Automatic PID Tuning: An Application of Unfalsified Control*

Myungsoo Jun and Michael G. Safonov
dpt. of Electrical Engineering — Systems
University of Southern California
Los Angeles, CA 90089-2563, USA

Abstract

In this paper, we give detailed procedures for using unfalsified control theory for real-time PID controller parameter tuning and adaptation. Related to the candidate-elimination algorithms of machine learning, our PID tuning technique does not need a plant model and makes PID gain selection possible by just using observed data. Simulation results are included.

1. INTRODUCTION

Traditional controller synthesis theories contain many assumptions about the plant. However, some assumptions which are made to simplify modeling procedure restrict applicability and some are so unrealistic that they make designs based on those assumptions unreliable. Modeling techniques such as [3, 8] were proposed to improve traditional modeling methods. However, these new modeling techniques still have some assumptions about the plant.

With unfalsified control theory [6, 7], you can design a controller that is consistent with a performance objective and measured past data without plant models or assumptions on the plant. The theory works by eliminating or pruning hypotheses that are not consistent with evolving experimental data. The hypotheses in question assert that members of a class of candidate controllers can meet prescribed closed-loop performance goals. The theory is related to certain artificial intelligence concepts, such as list-then-eliminate and candidate-elimination algorithms of machine learning theory [4]. In this paper, we use the unfalsified control theory to adaptively tune PID controller gains.

There are several methods for tuning PID gains. One of the most widely used methods for tuning PID gains manually is the Ziegler-Nichols method [9]. Åström et al. [1] proposed a procedure for automatic tuning of regulators of the PID type to specifications on phase and amplitude margin. Nishikawa et al. [5] proposed a method for determination of the control parameters based on the minimization of quadratic performance objectives in the time domain. However, it may be difficult to apply these methods for the automatic tuning of PID parameters to more complex systems because the rules are based, either implicitly or explicitly, on identifying approximate plant models. In this paper we describe a method based on unfalsified control theory for tuning PID parameters adaptively based on input-output data only, that is, without a plant model.

2. UNFALSIFIED CONTROL THEORY

Further details about unfalsified control theory are described in [6] and [7]. The formal definition of unfalsification and falsification is as follows:

Definition 1 [6] A controller is said to be falsified by measurement information if this information is sufficient to deduce that the performance specification \((r, y, u) \in \mathbb{R}^3\) would be violated if that controller were in the feedback loop. Otherwise, the controller is said to be unfalsified.

With the above definition of unfalsification and falsification we can state the following theorem in order to solve unfalsified control problem. Let the symbol \(K\) denote the set of triples \((r, y, u)\) that satisfy the equations that define the behavior of controller. Denote by \(P_{data}\) the set of triples \((r, y, u)\) consistent with past measurements of \((u, y)\) — cf. [6].

Theorem 1 [6] A control law \(K\) is unfalsified by measurement information set \(P_{data}\) if, and only if, for each triple \((r_0, y_0, u_0)\) \(\in P_{data} \cap K\), there exists at least one pair \((r_1, y_1)\) such that

\[ (r_0, y_1, u_1) \in P_{data} \cap K \cap \mathbb{T}_{spec} \] (1)

Fictitious reference signals occupy an important position in unfalsified control theory. Given measurements of plant input-output signals \(u, y\), there may correspond for each candidate controller, say \(K_i\), one or more fictitious reference signals \(\hat{r}_i(t)\). The \(\hat{r}_i\)'s are hypothetical signals that would have exactly reproduced the measured
data \((u, y)\) if the candidate controller \(K_i\) had been in the feedback loop during the entire time period over which the measured data \((u, y)\) was collected. Because the data \((u, y)\) may have been collected with a controller other than \(K_i\) in the feedback loop, the fictitious reference signal \(\hat{r}_i\) is in general not the same as the actual reference signal \(r(t)\). A candidate controller \(K_i\) is called causally-left-invertible if a unique values for its fictitious reference signal \(\hat{r}_i(t)\) is determined by past values of the open-loop data \(u(t)\) and \(y(t)\). Further details about fictitious reference signals can be found in [7].

