Proceedings Paper:

Aftab, M.S., Rossiter, J.A. orcid.org/0000-0002-1336-0633 and Panoutsos, G. (2022) Predictive functional control for difficult second-order dynamics with a simple pre-compensation strategy. In: Proceedings of 2022 UKACC 13th International Conference on Control (CONTROL). 2022 UKACC 13th International Conference on Control (CONTROL), 20-22 Apr 2022, Plymouth, UK. Institute of Electrical and Electronics Engineers , pp. 12-17. ISBN 9781665452014

https://doi.org/10.1109/Control55989.2022.9781367

Predictive Functional Control for Difficult Second-Order Dynamics with a Simple Pre-conditioning Strategy

Muhammad Saleheen Aftab[1] and John Anthony Rossiter[2]

***Abstract*— Predictive functional control (PFC) is a fairly straightforward model-based technique for controlling stable and monotonically convergent dynamics in a systematic fashion. However, owing to simplified design assumptions, the control performance generally degrades with oscillatory or unstable processes. This paper focuses on pre-stabilising such difficult dynamics, represented as second-order prediction models, before implementing the PFC. In this proposal, the pre-compensator is designed with a root locus method that shifts the undesirable open-loop poles to the stable break-in/breakaway position by varying compensator gain. It has been highlighted that such dynamics transformation enables PFC application in the standard manner by preserving design simplicity and intuitiveness in terms of parameter tuning and constraint handling. Two simulation examples are included to study the pros and cons of the proposal against the conventional PFC algorithm.**

***Index Terms*—predictive functional control, root locus, pre-compensation, constraint handling**

I. Introduction

Predictive functional control (PFC), since its introduction in the late 1970's [1], has emerged as the strongest competitor to the widely popular proportional-integral-derivative (PID) algorithm, especially for industrial process control. The advantages of PFC significantly outweigh those of PID in that it systematically handles process dead-times and constraints, which PID cannot without incorporating additional resources such as Smith predictor and anti wind-up techniques [2]. Moreover controller tuning in PFC distinctively relates to a physical characteristic i.e. system rise time which makes the tuning process comparatively meaningful. Consequently numerous successful PFC applications have been reported in the literature [3], [4].

PFC inherits most design attributes from the mainstream model predictive control (MPC) family [5]. Nevertheless it differs from other predictive control algorithms in the parametrisation of the future input which, in the case of PFC, is assumed as a linear combination of some simple basis functions [3], [6]. A polynomial basis function is usually employed whose order depends upon the characteristics of the set-point trajectory. Thus for a constant set-point, the future input parametrises to just one degree-of-freedom, eliminating the need for the complex optimisation routines generally associated with high-end MPC algorithms. This on one hand simplifies computations, but on the other hand necessitates heuristics to find a sub-optimal solution for the constrained predictive control problems. Unlike mainstream MPC, simple clipping or saturation has been the commonly deployed input constraint management protocol within PFC.

With stable first-order processes, using a constant future input to match the predicted output with the reference trajectory at a single coincidence point is sufficient to achieve any desirable target behaviour, provided the coincidence occurs one time-step ahead in future [4], [5]. Similar results are obtained with well damped higher-order systems although one-step ahead coincidence may not always be appropriate, especially if the prediction dynamics exhibit significant initial lag [7]. The closed-loop performance, however, deteriorates when oscillatory or divergent process dynamics are introduced [5], [7], [8]. This inefficacy relates to the insufficiency of the constant future input assumption alongside a single coincidence point that lacks flexibility to handle such difficult behaviour [9]. Nevertheless various design modifications have been proposed to handle difficult dynamics with PFC.

One proposal [8] implements input shaping which parameterises the future input so as to cancel the undesirable modes from the model predictions. This modification improves performance, but often results in aggressive control moves, limiting its practicality. A modified input shaping algorithm [9] ensures smooth and less aggressive control action but requires tedious offline computations that negate the core notion of simplicity associated with PFC. Another proposal [10] suggests decomposing the higher-order model into multiple first-order subsystems to benefit from simple tuning procedures. But such decomposition for oscillatory dynamics embeds complex number algebra into the computations which may not work easily with general purpose industrial PLCs [11].

Another alternative is to explicitly pre-condition the prediction model with some form of feedback compensation in order to obtain smooth and convergent prediction behaviour [12], [13]. This method is fairly common in mainstream MPC but its application in PFC is generally restricted to pre-stabilising first-order systems with simple proportional gain [14]–[16]. Researchers have pointed out that complex internal feedback compensators may complicate the constraint management process [4], [5].

