of the criterion's value at the critical points. Solving a set of polynomial equations can be effectively executed using a global Newton algorithm [2].

- Next, and as indicated before, we distinguish the first from the subsequent executions for the minimization tasks. The main difference between the first and the subsequent execution of the minimization is in the starting points for the global Newton algorithm, and the particular form the homotopy takes. Both are adapted as to exploit the prior solutions maximally, without sacrificing the ability of coping with large changes, perhaps due to disturbances.

3.1 Global Newton algorithm for reconstruction

3.1.1 Phase I, $t = 1$

In the first phase, $t = 1$, the search is for the trajectory $x_{[1,T]}$ that minimizes $R(x_{[1,T]}, u_{[1,T]}, y_{[1,T]})$ given the input and output sequences $u_{[1,T]}, y_{[1,T]}$. This is achieved as follows.

Select an initial condition $x^o(1)$ and define an initial guess over the horizon $t = 1, \cdots, T$ through the iteration:

$$x^o(t+1) = f(x^o(t), u(t));\ \ y^o(t) = h(x^o(t), u(t)). \tag{3.6}$$

Further, define

$$H(x_{[1,T]}) = D_1 R(x_{[1,T]}, u_{[1,T]}, y_{[1,T]}), \tag{3.7}$$

and for $s \in [0, 1]$

$$H_s(x_{[1,T]}) = H(x_{[1,T]}) - sH(x^o_{[1,T]}). \tag{3.8}$$

Let $s_k = 1 - k/N$, for some integers $k \leq N$ and $N > 0$. Observe that for $s = 1$, the initial guess $x^o_{[1,T]}$ is an exact zero of $H_1(x_{[1,T]})$.

Consider the iteration for $k = 0, 1, \cdots, N$

$$x^{k+1}_{[1,T]} = x^k_{[1,T]} - \left(DH(x^k_{[1,T]})\right)^{-1} H_{s_k}(x^k_{[1,T]}). \tag{3.9}$$

For a generic choice of $x^o(1)$, there exists a finite N, such that $x^N_{[1,T]}$ is an approximate zero[1] of $H(x_{[1,T]})$. Denote this estimate as $x^*_{[1,T]}$. The iteration number N is of the order of the square of a condition number and the square of the highest degree appearing in f, h. (See [2].) (All zeroes can be found in this manner, but in our situation there is generically but one.)

[1]Approximate zero as defined in [2], a vector such that continued iteration of a simple Newton algorithm provides guaranteed convergence to an exact zero.

3.1.2 Phase II, $t > 1$

The main difference is of course that for $t > 1$ we have a much better initial condition than the random selection proposed for Phase I. It is possible to make use of this, to adjust the homotopy in such a manner as to exploit this prior information. Moreover, as the input in this Phase is defined as to achieve regulation, this modification is critically important, as regulation may well prevent reconstruction in general.

Define as the initial guess for $x_{[t,t+T-1]}$, the sequence $x^o_{[t,t+T-1]} = [x^*_{[t,t+T-2]},\ f(x^*(t+T-2), u(t+T-2)]$, which consist of our best estimate to date, augmented with a one step ahead prediction to account for the step forward. Now consider the criterion ($s \in [0,1]$)

$$
\begin{aligned}
&G_s(x_{[t,t+T-1]}) = \\
&\quad R(x_{[t-1,t+T-2]}, u_{[t-1,t+T-2]}, y_{[t-1,t+T-2]}) \\
&+ (1-s)\|x(t+T-1) - f(x(t+T-2), u(t+T-2))\|_2^2 \\
&+ (1-s)r\|y(t+T-1) - h(x(t+T-1), u(t+T-1))\|_2^2 \\
&+ s\|x(t+T-1) - f(x^*(t+T-2), u(t+T-2))\|_2^2 \\
&+ sr\|h(x^*(t+T-1), u(t+T-1)) - \\
&\qquad\qquad h(x(t+T-1), u(t+T-1))\|_2^2.
\end{aligned} \tag{3.10}
$$

