5

From Basic to Advanced PI Controllers: A Complexity vs. Performance Comparison

Aldo Balestrino, Andrea Caiti, Vincenzo Calabró,
Emanuele Crisostomi and Alberto Landi
Department of Energy and Systems Engineering,
University of Pisa
Italy

1. Introduction

The history of PIDs dates back to the beginning of the twentieth century when preliminary works of [Sperry (1922)] and [Minorski (1922)] provided mathematical results for the control of the ship motion and of automatically steered bodies in general. In particular, Minorski was the first to introduce three-term controllers with Proportional-Integral-Derivative (PID) actions. The success of PID was fast and solid, as nowadays they still represent the most popular choice in most industrial process control applications. The main reasons for their success are:

- **Reduced number of parameters:** A process control engineer only has to tune a small number of parameters to make the PID work effectively.
- **Well-established tuning rules:** There are predefined and well-known methods for deciding the values of the PID regulators. The most popular methods were given by [Ziegler and Nichols (1942)], [Astrom and Hagglund (1984)], and more recently [Zhuang and Atherton (1993)] and [Luyben and Eskinat (1994)]. Different tuning methods are due to different control objectives (e.g. reference following, disturbance rejection) and different plants (e.g. first-order model, second-order model).
- **Good performances:** The main reason for PID success is of course that good control performances are usually obtained; thus, the control engineer might not be interested in developing more complicated and less intuitive control schemes to improve something that is already working fine.

The ideal equation of a PID controller is

$$u_{PID}(t) = k_p\left(r(t) - y(t)\right) + k_p\left\{\int_0^t \left[\frac{1}{T_i}\left(r(\tau) - y(\tau)\right)\right] d\tau + T_d \frac{d\left(r(t) - y(t)\right)}{dt}\right\}, \qquad (1)$$

where k_p is the proportional gain, k_p/T_i is the integral gain (sometimes denoted as k_i) and k_pT_d is the derivative gain (sometimes denoted as k_d). According to conventional notation $r(t)$, $y(t)$ and $u(t)$ denote respectively the reference, output and input signals. The error signal $r(t) - y(t)$ is sometimes denoted as $e(t)$. The classic feedback structure involving a PID controller is shown in Figure 1. Equation (1) is sometimes considered an ideal equation for PIDs, as

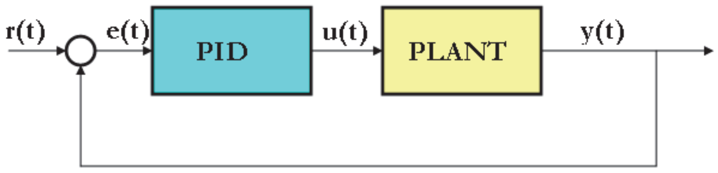


Fig. 1. Classic Feedback PID Control Scheme

usually a few tricks are required to avoid some typical well-known problems associated with ideal PIDs. For instance, a more realistic equation for PIDs is

$$u_{PID}(t) = k_p\left(b \cdot r(t) - y(t)\right) + k_p\left\{\int_0^t \left[\frac{1}{T_i}\left(r(\tau) - y(\tau)\right)\right] d\tau - T_d \frac{dy(t)}{dt}\right\}, \tag{2}$$

where the two main differences with the ideal equation (1) are the introduction of a set-point weighting factor b [Rasmussen (2009)] and the absence of the reference in the derivative term. The term b is used to mitigate kicking phenomena, that usually occur when there is an abrupt change in the reference signal $r(t)$. In the same circumstance, the derivative term of (1) is even larger, and a possible precaution is to relate the derivative only to the output signal. Assuming that only piecewise constant reference signals should be followed, the derivative term of (2) is equal to that of (1) after the transient stage.

Even the Equation (2) of a realistic PID is not always enough to obtain good control performances. One of the main drawbacks is that the PID's parameters are fixed, therefore if the plant's parameters vary in time then the fixed PID can not always represent the optimal solution. A simple way to deal with this problem is to use a relay auto-tuning procedure, i.e. the behaviour of the plant is continuously monitored, and the PID's parameters are automatically tuned in reaction to plant's changes.

A second drawback of conventional PIDs is that usually they are tuned either to have good tracking performances or to have good disturbance rejection. Of course, in practice both properties are desirable at the same time, thus requiring the design of suitable filters that separate the high-frequency components of noise from the low-frequency components of the reference signal. Alternatively, it is possible to design two PIDs which are optimal with respect to reference tracking and disturbance rejection respectively, and a smart device that switches between the two PIDs according to the most important objective in the particular moment.

A last problem of PIDs is that an anti-windup scheme must be adopted, especially in the common case that the inputs to the actuators are bounded by physical constraints, see for instance the tutorial [Peng et al. (1996)].

A consequence of the previous remarks is that some modifications to the basic structure of the PIDs might be required to achieve better control performances. A classic improvement is obtained by considering PID-like regulators whose parameters are not fixed, but are allowed to be time variant to maintain good control performances in different operating conditions. Moreover, it is clear that time varying parameters introduce more degrees of freedom in the control design, and in principle more flexibility in shaping the control response. Time variant PIDs can be designed according to several approaches well known in the literature: (a) fuzzy PIDs [Tang et al. (2001)]; in this case typically several fixed conventional PID controllers are designed for different operating conditions, and are then interpolated according to a set of fuzzy logic rules. Alternatively, fuzzy rules are used only to improve one component of the PID, for instance the set-point weighting term as in [Visioli (2004)]. (b) nonlinear PIDs; see

for instance [Haj-Ali and Ying (2004)] for a comparison with fuzzy PIDs. (c) variable structure PIDs, see for instance [Scottedward Hodel and Hall (2001)] or [Visioli (1999)] where a classic PID is modified by adding a feedforward term.
In this spirit, this work compares a traditional PI with three parameter varying PIs, characterised by an increasing level of complexity. As a support to the evalution of the control performance of each of the four regulators, conventional control indices, such as the integral of the absolute value or the square of the error signal, and the integral of the absolute value of the input and its derivative, have been used, as further detailed in Section 4. The final objective is to establish some thumb rules that quantify how much (or when) it is convenient to complicate the original conventional PI, in terms of improved control performances.