In this paper, we present a pre-stabilisation technique for challenging dynamic behaviour, represented as open-loop underdamped or unstable second-order models. The proposal implements concepts from root locus theory [17], [18], to shift undesirable open-loop poles to stable break-in/breakaway positions on the root loci, by varying compen-

[1],[2]Both authors are associated with the Department of Automatic Control and Systems Engineering, University of Sheffield, Mappin Street, S1 3JD, UK. Email Addresses: [1]msaftab1@sheffield.ac.uk, [2]j.a.rossiter@sheffield.ac.uk

sator gain. Mathematical expressions have been developed that enable pre-compensator design without requiring to plot and analyse root locus paths. Furthermore, it has been shown that the proposal keeps overall design simple and intuitive to benefit from the standard PFC tuning and constraint handling procedures.

The remainder of this paper is organised as follows: Section II formulates the problem and sets control objectives. The main methodology is presented in Sections III & IV where the pre-compensator and PFC designs are discussed in detail. Numerical studies follow next in Section V which discuss closed-loop performance and draw comparisons against the standard PFC. Finally the paper concludes in Section VI.

II. Problem Statement

Consider a difficult real world process characterised by a strictly proper second-order transfer function model

$$G(z) = \frac{b(z)}{a(z)} \tag{1}$$

where $a(z) = 1+a_1z^{-1}+a_2z^{-2}$ and $b(z) = b_1z^{-1}+b_2z^{-2}$. It is assumed that the open-loop model shows oscillatory or divergent dynamic behaviour i.e. the open-loop pole polynomial $a(z)$ either comprises a complex conjugate pole pair, see for example Fig. 1(a), or has at least one unstable mode, as shown in Figs. 1(b)-(c).

The problem addressed in this paper deals with designing PFC for the process modelled as $G(z)$. However, as stated earlier, conventional PFC may be less effective with such challenging dynamics. Therefore, the aim is first to stabilise model predictions with a simple internal feedback loop. Furthermore, the controller is expected to exhibit some degree of robustness against parametric uncertainty and/or unmodelled dynamics.

III. Pre-compensator Design

The primary objective of pre-compensation is to transform the undesirable open-loop dynamics into stable and, if possible, monotonically convergent prediction behaviour for straightforward implementation within a PFC framework. The proposed pre-compensation process, illustrated in Fig. 2, employs a simple feedback controller $C(z)$ to stabilise $G(z)$. The controller has the form $C(z) = KC_{in}(z)$, where K is the proportional gain with $K \in (-\infty, +\infty)$ and $C_{in}(z) = 1$ by default but may be designed as a lead or lag compensator if necessary. The internal feedback results in the following pre-compensated model:

$$T(z) = \frac{G(z)}{1 + KC_{in}(z)G(z)} \tag{2}$$

Next we present the design of the pre-compensator based on a root locus technique.

A. Preliminary Design via Root Locus

Root locus is a powerful graphical tool for control systems analysis and design [17]. It is generally used for the assessment of a system's closed-loop performance and relative stability as a function of various parameters, such as system gain and time constant. Here we wish to analyse the effect of varying K on the pre-compensated pole polynomial. Ultimately the goal is to identify such values of K that result in critically damped poles. Note that critical damping in root loci may only occur at the stable break-in/breakaway points.

A point on root locus curve where two poles exit the real axis and diverge to become a complex conjugate pair is known as the breakaway point. Conversely a point on the real axis at which a complex pole pair converges is called the break-in point. Intuitively one may obtain monotonically convergent predictions just by designing K at the break-in/breakaway point, provided it occurs within the stable range $0 < z < 1$. However, the open-loop zero dynamics can be significant in some cases and hence it may not always be possible to achieve a break-in/breakaway at acceptable locations. Let us examine the possible cases in detail.

Underdamped Poles. Consider the case of open-loop underdamped poles as shown in Fig. 1(a). Evidently there are three distinct regions for the system's zero location. It has been found that the poles would break-in within $0 < z < 1$ as long as the zero is either located in R_1 or R_3. However, a zero in R_2 i.e. in the vicinity of the open-loop poles may cause a break-in at $z < 0$. To solve this problem, we propose using a lag type $C_{in}(z)$ as follows:

$$C_{in}(z) = \frac{z + z_n}{z + z_0} \tag{3}$$

where $z_0 = -b_2/b_1$ is the open-loop zero whereas z_n is the new zero deliberately placed in R_1 (within the unit circle) away from the open-loop poles i.e. $z_n < z_0$. Note that this method may not work if the open-loop poles appear in the left half section of the unit circle.

One Unstable Pole. In this case, breakaway can only take place within $0 < z < 1$ if z_0 lies in either R_1 away from the stable pole p_1 or in R_4 away from the unstable pole p_2, see Fig. 1(b). However, a zero in the vicinity of either poles (while remaining in R_1 or R_4) might result in a breakaway outside the desirable range. Moreover, if z_0 lies in either R_2 or R_3 the model cannot be stabilised with simple gain K. A similar procedure as described above with (3) may be employed but only if the open-loop zero is stable i.e. located in R_2.