Where by omitting arguments in the G_s expression, we indicate that all other variables are considered constant during this process. Now, with some abuse of notation, define the equations of interest as

$$
H_s(x_{[t,t+T-1]}) = DG_s(x_{[t,t+T-1]}). \tag{3.11}
$$

Clearly for $s = 1$ the initial choice $x^o_{[t,t+T-1]}$ is optimal by construction. Which allows us to consider the same recursion as before to find the update. The important difference being, that the number of Newton iterations we need to consider is significantly reduced, as the initial guess should be very close to a global minimiser (for all values of $s \in [0,1]$). If the model/plant mismatch is large, the benefit of this modification will be severely eroded.

The update equation in Phase II is then, with $k = 1, \cdots M$, and $s_k = 1 - k/M$,

$$
\begin{aligned}
x^{k+1}_{[t,t+T-1]} =& x^k_{[t,t+T-1]} \\
&- \left(DH_{s_k}(x^k_{[t,t+T-1]})\right)^{-1} H_{s_k}(x^k_{[t,t+T-1]}).
\end{aligned} \tag{3.12}
$$

In the adaptive situation, it may be advised not to re-identify the parameters during the control phase. Indeed if regulation is achieved this is almost certainly going to lead to a non-informative input and output sequence, making full identification impossible. In Phase II these variables are simply fixed inside the reconstruction criterion. If re-identification is deemed essential, the control objective may have to be altered in order to achieve a sufficiently informative input and output sequence. There is no generally accepted way of achieving this.

3.2 Global Newton algorithm for control

A completely similar situation applies to the control phase of the algorithm. The above outline can be reproduced mutatis mutandis.

3.3 Comments about computational complexity

The computational complexity scales with the square of the highest degree in f or h, and with the square of a condition number [2, 1]. The latter is an inherent characterization of the problem at hand, indicating in a precise sense how well posed the problem actually is. The number of variables, significantly affected by the horizon over which the optimization is performed is at first sight absent from this computational complexity estimate. The number of variables does affect the condition number. In general the condition number may scale with the Bezout degree of the set of polynomials to be solved, which would indicate an exponential increase in complexity as the horizon is increased. However, the actual condition number is more related to the number of different zeroes the polynomial equations possess, and this number is typically substantially smaller than indicated by the Bezout degree.

4 Simulation example

Consider the Hénon map-inspired dynamics as an example system to be controlled. The plant is given by

$$\begin{aligned} x_{r,1}(t+1) &= 1 - 1.4x_{r,1}^2(t) + x_{r,2}(t), \\ x_{r,2}(t+1) &= 0.3x_{r,1}(t) + u(t) + \mu(t), \\ y(t) &= x_1(t). \end{aligned} \tag{4.13}$$

The sequence μ represents an input disturbance. The input u is a scalar valued sequence, y is the scalar valued output. The numbers are chosen such that the map exhibits a chaotic attractor under zero input and zero disturbance conditions.

The (parametrized) model, used to approximate the measurements and to develop the control law, is given by

$$
\begin{aligned}
x_1(t+1) &= 1-\theta_1(t)x_1^2(t)+x_2(t),\\
x_2(t+1) &= \theta_2(t)x_1(t)+u(t),\\
\theta_1(t+1) &= \theta_1(t),\\
\theta_2(t+1) &= \theta_2(t),\\
\hat{y}(t) &= x_1(t).
\end{aligned}
\tag{4.14}
$$

In the subsequent simulations, the control objective is to steer the output to zero, without using too much control effort.

We first verify if the reconstruction and control tasks are indeed reasonable.

