1
Introduction to Predictive Control

Model-based predictive control is a relatively new method in control engineering. The basic idea of the method is to consider and optimize the relevant variables, not only at the current time point but also during their course in the future. This goal is achieved first by a heuristic choice of the manipulated variable sequence and simulation of the future course of the process variables. If the future course of the controlled and the constrained variables is not satisfactory, then new manipulated variable sequences are tried out until the control behavior becomes satisfactory. The expression "predictive control" arises from a forecast of the variables. A process model is necessary to simulate the process; therefore, we have the attribute "model based". In acquiring knowledge of the predicted process variables, constraints on the manipulated, controlled, and other variables can be simply taken into account. Predictive control makes possible robust control, mostly at the expense of slower performance. These algorithms are particularly suitable for petrochemical plants, which are slow enough to allow the simulation of the future course of the process values to consider both the controlled variables and the fulfillment of the constraints.

In the sequel the basics of predictive control are dealt with, namely,

- preview of predictive control,
- manipulated, reference, and controlled signals,
- cost function of predictive control,
- receding horizon strategy,
- free and forced responses of the predicted controlled variable,
- minimization of the cost function.

Several simulation examples illustrate the predictive control principle and its advantage over proportional plus integral (plus derivative) (PI(D)) control for

- linear single-input, single-output (SISO) systems,
- linear multi-input, multi-output (MIMO) systems, and
- nonlinear processes.

Predictive Control in Process Engineering, First Edition. Robert Haber, Ruth Bars, and Ulrich Schmitz.

Finally, the possibility of handling constraints is demonstrated. Practical examples are not dealt with in this introductory chapter. They are discussed in Chapters 12 and 13.

1.1 Preview of Predictive Control

There is a fundamental difference between predictive control and conventional on–off or PID control:

- A conventional controller observes only the current (and remembers the past) process variables.
- A predictive controller observes the current and also the future process variables (and remembers the past variables).

Predictive thinking is more natural in everyday thinking, for example, during car driving one observes the future shape of the road, brakes if one is approaching a curve, pushes the gas pedal if one is nearing a hill, and decreases the speed if another, slower car appears in the field of vision. Figure 1.1 compares the two driver philosophies.

- Conventional control in driving would mean a driving style where the car driver looks only through the side windows. In a curve the driver can correct the trace following the position only after having observed an error.
- Any real driver on the route is a predictive controller, because he/she drives depending on the curvature and what he/she sees in advance in front of the car.

The longer the preview distance, the better the position control, but the calculations are more time consuming. The horizon length has to be increased with the car speed. Beyond a certain preview distance the control would not be better. A minimum sampling time is necessary, otherwise the car cannot follow the driver's commands in due time.

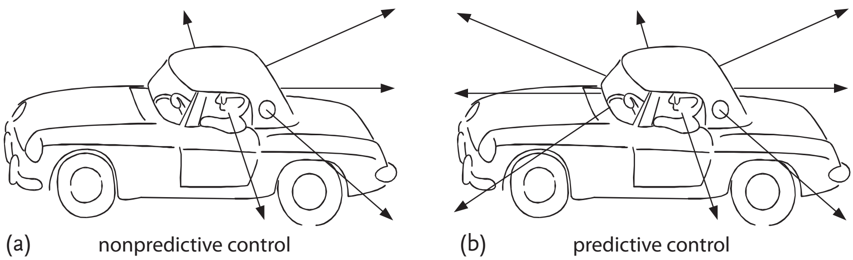


Figure 1.1 Car driving strategies.

The aim of control is to follow the reference signal and reject (which means eliminate) the effect of the disturbances. Therefore, the quality of the control depends on how these signals can be known in advance and also on the quality of the process model.

Sometimes there is no information about the future course of the reference signal or disturbance. Then the signal is assumed to remain constant, which is also a prediction, though it is not optimal.

1.1.1 Prediction of the Reference Value

In some cases the course of the future reference signal is known. Examples are:

- the product quality changes are planned in advance,
- the desired temperature in rooms in offices, schools, and so on, according to a schedule,
- the trajectory of a robot arm.

If the manipulated variable can be changed according to knowledge of the future reference signal course and before the change of the reference signal, then the desired value of the controlled variable can be achieved earlier than without this knowledge. Table 1.1 illustrates this fact for temperature control.

1.1.2 Prediction of the Disturbance

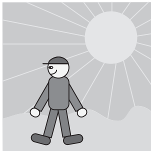
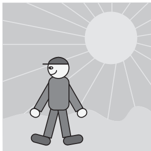
In some cases the course of the future disturbances is known. Examples are:

- weather forecast,
- electrical consumption forecast (schedule of broadcasting an event, when many people switch on their TV, lights, or heating).

Table 1.1 Decision about how to heat a school building before the teaching starts (8:00 in the morning).

Strategy	Heating before 8:00	Pupils in the class at 8:10
Decision based on the current temperature set point	Heating according to the current (night) demand	Freezing pupils can not learn
Decision based on the current and future temperature set point	Heating starts at about 6:30 as the building warms up slowly	Pupils learn in a pleasant climate

Table 1.2 Decision about taking or not taking an umbrella on an excursion.

	Current weather situation	Future weather situation	
Decision based on the current weather situation			Without a forecast, one gets into difficulties
Decision based on the weather forecast			One can handle disturbances by planning in advance (with a forecast)

If the manipulated variable can be changed with knowledge of the future course of the disturbances and before the disturbances occur, then the desired course of the controlled variable can be achieved earlier than without this knowledge. Table 1.2 illustrates this fact for “weather control” in everyday life. The advantage of predictive control is obvious: one will not get wet if an umbrella is taken on an excursion.

1.2 Manipulated, Reference, and Controlled Signals

Figure 1.2 shows the course of the manipulated, reference, and controlled signals during the control. The following symbols are introduced:

u:	manipulated variable (also called control signal)
y, y_r:	controlled and reference signals
$\hat{y}$:	predicted controlled signal
t, k:	current continuous and discrete times
ΔT:	sampling time
d:	discrete physical dead time relative to the sampling time.

Predictive control performs the following tasks:

- minimizes the control error several steps ahead of the current time point (between $k + N_1$ and $k + N_2$),
- penalizes the control increments several steps $(n_u - 1)$ ahead of current time point (e.g., to eliminate valve wear),
- takes into account limitations in the control, controlled, and other computed (e.g., state) variables.

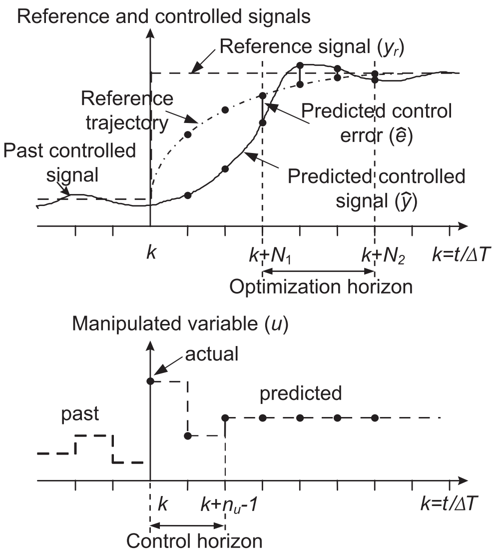


Figure 1.2 Manipulated, reference, and controlled signals during the control.

A sudden, for example, stepwise change of the reference signal can produce a too large, nonrealizable change in the manipulated variable. Sometimes a reference trajectory is introduced, which is the filtered value of a set value (reference signal) change. In the sequel only the reference signal and not the reference trajectory will be used, unless it is mentioned explicitly.

The future course of the controlled signal can be calculated only if a model of the process is known. Therefore, predictive control is often called model-based predictive control.

The difference between predictive and nonpredictive control is shown in Figures 1.3 and 1.4. Nonpredictive control (like PI(D) control) works with current (and through the internal memory also with past) values, whereas predictive control considers also future reference and/or measurable or observable disturbance and predicted manipulated and controlled signal sequences. (The connections with the predicted signals are drawn with double arrows as these signals usually contain several values. The selector generates the current control signal from the calculated manipulated variable sequence.)