Two Unstable Poles. With two unstable poles, as shown in Fig. 1(c), the only possibility to get a break-in within the desirable range $0 < z < 1$ is to have the open-loop zero z_0 located within R_2 near the stability boundary. If z_0 is in R_3 then it may or may not be possible to stabilise the model. In all other scenarios, this method would fail to stabilise prediction dynamics. However, if z_0 is located in the stable portion of R_1, then it is possible to employ a lead-type $C_{in}(z)$ similar to (3) but with $z_n > z_0$ in order to replace the open-loop zero with the new one placed in R_2 near $z = 1$.

B. Design Procedure

It should be obvious from the preceding discussion that the efficacy of the preliminary design is strongly linked to the

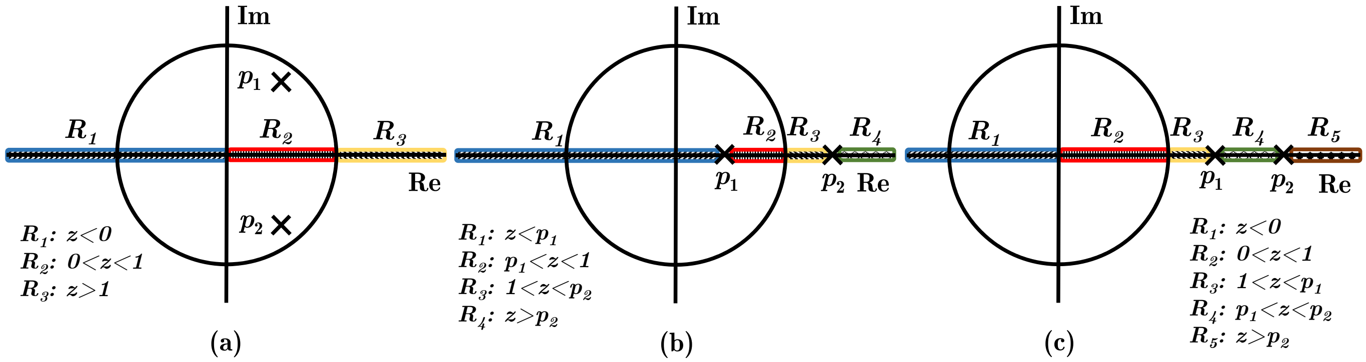

Fig. 1. Pole-zero map of $G_0(z)$ with (a) complex pole pair, (b) one unstable pole, and (c) two unstable poles; R_i's represent the possible zero positions.

actual pole-zero mapping of $G(z)$. Nevertheless, the concept can be generalised mathematically for any second-order pole-zero configuration as illustrated in Fig. 1. With $C_{in}(z) = 1$, the compensated pole polynomial $1 + KG(z) = 0$ implies:

$$K = -\frac{1}{G(z)} = -\frac{a(z)}{b(z)} \tag{4}$$

The break-in/breakaway points are stationary in nature, thus in order to find them K is differentiated with respect to z and equated to '0' [17]:

$$\left.\frac{dK}{dz}\right|_{z=\sigma} = -\frac{d}{dz}\left[\frac{a(z)}{b(z)}\right]_{z=\sigma} = 0$$

where σ represents the point(s) at which break-in/breakaway occurs. This implies:

$$\begin{aligned} & b(\sigma)a'(\sigma) - b'(\sigma)a(\sigma) = 0 \\ \implies & (b_1\sigma + b_2)(2\sigma + a_1) - (b_1)(\sigma^2 + a_1\sigma + a_2) = 0 \\ \implies & b_1\sigma^2 + 2b_2\sigma + (a_1b_2 - a_2b_1) = 0. \end{aligned} \tag{5}$$

Lemma 1: For a given pole polynomial $a(z)$, σ is a non-linear function of the open-loop zero z_0.

Proof: Equation (5) is in the standard quadratic form and can be solved analytically:

$$\begin{aligned} \sigma_{1,2} &= -\frac{b_2}{b_1} \pm \frac{1}{b_1}\sqrt{a_2{b_1}^2 - a_1b_1b_2 + {b_2}^2} \\ &= \left(-\frac{b_2}{b_1}\right) \pm \sqrt{a_2 + a_1\left(-\frac{b_2}{b_1}\right) + \left(-\frac{b_2}{b_1}\right)^2} \\ \implies \sigma_{1,2} &= z_0 \pm \sqrt{a_2 + a_1z_0 + {z_0}^2}. \end{aligned} \tag{6}$$

Hence for a given $a(z)$, σ is a non-linear function of z_0. ■

Remark 1: When there is no finite zero i.e. $b_1 = 0$, (5) reduces to $\sigma = -0.5a_1$. Thus there can only be either a break-in or breakaway point (depending on the poles) but not both.

