Predictive Control for Time-Delayed Switching Control Systems

Bobby L. Shields
Eric J. Barth
Michael Goldfarb

Department of Mechanical Engineering
Vanderbilt University
Nashville, TN 37235

Version: *September 13, 2005*

To appear in the
ASME Journal of Dynamic Systems, Measurement, and Control, vol. 128, no. 2, 2006.

Abstract

A methodology is proposed for the control of switching systems characterized by linear system dynamics with a time delay in the input channel. The method incorporates a state predictor that at each switching period determines the effect that the next control input will have on the future output of the system, and chooses the input that will take the system closest to the desired future state. The resulting control action is sub-optimal, but is computationally tractable and shown to provide a bounded tracking error for stable plants. The proposed predictive control methodology is demonstrated on a hot gas pressurization system that tracks a desired pressure trajectory via an on/off solenoid valve.

1 Introduction

This paper is motivated by the need to control a switching control system that is characterized by linear dynamics with a time-delayed input. A switching control system (as defined herein) is one that attempts to track a continuous output through the selection of a finite number of discrete-valued control inputs $u \in \{u_1, u_2, \ldots, u_p\}$, where each input value u_i is a fixed value and p is typically a small number. The level of control effort in a switching control system is thus primarily a function of the duration of the control input rather than the amplitude. As such, one perspective on the control of a continuous output via a switching input is to view the control action as one in which proportionality in time, not amplitude, governs the continuous output. Proportionality in time, however, requires knowledge of the future, which is generally not available in real-time control systems. Adequate knowledge of the future, however, and thus adequate proportionality in time, can be provided by incorporating a predictive control approach. Though several predictive control techniques have been developed [1-5], a common theme among them is the use of knowledge of the system model together with knowledge of the past states and the future desired trajectory to generate the present control effort. These notions are often used with a receding horizon strategy, in which the "future" is considered only for some finite time that captures the transient nature of the system dynamics. Generalized predictive control, presented by Clarke et al. [4], is a predictive control approach representative of [1-5] that enables the computation of a control input that optimizes a linear quadratic performance index. Specifically, for a linear time-invariant system, the sequence of future control inputs over a given prediction horizon can be computed in a closed-form solution, provided knowledge of the past inputs and outputs, in addition to the future knowledge of the desired trajectory up to the prediction horizon. Given accurate models, such methods have been shown to provide significant improvement in closed-loop control performance relative to non-model-based feedback control [5].

The existing body of literature on predictive control has focused on the treatment of continuous-valued control efforts rather than discrete-valued inputs, since the case of continuously varying control renders a closed-form solution that makes the optimization problem computationally tractable. For the case of discrete-valued control inputs, however, the closed-form analytical solution for the future control sequence is not available. Tsang and Clarke [6] treated the case of optimal bang-bang predictive control by exhaustively searching the possible permutations of control input sequences over the prediction horizon. Since the corresponding number of permutations is the number of discrete control values raised to a power corresponding to the number of time steps included in the prediction horizon, such a proposed exhaustive search can quickly become computationally prohibitive. For example, for a bang-bang control system (i.e., two discrete control values) with a prediction horizon of twenty samples, more than one million possible control sequences will exist. A bang-off-bang control system (i.e., three discrete control values) with a similar horizon would entail one billion possible permutations. Such an approach is therefore limited to a small number of prediction time steps and a low number of control choices. Sprock and Hsu [7] proposed a predictive control approach for switching controllers that circumvents the issue of optimal sequence computation by using a single-sample prediction horizon. This limiting case of prediction horizon severely limits the capacity of the predictive controller, since in order for the prediction to be effective, the prediction horizon should be on the order of the characteristic time of the system dynamics, yet for purposes of stability, the sampling period should be much shorter. Thus, a prediction horizon of one sample, as proposed by Sprock and Hsu, either impairs the control performance (in the case of a sampling period that is small relative to the characteristic time of the system), or impairs the control stability (in the case of a sampling period that is large relative to the characteristic time of the system). Finally, neither Tsang and Clarke nor Sprock and Hsu treat the case of time-delayed inputs.

This paper proposes a methodology suitable for the real-time control of switching systems with time-delayed inputs. Specifically, a predictive control methodology is formulated that enables tracking control for a switching control system with a finite number of time-delayed discrete-valued inputs. The proposed controller predicts the states of the system at an appropriate future time horizon for each of the present-time discrete-valued input candidates. The error between the predicted states and the future desired states is calculated for each of the inputs and the input candidate that produces the minimum predicted Lyapunov function is chosen. Due to the requirement for closed-form prediction, the method is restricted to linear systems and requires measurement of the full state. Though the discrete-valued nature of the control input precludes guarantees of asymptotic convergence, the method is shown to have a bounded tracking error for stable plants.