### 3. PID CONTROLLER

PID control is used commonly in industrial and aerospace applications. It is its simplicity and performance characteristics that make PID popular. The ideal PID controller can be expressed as

\[
u = \left( k_p + \frac{k_i}{s} \right) (r - y) - \frac{s k_D}{s + 1} y
\]

with approximated derivative term is shown in Fig. 1.

The ideal PID controller has an improper transfer function. It is hard to exactly implement the derivative part. Thus, an approximated derivative \(\frac{s k_D}{s + 1}\) is used in realization where \(\varepsilon\) is a small number. A PID controller

\[
u = \left( k_p + \frac{k_i}{s} \right) (r - y) - \frac{s k_D}{s + 1} y
\]

standard PID controllers have gains \(k_p > 0, k_i \geq 0, k_D \geq 0\). They are always causally left invertible, which means that, given past values of \(u(t)\) and \(y(t)\), there is a unique fictitious reference signal \(\hat{r}_i(t)\) associated with each controller \(K_i\). The signal \(\hat{r}_i(t)\) can be reliably computed in real-time by filtering the measurement data \((u, y)\) via the following expression (obtained by rearranging Eq. (2) with appropriate substitutions):

\[
\hat{r}_i = y + s k_p + k_i \left( u + \frac{s k_D}{s + 1} y \right).
\]

A bank of such filters (one for each \(i \in I\)) may be used to generate the fictitious reference signals \(\hat{r}_i(t)\) in real-time — see Fig. 2. One filter is required for each candidate controller \(K_i\) (i.e., for each distinct triple of candidate PID gains \((k_p, k_i, k_D)\)).

We denote by \([A_i, B_i, C_i, D_i]\) a state-space realization of filter associated with the \(i\)-th candidate controller transfer function \(K_i, (i \in I)\). That is, \([A_i, B_i, C_i, D_i]\) is a state-space realization of the system in Fig. 2 with the values of \((k_p, k_i, k_D)\) associated with the \(i\)-th controller \(K_i\) inserted. The state vector is represented by \(\hat{x}_i(t)\).

### 4. CONTROLLER PARAMETER ADAPTATION

The main difference between unfalsified control and other adaptive methods is that one can adjust controller parameters in unfalsified control based on measured data alone without any assumptions about the plant. Our algorithm for tuning PID gains uses only measured past data in adapting its gains. While, in principle, the unfalsified control theory allows for the set \(K\) to be an arbitrary subset of \(\mathbb{R}^n\) where \(n\) is the number of controller parameters to be adjusted, we discretize candidate controller set \(K\) so that it has only a finite number of elements in order to simplify computations.

At each time \(r\), the performance specification set \(\mathcal{T}_{\text{spec}}\) consists of the set of triples \((r, y, u)\) satisfying an integral performance inequality of the form

\[
J(t) \leq -\rho + \int_0^t \mathcal{T}_{\text{spec}}(r(t), y(t), u(t)) dt \leq 0, \forall t \in [0, r]
\]

where \(\rho \geq 0\) and \(\mathcal{T}_{\text{spec}}(\cdot, \cdot, \cdot)\) are chosen by the designer.

By Theorem 1, the \(i\)-th candidate PID controller \(K_i\) is unfalsified at time \(r\) by plant data \((u(t), y(t), t \in [0, r])\) if, and only if,

\[
J(i, t) \leq 0, \forall t \in [0, r]
\]
where
\[
\hat{J}(i, t) = -\rho + \int_0^t T_{\text{spec}}(\tilde{r}_i(t), y(t), u(t)) dt, \quad \forall t \in [0, \tau],
\]
(5)
u(t), y(t), (t \in [0, \tau]) \text{ is measured past plant data, and} \tilde{r}_i(t) \text{ denotes the fictitious reference signal for the } i\text{-th controller } K_i \text{ — see Fig. 2.}