If any of the σ_i's from (6) lies within $0 < z < 1$, then one may find the corresponding gain value K using:

$$K = -\frac{a(\sigma_i)}{b(\sigma_i)}; \quad i = 1 \text{ or } 2. \tag{7}$$

This completes the pre-compensator design. But what happens when none of the σ_i's is present within the right half of the unit circle? In this case, one may first check if employing a lead or lag type compensator such as (3) would suffice. Theorem 1 establishes conditions on the usability of $C_{in}(z)$ in such cases.

Theorem 1: The prediction model $G(z)$ can be stabilised with a lead or lag type compensator $C_{in}(z)$, such as (3), if in addition to $|z_0| < 1$,

$$|\sigma_d^2 - a_2| < |2\sigma_d + a_1|$$

where σ_d is the desired break-in/breakaway point such that $0 < \sigma_d < 1$.

Proof: Evidently the purpose of lead/lag compensation here is to replace the open-loop zero z_0 with the new zero z_n at a desirable location within the unit circle. Thus for guaranteed internal stability, $C_{in}(z)$ can only be designed if the old zero z_0 and the new zero z_n both lie inside the unit circle i.e. $|z| < 1$. Subsequently one may re-write (6) as follows:

$$\begin{aligned} \sigma_d &= z_n \pm \sqrt{a_2 + a_1z_n + z_n^2} \\ \implies (\sigma_d - z_n)^2 &= \left(\pm\sqrt{a_2 + a_1z_n + z_n^2}\right)^2 \end{aligned}$$

which after a few simple manipulations becomes,

$$z_n = \frac{\sigma_d^2 - a_2}{2\sigma_d + a_1}. \tag{8}$$

Since $|z_n| < 1$, this means

$$|\sigma_d^2 - a_2| < |2\sigma_d + a_1| \tag{9}$$

for $0 < \sigma_d < 1$. ■

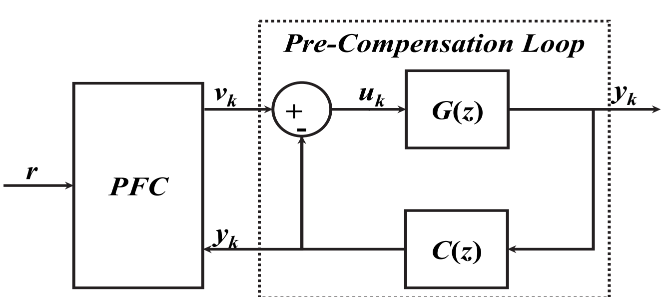

Fig. 2. Pre-conditioned PFC (PPFC) with internal feedback loop.

Equation (8) can validate whether designing $C_{in}(z)$ as lead or lag type compensator would be worthwhile. One may plot z_n as a function of σ_d to find a suitable zero that enforces break-in/breakaway within $0 < z < 1$ (see Fig. 5 for example). If such z_n exists, then K can be evaluated from (4) by replacing $b(z)$ with the new compensated polynomial $\beta(z) = b_1 z^{-1} + b_1 z_n z^{-2}$. This will give the following pre-compensated model.

$$T(z) = \frac{\beta(z)}{\alpha(z)} = \frac{b_1 z^{-1} + b_1 z_n z^{-2}}{1 - 2\sigma_d z^{-1} + \sigma_d^2 z^{-2}} \tag{10}$$

Remark 2: It should be emphasised that pre-compensation only stabilises the prediction model, whereas attributes such as transient performance, offset-free tracking, dead-time and constraints are managed by the outer PFC loop, as shown in Fig. 2.

IV. Design of Pre-Conditioned PFC

The Pre-conditioned PFC (PPFC) algorithm, similar to the original PFC, attempts to match the predicted response with an ideal (first-order) behaviour at the single coincidence point n_y with constant control moves. This process is repeated at each time step and owing to receding horizon, a virtual feedback is established that moves the plant output closer to the target. This convergence depends upon the desired behaviour and can be implemented as a first-order pole ρ. Assume that the ideal n_y-step ahead prediction based on first-order response is given as:

$$y_{k+n_y|k} = r - (r - y_k)\rho^{n_y} \tag{11}$$

where r is the constant set-point and y_k is the measured plant output. The n_y steps ahead output predictions are derived from the pre-stabilised model $T(z)$ such that:

$$\hat{y}_{k+n_y|k} = H \underrightarrow{v}_k + P \underleftarrow{v}_{k-1} + Q \underleftarrow{\hat{y}}_k \tag{12}$$

where H, P and Q depend on the model parameters. For a generic N^{th} order model:

$$\underrightarrow{v}_k = \begin{bmatrix} v_k \\ v_{k+1} \\ \vdots \\ v_{k+n_y-1} \end{bmatrix}; \underleftarrow{v}_{k-1} = \begin{bmatrix} v_{k-1} \\ v_{k-2} \\ \vdots \\ v_{k-N+1} \end{bmatrix}; \underleftarrow{\hat{y}}_k = \begin{bmatrix} \hat{y}_k \\ \hat{y}_{k-1} \\ \vdots \\ \hat{y}_{k-N+1} \end{bmatrix}$$

With constant control values throughout the coincidence horizon i.e. $v_{k+i} = v_k, \forall i > 0$, we obtain the PPFC control law from (11)-(12) as follows:

$$v_k = \frac{r - (r - y_k)\rho^{n_y} - (P \underleftarrow{v}_{k-1} + Q \underleftarrow{\hat{y}}_k)}{h} \tag{13}$$

where $h = \sum_{j=1}^{n_y} H_j$ and H_j is the j^{th} element of H.

Remark 3: For clarity of presentation, the PPFC control law derived above does not include algebra relevant to offset-free tracking and dead-times. The numerical examples, nonetheless, include these details. See [5] for further information.

Since v_k is the input to $T(z)$, we must also determine u_k for plant actuation and constraint management. Theorem 2 establishes the relationship between v_k and u_k.

Theorem 2: The PPFC control input v_k and the plant input u_k are related as follows:

$$u_k = \hat{a} \underleftarrow{v}_k + \hat{\alpha} \underleftarrow{u}_{k-1}$$

where vectors $\hat{a}$ and $\hat{\alpha}$ contain the suitable coefficients of $a(z)$ and $\alpha(z)$ respectively.

Proof: With reference to Fig. 2, $u_k = v_k - C(z)y_k$ and since $y_k = G(z)u_k$ and $C(z) = KC_{in}(z)$, we get:

$$u_k = v_k - KC_{in}(z)G(z)u_k$$

which implies,

$$\left[1 + KC_{in}(z)G(z)\right] u_k = v_k$$

But $1 + KC_{in}(z)G(z) = \alpha(z)/a(z)$. Therefore $\alpha(z)u_k = a(z)v_k$ implies:

$$u_k = a(z)v_k + 2\sigma_d z^{-1} u_k - \sigma_d^2 z^{-2} u_k$$

Or equivalently in the time-domain:

$$u_k = \hat{a} \underleftarrow{v}_k + \hat{\alpha} \underleftarrow{u}_{k-1} \tag{14}$$

where $\hat{a} = [1 \quad a_1 \quad a_2]$ and $\hat{\alpha} = [2\sigma_d \quad -\sigma_d^2]$. ■

Remark 4: Constraint handling is one of the key features of conventional PFC, and the techniques to do so are well established in literature [4], [5]. While it is obvious that constraint management after pre-compensation is slightly more expensive, the associated algebra and coding are still fairly benign, see for instance [19], [20], where the impact of pre-stabilisation on constraint validation is discussed in detail. In this paper, we will employ these results directly in the simulation examples.

Remark 5: Pre-stabilisation helps selecting the coincidence point n_y for difficult dynamics in a straightforward manner, based on the conjecture presented in [7]. As per the recommendation, n_y lies within the time range when the pre-stabilised step response rises from 40% to 80% with significant gradient. As for finding ρ, one may overlay several first-order responses on the step response to identify which target behaviour coincides within the mentioned n_y range. See, for instance, Fig. 3.

V. Numerical Examples

This section investigates the efficacy of the proposal with two numerical examples. Example 1 illustrates the case of an oscillatory higher-order process which is modelled as a second-order underdamped system for the Pre-conditioned PFC implementation. Example 2, on the other hand, demonstrates the PPFC design for a second-order unstable system when a simple proportional gain alone is ineffective due to dominant open-loop zero dynamics. Details follow next.