The development of a controller for this class of system was motivated by related work by the authors on the development of a hot gas pressurization system that utilizes control of a binary on/off liquid valve to track a desired pressurization curve through the catalytic decomposition of the monopropellant hydrogen peroxide [8]. As such, the effectiveness of the proposed predictive control methodology is demonstrated on the hot gas pressurization system.

2 Predictive Controller

2.1 Controller Design

Convolution describes the mathematical process by which the response of a linear system to an arbitrary input can be constructed from a summation of weighted impulse responses of that system. Assuming that the switching period of a switching control system is considerably faster than the characteristic time of the system dynamics (as is normally the case), the discrete-valued control input at each switching period can be considered as an impulse. The response of the switching control system can thus be viewed as the summation of a sequence of the system's impulse responses. Based loosely upon this notion, the proposed switching controller incorporates a predictor that at each switching period convolves the effect of each next possible discrete control choice with the effect of past control choices that have not yet fully affected the output of the system, and chooses the input that will take the system closest to the desired future state. For each switching period, the controller considers only the effect of the next input on the future state error, rather than the effects of all future inputs on the future state error, and as such assumes that all future inputs beyond the next are zero. The result is a sub-optimal solution, but one that is shown to provide a stable and bounded error dynamic for stable plants, and one that is computationally tractable. Specifically, the amount of computation required for this sub-optimal approach scales linearly with the number of discrete input values and is independent of the number of samples involved in the prediction horizon, in contrast to the previously described optimal approach, which scales exponentially with both number of samples and number of inputs.

In order to develop a predictor for the proposed control methodology, consider a continuous, linear time-invariant state space model with time delay T_u in the input, described by

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t - T_u) \tag{1}$$

where $\mathbf{x}^{\mathrm{T}} = [x \quad \dot{x} \quad \ldots \quad x^{(n)}] \in \Re^{n}$, u is a scalar input, $\mathbf{A} \in \Re^{nxn}$, and $\mathbf{B} \in \Re^{nx1}$. The proposed control approach requires that the response of the system be predicted for the prediction horizon, T_u+T_d, where T_d is a duration that is representative of the characteristic time of the system (i.e., barring any future input, the system would presumably near some steady state over the period T_d). For an arbitrary input, the response can be obtained in closed form as

$$\mathbf{x}(t + T_u + T_d) = e^{\mathbf{A}(T_u+T_d)}\mathbf{x}(t) + \int_{t}^{t+T_u+T_d} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}u(\tau - T_u)d\tau \tag{2}$$

where the first term on the right hand side is the unforced component and the second the forced component of the predicted response. At each time step, the controller predicts the future state of the system for each possible candidate input, assuming all other future inputs are zero (as previously described). As such, the input $u(t)$ will be some constant value for the interval [t, t+T_s], where T_s is the switching period, and then will be assumed to be zero for the remainder of the prediction horizon [t+T_s, t+T_d]. Note that even though the form of the input is assumed only through time t+T_d, because of the time delay in the input channel T_u, the assumed input sequence will affect states through time t+T_u+T_d. Thus, for each candidate control input u_i, the predicted future state $\hat{\mathbf{x}}_{u_i}(t + T_u + T_d)$, assuming all inputs after time t+T_u+T_s are zero, can be described by

$$\hat{\mathbf{x}}_{u_i}(t + T_u + T_d) = e^{\mathbf{A}(T_u+T_d)}\mathbf{x}(t) + \int_{t}^{t+T_u} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}u(\tau - T_u)d\tau + \int_{t+T_u}^{t+T_u+T_s} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}u_i d\tau \tag{3}$$

where the first term on the right hand side represents the unforced response, the second term accounts for all of the inputs that have already occurred but have yet to affect the system due to the time delay, and the third term is the effect that the next input will have on the future state. Once the future states are predicted for each possible discrete-valued control input u_i, the next control input corresponding to the minimum weighted future state error is selected (i.e., the predicted error response closest to the desired error response is selected). This procedure effectively minimizes the magnitude of the predicted Lyapunov function at the prediction horizon. Specifically, minimizing the weighted state error at the prediction horizon corresponds to minimizing the magnitude of a sliding surface described by

$$s(t) = \left(\tfrac{d}{dt} + \lambda\right)^{n-1} e(t) \tag{4}$$

where n is the order of the system, λ is a gain that determines the state error weighting, and $e(t) = x(t) - x_d(t)$ is the tracking error where $x_d(t)$ is the desired output. If a Lyapunov function is defined as

$$V = \tfrac{1}{2} s^2 , \tag{5}$$

then minimizing the weighted error in the state at the prediction horizon is equivalent to minimizing the Lyapunov function at that future time. The control law therefore amounts to choosing the next control input that minimizes the predicted value of the Lyapunov function at the prediction horizon, and can be stated mathematically as