Observe that given if the parameters are known, it would be straightforward to compute a regulating input sequence (using the predictor form of the model) as follows:

$$
\begin{aligned}
y(t+2) &= 1-\theta_1 y^2(t+1)+\theta_2 y(t)+u(t)\\
y(t+2) &= 1-\theta_1(1-\theta_1 y^2(t)+\theta_2 y(t-1)\\
&\quad +u(t-1))^2+\theta_2 y(t)+u(t)
\end{aligned}
\tag{4.15}
$$

The unique control law that achieves dead beat regulation is given by

$$
u(t) = -1-\theta_2 y(t)+\theta_1(1-\theta_1 y^2(t)+\theta_2 y(t-1)+u(t-1))^2. \tag{4.16}
$$

Strictly speaking one has to replace $y(t)$ by $1-\theta_1 y^2(t-1)+\theta_2 y(t-2)+u(t-2)$, its expression involving only past outputs and inputs, as we insisted that $u(t)$ would only depend on past inputs and outputs.

An alternative state space control law could be $u(t) = -1-\theta_2 x_1(t)+\theta_1(1-\theta_1 x_1^2(t)+x_2(t))^2$, where $x_1(t)$ and $x_2(t)$ are replaced by their estimates from the identification step.

The discussed adaptive algorithm takes on a different form, and approximates the dead beat control as $c \to 0$. In general minisation of R and C will not lead to dead beat control, as can be clearly seen from the simulations.

Regulating the output leads to an equilibrium which is unique in this case with $(x_1, x_2) \equiv (0, -1)$. It is achieved with constant input $u \equiv -1$.

This demonstrates that the control task of output regulation can be achieved over any horizon $H > 1$. A unique control input sequence will be found in the control step.

It can also be verified that the (model's) state $(x_1, x_2, \theta_1, \theta_2)$ can be reconstructed from output and input measurements under generic conditions. Unless the output is constant[2] $x_1(t)$, $x_2(t)$, θ_1 and θ_2 are directly identifiable from $y(t), \cdots y(t+3)$ given $u(t), u(t+1)$. It is clear that in this case output regulation renders the (complete) reconstruction task unfeasible. Nevertheless, the partial state $x_1(t), x_2(t)$ remains observable from $y(t), y(t+1)$ because $x_1(t) = y(t)$ and $x_2(t) = y(t+1) - 1 + \theta_1 y^2(t)$.

From the above discussion, we conclude that the proposed algorithm will indeed be feasible, provided in Phase II we do not reconstruct the whole state but only the partial state $x_1(t), x_2(t)$.

[2]More precisely, identifiability requires that $y^2(t+2)y(t) \neq y^3(t+1)$.

Figure 1 summarises a typical simulation result. The simulation situation is determined by zero state initial conditions and zero parameter conditions for the model, for the actual plant the initial conditions are $(0.1, 0.7)$, with parameters as above. The input is set to zero for both model and plant over the first identification horizon $u(t) = 0, t = 1, \cdots, 20$. The noise μ is an i.i.d. uniformly distributed over $(-0.05, 0.05)$, which represents roughly 5% input error. The identification horizon is $T = 20$ and the control horizon is $H = 5$. The identification criterion weight $r = 1$, the control input weight in the control criterion is set at $c = 0.1$. Figure 1 shows the time responses of the system states and the corresponding estimated state sequence. The overall behaviour is quite acceptable, with the output nearly regulated at zero, it is of the order of the input disturbance. In general it was found that the proposed identification and control algorithm is very robust with respect to both model, and signal perturbations.

5 Concluding comments

We have presented an in-principle algorithm for control of non-linear systems described by polynomial maps. Both adaptive and non-adaptive problems are approached in the same framework.

The characterization of the condition number of the set of polynomial equations to be solved in the reconstruction/control phase would provide us with a clear indication of the difficulty a particular problem poses. This is under investigation.

Equally clear is that the reconstruction/control task needs some form of supervision to avoid ill conditioning. The information about how well posed the task at hand really is, is implicitly available during the minimization stages. How to exploit this information to advantage is an open question.