1.3 Cost Function of Predictive Control

Any reasonable criterion can be defined to be achieved by the predictive controller. Some possible aims may be:

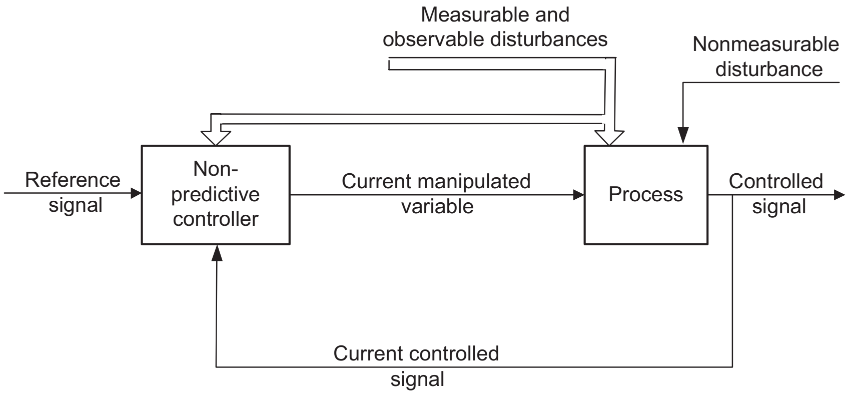


Figure 1.3 Block scheme of a nonpredictive controller.

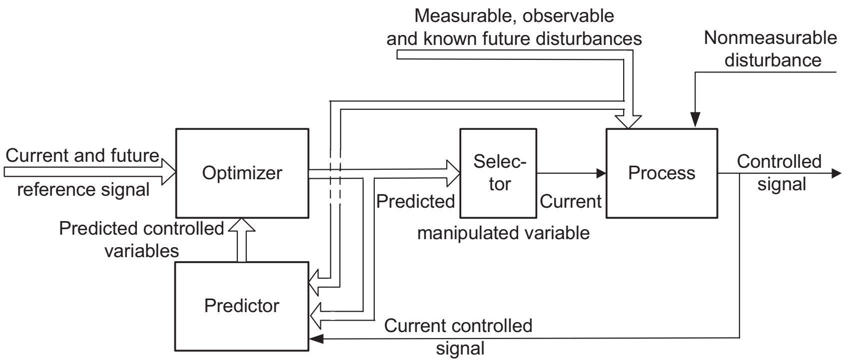


Figure 1.4 Block scheme of a predictive controller.

- fastest control,
- fastest control without overshoot in the controlled signal,
- fastest control with limitation of the manipulated signal, and so on.

A possible criterion of predictive control is to minimize a quadratic cost function of the control error and the manipulated variable increments during the corresponding prediction horizons. Clarke *et al.* [2] derived the control algorithm called Generalized Predictive Control (GPC) for linear input/output models. In the unconstrained case, the solution is explicit.

The quadratic cost function for the SISO case is

$$J = \sum_{i=N_1}^{N_2} \lambda_{yi} \left[y_r(k+i) - \hat{y}(k+i|k) \right]^2 + \sum_{j=1}^{n_u} \lambda_{uj} \Delta u^2(k+j-1) \Rightarrow \min_{\Delta u} , \tag{1.1}$$

with the notation

$y_r(k+i|k)$: reference signal i steps ahead of the current time,
$\hat{y}(k+i|k)$: predicted (and controlled) output signal i steps ahead,
$\Delta u(k+i)$: controller output increment i steps ahead,

where $(k+i|k)$ denotes that the future signal is predicted on the basis of the information available till the current time point k.

The tuning parameters of the control algorithm are:

N_1: first point of the prediction horizon beyond the current time,
N_2: last point of the prediction horizon beyond the current time,
n_u: length of the control horizon (the number of supposed consecutive changes in the control signal),
$\lambda_{y,N_1},\ldots,\lambda_{y,N_2}$: weighting factors of the control error, usually assumed to be equal to 1 ($\lambda_y = 1$ in the SISO case),
$\lambda_{u1},\ldots,\lambda_{u,n_u}$: weighting factors of the control increments, usually assumed to be equal (and denoted then by λ_u).

As the current manipulated variable can influence the controlled signal only after the dead time, the first and last points of the control error horizon are considered by n_{e1} and n_{e2}:

n_{e1}: first point of the prediction horizon beyond the current time point and the dead time,
n_{e2}: last point of the prediction horizon beyond the current time point and the dead time

$$n_{e1} = N_1 - d - 1\,; \quad n_{e2} = N_2 - d - 1\,. \tag{1.2}$$

With these denotations (1.1) becomes

$$J = \sum_{n_e=n_{e1}}^{n_{e2}} \left[y_r(k+d+1+n_e) - \hat{y}(k+d+1+n_e|k)\right]^2 + \sum_{j=1}^{n_u} \lambda_u \Delta u^2(k+j-1) \Rightarrow \min_{\Delta u(k)}\,. \tag{1.3}$$

The cost function consists of two parts:

- costs due to control error during the control error horizon, which is also called the optimization or prediction horizon,
- costs to penalize the control signal increments during the manipulated variable horizon, which is also called the control horizon.

After the control horizon, that is, after n_u steps, the manipulated variable is kept constant. That means that if the reference signal is a constant value, then the last manipulated variable is the steady-state value of the corresponding manipulated variable.

The above cost function can be minimized with knowledge of the process model for different controller parameters. This will be done for linear SISO processes in Chapter 5 and for MIMO processes in Chapter 7. Now let us consider a practical example of a complex task.

Example 1.1 Control of the economy: decision about increase or reduction of taxes

The global goal is to maximize the satisfaction of the citizens (as the government would like to win the next parliamentary election). More precisely, sometimes this global goal is reduced to a current goal of maximizing the satisfaction of the citizens on the day of the election (without consideration of the problems after the election, e.g., guaranteeing pensions).

The *target (cost) function* consists of several parts:

- maximizing the incomes of the state,
- maximizing the incomes of the citizens (satisfaction feeling),
- minimizing the working time (satisfaction feeling), and so on.

The following *variables* are defined:

- *Controlled variable*: satisfaction of the citizens
- *Disturbances*: effects of the world economy
- *Manipulated signal*: tax change (increase or reduction)
- *Constraints*: Each citizen must receive the subsistence level, daily working time maximum 8 h, and so on.

Model-based prognosis: To compute the target function some years ahead, models are used which simulate (predict) the consequences of a tax change.

Model-based control: On the basis of minimization of the target function, a new, optimal tax (manipulated variable) is computed.

Sampling time of discrete-time control: The tax is changed each 1 January, which means $\Delta T = 1$ year.

A sequence of tax changes is calculated for the next years, but only the current tax change is realized. The calculation is repeated every year in the knowledge of the current (measured) values, considering also the situation of the world market.

1.4 Reference Signal and Disturbance Preview, Receding Horizon, One-Step-Ahead, and Long-Range Optimal Control

In the sequel the basic principles of predictive control are illustrated by some simulation results.

As predictive control minimizes the future control error, the future values of the reference signal have to be known. There are two possibilities:

- the future reference signal course is known;
- the future reference signal course is not known.

If there is no other information, then the current set value is assumed to be constant in the future.

Example 1.2 Predictive control of a linear third-order process without knowing the future reference signal

Figure 1.5 shows the predictive control of a linear third-order process without a reference signal and disturbance prediction. The process parameters are as follows: static gain $K_p = 1$, and three equal time constants of $T_1 = 1/3$ s. The set value is increased stepwise at $t = 1$ s from 0 to 1 and a load disturbance of -1 is added to the input of the process at $t = 6$ s. The sampling time is $\Delta T = 0.1$ s and the controller parameters are $n_{e1} = 0$, $n_{e2} = 9$, $n_u = 3$, and $\lambda_u = 0.1$.