$$u(t^+) = u_i \;\text{ where }\; \hat{V}_i(t + T_u + T_d) = \min\left(\hat{V}_j(t + T_u + T_d)\right) \text{ for } \; j = \{1, 2, \ldots, p\} \tag{6}$$

where u_i represents each of the possible control inputs for the next time step, $\hat{V}_i(t + T_u + T_d)$ is the predicted value of the Lyapunov function at the prediction horizon corresponding to the input choice u_i (but assuming all inputs in the time interval $[t + T_u + T_s, t + T_u + T_d]$ are zero), and p represents the number of possible control input choices. Note that while the predictor provides the values of $\mathbf{x}(t + T_u + T_d)$ for the evaluation of the control law (6), prediction of s in Eq. (4)

technically requires knowledge of $x_d, \dot{x}_d, \ldots x_d^{(n-1)}$ up to future time $t+T_u+T_d$. In real-time implementation these future desired values are generally not known, and as such the error given in Eq. (4) can alternately be constructed using a combination of the predicted states and the present desired reference trajectory $x_d, \dot{x}_d, \ldots x_d^{(n-1)}$, so that the predicted error is defined by

$$\hat{e}(t+T_u+T_d) = \hat{x}(t+T_u+T_d) - x_d(t) \tag{7}$$

Therefore, for real-time applications for which the future desired reference trajectory is not known, the predicted value of the Lyapunov function for each discrete-valued control input is given by

$$\hat{V}_i(t+T_u+T_d) = \frac{1}{2}\left(\left(\tfrac{d}{dt}+\lambda\right)^{n-1}\left(\hat{x}_{u_i}(t+T_u+T_d) - x_d(t)\right)\right)^2 \tag{8}$$

The controller is thus constructed from the combination of Eqs. (3), (6), and (8). Note that, though the controller will not require any future information, perfect control performance (i.e., a control that will result in a zero error) will not result in perfect tracking, but rather will result in the actual output x tracking a delayed version of the desired output x_d, delayed by the time delay and the prediction horizon, T_u+T_d.

Finally, note that while Eqs. (3), (6), and (8) describe the controller in the context of a single-input system, the method is easily extended to the multi-input case. For the multi-input case where $\mathbf{u} \in \Re^m$, the subscript p no longer represents the number of discrete-valued choices of the single input, but rather represents the total number of discrete-valued combinations of input vectors. Thus, for a three-input system in which each input can assume two possible values, the control vector $\mathbf{u}$ can assume eight possible combinations ($p = 2^3$), and thus eight possible values for the future Lyapunov function must be computed in order to select the next control vector.

Note also that the method does not require that each element of the input vector assume a uniform number of choices, but rather is concerned only with the total number of input combinations (i.e., some elements may have two possible choices, some may have three, etc.).

2.2 Controller Stability

The variable *s*, as defined by Eq. (4), acts as the forcing term in the stable tracking error dynamics. If the control law results in a bounded value of *s*, then the tracking error will remain bounded. The value of *s* at time $t+T_u+T_d$ is given by:

$$s(t+T_u+T_d)=\left(\tfrac{d}{dt}+\lambda\right)^{n-1}e(t+T_u+T_d) \tag{9}$$

Since the term $\left(\tfrac{d}{dt}+\lambda\right)^{n-1}$ represents a stable linear dynamic with real repeated poles, the magnitude of *s* at time $t+T_u+T_d$ is bounded by the error at that point in time, which can be expressed as:

$$e(t+T_u+T_d)=x(t+T_u+T_d)-x_d(t) \tag{10}$$

where the use of the desired motion at present time *t* is consistent with the causal form of Eq. (8). Note that the desired trajectory $x_d(t)$ is bounded by assumption. The output $x(t+T_u+T_d)$ is given by:

$$x(t+T_u+T_d)=\mathbf{Cx}(t+T_u+T_d) \tag{11}$$

where

$$\mathbf{C}=\left\lfloor \mathbf{0}_{1x(n-1)} \quad 1 \right\rfloor \tag{12}$$

and

$$\begin{aligned}\mathbf{x}(t+T_u+T_d) &= e^{\mathbf{A}(T_u+T_d)}\mathbf{x}(t)+\int_t^{t+T_u+T_d} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}u(\tau-T_u)d\tau \\ &= e^{\mathbf{A}(T_u+T_d)}\mathbf{x}(t)+\int_t^{t+T_u+T_s} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}u(\tau-T_u)d\tau+\int_{t+T_u+T_s}^{t+T_u+T_d} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}u(\tau-T_u)d\tau\end{aligned} \tag{13}$$

This expression can be rewritten as:

$$\mathbf{x}(t+T_u+T_d)=\hat{\mathbf{x}}(t+T_u+T_d)+\tilde{\mathbf{x}}(t+T_u+T_d) \tag{14}$$

where

$$\hat{\mathbf{x}}(t+T_u+T_d)=e^{\mathbf{A}(T_u+T_d)}\mathbf{x}(t)+\int_t^{t+T_u+T_s} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}u(\tau-T_u)d\tau \tag{15}$$

and

$$\tilde{\mathbf{x}}(t+T_u+T_d)=\int_{t+T_u+T_s}^{t+T_u+T_d} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}u(\tau-T_u)d\tau \tag{16}$$