The discussion in this paper is restricted to PIs rather than PIDs for two main reasons:

- Many industrial controllers are simple PIs.
- The design of the derivative action simply follows the design of the other components. Therefore the derivative component can be introduced without affecting the general results of this work. Simple rules to introduce the derivative component can be found in the recent reference [Leva and Maggio (2011)].

Furthermore, the discussion is here restricted to linear time invariant systems, as in practice most industrial plants can be represented in such a form, eventually after a linearisation step around the desired operating point. While PI are successful in most stabilization and control problems, from a theoretical perspective there is a class of linear systems which are not stabilizable via output PI feedback; however, such examples will not be trated in this chapter.
This paper is organised as follows: next section introduces the 3 PIs that will be thoroughly compared with a conventional PI in benchmark examples. Section 3 illustrates the tuning procedures for all PIs. Section 4 compares the PI regulators in a challenging noisy reference tracking example, while Section 5 is dedicated to a realistic example. In the last section final conclusions are given and future work is outlined.

2. PI-like regulators

This sections presents the four regulators that will be later compared in challenging control problems.

2.1 Conventional PI controller

The conventional PI is described by Equation 1, without the derivative term, i.e.

$$u_{PI}(t) = k_p \left(b \cdot r(t) - y(t)\right) + k_p \left\{ \int_0^t \left[\frac{1}{T_i} \left(r(\tau) - y(\tau)\right) \right] d\tau \right\}, \tag{3}$$

where all parameters are constant.

2.2 Variable Integral PI

A simple modification of the standard PI was proposed in [Balestrino et al. (2009)] and is here reproposed as a term of comparison as it represents a good trade-off between performances and complexity. In particular, the Variable Integral PI (VIPI), provides a control action equal to

$$u_{VIPI}(t) = k_p \left\{ (b \cdot r(t) - y(t)) + \int_0^t \left[\frac{1}{T_i} \left(e(\tau) \cdot \exp^{-\frac{e(\tau)^2}{2\sigma^2}} \right) \right] d\tau \right\}, \tag{4}$$

where σ is a further tuning parameter. The main difference with Equation (3) is that the integral gain is not simply k_p/T_i, but $k_p/T_i \cdot \exp^{-\frac{e(\tau)^2}{2\sigma^2}}$, and thus it is not fixed anymore, but depends on the instant error. The motivation of Equation (4) is that the integral gain is required to obtain a zero steady-state error, and therefore a precise and accurate tracking of the reference signal. However, in the transient stage when the error is large, it might be useless or at least misleading to perform accurate and refined control actions using the integral component. Moreover, this use of the integral action might also lead to windup problems. Accordingly, Equation (4) encourages the use of the proportional action to get close to the reference signal (as in presence of large errors the integral gain is close to zero), while the integral action recovers the nominal value k_p/T_i when the error is close to zero. For this purpose, the tuning parameter σ^2 plays the role of choosing when the integral action has to play an important role, as it corresponds to the variance of the Gaussian distribution centered around zero error, as in Figure 2. A consequence of VIPI is that the control effort is always inferior to that of a conventional PI as the integral action can never be greater than the nominal one.

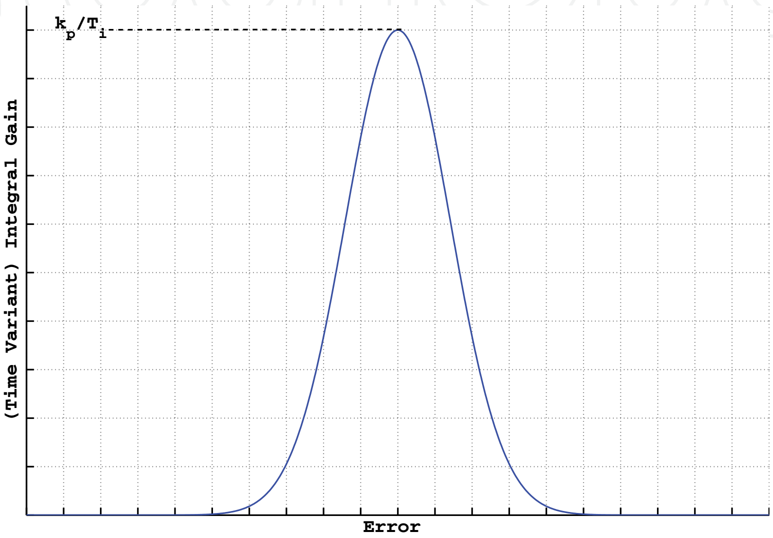


Fig. 2. The value of the integral gain depends on the error

2.3 API controller

The acronym API denotes an Adaptive PI controller, whose parameters are not fixed but change, i.e. adapt, in reaction to different operating conditions (e.g. plant parameter changes, ageing phenomena, input uncertainties). Adaptive controllers have a long history, but their use in practical applications has been limited by their usually high demanding computational requirements; indeed, computations like matrix inversions can not be easily embedded in real-time applications, as they might cause runtime faults and consequently even lead to plant damages.

More recently, adaptive techniques have been applied to PI controllers, as their simple structure is very attractive, by including additional adaptive terms to extend and robustify such controllers. One example is given by [Fisher (2009)], where the authors compare three different controllers: a classic PI, an Adaptive PI and a P-FI which is a Proportional+Fuzzy Integral term controller. In this paper, we use the second controller (API) as a term of comparison in our examples, because it is characterised by accurate and robust tracking performances. The main property of adaptive controllers is that parameters are not fixed, but vary in time searching for an optimal configuration. In [Fisher (2009)] the controller parameters update law is described by

$$\dot{k}_p = -\gamma_p k_p + \beta_p e^2 \tag{5}$$

$$\dot{k}_i = -\gamma_i k_i + \beta_i e \int_0^t e(\tau) d\tau \tag{6}$$

with positive constant parameters $\gamma_p, \gamma_i, \beta_p, \beta_i$; the resulting control law is as usual

$$u_{API}(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau \tag{7}$$

The rationale of this adaptive PI control is that the updating law is composed by a

$$\text{dissipative term} \begin{cases} -\gamma_p k_p \\ -\gamma_i k_i \end{cases} \tag{8}$$

and an

$$\text{anti-dissipative term} \begin{cases} \beta_p e^2 \\ \beta_i e \int_0^t e(\tau) d\tau \end{cases} . \tag{9}$$

The *dissipative term* is used to decrease the value of the corresponding gain, once that the *anti-dissipative terms* becomes small. For instance, a large error will cause an increase of the proportional gain through the anti-dissipative term; thus the error will decrease, and when close to zero ($e \approx 0$), the proportional gain decreases exponentially with decay rate γ_p.