Example 1.3 Predictive control of a linear third-order process knowing the future reference signal

Figure 1.6 shows the predictive control of a linear third-order process if the future reference signal course is known for $n_{y_r,pre}$ steps. All other parameters are as in Example 1.2. In the case of the reference signal preview, the control starts $n_{y_r,pre}$ steps before the set point change and achieves the new set value earlier than without a preview. As is seen, a part of the control error after the set point change is shifted to before the set point change. As the disturbance is not known in advance, its compensation starts after its occurrence (and measurement).

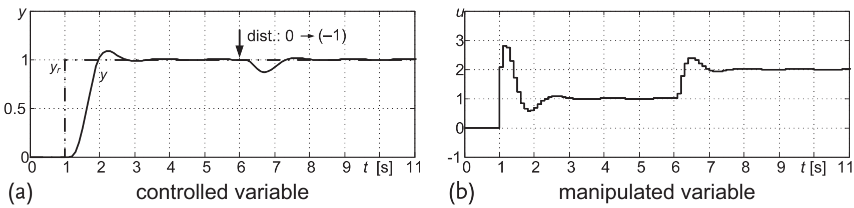


Figure 1.5 Predictive control of a linear third-order process without knowing the future reference signal.

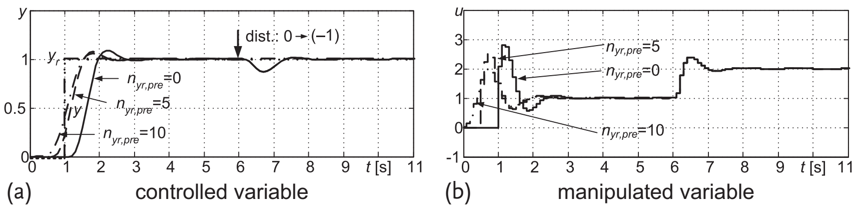


Figure 1.6 Predictive control of a linear third-order process knowing the future reference signal for some $n_{y_r,pre}$ steps in advance.

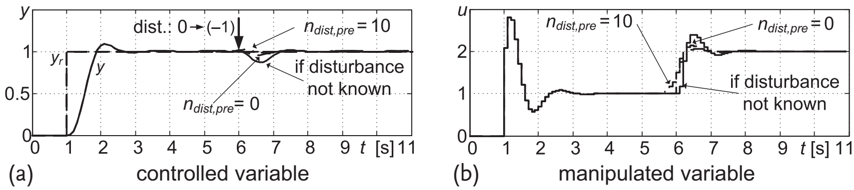


Figure 1.7 Predictive control of a linear third-order process without knowing the future reference signal but knowing the disturbance $n_{dist,pre}$ steps ahead.

Disturbances can be divided into three groups:

- *Nonmeasurable (or unobservable) disturbances*
  Only the controlled output signal is measured and used in the control algorithm.
- *Measurable (or observable) disturbances*
  Both the disturbance and the controlled output signals are measured and used in the control algorithm. The control starts only if the disturbance occurred. If the process has a long delay and/or dead time, the manipulated signal can compensate for the effect of the disturbance with delay.
- *Future course of the measurable (or observable) disturbances is known*
  The control can start in advance to compensate for the disturbance. In the optimal case, the disturbance does not influence the controlled output.

Example 1.4 Predictive control of a linear third-order process knowing the disturbance in advance

Figure 1.7 shows the predictive control of a linear third-order process if the future course of the disturbance signal is known for $n_{dist,pre}$ steps. All other parameters are as in Example 1.2. In the case of a preview, the control starts $n_{dist,pre}$ steps before the disturbance change and compensates for the disturbance earlier than without a preview. As is seen, a control error occurs before the appearance of the disturbance and acts to eliminate it. As the future course of the reference signal is not known in advance, the control starts to move to the new set point after the change of the reference signal. For comparison, the controlled and the manipulated signals are also plotted for the case when the disturbance is not known.

In Example 1.1 it was mentioned that a sequence of tax changes is calculated for the next years but only the current tax change is realized. The calculation is repeated every year.

Receding horizon control strategy: The manipulated variable and its future values are computed in each control sampling step. Only the current manipulated signal is realized. The advantage is that changes in the model, in the reference signal, in the disturbances, and/or in the constraints during the prediction horizon (which means in the future) can be actually considered.

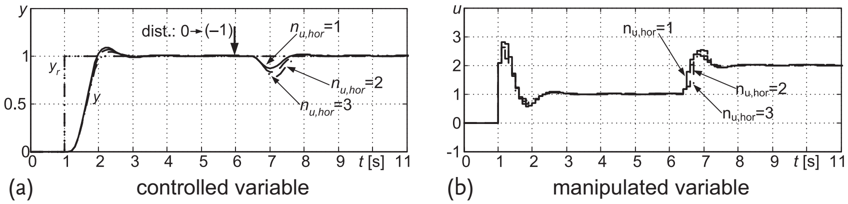


Figure 1.8 Predictive control of a linear third-order process with and without a receding horizon, without knowing the future reference and disturbance signal.

Example 1.5 Control of a disturbed plant without and with the receding horizon technique

The control scenario is the same as in Example 1.2. The variable $n_{u,hor}$ shows how many elements of the manipulated variable are taken from the control signal sequence without a new calculation. $n_{u,hor} = 1$ corresponds to the receding horizon case. As is seen, the control is faster if the manipulated variable is calculated in every control step in the case of a disturbance (shown in Figure 1.8) or of a process parameter change (not shown here).

Depending on the control error horizon bounds, one can distinguish two types of cost function:

- *One-step-ahead control*:
  The start and the end of the control error horizon are equal; the cost function is optimized only at one time point in the future ($n_{e1} = n_{e2}$).
- *Long-range optimal control*:
  The end point of the control error horizon is larger than the start point; the cost function is optimized at more time points in the future ($n_{e2} > n_{e1}$).

Example 1.6 One-step-ahead and long-range optimal control of a linear third-order process

Figure 1.9 compares the one-step-ahead and long-range optimal control of a linear third-order process if the future reference signal course is unknown. All other parameters are as in Example 1.2. The one-step ahead control with a short prediction $n_{e1} = n_{e2} = 1$ caused overshoot and oscillations, and with a long prediction $n_{e1} = n_{e2} = 9$ it was very slow. The overshoot could be decreased and the control becomes faster if a long-range optimal control is used with nearly the same start and end values of the control error horizon as in the one-step-ahead cases before.

The simulations show that

- a too short prediction horizon may cause overshoot for higher-order systems;
- a too long prediction horizon results in slow control without overshoot;
- a long-range optimal control with a low starting value and high end value of the horizon may lead to fast control with a small overshoot.

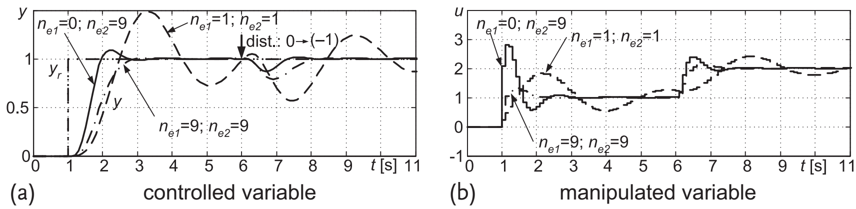


Figure 1.9 One-step-ahead and long-range optimal predictive control of a linear third-order process without knowing the future reference and disturbance signals.

1.5 Free and Forced Responses of the Predicted Controlled Variable

The predicted process response $\hat{y}(k+1+d+n_e|k)$ is the effect of the free and the forced response (Figure 1.10):

- The free response $\hat{y}_{free}(k+1+d+n_e|k)$ is obtained if the last value of the manipulated signal is kept unchanged.
  - $u(k-1) = u(k) = u(k+1) = \ldots = u(k+n_u-1)$ or $\Delta u(k+j-1) = 0$ with $j = 1, 2, \ldots, n_u$
  - with the initial values $y(k), y(k-1), \ldots, y(k-n)$ (n is the model order)
- The forced response $\hat{y}_{forc}(k+1+d+n_e|k)$ is the effect of the consecutive changes in the manipulated variable at the current and future time points.
  - $\Delta u(k), \Delta u(k+1), \ldots, \Delta u(k+n_u-1)$, and $\Delta u(k+i) = 0$ for $i \geq n_u$
  - with the initial values $y(k) = 0$, $y(k-1) = 0, \ldots, y(k-n) = 0$.