Based on Eqs. (9-16), if Eqs. (15) and (16) are stable systems with bounded inputs, then Eq. (9) will be bounded, and as a result the tracking error will also be bounded. Equation (15) represents the predicted state used in the control law, which is the predicted state assuming no control input in the time interval $[t+T_u+T_s, t+T_u+T_d]$. Equation (16) represents the error in the state prediction, which specifically represents the difference in future state based on any nonzero inputs that occur in the time interval $[t+T_u+T_s, t+T_u+T_d]$. If the plant is assumed to be stable, then in the case that the input is zero, the predicted state (as given by Eq. (15)) will be bounded and will not grow without bound. Since the controller chooses the next control input to maximize the convergence between Eq. (15) and the desired state, and since the desired state is assumed bounded, the control law as given by Eqs. (3, 6, and 8) guarantees that Eq. (15) will remain bounded. The control law does not, however, consider the evolution of the component of the state described by Eq. (16). Despite this, the component of the state described by Eq. (16) is bounded, with the upper bound being the case in which the input is maximum for the time period $[t + T_u + T_s, t + T_u + T_d]$. Since the bound on this term is independent of the state (i.e., it does not scale with some norm of the state vector), it represents a forcing term in a stable of an equivalent difference equation, and therefore has no influence on the stability of the system as long as it remains bounded. Likewise, the homogeneous portion of this equivalent difference equation is effectively given by Eq. (15) which, as previously argued above, is stable.

Therefore, for the assumed case of a stable plant, the proposed control approach will provide bounded error dynamics. In reference to tracking performance, the control will in general provide an error forcing that is smaller than the bounded value previously described. Specifically, the bound on Eq. (9) consists of a component based on the bound on Eq. (15) and a component based on the bound on Eq. (16). The control law will select an input in the interval $[t + T_u, t + T_u + T_s]$ that will minimize the magnitude of the component corresponding to Eq. (15), and as such, the magnitude of $s(t + T_u + T_d)$ will in general be smaller than its bound.

It should be noted that the authors have not as yet constructed a proof that describes the conditions under which the proposed control approach provides stable tracking for a metastable or unstable plant. Despite this, the following example experimentally demonstrates the proposed control approach for the stable tracking of a metastable plant.

3 Example: Predictive Switching Control of a Hot Gas Pressurization System

As previously mentioned, the development of the proposed controller was motivated by related work by the authors on the development of a hot gas actuation system. Recent work by Goldfarb et al. [8] presented an experimental monopropellant-powered actuation concept designed to address the energy and power density requirements imposed by a self-powered human-scale robot. One configuration of this system controls the pressure in each side of a double-acting pneumatic cylinder via the injection of a liquid monopropellant through a catalyst bed. The catalyst bed decomposes the liquid into a hot gas, which in turn pressurizes the respective cylinder chamber. Chamber pressurization is thus controlled by the flow rate of the liquid monopropellant, which is controlled via a binary on/off solenoid valve. Depressurization is accomplished by exhausting the chamber, and is controlled with a proportional exhaust valve. The pressurization component of this system thus involves a discrete-valued (on/off) control input, a non-negligible transport delay of the propellant between the valve and the catalytic material, a reaction dynamic of the chemical decomposition, and a pressurization dynamic of the hot gas filling the actuator. The following section describes the implementation of the proposed control methodology on this pressurization system.

In order to isolate the pressurization component of the system and test the proposed control methodology, an experimental setup was constructed as shown schematically in Fig. 1. In this setup, a monopropellant consisting of 70% hydrogen peroxide and 30% water (by weight) was stored in a stainless steel blowdown propellant tank, and was metered through a 2-way solenoid-actuated fuel valve (Parker/General Valve Series 9), through a catalyst pack, and into a stainless steel tank. The catalyst pack was constructed in-house and consists essentially of a 5 cm (2 in) long, 1 cm (0.375 in) diameter stainless steel tube packed with catalyst material. A pressure sensor (Omega model PX32B1-250GV) was used to measure the tank pressure. The control objective was to track a desired pressurization trajectory via switching control of the liquid solenoid valve. With the aid of a spike-and-hold electronic driver, the solenoid-actuated valve was capable of switching on and off in less than 400 microseconds, and therefore a switching period T_s of one millisecond was used in the controller, during which time the control command was approximated as constant.