2.4 FAPI controller

Similarly to many other recent approaches, we also propose here a Fuzzy variant of the Adaptive PI (FAPI). Fuzzy approximation property has been widely and successfully used in robotics and control theory, to handle model uncertainties and external unpredictable disturbances. A large number of controllers use the Wang universal approximation theorem [Wang (1997)], to design nonlinear integral terms to improve performance indices and address robustness issues. However, in many cases, as shown in [Fisher (2009)], the involved additional computational efforts do not match significative performance improvements, thus not making fuzzy techniques particularly attractive.

Here we present a different novel approach to fuzzy controllers, where the simplicity of the conventional PI regulator, the interesting idea of the VIPI integral action and the robustness properties of adaptive PI controllers, are all combined together into a single Fuzzy-Adaptive PI Control (FAPI).

Next section is dedicated to recall the basic ideas of Fuzzy Logic Theory that, in the following section, will be used to implement the FAPI controller, which is one of the main contributions of this paper.

2.4.1 Fuzzy logic theory background

A fuzzy set A on a domain X is a set defined by the membership function $\mu_A(x)$ which is a mapping from the domain X into the unit interval:

$$\mu_A(\cdot) : \mathrm{X} \rightarrow [0,1]. \tag{10}$$

There are several ways to define a fuzzy set, in particular we define it here using the analytic description of its membership function $\mu_A(x) = f(x)$. For instance (see Fig. 3), the *triangular* membership function can be described as:

$$\mu(x; a, b, c) = \max\left(0, \min\left(\frac{x-a}{b-a}, 1, \frac{c-x}{c-b}\right)\right) \tag{11}$$

where a,b and c are parameters that is related to the coordinates of the triangle's vertices, whereas a Gaussian membership function can be described as

$$\mu(x; \eta, \sigma) = \exp\left[-\left(\frac{x-\eta}{\sigma}\right)^2\right]. \tag{12}$$

A static or dynamic system which makes use of fuzzy sets and the corresponding mathematical framework is called a *fuzzy system*. In order to derivate the FAPI controller updating law, it is necessary to define the intersection of fuzzy sets (connective *AND*) , obtained by considering a function $t : [0,1] \times [0,1] \rightarrow [0,1]$ that transforms the membership functions of fuzzy sets A and B into the membership function of the intersection of A and B, that is:

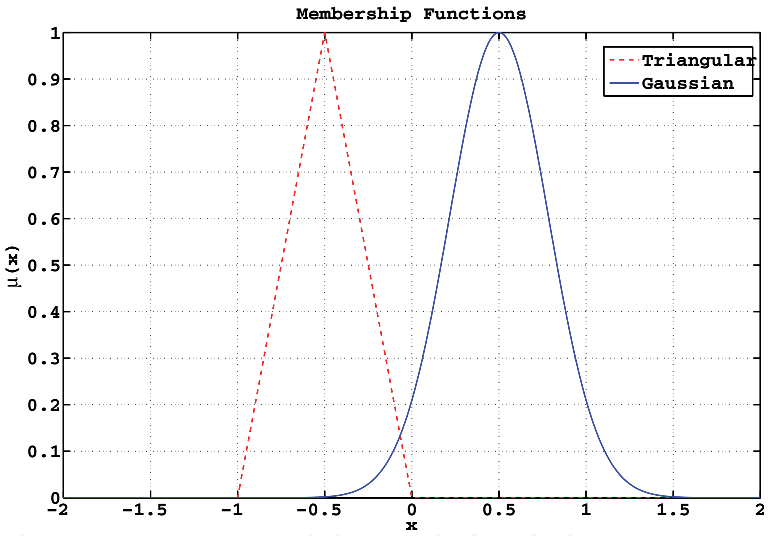


Fig. 3. Example of Membership Functions: Triangular ($a = -1, b = -0.5, c = 0$) and Gaussian ($\eta = 0.5, \sigma = 0.4$)

$$t\,[\mu_A(x), \mu_B(x)] = \mu_{A\cap B}(x). \tag{13}$$

A function t can be qualified as an intersection function, if it satisfies at least the following four requirements:

$$\begin{array}{cc} t(0,0) = 0, \quad t(a,1) = t(1,a) = a & \text{boundary condition} \\ t(a,b) = t(b,a) & \text{commutativity} \\ t(a,b) \leq t(a',b'), \quad \forall a \leq a',\ b \leq b' & \text{monotonicity} \\ t(t(a,b),c) = t(a,t(b,c)) & \text{associativity} \end{array}. \tag{14}$$

In the following analysis, the probabilistic connective *AND* will be used:

$$\mu_{A\cap B}(x) = \mu_A(x)\mu_B(x). \tag{15}$$

The most common fuzzy systems are defined by means of if-then rules: *rule-based fuzzy systems*. In the rule-based fuzzy systems, the relationships between variables are represented in the following general form:

if *antecedent proposition* **then** *consequent proposition*.

A *fuzzy proposition* is a statement like "x is big" where "big" is a *linguistic label*, defined by a fuzzy set on the universe of discourse of variable x. In the *linguistic fuzzy model* developed by [Zadeh (1978)] and [Mamdani (1977)], both the antecedent and the consequent are fuzzy propositions:

$$\mathcal{R}_i : \textbf{if } x \text{ is } A_i \textbf{ then } y \text{ is } B_i\ , \quad i = 1, ..., L, \tag{16}$$

where L is the number of propositions (rules). Here x is the input (antecedent) *linguistic variable*, and A_i are the antecedent *linguistic terms* (labels). Similarly, y is the output (consequent) linguistic variable and B_i are the consequent linguistic terms. The linguistic terms A_i,B_i are always fuzzy sets. After fuzzy theory gained popularity, many control problems have been recasted into control of Takagi-Sugeno-Kang (TSK) models:

$$\mathcal{R}_i : \textbf{if } x \textit{ is } A_i \textbf{ then } y = f_i(x)\ , \quad i = 1, ..., L \tag{17}$$