A. Example 1

Consider an underdamped process,

$$G_1 = \frac{0.065z^{-1} + 0.26z^{-2}}{1 - 1.35z^{-2} + 1.158z^{-2} - 0.28z^{-3}} \cdot z^{-5} \tag{15}$$

with an open-loop zero $z_0 = -4$, a real pole at $z = 0.35$, a complex conjugate pole pair $p_{1,2} = 0.5 \pm j0.742$ and a

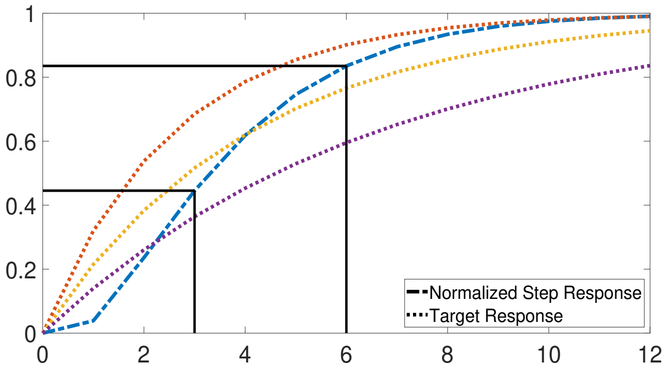

Fig. 3. Target responses with $\rho = [0.68, 0.79, 0.86]$ overlaying the normalised step response of T_1.

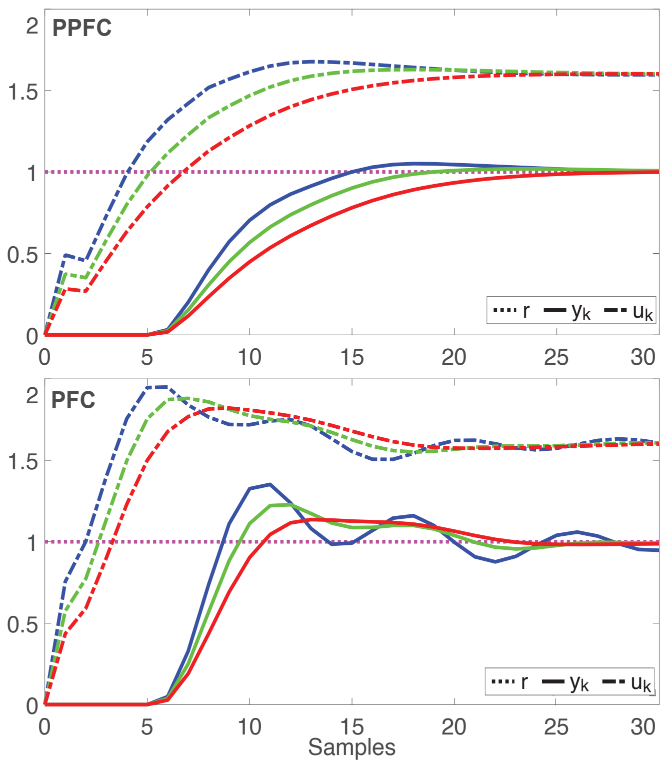

Fig. 4. PPFC vs. PFC with unmodelled pole at $z = 0.35$ for model G_1, $n_y = 4$ and $\rho = [0.68(blue), 0.8(green), 0.86(red)]$.

dead-time of $m = 5$ samples. To apply the proposed pre-compensation, only the dominant second-order dynamics are considered after neglecting the non-dominant pole at $z = 0.35$ from the prediction model. Nevertheless, it is included in the plant simulation to analyse robustness.

The break-in/breakaway point calculated from (6) suggests $\sigma_d = 0.561$ with the corresponding $K = -1.214$. In this example, lag or lead type compensation is not needed i.e. $C_{in}(z) = 1$ and thus $C(z) = -1.214$. This gives the following delay-free second-order pre-compensated model:

$$T_1 = \frac{0.1z^{-1} + 0.4z^{-2}}{1 - 1.121z^{-1} + 0.314z^{-2}} \tag{16}$$

Next we determine the appropriate n_y and ρ by plotting the normalised step response of T_1 overlaying several desired first-order responses with differing ρ's, as shown in Fig. 3. The plot suggests $3 \leq n_y \leq 6$ as a suitable coincidence horizon window. Note that target dynamics with $\rho = 0.68$ or $\rho = 0.86$ do not match predicted behaviour within the desirable n_y range and hence would need over-actuation or under-actuation to enforce an intercept. However, a sensible choice would be $\rho = 0.79$ which gets an exact match at $n_y = 4$.

Efficacy of the PPFC algorithm is obvious with the closed-loop performance shown in Fig. 4. Even in the presence of the unmodelled pole ($z = 0.35$), the PPFC plant output (upper figure) is smooth and oscillation-free, and strongly linked to the corresponding ρ, although control input for faster target dynamics is relatively aggressive as expected. The conventional PFC (lower figure) appears ineffective in damping oscillations even with fairly aggressive control moves in the transient region. In practice, this may lead to actuator saturation resulting in unacceptable control performance.