In order to develop a linear model of the pressure dynamics of this system from liquid valve input to tank pressure output, a series of one millisecond pulses was commanded to the valve and the resulting tank pressure was measured, as shown in Fig. 2. Note that the pulse inputs are shown at the bottom of the plot for reference, though the amplitude of each pulse is unitless. This series of pulse responses was then averaged to find the average pressure response to an input pulse, which is shown in Fig. 3. Note that the time scale of the input pulse is also shown in the figure for reference, though the amplitude is unitless. The average impulse response can be described by a second order model with a time delay in the input channel. This model, which is also shown in Fig. 3, is given by

$$\frac{P(s)}{U(s)} = \frac{ke^{-sT_u}}{s(\tau_c s + 1)} \tag{17}$$

where $P(s)$ and $U(s)$ represents the Laplace transforms of the chamber pressure and the input from the liquid valve, respectively, the parameters k and τ_c represent the gain and time constant of the reaction dynamics, and e^{-sT_u} represents the transport lag associated with the propellant reaching the catalyst. Note that in Eq. (17), the variable s represents the Laplace variable, but represents the sliding surface in all other references in this paper. It should be noted that a third order model consisting of a free integrator and a damped complex pole pair provided a somewhat better fit than the second order description, but since full state feedback is required, the authors opted for the second order description to avoid measurement of any higher derivatives of pressure. The model parameters were determined empirically from the average pulse response to be

$$
\begin{aligned}
k &= 16{,}500 \text{ kPa/sec} \quad (2400 \text{ psi/sec}) \\
\tau_c &= 6 \text{ msec} \\
T_u &= 5 \text{ msec}
\end{aligned}
\tag{18}
$$

In order to implement the proposed control methodology, the time-domain representation of the system dynamics

$$\tau_c \ddot{P}(t) + \dot{P}(t) = ku(t - T_u) \tag{19}$$

was converted to a state-space representation, where the states are given by $x_1(t) = P(t)$ and $x_2(t) = \dot{P}(t)$, and the **A** and **B** matrices given by

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & -1/\tau_c \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0 \\ k/\tau_c \end{bmatrix} \tag{20}$$

The predictor, given by Eq. (3), was evaluated for both input possibilities at each time step. As such, the next control input could either correspond to turning the valve off (i.e., $u_1 = 0$), in which case the predicted future state would be given by:

$$\hat{\mathbf{x}}_{u_1}(t + T_u + T_d) = e^{\mathbf{A}(T_u + T_d)}\mathbf{x}(t) + \int_t^{t+T_u} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}u(\tau - T_u)d\tau \tag{21a}$$

or could correspond to turning the valve on (i.e., $u_2 = 1$), in which case the predicted future state would be given by

$$\hat{\mathbf{x}}_{u_2}(t + T_u + T_d) = e^{\mathbf{A}(T_u + T_d)}\mathbf{x}(t) + \int_t^{t+T_u} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}u(\tau - T_u)d\tau + \int_{t+T_u}^{t+T_u+T_s} e^{\mathbf{A}(t+T_u+T_d-\tau)}\mathbf{B}d\tau \tag{21b}$$

Note that the two predictions differ only in the last term. Upon computing both predictions, the predicted Lyapunov function given by Eq. (8) was computed for both cases, and the control input corresponding to the smallest predicted Lyapunov function was chosen as the next control input, as described by the control law Eq. (6). The prediction of the weighted state error for each case was computed in real time as shown in the block diagram of Fig. 4. Note that the predictor requires only the current state measurements, $P(t)$ and $\dot{P}(t)$, the current control input $u(t)$, and the current desired pressure and its derivative, $P_d(t)$ and $\dot{P}_d(t)$.

The controller shown in Fig. 4 was implemented on the setup shown in Fig. 1 using the model described by Eqs. (18) through (20). Given the impulse response represented by Fig. 3, the prediction horizon T_d was chosen to be 20 milliseconds, which spans the significant transients of the system dynamics. Based on the dynamics of the plant (as given by Eqs. (18-19)) and experimental tuning, the single control gain was chosen as $\lambda = 2500$ rad/sec. Fig. 5 shows the tracking results for a single (half-period) sinusoidal pressurization trace, which demonstrates accurate tracking within the capacity of the control system to do so (recall from Fig. 3 that the minimum step size, and thus the resolution of the controller, is approximately 2 psi). Fig. 6 shows the tracking results for the same trajectory, but without including the time delay in the prediction. The poor tracking performance (relative to Fig. 5) demonstrates the importance of including the time delay in the performance of the predictive controller. Finally, Fig. 7 illustrates cyclic tracking at various frequencies, where the depressurization is achieved with a proportional exhaust valve, not shown in the schematic of Fig. 1. Note that the higher frequencies of tracking demonstrate the dynamic capacity of the predictive control method.