which is a particular case of the general fuzzy model (16), obtained when the consequent fuzzy sets B_i are functions of the variable x. In systems and control theory, TSK models are frequently used to model nonlinear systems over a fuzzy space. The resulting TSK model can efficiently clone the nonlinear system or alternatively, approximate it over a defined domain. For such a nonlinear systems representation, stability and synthesis of controllers and observers can be expressed in terms of Linear Matrix Inequalities, which in turn can be solved adopting convex optimization techniques as shown in [Tanaka (2001)]. It is important to mention that the output of a fuzzy system can be obtained using different *defuzzification* methods. In the remainder of this chapter we will use the following TSK model:

$$\mathcal{R}_i : \textbf{if } x_1 \textit{ is } A_{i1} \textit{ and ... } x_n \textit{ is } A_{in} \textbf{ then } y = f_i(x)\ , \quad i = 1, ..., L \tag{18}$$

where we consider that each rule has an antecedent proposition obtained by intersecting n fuzzy sets. The output can be evaluated by considering the Center of Gravity defuzzification method

$$y = \sum_{i=1}^{L} \alpha_i f_i(x) \tag{19}$$

where

$$\alpha(t) = (\alpha_1(t), ..., \alpha_L(t)), \qquad \alpha_i(t) = \frac{\beta_i(t)}{\sum_{i=1}^{L} \beta_i(t)},$$

$$\beta_i(t) = \prod_{j=1}^{n} \mu_{Aij}(x). \tag{20}$$

2.4.2 FAPI parameters update law

According to the discussion on fuzzy sets and rules introduced in the previous section, we introduce now the controller parameters update laws:

$$\text{IF error is SMALL, then } \dot{k}_p = -\beta_p k_p \tag{21}$$

$$\text{IF error is MEDIUM, then } \dot{k}_p = -\gamma_p (k_p - k_p^*) \tag{22}$$

$$\text{IF error is LARGE, then } \dot{k}_p = \alpha_p k_p^* e^2 \tag{23}$$

for the proportional gain k_p, while for the integral action we have

$$\text{IF error is LARGE, then } \dot{k}_i = -\beta_i k_i \tag{24}$$

$$\text{IF error is MEDIUM, then } \dot{k}_i = -\gamma_i (k_i - k_i^*) \tag{25}$$

$$\text{IF error is SMALL, then } \dot{k}_i = \alpha_i k_i^* e \int_0^t e(\tau) d\tau. \tag{26}$$

The main difference with respect to the API regulator is the presence of the two terms k_p^* and k_i^* that are the gains of a reference model regulator K^*. In order to compute the corresponding time-varying gain, we will consider a single Gaussian membership function $\mu_S(e)$ defined over the error domain to identify the fuzzy set SMALL (S), and also the fuzzy sets MEDIUM (M) and LARGE (L) as follows:

$$\mu_S = e^{-\left(\frac{x}{\sigma}\right)^2}, \qquad \mu_L(e) = 1 - \mu_S(e), \qquad \mu_M(e) = \mu_{S \cap L}(e) = \mu_S(e) \cdot \mu_L(e) \tag{27}$$

The philosophy of shaping the control effort on the basis of the error value is analogous to that of the previously introduced VIPI. The resulting k_p gain law is obtained as

$$\dot{k}_p = \frac{1}{1 + \mu_M(e)} \left(\alpha_p \mu_L(e) k_p^* e^2 - \beta_p \mu_S(e) k_p - \gamma_p \mu_M(e)(k_p - k_p^*) \right) \tag{28}$$

while the integral gain k_i law is

$$\dot{k}_i = \frac{1}{1 + \mu_M(e)} \left(\alpha_i \mu_S(e) k_i^* e \int_0^t e(\tau) d\tau - \beta_i \mu_L(e) k_i - \gamma_i \mu_M(e)(k_i - k_i^*). \right) \tag{29}$$

Each updating law it is composed of three terms:

$$\text{dissipative term} \begin{cases} -\beta_p \mu_S(e) k_p \\ -\beta_i \mu_L(e) k_i \end{cases} \tag{30}$$

used to decrease the (absolute) value of the gains,

$$\text{anti-dissipative term} \begin{cases} \alpha_p \mu_L(e) k_p^* e^2 \\ \alpha_i \mu_S(e) k_i^* e \int_0^t e(\tau) d\tau \end{cases} \tag{31}$$

used to increase the gain values analogously to the API control philosophy, and

$$\text{model reference tracking term} \begin{cases} -\gamma_p \mu_M(e)(k_p - k_p^*) \\ -\gamma_i \mu_M(e)(k_i - k_i^*) \end{cases} \tag{32}$$

used to force the adapting law to generate controller gains sufficiently close to the ideal controller K^*. In the end, the control law is as usual

$$u_{FAPI}(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau \tag{33}$$

In practice, when the error is large, the parameter update laws make the proportional gain increase due to its anti-dissipative term, while the integral action progressively disappears. This leads to a fast response (high proportional gain). On the other hand, when the error is small, the proportional gain is subject to the dissipative term and gets negligible values, while the integral component grows. This will result in a disturbance rejection behaviour. In any moment, good performances are guaranteed by the third term that makes the PI close to the model reference controller K^*.

Remark: Both the API controller developed in [Fisher (2009)] and the FAPI controller shown here are not symmetrical with respect to the error signal as their update rules are a function of the error, and thus depend on its sign. As a consequence, they can behave differently if the reference signal is larger or smaller than the actual output of the plant.

3. Tuning methods

3.1 Tuning of the conventional PI

In this paper we tune the conventional PI using Zhuang-Atherton optimal parameters [Zhuang and Atherton (1993)]. In particular we use the values of Table 1 of [Zhuang and Atherton (1993)], which correspond to PI tuning formulae for set-point changes in the case of first-order plus dead time plant model, optimised in order to minimise the Integral of the Square Error (ISE) signal. The set-point weighting factor is usually not used (i.e. $b = 1$), as in the examples a time-varying reference signal is used.

3.2 Tuning of the VIPI

Tuning of the VIPI is a two-step procedure:

1. Conventional tuning is first performed, and values of k_p and T_i are found according to the procedure outlined in Section 3.1.
2. The further parameter σ is computed to decide at which point the integral action should come into action. Namely, the integral action must already be active when the error is equal to the steady-state error obtained using only the proportional action.