For linear processes the predicted controlled output can be calculated as the sum of the free and forced responses (superposition principle). Unfortunately, the superposition is not valid for nonlinear processes.

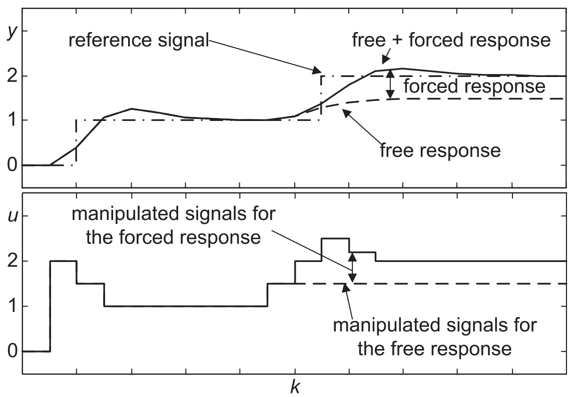


Figure 1.10 Splitting the predicted controlled variable into a free and a forced response for linear systems.

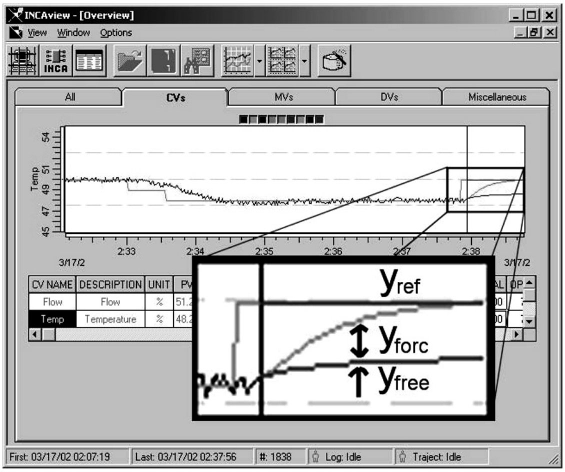


Figure 1.11 Preview of the reference value, free response, and controlled variable by the program INCA from IPCOS Technologies.

Figure 1.11 shows the prediction of the free response and the predicted controlled variable (as the sum of the free and the forced response) in a display hard copy of the commercial program INCA from IPCOS Technologies, Boxtel, The Netherlands. The operator can see a preview of the above-mentioned variables and can stop the automatic control, change the set value, or change the manipulated variable if the predicted values are not satisfactory.

1.6 Minimization of the Cost Function

The controlled output also depends on future manipulated variable values. Accordingly, future values of the current manipulated variable have to be optimized as well. The sequence of the changes in the manipulated variable to be calculated is

$$\begin{aligned}\Delta\mathbf{u} &= \left[\Delta u(k|k), \Delta u(k+1|k), \dots, \Delta u(k+n_u-1|k)\right]^T \\ &\equiv \left[\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+n_u-1)\right]^T \end{aligned} \tag{1.4}$$

and the first term of vector (1.4) is used as the current manipulated variable. For simplicity, $\Delta u(k+i|k)$ is written as $\Delta u(k+i)$ in the sequel, although the future manipulated variable is calculated at the current time point k.

1.6.1 Minimization Algorithms for Nonlinear Processes with or without Constraints

Generally there is no analytical solution: the cost function is computed by simulation in the prediction horizon for all sequences of the manipulated variable and the manipulated variable sequence is calculated by a numerical algorithm such as

- the simplex or gradient method (this is faster, however sometimes no global minimum is found),
- an evolutionary algorithm (this is slower, but mostly the global minimum is found).

The initial value for the minimization can be the manipulated signal sequence in the previous sampling step.

1.6.2 Minimization of the Quadratic Cost Function for Linear Processes without Constraints

The free response can be calculated with knowledge of the model parameters from the current and past input/output values. The forced response is a linear function of the manipulated variable sequence in the future horizon. A quadratic cost function of the manipulated signal sequence can be minimized analytically without iteration if there are no constraints.

Table 1.3 summarizes the cases mentioned.

Figure 1.12 shows the general structure of a predictive controller. If the minimization of the cost function leads to an analytical solution, then the control algorithm is a difference equation like with PID control. Then the online computation

Table 1.3 Comparison of the minimization algorithms.

	Linear model and no technological constraints	Linear model with technological constraints	Nonlinear model with or without technological constraints
Model type	Linear	Linear	Nonlinear
Hard constraint	No	Possible	Possible
Prediction of the controlled signal	Sum of the free and forced system responses	Direct calculation or sum of the free and forced system responses	Direct calculation
Minimization algorithm	Analytical solution	Minimization of the cost function in an iterative way	Minimization of the cost function in an iterative way
Computational demand	Low	High	Very high

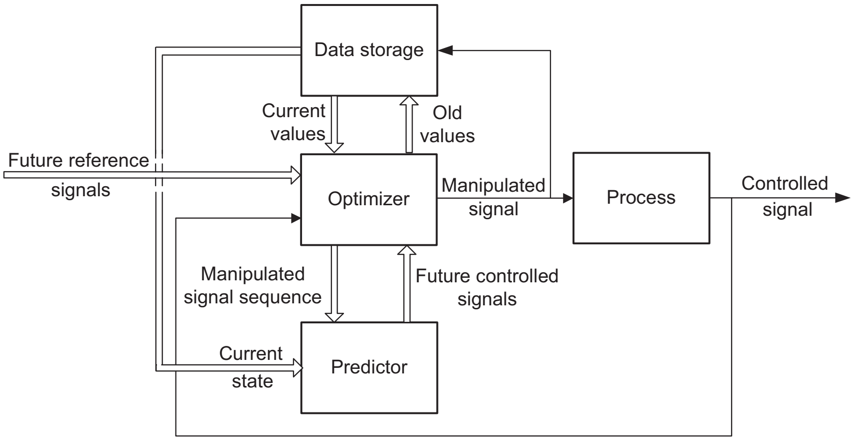


Figure 1.12 Structure of a predictive controller.

time is comparable with that of PID control, which is usually negligible related to the sampling time. If iterative minimization is necessary, then the online computational demand is higher, depending on the dimension of the minimization problem and on the algorithm applied. There are some algorithms known for linear processes with hard constraints which work with a series of a priori computed control laws and with knowledge of the state variables the linear control algorithm has to search for in a lookup table.

1.7 Simple Tuning Rules of Predictive Control

The advantage of predictive control over PI(D) control is obvious when the process has nonaperiodic characteristics or it contains significant dead time. This is illustrated by two examples.

Example 1.7 Level control in a tank and in a boiler

The relation between the water flow and the level in a tank results in an aperiodic process (without boiling water), and the level control is an easy task. In a boiler, however, the cold water increase leads temporarily to a decrease of the level as bubbles in the boiling water collapse. If the water feed becomes warmer, the level increases and achieves its new, higher steady-state value. Such processes are called inverse repeat or non-minimum-phase processes. Figure 1.13 shows both cases with the step responses. The tanks and boilers in Figure 1.13 also show the sequences of the level changes.