4 Conclusions

A control methodology was proposed for the control of switching systems characterized by linear system dynamics with a time delay in the input channel. The method incorporates a state predictor that at each switching period determines the effect that the next control input will have on the future output of the system, and chooses the input that will take the system closest to the desired future state. The resulting control action is sub-optimal, but is computationally tractable and shown to provide a bounded tracking error for stable plants. Though the conditions of stability for metastable and unstable plants are not explicitly derived, an experimental demonstration indicates stable and accurate tracking of a metastable plant.

References

[1] Richalet, J., Rault, A., Testud, J.L., Papon, J., 1978, "Model Predictive Heuristic Control: Applications to Industrial Processes," *Automatica*, vol. 14, no. 5, pp. 413-428.

[2] Reid, J.G., Chaffin, B.E., and Silverthorn, J.T., 1981, "Output Predictive Algorithmic Control: Precision Tracking with Applications to Terrain Following," *Journal of Guidance and Control*, vol. 4, no. 5, pp. 502-509.

[3] Rouhani, R., and Mehra, R.K., 1982, "Model Algorithmic Control (MAC): Basic Theoritical Properties," *Automatica*, vol. 18, no. 4, pp. 401-414.

[4] Clarke, D.W., Mohtadi, C., and Tuffs, P.S., 1987, "Generalized Predictive Control – Parts I and II," *Automatica*, vol. 23, no. 2, pp. 137-160.

[5] Hess, R. A. and Jung, Y.C., 1988, "Generalized Predictive Control of Dynamic Systems," *Proceedings of the IEEE Conference on Systems, Man, and Cybernetics*, vol. 2, pp.844-849.

[6] Tsang, T.T.C. and Clarke, D.W., 1988, "Generalized Predictive Control with Input Constraints," *IEEE Proceedings – Control Theory and Applications*, vol. 135, issue 6, pp. 451-460.

[7] Sprock, D. and Hsu, P., 1997, "Predictive Discrete Time Control of Switch-Mode Applications," *Power Electronics Specialists Conference Record*, vol. 1, pp. 175-181.

[8] Goldfarb, M., Barth, E.J., Gogola, M.A. and Wehrmeyer, J.A., 2003, "Design and Energetic Characterization of a Liquid-Propellant-Powered Actuator for Self-Powered Robots." *IEEE/ASME Transactions on Mechatronics*, vol. 8, no. 2, pp. 254-262.

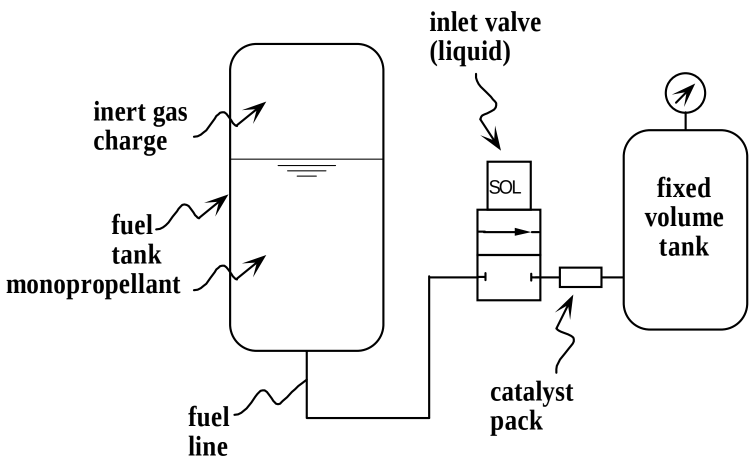

Fig. 1. Schematic of the experimental setup.

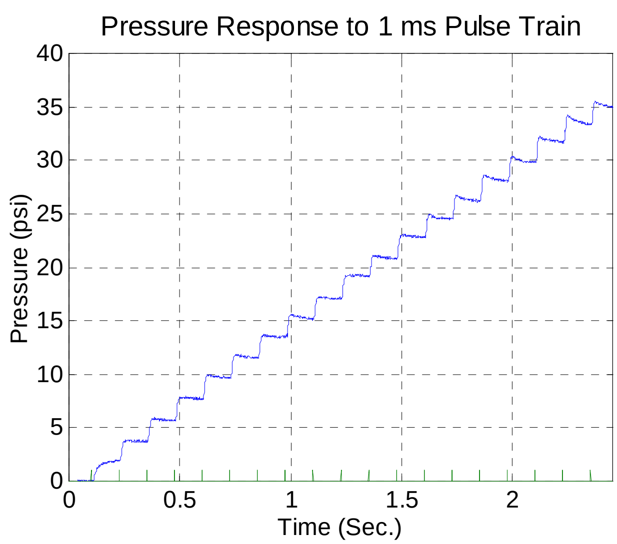

Fig. 2. Measured pressure response to a series of 1 ms valve commands.