Example :

Let us consider a plant described by the transfer function

$$G(s) = \frac{4}{s^2 + 4s + 4} \tag{34}$$

and let us design a classic PI characterised by $k_p = 6.122$, $T_i = 0.606$ and $b = 1$. Then the step response of the VIPI for different values of $\sigma = 0.1, 0.15, 0.25, 0.5, 1, 5$ are shown in Figure 4. As can be appreciated in Figure 4, the step response is contained between the one obtained using a single proportional controller, which is recovered from Equation (4) when σ tends to zero, and that of the conventional PI, which is recovered from Equation (4) when σ has large values (in practice they coincide already for $\sigma = 5$).

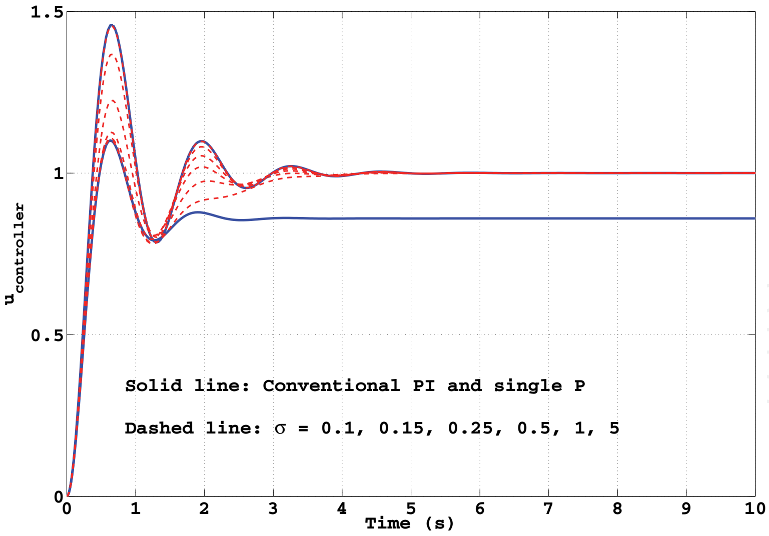


Fig. 4. Different step responses as a function of the free parameter σ of the VISI. The step response is contained between the one obtained using a single proportional controller (i.e. $\sigma \to 0$) and high values of the parameter. In this case, the step response when $\sigma = 5$ already coincides with the one obtained with the nominal PI.

3.3 Tuning of the API

Tuning of adaptive controllers is simpler than other PIs as the inner adaptive capacity allows the API to recover good performances against non optimal initial tunings. However, APIs are characterised by more degrees of freedom, e.g. parameters in the updating rules. For the purpose of the example shown in the following sections, the adaptive PI control parameters γ and β have been optimally tuned (using genetic algorithms) in order to get a good trade-off between tracking and disturbance rejection. Particular care is required to handle the anti-dissipative terms, which might yield to instability problems when a fault occurs. In fact, the anti-dissipative term should be neglected only when the error is close to zero.

3.4 Tuning of the FAPI

The FAPI controller parameters α, β, γ must be tuned, after a desired target controller K^* is chosen. In this case, we use a conventional PI tuned according to Zhuang-Atherton rules (see Section 3.1) as a reference model. Then, the parameters can be tuned keeping in mind that each parameter directly affects a different controller property:

- α: Adapting
- β: Low Gain Trend
- γ: K^* Model Reference Tracking.

Therefore, parameters are chosen in function of whether the priority objective is fast response to variations, or no overshoots or adherence to the ideal model controller. Particular care should be used in tuning α, that should be small in presence of significative system delays.

4. Comparison of the four PIs

As a preliminary comparison the step-responses of the four controllers are compared. Then, in the following sections, a more challenging example and a realistic scenario are simulated to further establish the differences among the proposed PI regulators. The step response of the four controllers is shown in Figure 5, in the case of the system plant (34). The shown comparison is performed after a transient time given to the adaptive controllers to adapt their parameters, and after Zhuang-Atherton tuning procedure for the other two controllers [Zhuang and Atherton (1993)]. The control performances of the four regulators are also compared in Table 1 to further distinguish and classify the proposed regulators, where the following well known control indices were used

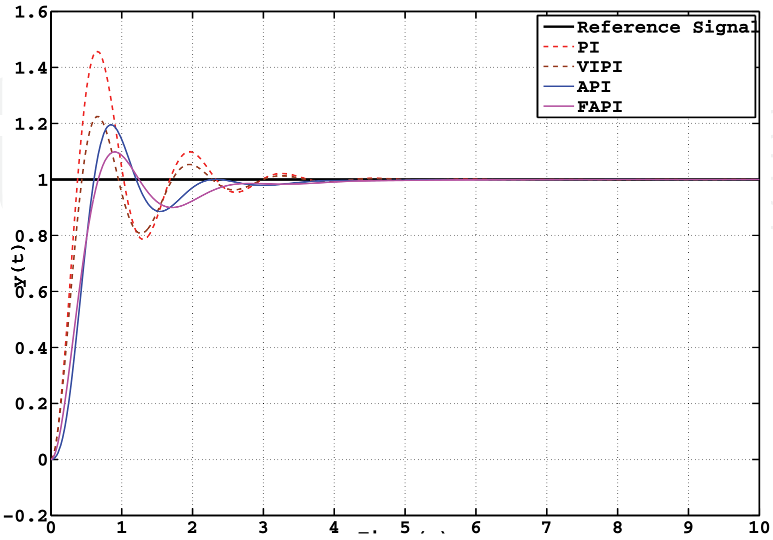


Fig. 5. Comparison of the four PI controllers in terms of the step response.

- **IAE:** Integral of the Absolute value of the Error, $IAE = \int_0^t |e(\tau)|\, d\tau$
- **ISE:** Integral of the Square Error, $ISE = \int_0^t (e(\tau))^2\, d\tau$
- **IAU:** Integral of the Absolute value of the input u , $IAU = \int_0^t |u(\tau)|\, d\tau$
- **IADU:** Integral of the Absolute value of the Derivative of the input u , $IADU = \int_0^t \left| \frac{du(\tau)}{d\tau} \right| d\tau$

	IAE	ISE	IAU	IADU
PI	0.58	0.26	16.75	22.20
VIPI	**0.46**	**0.21**	16.10	18.42
API	0.54	0.32	15.59	11.72
FAPI	0.50	0.27	**15.47**	**8.69**

Table 1. Comparison of the four controllers in terms of the Step Response. The best values of the indices have been highlighted in grey. The FAPI requires the least control effort, while the VIPI has the best overall control performances.