The sampling time is $\Delta T = 0.1$ s. First, a PID controller is tuned according to Chien *et al.* [1] for the tank level control. The simulated process is approximated by a first-order lag element with dead time and the following parameters: static gain

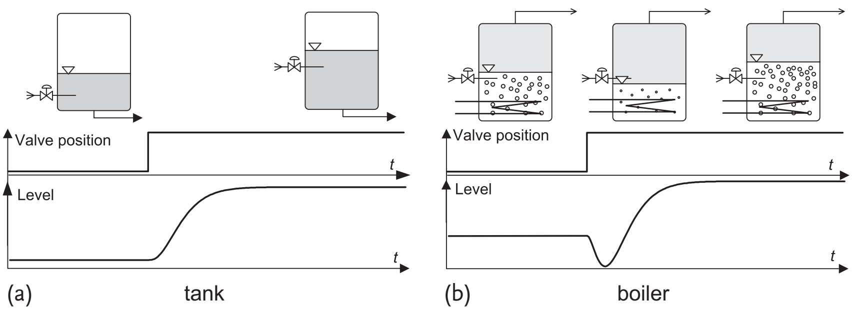


Figure 1.13 Level step responses of a tank and a boiler.

$K_p = 1$, apparent dead time $T_L = 0.2$ s, and apparent time constant $T_T = 1.4$ s (of course, such a process is much slower in practice). The controller parameters are as follows:

$$K_c = \frac{0.6}{K_p} \cdot \frac{T_T}{T_L} = \frac{0.6}{1} \cdot \frac{1.4\,\text{s}}{0.2\,\text{s}} = 4.5\;;$$
$$T_I = T_T = 1.4\,\text{s}\;; \quad T_D = 0.5 \cdot T_T = 0.5 \cdot 0.2\,\text{s} = 0.1\,\text{s}\;.$$

The set value was increased stepwise at $t = 1$ s from 0 to 1 and a step disturbance was added to the input of the process at $t = 11$ s from 0 to -1. The control with an overshoot of about 50% is seen in Figure 1.14.

The same process was controlled by GPC. All controller parameters were selected with their minimum value, $n_{e1} = 0$, $n_u = 1$, and $\lambda_u = 0$, except for the end of the control error horizon, $n_{e2} = 30$. The fast, aperiodic control is seen in Figure 1.15.

Now the level controller is designed for the boiler. Again, first a PID controller is tuned according to Chien *et al.* [1]. The inverse-response step response is approximated as an aperiodic process with dead time. The process parameters are given by the static gain $K_p = 1$, the apparent dead time $T_L = 1.1$ s, and the apparent time constant $T_T = 0.8$ s. The controller parameters are as follows:

$$K_c = \frac{0.6}{K_p} \cdot \frac{T_T}{T_L} = \frac{0.6}{1} \cdot \frac{0.8\,\text{s}}{1.1\,\text{s}} = 0.436\;;$$
$$T_I = T_T = 0.8\,\text{s}\;; \quad T_D = 0.5 \cdot T_T = 0.5 \cdot 1.1\,\text{s} = 0.55\,\text{s}\;.$$

The set value and the disturbance were changed as before. The control behavior became significantly slow and oscillating (Figure 1.16).

The same process was controlled by GPC. The controller parameters were selected as before with the tank level control, only the control error horizon started at $n_{e1} = 11$ because of the apparent dead time $T_L = 1.1$ s. The control became fast and aperiodic (Figure 1.17).

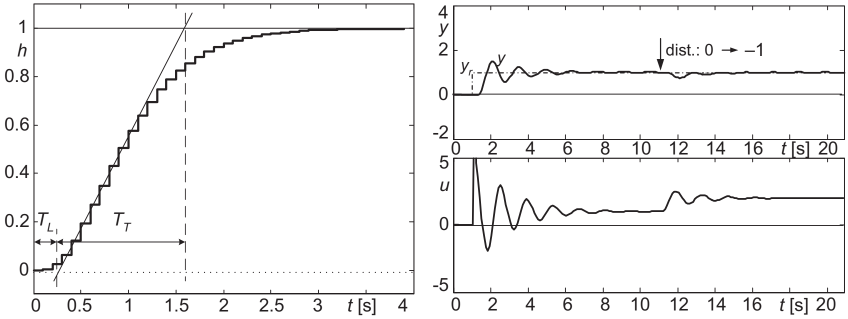


Figure 1.14 PID control of an aperiodic process (tank level).

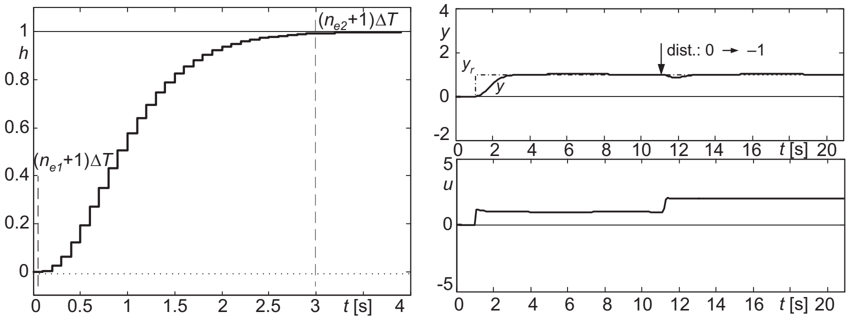


Figure 1.15 GPC of an aperiodic process (tank level).

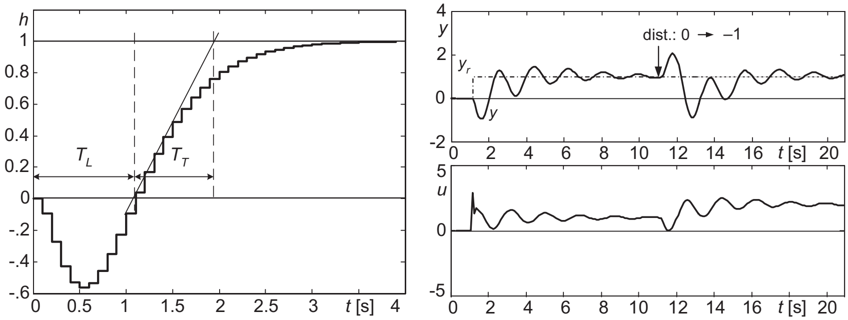


Figure 1.16 PID control of an inverse-response process (boiler level).

The example shows that GPC can be tuned more easily than a PI(D) controller and the controller parameters can be derived from the physical parameters of the step response.

- The control error horizon should be started immediately after the dead time (with inverse-response characteristics the time duration of the initial inverse response is considered as a dead time).

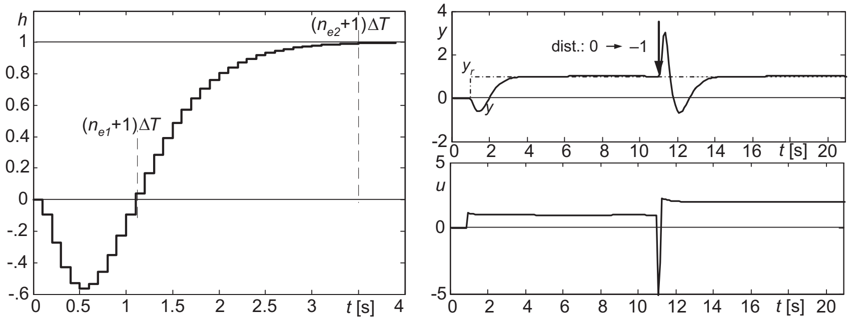


Figure 1.17 GPC of an inverse-response process (boiler level).

- The control error horizon should be finished at the settling time of the open-loop step response (it can be chosen to be longer, but this has no effect on the control behavior any more).

1.8
Control of Different Linear SISO Processes

Predictive control can be used for different process types.

Example 1.8 Predictive control of different linear SISO processes

Table 1.4 shows the open-loop step responses of different linear processes to a unit step at $t = 1$ s and the control of a set value change from 0 to 1 at $t = 1$ s for given values of the tuning parameters.