Discretizing time, we may recursively compute each of the fictitious reference signals \( \tilde{r}_i(k\Delta t) \) and its corresponding cost \( J(i, k\Delta t) \) at each time \( \tau = k\Delta t \). We use MATLAB function \texttt{c2d.m} to discretize the fictitious reference signal generating systems depicted in Fig. 2. Eq. (5) is discretized as
\[
\hat{J}(i, k\Delta t) = \hat{J}(i, (k-1)\Delta t) + \int_{(k-1)\Delta t}^{k\Delta t} T_{\text{spec}}(\tilde{r}_i(t), y(t), u(t)) dt
\]
(6)
\[
\approx \hat{J}(i, (k-1)\Delta t) + \frac{1}{2}\Delta t \cdot \{ T_{\text{spec}}(\tilde{r}_i((k-1)\Delta t), y(k\Delta t), u(k\Delta t)) + T_{\text{spec}}(\tilde{r}_i((k-1)\Delta t), y((k-1)\Delta t), u((k-1)\Delta t)) \}
\]
when \( \rho = 0 \).

The adaptation algorithm is as follows and detailed calculation procedure is described in Section 5.

Algorithm 1 (Controller Unfalsification Procedure)

\begin{itemize}
  \item \textbf{INITIAL SETTING:}
    \begin{itemize}
      \item a finite set \( K \) of \( m \) controller candidates \( K_i, i \in I \triangleq \{1, \ldots, m\} \).
      \item performance functional \( T_{\text{spec}}(\cdot, \cdot, \cdot) \)
      \item sampling time \( \Delta t \).
      \item the values of \( \epsilon, \rho \) and \( \sigma \) (see Eq. (9)).
      \item initial time \( k = 0 \).
      \item initial consistency criterion \( \hat{J}(i, 0) = 0, i = 1, \ldots, m \).
      \item initial controller \( K_m \).
    \end{itemize}
  \item \textbf{PROCEDURE (at each time } \tau = k\Delta t \text{):}
    \begin{enumerate}
      \item Measure \( u(k\Delta t) \) and \( y(k\Delta t) \).
      \item For each \( i \in I \),
        \begin{enumerate}
          \item calculate \( \tilde{r}_i((k+1)\Delta t) \) and \( \tilde{r}_i(k\Delta t) \) using a MATLAB \texttt{c2d.m} discretized approximation of Fig. 2,
          \item calculate \( \hat{J}(i, k\Delta t) \) using Eq. (7), and
        \end{enumerate}
    \end{enumerate}
  \end{itemize}
\end{itemize}

\textbf{c) if } \hat{J}(i, k\Delta t) > 0, \text{ then delete the controller index element } i \text{ from } I \text{ (since } K_i \text{ has been falsified by the measured data up to time } k\Delta t); \text{ else continue.}

9. If the set \( I \) is empty, then terminate algorithm; else, set the current controller to \( K_{i(k)} \), \( i(k) = \text{arg min}_{i \in I} \hat{J}(i, k\Delta t) \), \( i \in I \) and increment time \( k \leftarrow k + 1 \), go to step 1 and repeat.

If the set \( I \) becomes the empty set, the algorithm terminates because all controllers in \( K \) are falsified. In this case we have to either relax the performance specification or augment the set \( K \) with additional controller candidates.

In general, many candidate controllers will be falsified and discarded even before they are ever inserted into the feedback loop. Consequently, the algorithm often converges quite quickly. The current controller \( K_{i(k\Delta t)} \) remains in the loop so long as it remains unfalsified by the past data. If at some time \( k\Delta t \) the current controller becomes falsified by new data \( (u(k\Delta t), y(k\Delta t)) \), then the algorithm switches to a new controller \( K_{i} \) which we chose to be the one that has the largest index \( i(k\Delta t) \) among the as yet unfalsified controller candidates \( K_i \) in \( K \).

5. SIMULATION

In this section we describe a simulation of PID controller parameter adaptation using unfalsified control theory. The simulation shown was conducted with no disturbance, no noise and zero-initial conditions, though this is not essential.