4.1 A more challenging example

The performances of the four controllers are again compared in a more challenging scenario where the plant transfer equation is the same (i.e. Equation (34)), but the reference signal is composed of a periodic sinusoidal component and of a pulse wave, plus a filtered Gaussian random signal $n(t)$ added to simulate sensor noise (i.e. $e(t) = r(t) - y(t) - n(t)$). As a consequence, this simulation is tailored on purpose to compare the robustness and disturbance rejection performances of the four controllers. The ability of the four controllers to track the reference signal despite the sensor noise is shown in Figure 6. Again, the comparison has been performed after some time that was required by the adaptive controllers to reach a steady-state behaviour. As illustrated in Figure 6, the conventional PI and the modified VIPI apparently have the best performance in terms of tracking, however, as better shown in Figure 7, the adaptive controllers, and especially the FAPI, are characterised by a less demanding input signal. This is particularly important because the input signal is usually required to vary slowly in time, to avoid actuators' stress.

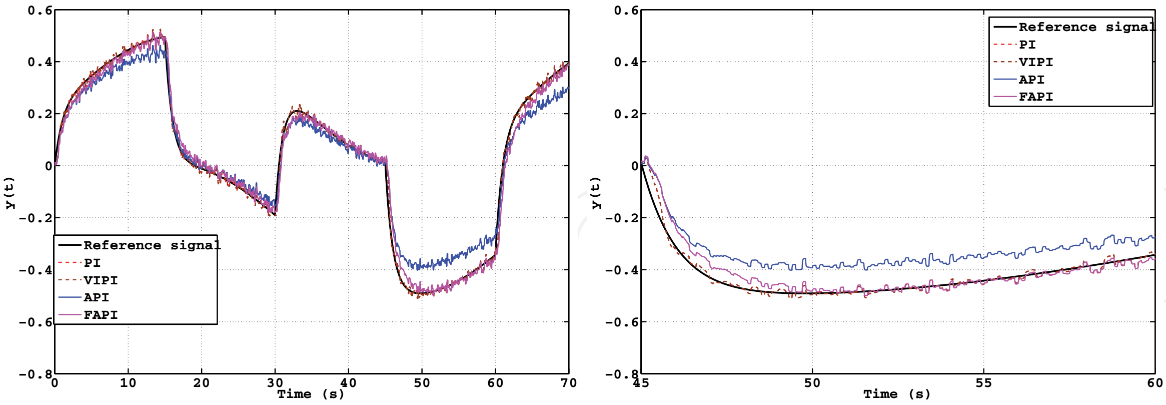


Fig. 6. Comparison of the four PI controllers in presence of a varying reference signal and sensor noise. This simulation aims at comparing the disturbance rejection abilities of the four controllers. On the left a long time interval, and a zoom is shown on the right. The API exhibits the worst tracking capabilities.

Remark: In this example, the plant is required to follow small variations of the reference signal, therefore the error is usually small and the integral action of the VIPI is constantly set to the nominal value. As a consequence, the PI and the VIPI provide (almost) identical results.

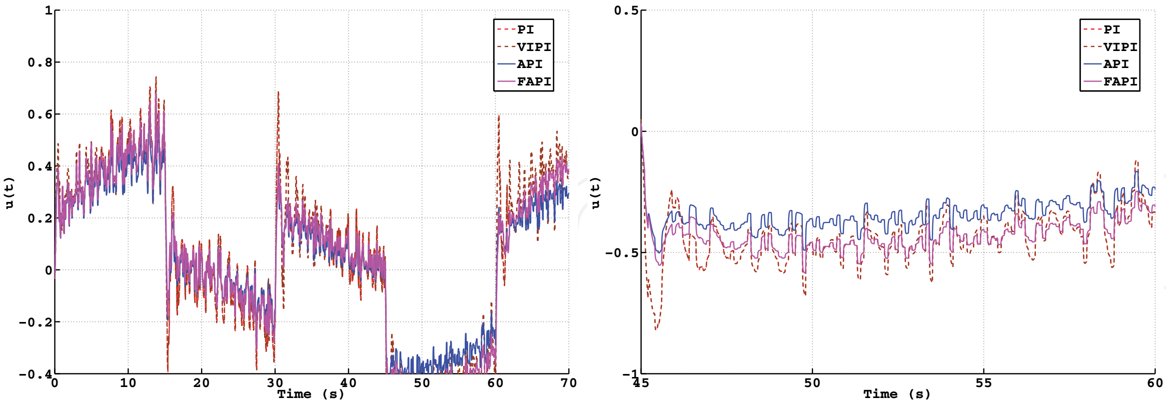


Fig. 7. Comparison of the four PI controllers in presence of a varying reference signal and sensor noise. This simulation shows the control effort of the four controllers. Clearly the FAPI is the most convenient one, as actuators are less stressed. On the left a long time interval, while on the right a shorter time interval is shown.

4.2 A realistic example: Ship course control

Let us consider a 3DoF model of a low-speed marine vessel [Fossen (2002)]:

$$M\dot{\nu} + C(\nu)\nu + D\nu = \tau + J^T(\eta)\tau_d \tag{35}$$

$$\dot{\eta} = J(\eta)\nu \tag{36}$$

where

- M represents the generalized mass-inertia matrix, including the added-masses contribution
- $C(\nu)$ contains the Coriolis-centripetal effects
- D represent the linear approximation of hydrodynamic drag
- τ is the generalized force-torque applied to the 3DoF model expressed in the body-fixed reference frame
- τ_d is an external disturbance expressed in the navigation referenceframe
- $\nu = [u, v, r]^T \in \mathbb{R}^3$ is the state variable related to the surge, sway and yaw rate speed
- $\eta = [p_n, p_e, \psi] \in \mathbb{R}^3$ represents the position and the orientation of the vessel with respect to the navigation frame
- $J(\eta)$ is the Jacobian matrix which relates body-fixed reference frame to navigation reference frame:

$$J(\eta) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{37}$$

Let us assume that the vessel is moving at constant speed u_0 , and $\sqrt{u_0^2 + v^2} \approx u_0$, then the previous 3DoF model can be decoupled into longitudinal and manoeuvring subsystems. Here we will analyse the manoeuvring subsystem in order to obtain a course control for a vessel equipped with a single rudder. For low surge speed, in addition the Eq. (35) can be approximated by:

$$\bar{M}\dot{\bar{v}} + N(u_0)\bar{v} = b\delta \tag{38}$$

where $\bar{v} = [v, r]^T$, $b = -[Y_\delta, N_\delta]^T \in \mathbb{R}^2$ and

$$\bar{M} = \begin{bmatrix} m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ mx_g - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix}, \quad N(u_0) = \begin{bmatrix} -Y_v & mu_0 - Y_r \\ -N_v & mx_g u_0 - N_r \end{bmatrix} \tag{39}$$

where the parameters Y_δ, N_δ are used to model the force and the torque generated by the rudder, $Y_{\dot{v}}, Y_{\dot{r}}, N_{\dot{r}}$ are parameters related to the added-masses, m, x_g, I_z are parameter of the rigid-body (mass, center of gravity and moment of inertia, respectively), Y_v, Y_r, N_v, N_r are coefficients related to the drag effects and δ is the rudder deflection. The equivalent state-space model of (38) can be found by observing that:

$$\dot{\bar{v}} = -\bar{M}^{-1}N(u_0)\bar{v} + \bar{M}^{-1}b\delta = A\bar{v} + B\delta \tag{40}$$

Considering the the parameters of the CyberShip II experimentally estimated in Fossen (2004), choosing a constant speed of $u_0 = 1.5m/s \approx 3knots$ and defining the output $y = r =$

$C_r\bar{\nu},\ C_r = [0,\ 1] \in \mathbb{R}^2$, the following second linear time invariant system, also referred as Nomoto $2nd$ order model is obtained:

$$G_r(s) = C_r\left(sI - A\right)^{-1}B = \frac{r(s)}{\delta(s)} = \frac{-0.09185s - 0.002137}{s^2 + 0.8165s + 0.04882} \tag{41}$$

Since the course angle derivative is related to the yaw-rate as $\dot{\psi} = r$, we can finally derive the course model for the CyberShip II as:

$$G_\psi(s) = \frac{\psi(s)}{\delta(s)} = \frac{1}{s}G_r(s) = \frac{-0.09185s - 0.002137}{s^3 + 0.8165s^2 + 0.04882s} \tag{42}$$

The controller parameters used in the course-control problem are summarised in Table 2.

	ZA	VIPI	API	FAPI
K_p^*	7.7220	7.7220	-	7.7220
$K_i^* = K_p^*/T_i^*$	0.0978	0.0978	-	0.0978
σ	-	0.5	-	0.25
β_p	-	-	1.1612	0.0087
β_i	-	-	1.1343	0.1206
γ_p	-	-	0.0151	0.1142
γ_i	-	-	0.1363	0.1671
α_p	-	-	-	0.0011
α_i	-	-	-	0.7126

Table 2. Course Control Problem: controller parameters used in the simulation.

Note that we are not handling actuator saturations and limitations of the input rate. However, in order to use efficiently those controllers with such limitations the adoption of anti-windup systems and reference filters is strongly recommended. In practice, the use of a frequency-shaped reference signal causes a smoother and less demanding control action which is expected to satisfy the actuator limitations.

The four controllers are compared in the challenging scenario described in Figure 8. In this simulation we assume that the reference signal is a desired course angle (i.e. not a step reference, as it is not realistic in this context as previously remarked). Disturbance is modeled with two components: a filtered Gaussian noise, of the order of $2-3°$; and an aperiodic square pulse which refers to unpredictable external disturbance (e.g. wave current, wind gust). It is possible to note from Figure 8 that the API controller not always provide a satisfactory tracking of the reference signal. On the other hand, the other controllers have similar good performances, but the FAPI is characterised by a reduced control effort.

5. Conclusion

This chapter gives a comparison between a conventional PI regulator tuned according to Zhuang-Atherton rules with three less conventional controllers: a variable integral component PI (VIPI), an adaptive PI (API) and a fuzzy adaptive PI (FAPI). The VIPI is characterised by one time variant parameter, i.e. the integral one, and only one more degree of freedom (the parameter σ). Both the API and the FAPI have two time variant parameters and more degrees of freedom, as for instance the dissipative and anti-dissipative coefficients that regulate the parameters' update laws.

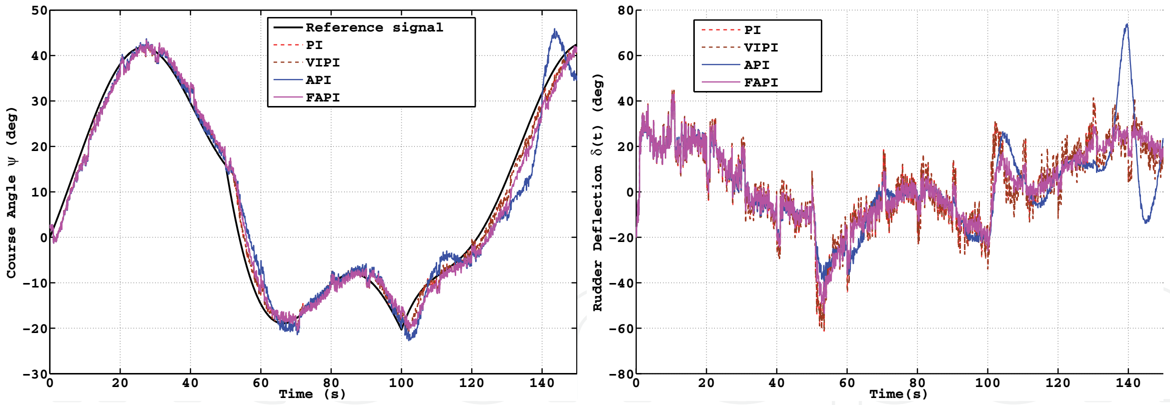


Fig. 8. Comparison of the four PI controllers in response to a course angle reference signal (on the left). Realistic disturbances are taken into consideration. On the right, the control effort of the four controllers.