1. *Oscillating process (second order)*:
   Process parameters: static gain $K_p = 1$, damping factor $\xi = 0.5$, and time constant $T_2 = 2$ s. The transfer function of the process is

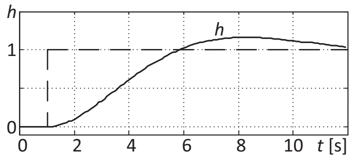
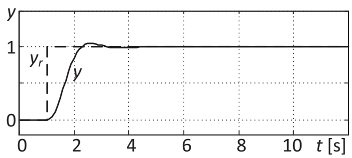
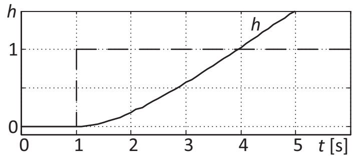
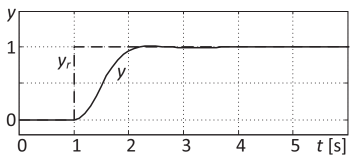
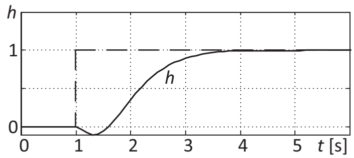
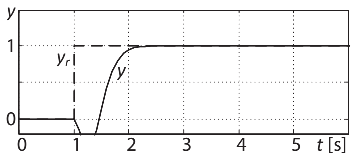
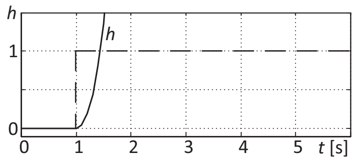
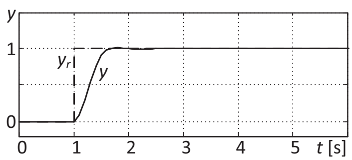
$$G(s) = \frac{1}{1 + 2 \cdot 0.5 \cdot 2s + 2^2 s^2} \ .$$

   where s denotes the Laplace operator.
   Controller parameters: $\Delta T = 0.2$ s, $n_{e1} = 0$, $n_{e2} = 19$, $n_u = 5$, and $\lambda_u = 0.01$.
2. *Integrating process*:
   Process parameters: integrating time constant $T_I = 2$ s and time constant $T_1 = 1$ s:

$$G(s) = \frac{1}{2s(1 + s)} \ .$$

   Controller parameters: $\Delta T = 0.2$ s, $n_{e1} = 0$, $n_{e2} = 19$, $n_u = 5$, and $\lambda_u = 0.01$.

Table 1.4 Control of different linear processes.

No.	Open-loop step response	Set value and controlled signal
1. Oscillating		
2. Integrating		
3. Inverse response		
4. Unstable		

3. *Process with inverse-response characteristics*:
   Process parameters: static gain $K_p = 1$, three equal time constants $T_1 = T_2 = T_3 = 1/3$ s, and time constant corresponding to non-minimum-phase zero $\tau = 1/3$ s:

   $$G(s) = \frac{1 - (1/3)s}{(1 + (1/3)s)^3} \, .$$

   Controller parameters: $\Delta T = 0.2$ s, $n_{e1} = 5$, $n_{e2} = 19$, $n_u = 3$, and $\lambda_u = 0.01$.
4. *Unstable process*:
   Process parameters: static gain $K_p = 1$ and time constants $T_1 = 1/3$ s and $T_2 = -1/3$ s:

   $$G(s) = \frac{1}{(1 + (1/3)s)\,(1 - (1/3)s)} \, .$$

   Controller parameters: $\Delta T = 0.2$ s, $n_{e1} = 0$, $n_{e2} = 9$, $n_u = 3$, and $\lambda_u = 0.1$.

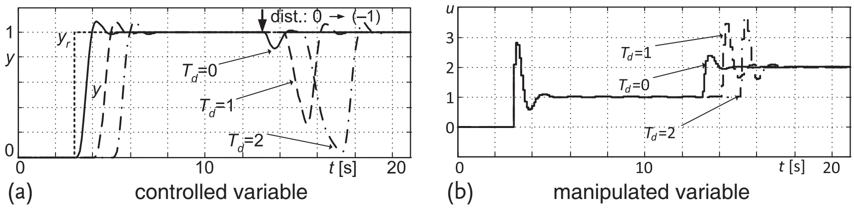


Figure 1.18 Predictive control of a linear third-order process with different dead times without knowing the future reference signal.

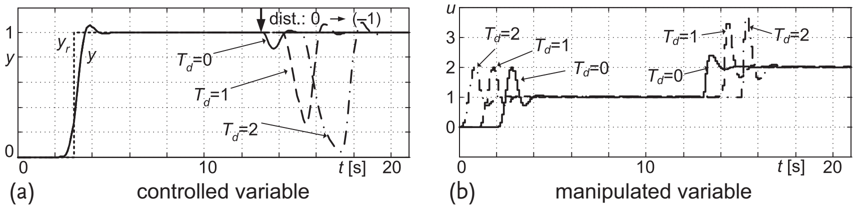


Figure 1.19 Predictive control of a linear, third-order process with dead time with knowledge of the future reference signal.

As can be seen, different processes can be controlled fast and nearly without overshoot. Control of the unstable process is possible without any problems.

The next example shows the control of a dead-time process.

Example 1.9 Control of processes with different dead times

Figure 1.18 shows the predictive control of the same linear third-order process without dead time ($T_d = 0$) as in Example 1.2. As the future course of the reference signal is not known, the manipulated signal changes only after the change in the set value. Furthermore, the control of the same process with dead time is also shown for $T_d = 1$ s and $T_d = 2$ s. As is seen, the controlled signal is shifted by the dead time and the manipulated variable is the same in all cases until the time point when the disturbance appears.

The simulations are repeated for the case when the future course of the reference signal is known, see Figure 1.19. Now, all controlled signals for the reference tracking control are identical and the manipulated variable starts before the set value change by the process dead time.

A process with known dead time can be controlled as if the dead time were not present if the future reference values are known in advance. (This is also valid for the disturbance signal, but in the simulated cases the disturbance was not assumed to be known in advance.) Predictive control has – in this respect – features similar to those of the Smith predictor. However, predictive control is more robust than the Smith predictor (see also Section 1.12).

1.9 Control of Different Linear MIMO Processes

The next example shows how easily a MIMO process can be controlled according to the predictive control principle minimizing a quadratic cost function. The simulated process is of two-input, two-output (TITO) type. Both terms of the cost function of the SISO process (1.1) had to be extended by similar terms for the second manipulated and controlled variable.

Example 1.10 Predictive control of different TITO processes

Figure 1.20 shows three different structures:

1. two SISO processes,
2. a TITO process,
3. another TITO process where the main channels and the coupling terms are interchanged.

All subprocesses have the same three time constants $T_1 = T_2 = T_3 = 1/3$ s. The static gains of the coupling terms are smaller than those of the main channels, $K_{p11} = 1$, $K_{p12} = 0.25$, $K_{p21} = 0.5$, $K_{p22} = 1$, and the dead times of the coupling terms are bigger than those of the main channels, $T_{d11} = 0.4$ s, $T_{d12} = 0.8$ s, $T_{d21} = 1.0$ s, $T_{d22} = 0.6$ s.

With traditional PI(D) design, first a decoupling has to be designed and afterwards the PI(D) parameters have to be determined. With predictive control, solely the start and end points of both control error horizons and the lengths of the manipulated variable horizons have to be given – in addition to the process model. The sampling time was $\Delta T = 0.2$ s and the controller parameters were chosen for both variables as in Example 1.2 for the SISO third-order process: $n_{e1} = 0$, $n_{e2} = 9$, $n_u = 3$, and $\lambda_u = 0.1$.

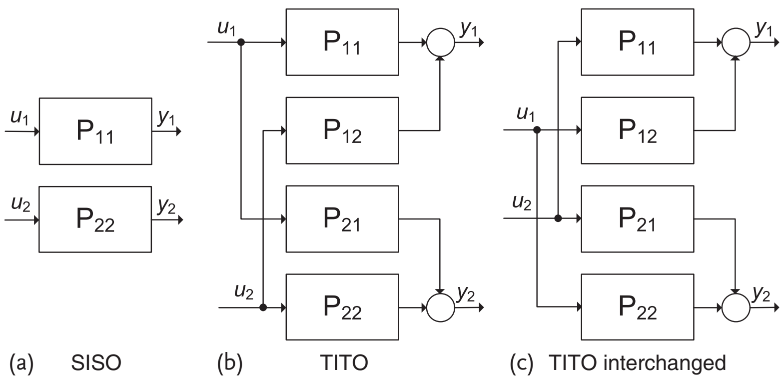


Figure 1.20 SISO and different TITO processes.