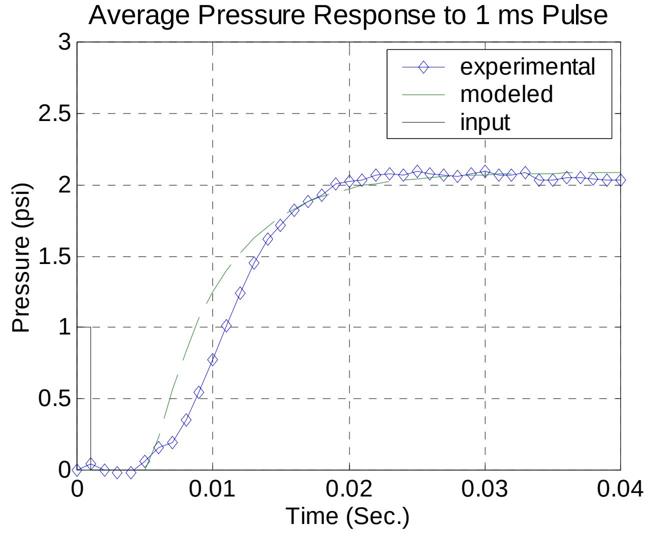

Fig. 3. Average pressure response to a 1 ms pulse.

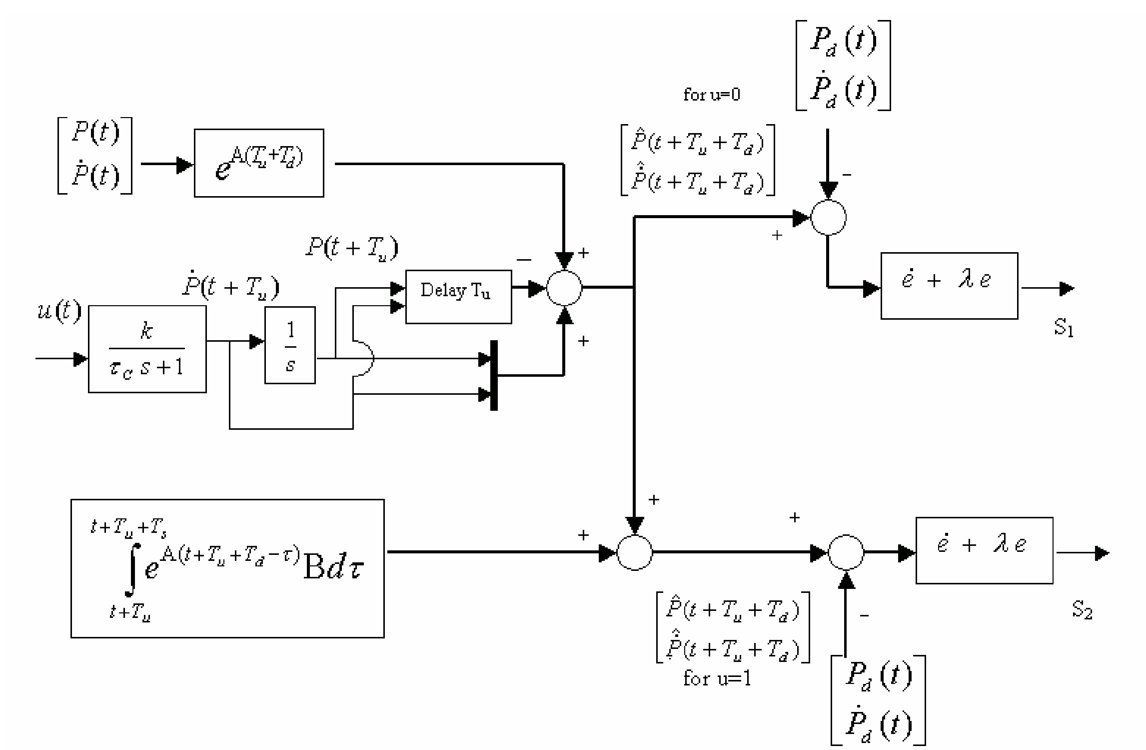

Fig. 4. Block diagram of the predictive controller.

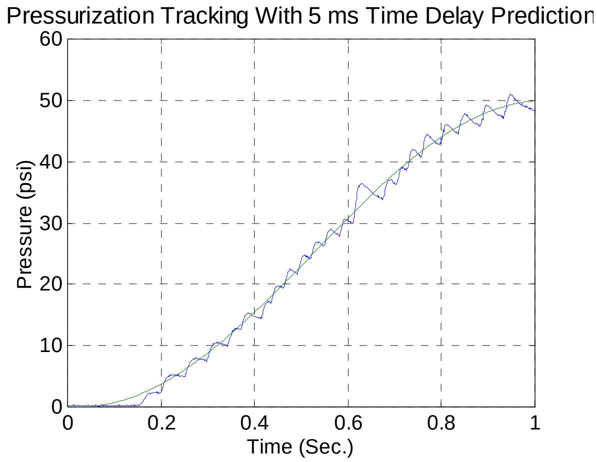

Fig. 5. Sinusoidal pressure tracking with the predictive switching controller.