One of the important difficulties exhibited by polynomial systems, and that is not present in linear systems, is the issue of choice in the input calculations (this problem was avoided in the example). Several, equally acceptable from the minimization perspective, and thus competing control actions may co-exist. Which one should be implemented? As illustrated in [4] not every choice is guaranteed to lead to a well posed control problem ad infinitum. This must be further investigated.

We presented an in-principle approach to self-tuning control for a large class of nonlinear systems described by polynomial maps. We believe it provides us with a framework to analyse feasibility and well-posedness as well as giving us the opportunity to classify problems according to computational complexity.

6 Acknowledgement

Most of this work was completed when Iven Mareels was visiting the Technical University of Twente, Enschede, The Netherlands. He acknowledges the hospitality of his host Dr.

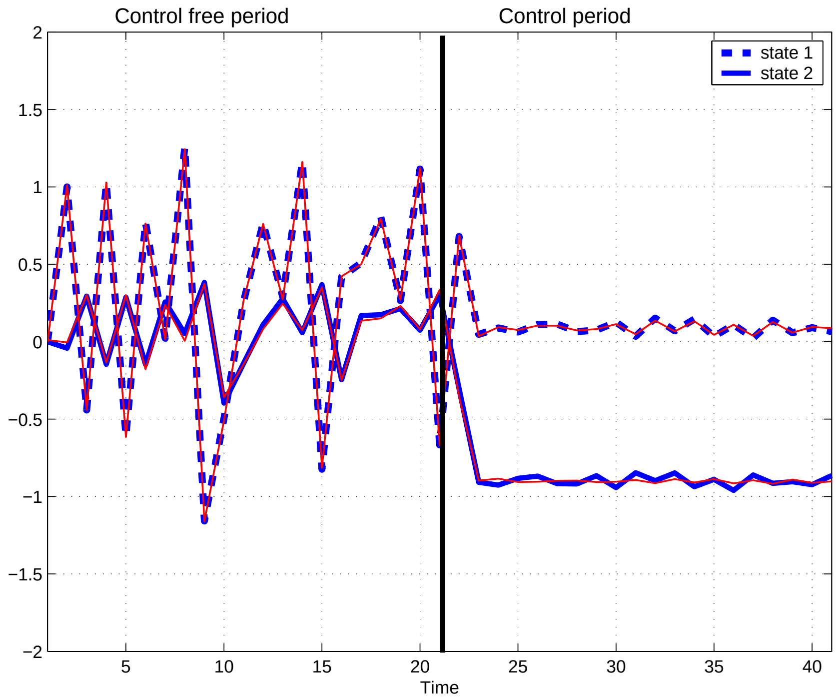

Figure 1: Controlled system response

Jan Willem Polderman, the Department of Applied Mathematics, the Technical University Twente, as well as support received by the NWO (Dutch Fund for Scientific Research) that enabled the visit.

References

[1] Iven Mareels, "Complexity Issues in Synchronisation", *Proc 40 th IEEE Conference on Decision and Control*, Orlando, USA, Dec 2001. ISBN-0-7803-7063-5

[2] Blum, L., Cucker, F., Shub, M. & Smale, S. [1998] *Complexity and real computation*, Springer, New York, Berlin.

[3] Moraal, P. & Grizzle, J.W. [1995] "Observer design for nonlinear systems with discrete-time measurements", *IEEE Trans on Aut Cont*, **40**, (3), 395-404.

[4] G. Bastin, F. Jarachi, I.M.Y. Mareels, "Output deadbeat control of nonlinear discrete-time systems with one-dimensional zero dynamics: global stability conditions",*IEEE Trans Aut Cont*, June 99, Vol.44, No 6, pp 1262-1266.

[5] J. Stark, 1999, "Delay Embeddings of Forced Systems: I Deterministic Forcing", *J. Nonlinear Sci.*, **Vol 9**, 255-332.