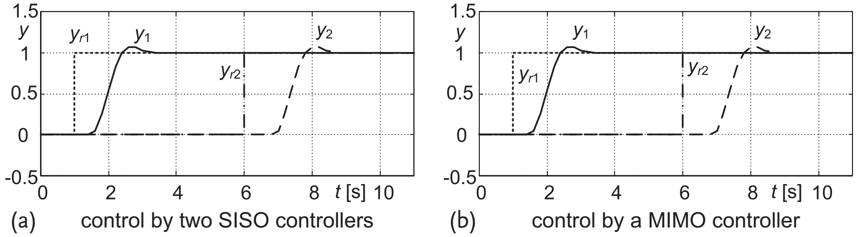


Figure 1.21 Predictive control of two SISO processes (process (a) in Figure 1.20).

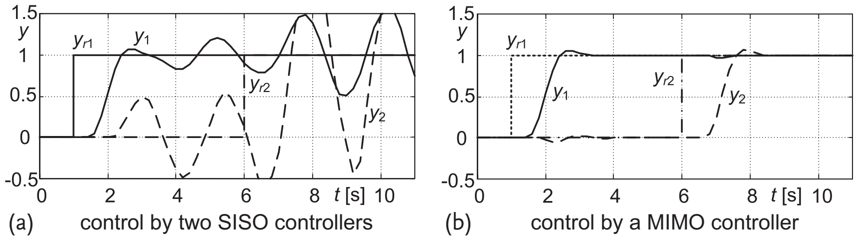


Figure 1.22 Predictive control of the TITO process (process (b) in Figure 1.20).

In the sequel two cases are compared:

- control by two independent (decentralized) SISO controllers,
- control by a TITO controller.

Figure 1.21 shows the SISO and MIMO predictive control of the two noncoupled processes. Of course, there is no difference between the control behaviors.

Figure 1.22 shows the SISO and MIMO control of the two coupled processes. As expected, the SISO control is bad. With MIMO control the controlled signals are very similar to those of the uncoupled case; the MIMO predictive control decouples the process automatically.

Figure 1.23 shows the SISO and MIMO control of the coupled processes if the main and coupling terms are interchanged. Because of the longer dead times of the coupling terms than those of the main channels, the SISO control is very bad (unstable). The MIMO control results in the same control behavior as in case when the main and coupling channels were not interchanged. The only difference is in the course of the manipulated variables (not shown here).

Example 1.10 has shown that

- a predictive controller can be tuned very easily for MIMO processes using similar controller parameters now for more variables,
- predictive MIMO control decouples the controlled variables and the degree of the decoupling can be influenced by the choice of the controller parameters (not shown here).

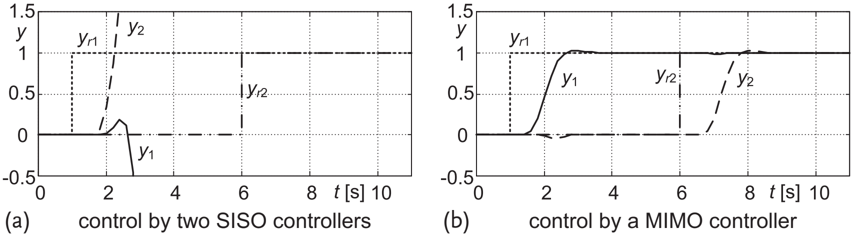


Figure 1.23 Predictive control of the TITO process (process (c) in Figure 1.20).

1.10 Control of Nonlinear Processes

Any predictive controller consists of a predictor, as shown in Figures 1.4 and 1.12. As a prediction can be performed by repeated simulations, this technique can be used both for linear and for nonlinear processes. (For linear systems more effective ways exist to calculate the predicted model output. This technique is dealt with later on.)

Example 1.11 Prediction of a linear and a nonlinear model

Consider the first-order nonlinear (bilinear) difference equation

$$y(k) = -a_1 y(k-1) + b_1 u(k-1) + c_1 u(k-1) y(k-1) \,.$$

The one-step-ahead prediction is a similar equation:

$$y(k+1) = -a_1 y(k) + b_1 u(k) + c_1 u(k) y(k) \,.$$

The two-steps-ahead prediction is obtained by shifting the one-step-ahead predictive equation and substituting $y(k+1)$ from the one-step-ahead predictive equation:

$$\begin{aligned} y(k+2) &= -a_1 y(k+1) + b_1 u(k+1) + c_1 u(k+1) y(k+1) \\ &= -a_1 \left[-a_1 y(k) + b_1 u(k) + c_1 u(k) y(k)\right] \\ &\quad + b_1 u(k+1) + c_1 u(k+1) \left[-a_1 y(k) + b_1 u(k) + c_1 u(k) y(k)\right] . \end{aligned}$$

This equation is predictive, as $y(k+2)$ depends only on known measured values $y(k)$ and $u(k)$ and a future input value $u(k+1)$. The method for calculation of the predictive equation is the same for the nonlinear case $c_1 \neq 0$ and for the linear case $c_1 = 0$.

1.11 Control under Constraints

During the control different variables can be constrained. Some typical constraints concerning the manipulated and the controlled variable are shown in Figure 1.24. Another variable, for example, a state variable, can be restricted as well.

In this case the cost function (1.1) should be minimized under constraints. Alternatively, the cost function can be extended by a quadratic term that weights the constraint violation. In the case of the so-called soft constraints, the unconstrained minimization of the cost function is an easy task.

Example 1.12 Predictive control with constraints

Figure 1.25 is the same as Figure 1.5 for Example 1.10 where a linear third-order process was controlled without constraints.

In Figure 1.26 the manipulated signal is limited to the interval $0 \le u(k) \le u_{up} = 2$. As is seen, the control becomes a bit slower, mainly the compensation of the disturbance after $t = 6$ s; however, no steady-state error occurs.

In Figure 1.27 the controlled output signal is limited to the interval $0 \le y(k) \le y_{up} = 0.9$. As expected, the controlled variable never achieves the desired set value.

In both simulated cases constraint handling was very effective and the control remained relatively fast. As with predictive control, the future course of the manipulated and controlled variables is simulated in every step, and constraint handling can be performed online with little additional computational demand. However, in this case GPC does not have an analytical form.

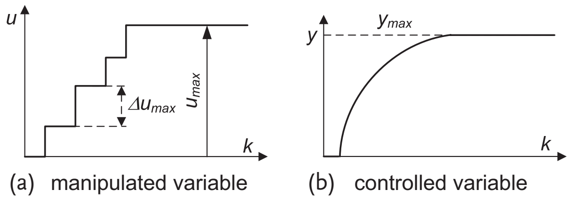


Figure 1.24 Constraints.

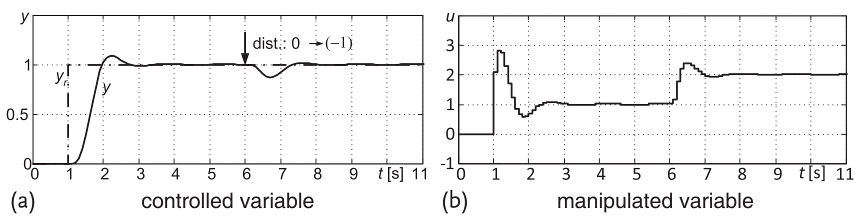


Figure 1.25 Predictive control of a linear third-order process without constraints.