Simulations show that the VIPI generally outperforms the simple PI both in terms of the control effort, which is always inferior, and in terms of settling time. The VIPI is very convenient, as it only contains one more parameter than the conventional PI, and better performances are usually achieved without requiring a complex tuning procedure for the extra parameter. On the other hand, the adaptive controllers require a more laborious tuning procedure (as more parameters are involved), and not always the control performance is so satisfactory, especially for the API, at least for the proposed examples. However, the FAPI, although provides similar control results to the PI and the VIPI, is characterised by a reduced small effort, both in terms of the absolute value and its derivative; for this reason it is particularly suitable in particular control applications: for instance when control components with moving parts are involved (e.g. valves) frequent fluctuations of the control action should be avoided to skip the high expenses of valve wear and maintenance programs.
Ongoing and future work will follow several directions:

- Robustness performances will be further investigated, so to account for time variant process plants. In some industrial applications, the plant coefficients change according to different factors (e.g. temperature, age, wear and tear of the machines).
- The controllers can be further compared on their ability to prevent wind-up phenomena.
- The proposed framework can be easily extended to decentralised Multiple Input Multiple Output (MIMO) control problems.
- The FAPI controller exhibits the best performance in terms of control effort, and for this reason it will be used in a real application in underwater robotics.

6. References

E. Sperry, *Automatic steering*, Society of Naval Architects and Marine Engineers, 1922.

N. Minorski, *Directional stability of automatically steered bodies*, Journal of American Society of Naval Engineers, 1922.

J. Ziegler and N. Nichols, *Optimum settings for automatic controller*, Trans. ASME, vol. 75, pp.827–833, 1942.

K. Astrom and E. Hagglund, *Adaptive tuning of simple regulators with specifications on phase and amplitude margins*, Automatica, vol. 20, pp. 645–651, 1984.

M. Zhuang and D.P. Atherton, *Automatic tuning of optimum PID controllers*, IEE Proceedings D on Control Theory and Applications, vol. 140, pp. 216–224, 1993.

W. Luyben and E. Eskinat, *Nonlinear auto-tune identification*, International Journal of Control, vol. 59, pp.595–626, 1994.

H. Rasmussen, *Automatic tuning of pid-regulators*, Textbook, Department of Control Engineering, Aalborg University, Denmark, 2009.

Y. Peng, D. Vrancic and R. Hanus, *Anti-windup, bumpless, and CT techniques for PID controllers*, IEEE Control systems magazine, vol. 16, pp.48–56, 1996.

K. Tang, K. Man, G. Chen and S. Kwong, *An optimal fuzzy PID controller*, IEEE Transactions on Industrial Electronics, vol. 48, pp. 757–765, 2001.

A. Visioli, *A new design for a PID plus feedforward controller*, Journal of Process Control, vol. 14, pp. 457–463, 2004.

A. Haj-Ali and H. Ying, *Structural analysis of fuzzy controllers with nonlinear input fuzzy sets in relation to nonlinear PID control with variable gain*, Automatica, vol. 40, pp. 1551–1559, 2004.

A. Scottedward Hodel and C.E. Hall, *Variable-Structure PID control to prevent integrator windup*, IEEE Transactions on Industrial Electronics, vol. 48, no. 2, 2001.

A. Visioli, *Fuzzy logic based set-point weight tuning of PID controllers*, IEEE Transactions on Systems, Man, and Cybernetics - Part A, vol. 29, no. 6, pp. 587–592, 1999.

A. Leva and M. Maggio, *A systematic way to extend ideal PID tuning rules to the real structure*, Journal of Process Control, vol. 21, pp. 130–136, 2011.

A. Balestrino, V. Biagini, P. Bolognesi and E. Crisostomi, *Advanced variable structure PI controllers*, IEEE Conference on Emerging Technologies and Factory Automation (ETFA), 2009.

A.D. Fisher, J.H. VanZwieten and T.S. VanZwieten, *Adaptive Control of Small Outboard-Powered Boats for Survey Applications*, OCEANS 2009, MTS/IEEE Biloxi - Marine Technology for Our Future: Global and Local Challenges, 2009.

L.X. Wang, *A Course in Fuzzy Systems and Control*, Prentice Hall,1997.

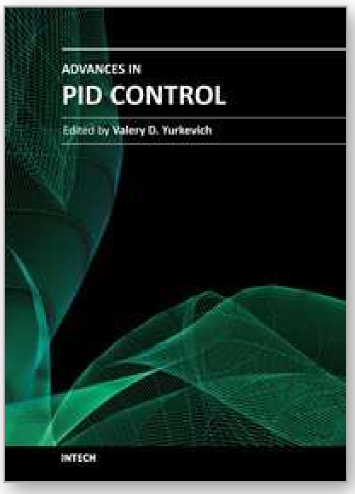
L.A. Zadeh, *Fuzzy Sets as basis for a theory of possibility.*, Fuzzy Sets and System 1, 1978.

E.H. Mamdani, *Application of Fuzzy Logic to Approximate Reasoning Using Linguistic Synthesis*, IEEE Transaction on Computers, 1977.

K. Tanaka and H.O. Wang, *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach*, John Wiley and Sons, 2001.

T.I. Fossen, *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*, Marine Cybernetics, 2002.

T. I. Fossen, *Modeling, Identification, and Adaptive Maneuvering of CyberShip II: A complete design with experiments*, Proc. of the IFAC CAMS'04, Ancona, Italy.



ADVANCES IN

PID CONTROL

Edited by Valery D. Yurkevich

INTECH

Advances in PID Control

Edited by Dr. Valery D. Yurkevich

ISBN 978-953-307-267-8

Hard cover, 274 pages

Publisher InTech

Published online 06, September, 2011

Published in print edition September, 2011

Since the foundation and up to the current state-of-the-art in control engineering, the problems of PID control steadily attract great attention of numerous researchers and remain inexhaustible source of new ideas for process of control system design and industrial applications. PID control effectiveness is usually caused by the nature of dynamical processes, conditioned that the majority of the industrial dynamical processes are well described by simple dynamic model of the first or second order. The efficacy of PID controllers vastly falls in case of complicated dynamics, nonlinearities, and varying parameters of the plant. This gives a pulse to further researches in the field of PID control. Consequently, the problems of advanced PID control system design methodologies, rules of adaptive PID control, self-tuning procedures, and particularly robustness and transient performance for nonlinear systems, still remain as the areas of the lively interests for many scientists and researchers at the present time. The recent research results presented in this book provide new ideas for improved performance of PID control applications.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821