The performance specification set \( T_{\text{spec}} \) is taken to be the set of \( (r, y, u) \in L_2\times L_2 \times L_2 \), which for all \( \tau \geq 0 \) satisfy the inequality
\[
\|w_1 \ast (r - y)\|^2 + \|w_2 \ast u\|^2 \leq \sigma^2 \tau \leq \|r\|^2 + \rho
\]
(8)
where \( \|f\|^2 = \int_0^\tau |f(t)|^2 dt \), and \( \ast \) denotes the convolution operator. Design parameters are \( \sigma \) (a constant representing the r.m.s. effects of noise on the cost), and the signals \( w_1 \) and \( w_2 \) are weighting filters. Therefore, \( T_{\text{spec}}(r(t), y(t), u(t)) \) is
\[
T_{\text{spec}}(r(t), y(t), u(t)) = \|w_1 \ast (r(t) - y(t))\|^2 + \|w_2 \ast u(t)\|^2 - \sigma^2 \tau - \|r(t)\|^2
\]
(9)
The simulation is conducted as follows: At each sampling time \( \tau = k\Delta t \), the data \( u(k\Delta t) \) and \( y(k\Delta t) \) are measured. Then, the controller unfalsification procedure (Algorithm 1) is invoked to determine which, if any, previously unfalsified controllers are now falsified based on the consistency test
\[
\hat{J}(i, k\Delta t) \leq 0.
\]
(10)
The consistency criterion \( \hat{J}(i, k\Delta t) \) is computed based on the discrete-time approximation (7) and MATLAB \texttt{c2d.m}
Figure 3: Plots of signals $y(t)$ and $u(t)$ when the states of the controller are not reset at switching time. Poor transients with spikes can occur if we fail to properly reset controller states at switching time.

Figure 4: Simulation results showing good transient response with correctly reset controller states

Figure 5: Simulation results showing the changes in controller gains

The simulation was carried out using Simulink. The results are as shown in Fig. 4 and Fig. 5, and the Simulink model used in simulation is in Fig. 6. The figure shows two times at which gain switching occurs. The values of $k_P$ and $k_I$ switch at the first switch time, and at the second switch all three gains $k_P$, $k_I$, and $k_D$ change. At each switching time, the current controller is falsified and a new, as yet unfalsified controller is switched into the closed loop. The final values of the controller parameters are $k_P = 80$, $k_I = 50$ and $k_D = 0.5$. The final number of unfalsified elements of the set $K$ is 12.

6. DISCUSSION

While the simulation shown in Fig. 4 was conducted assuming no noise ($\sigma = 0$) and zero initial conditions ($\rho = 0$), the algorithm is actually fairly robust to noise and initial state perturbations. However, if the noise or
Figure 6: Simulink model used in simulation

Initial conditions are very large, then it may sometimes be necessary to use non-zero values for \( p \) and/or \( \sigma \) in the performance specification (8).

If a plant is slowly time-varying or subject to occasional abrupt changes, the far past data may not contain much information about current plant. In such cases, either an exponential forgetting factor or a finite-memory data-window should be introduced.

While the simulation only shows the result for the case in which a step command \( r(t) \) is the input, the algorithm also works when for inputs other than a step signal. It is important only that the input have sufficient strength and spectral breadth to allow candidate controllers \( K_i \) to be reliably ordered by the performance specification functional \( J(t, t) \).

7. CONCLUSION

In this paper, we described in detail how to adaptively tune the parameters of a PID controller in real-time using unfalsified control theory. An advantage of this approach is that no plant model is required. We need only real-time measurements of input-output data \( (u, y) \) from the plant. Thus we may apply this method to distributed parameter systems, to nonlinear time-varying plants as well as to high-order linear time-invariant plants.

Irrespective of how complicated the plant may be, the PID controllers themselves are not very complicated and we can easily compute the unfalsified controller parameters at each time via the recursive procedure described in Section 4. A limitation of our procedure is that the set \( K \) of unfalsified controllers may shrink to a null set if there are no PID controllers in \( K \) that are capable of meeting the performance specification (4). But when this is not the case, convergence of the algorithm is typically rapid and sure-footed. Our experience with simulations has been that convergence is usually so rapid that satisfactory transient response is obtained on the first try even with no prior knowledge of the plant.

References