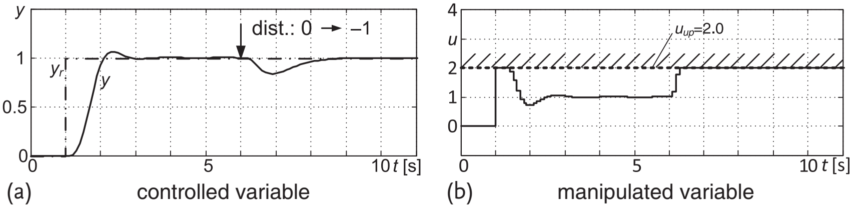


Figure 1.26 Predictive control of a linear third-order process if the manipulated variable is limited below 2.

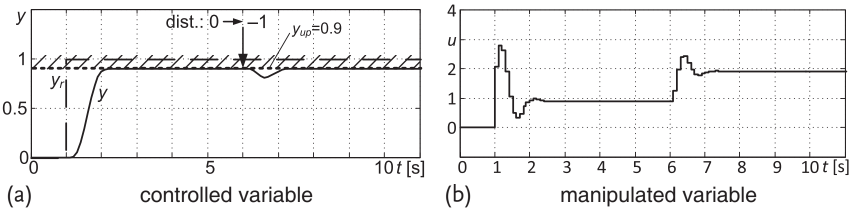


Figure 1.27 Predictive control of a linear third-order process if the controlled variable is limited below 0.9.

1.12 Robustness

Predictive control is usually more robust to parameter changes than PI(D) control.

Example 1.13 Comparison of predictive control and PID control

This example shows control of the same linear third-order process as in Example 1.2 but with dead time $T_{d,m} = 1$ s. The weighting factor is raised from 0.1 to 10.0 to make the GPC similar to PID control. The GPC tuning parameters are $n_{e1} = 0$, $n_{e2} = 9$, $n_u = 3$, and $\lambda_u = 10$. The PID control is tuned manually ($K_p = 0.7$, $T_I = 1.2$ s, and $T_D = 0.5$ s), because the usual tuning rules do not work very well for processes with dead time. Figure 1.28 shows the control with the process dead time equal to the model dead time $T_{d,m} = 1$ s and with a value increased by 10%, $T_{d,p} = 1.1$ s.

Both controls show mostly the same sensitivity to the parameter change, even if the PID controller generally produces more oscillations, bigger manipulated variable changes (the initial change was from 0 to 4.22), and needs more considerations for the tuning.

The next example shows a special case: the control of a dead-time process with a PID controller using a Smith controller vs. predictive control.

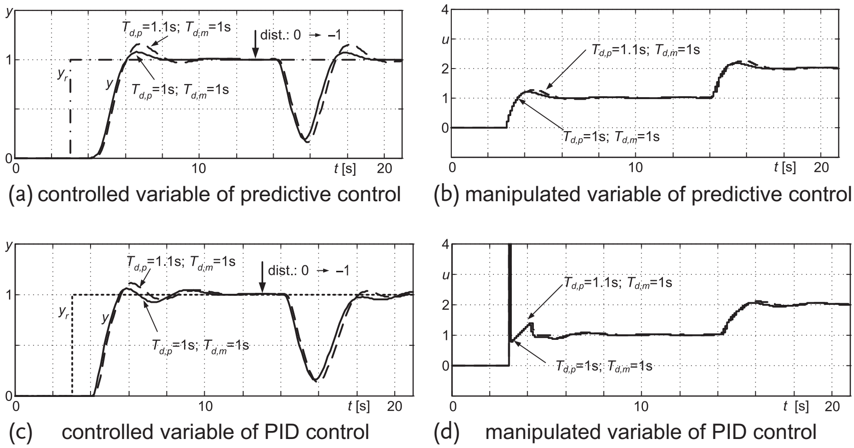


Figure 1.28 Control of a linear third-order dead-time process with dead-time mismatch.

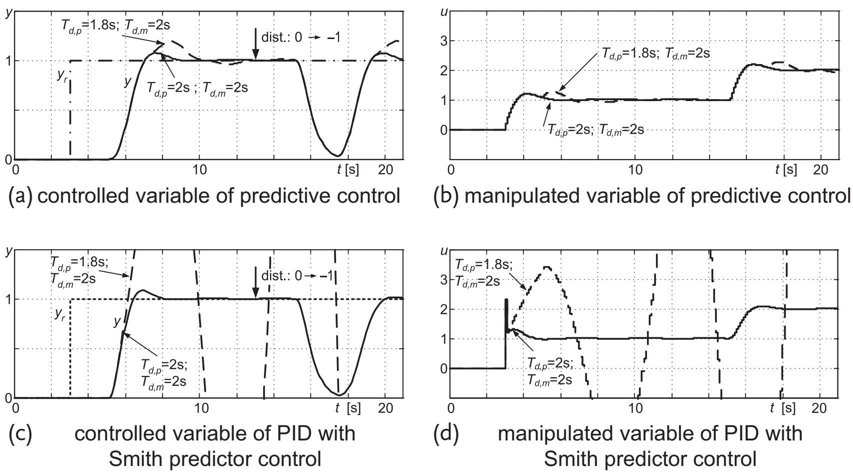


Figure 1.29 Control of a linear third-order dead time process and with dead-time mismatch.

Example 1.14 Comparison of predictive control and PID control using a Smith predictor

Figure 1.29 shows the predictive control of the same linear third-order process as in Example 1.2 but with dead time $T_{d,m} = 2\,\mathrm{s}$ and the weighting factor raised from 0.1 to 10.0 to make the GPC as slow as the PID control. The predictive controller parameters are as in Example 1.2. The same process was controlled by a PID controller with a Smith predictor as well. The PID controller was tuned for fast aperiodic control according to the tuning rule of the *T*-sum rule of Kuhn [3]:

$$K_p = 1.0\,; \quad T_\Sigma = 1.0\,\mathrm{s}\,; \quad K_c = \frac{1}{K_p} = \frac{1}{1.0} = 1.0\,;$$

$$T_I = \frac{2}{3} \cdot T_\Sigma = \frac{2}{3} \cdot 1.0\,\mathrm{s} = \frac{2}{3}\,\mathrm{s}\,; \quad T_D = \frac{1}{6} \cdot T_\Sigma = \frac{1}{6} \cdot 1.0\,\mathrm{s} = \frac{1}{6}\,\mathrm{s}\,.$$

Here T_Σ is the sum of the time constants.

Now, the simulation was repeated with the dead time of the processes reduced by 10% (the model remained unchanged) ($T_{d,m} = 2\,\mathrm{s}$, $T_{d,p} = 1.8\,\mathrm{s}$).

What Example 1.14 shows is generally valid: predictive control is usually more robust than PI(D) control, not only if a Smith predictor is used. (There are some methods that ensure enhanced robustness for both predictive and PI(D) control algorithms.)

1.13 Summary

The above-mentioned considerations, the industrial experiences, and the literature show that predictive control is to be preferred to PID control if:

- the future course of the reference signal is known,
- the future course of the disturbances is known,
- the the process has a long dead time,
- the process has inverse-response (non-minimum-phase) characteristics,
- the process is unstable,
- constraints are to be considered,
- the process is nonlinear,
- the process parameters may change during the control,
- several control variables are to be controlled simultaneously,
- decoupling of a MIMO process is desired.

The advantages of predictive control are as follows:

- simple controller tuning based on physical process parameters,
- robust behavior against model parameter and disturbance changes,
- applicable both for input/output and for state space models,
- nonparametric models, such as finite impulse response and finite step response models, can be used,
- predictive control works with physically interpretable parameters and therefore this algorithm can be easily understood by engineers and operators.

References

1 Chien, K.L., Hrones, J.A., and Reswick, J.B. (1952) On the automatic control of generalized passive systems. *Transactions ASME*, **74**, 175–185.

2 Clarke, D.W., Mohtadi, C., and Tuffs, P.S. (1987) Generalized predictive control. Part I. The basic algorithm. *Automatica*, **23**(2), 137–148.

3 Kuhn, U. (1995) Eine praxisnahe Einstellregel für PID-Regler: Die T-Summenregel (A practical tuning rule for PID controllers: the T-sum rule). *Automatisierungstechnische Praxis*, **37**(5), 10–16.