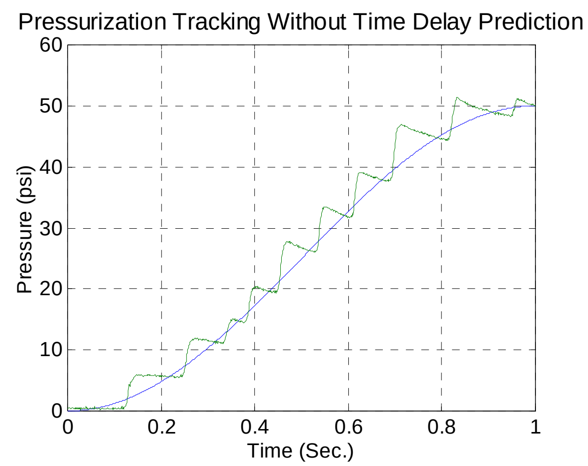

Fig. 6. Sinusoidal pressure tracking for the case of neglecting transport lag in the predictive controller.

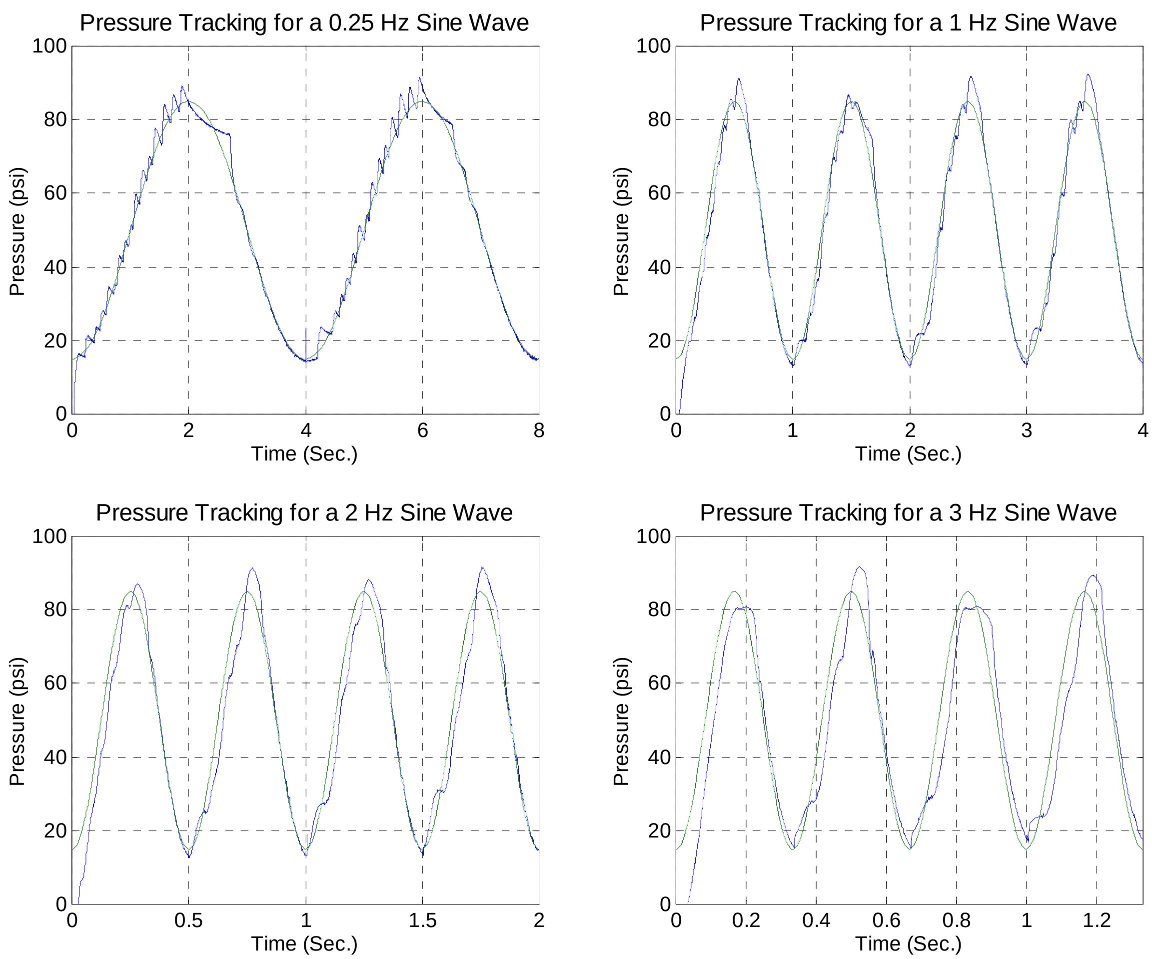

Fig. 7. Pressure tracking performance for sinusoidal pressure tracking.