B. Example 2

Consider an open-loop unstable system,

$$G_2 = \frac{1.5z^{-1} - 1.2z^{-2}}{1 - 1.5z^{-1} + 0.44z^{-2}}; \quad |u_k| \leq 0.205 \tag{17}$$

with $z_0 = 0.8$ located between the open-loop poles at $p_1 = 0.4$ and $p_2 = 1.1$. Clearly no break-in/breakaway is possible with simple proportional compensation and therefore $C_{in}(z)$ must be designed to stabilise the model. Fig. 5 plots the new zero z_n as a function of σ_d for G_2. It is evident that a break-in/breakaway within the right half of unit circle can be enforced with a lag-type $C_{in}(z)$. Notice that faster pre-compensated dynamics ($\sigma_d \leq 0.4$) may be obtained with $z_n \approx 0.3$. However, such dynamics may not be appropriate owing to aggressive control action, potentially causing constraint violation. Therefore, we select $z_n = 0$ for $\sigma_d = 0.6633$. The corresponding gain value is then found as $K = 0.1155$. Thus $C(z) = 0.1155z/(z - 0.8)$ and the pre-compensated model is:

$$T_2 = \frac{1.5z^{-1} - 1.2z^{-2}}{1 - 1.327z^{-1} + 0.44z^{-2}}. \tag{18}$$

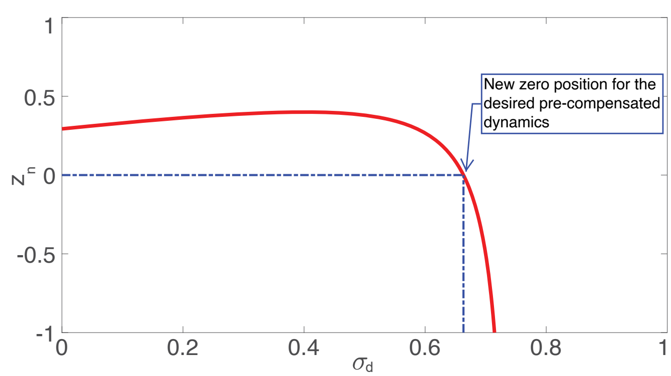

Fig. 5. Plot of z_n ($|z_n| < 1$) vs σ_d ($0 < \sigma_d < 1$) for G_2; $z_n = 0$ with $\sigma_d = 0.6633$ selected for pre-compensation.

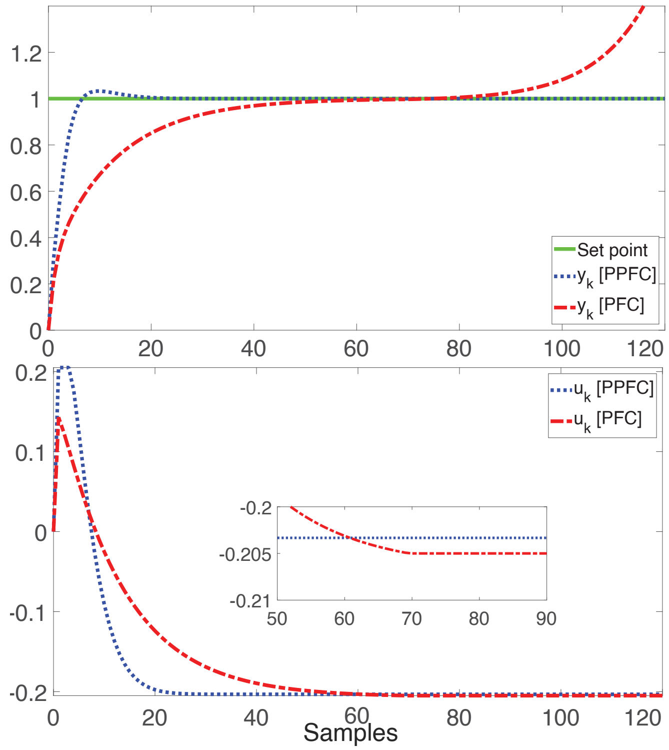

Fig. 6. Closed-loop performance comparison with PPFC ($blue$) and PFC (red) for G_2 including parametric uncertainty (no uncertainty in unstable pole [21]); $n_y = 5$ and $\rho = 0.85$.

Next $n_y = 5$ and $\rho = 0.85$ are obtained with a similar procedure as described in Example 1. The closed-loop performance is compared and contrasted for both PPFC and conventional PFC algorithms in Fig. 6, with deliberately added parametric uncertainty (assuming no uncertainty in the unstable pole [21]). Due to unreliable open-loop divergent predictions, the closed-loop output with PFC destabilises around the 80^{th} sample. Coincidently, the corresponding control input also saturates at $u_{min} = -0.205$ due to constraint violation around that time. On the other hand, the proposed PPFC algorithm keeps the system output smooth and stable while maintaining robustness against uncertainty and without violating input constraints.

VI. Conclusion

A root locus based pre-conditioning strategy for predictive functional control of difficult dynamic processes is presented. The proposal is fairly generic and based on the fact that a majority of real-world processes can be adequately represented as dominant second-order dynamics for which simple tailored solutions are well understood. The main idea is to form smooth and well-damped predictions with an internal feedback compensator designed to enforce break-in/breakaway at the desired closed-loop poles. We have shown that a simple proportional gain is generally sufficient, however, a lead or lag type compensator may also be needed with some challenging pole-zero configurations. Moreover the proposed PPFC design preserves the inherent simplicity and intuitiveness of the original PFC, including the standard parameter tuning and constraint management procedures.

The proposal has shown promising results for both oscillatory and unstable processes in the presence of uncertainty. Nevertheless, future work will focus on more formal analysis of the closed-loop characteristics against disturbances, sensor noise and modelling errors, along with an analysis of efficacy in real world industrial applications.

References

[1] J. Richalet, A. Rault, J. Testud, and J. Papon, "Model predictive heuristic control," *Automatica (Journal of IFAC)*, vol. 14, no. 5, pp. 413–428, 1978.

[2] R. Haber, J. A. Rossiter, and K. Zabet, "An alternative for pid control: Predictive functional control-a tutorial," in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 6935–6940.

[3] J. Richalet and D. O'Donovan, "Elementary predictive functional control: A tutorial," in *2011 International Symposium on Advanced Control of Industrial Processes (ADCONIP)*, May 2011, pp. 306–313.

[4] J. Richalet and D. O'Donovan, *Predictive functional control: principles and industrial applications*. Springer Science & Business Media, 2009.

[5] J. Rossiter, *A first course in predictive control*. CRC Press, 2018.

[6] M. Khadir and J. Ringwood, "Extension of first order predictive functional controllers to handle higher order internal models," *International Journal of Applied Mathematics and Computer Science*, vol. 18, no. 2, pp. 229–239, jun 2008.

[7] J. Rossiter and R. Haber, "The effect of coincidence horizon on predictive functional control," *Processes*, vol. 3, no. 1, pp. 25–45, 2015.

[8] J. Rossiter, "Input shaping for pfc: how and why?" *Journal of control and decision*, vol. 3, no. 2, pp. 105–118, 2016.

[9] M. Abdullah and J. Rossiter, "Input shaping predictive functional control for different types of challenging dynamics processes," *Processes*, vol. 6, no. 8, p. 118, 2018.

[10] J. Rossiter, R. Haber, and K. Zabet, "Pole-placement predictive functional control for over-damped systems with real poles," *ISA transactions*, vol. 61, pp. 229–239, 2016.

[11] K. Zabet, J. Rossiter, R. Haber, and M. Abdullah, "Pole-placement predictive functional control for under-damped systems with real numbers algebra," *ISA transactions*, vol. 71, pp. 403–414, 2017.

[12] J. A. Rossiter, B. Kouvaritakis, and M. Rice, "A numerically robust state-space approach to stable-predictive control strategies," *Automatica*, vol. 34, no. 1, pp. 65–73, 1998.

[13] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[14] M. S. Aftab, J. A. Rossiter, and Z. Zhang, "Predictive Functional Control for Unstable First-Order Dynamic Systems," in *CONTROLO 2020*, ser. Lecture Notes in Electrical Engineering, J. A. Gonçalves, M. Braz-César, and J. P. Coelho, Eds., vol. 695. Cham: Springer International Publishing, 2021, pp. 12–22.

[15] M. Abdullah and J. A. Rossiter, "Alternative method for predictive functional control to handle an integrating process," in *2018 UKACC 12th International Conference on Control (CONTROL)*. IEEE, 2018, pp. 26–31.

[16] Z. Zhang, J. Rossiter, L. Xie, and H. Su, "Predictive functional control for integral systems," in *International Symposium on Process System Engineering*, 2018.

[17] K. Ogata *et al.*, *Discrete-time control systems*. Prentice Hall Englewood Cliffs, NJ, 1995, vol. 2.

[18] N. S. Nise, *Control Systems Engineering, Sixth Edition*. John Wiley & Sons, 2007.

[19] M. S. Aftab and J. A. Rossiter, "Pre-stabilised predictive functional control for open-loop unstable dynamic systems," in *Proceedings of 7th IFAC Conference on Nonlinear Model Predictive Control*, 2021.

[20] M. S. Aftab and J. A. Rossiter, "Predictive functional control with explicit pre-conditioning for oscillatory dynamic systems," in *Proceedings of 2021 European Control Conference*, 2021.

[21] S. Skogestad and I. Postlethwaite, *Multivariable feedback control: analysis and design*. Wiley New York, 2007, vol. 2.