machines

MDPI

Article

Design and Implementation of a Recursive Feedforward-Based Virtual Reference Feedback Tuning (VRFT) Controller for Temperature Uniformity Control Applications

Juan Gabriel Araque [1], Luis Angel [1], Jairo Viola [2,*] and Yangquan Chen [2]

[1] Electronic Engineering, Pontifical Bolivarian University, Medellín 050031, Colombia; araquemora@gmail.com (J.G.A.); luis.angel@upb.edu.co (L.A.)
[2] School of Engineering, University of California, Merced, CA 95343, USA; ychen53@ucmerced.edu
* Correspondence: jviola@ucmerced.edu

Abstract: Data-driven controller synthesis methods use input/output information to find the coefficients of a proposed control architecture. Virtual Reference Feedback Tuning (VRFT) is one of the most popular frameworks due to its simplicity and one-shoot synthesis style based on open-loop system response for classic regulators such as PI or PID. This paper presents a recursive VRFT framework to extend VRFT into high-order controllers with more complex structures. The framework first defines a reference model and controller structure, then uses the open-loop data to compute the virtual reference and error signals, and, finally, uses these to find the controller parameters via an optimization algorithm. Likewise, the recursive VRFT controller performance is improved by adding a model-based feedforward loop to improve reference signal tracking. The recursive method is tested to design a temperature uniformity control system. The obtained results show that the recursive VRFT with a feedforward improves the system response while allowing more complex controller synthesis.

Keywords: recursive VRFT; model-reference feedforward; process control

Citation: Araque, J.G.; Angel, L.; Viola, J.; Chen, Y. Design and Implementation of a Recursive Feedforward-Based Virtual Reference Feedback Tuning (VRFT) Controller for Temperature Uniformity Control Applications. *Machines* **2023**, *11*, 975. https://doi.org/10.3390/machines11100975

Academic Editors: Jian Wu, Xiangkun He and Guangfei Xu

Received: 26 September 2023
Revised: 15 October 2023
Accepted: 17 October 2023
Published: 20 October 2023

1. Introduction

Model-based control methods cover an important part of control systems design. These are supported by mathematical process models for the design and synthesis of a closed-loop controller [1]. However, obtaining an acceptable process model can be challenging in processes with uncertainty and non-linear behavior, compromising the system's stability and performance [1–4]. This is especially relevant for processes that face variable environments that may require not only an acceptable model but also a constant parameter adaptation using optimization algorithms or adaptive control methodologies, such as [5–11]. Thus, the interest in data-driven control methods has been increasingly motivated by the increasing availability of process data given new technologies such as IoT, edge computing, or cheap sensing [12]. Some of these methods include Iterative Feedback Tuning [13], dynamic mode decomposition [14], Iterative Learning Control [15], extremum seeking [16], model-free adaptive control [17], unfalsified control [18], and simultaneous perturbation stochastic approximation [19].

One of the most studied data-driven control methods is the Virtual Reference Feedback Tuning (VRFT). It uses input–output data from a process to generate a virtual reference signal based on a desired reference model to find the optimal parameters that minimize the difference with the desired behavior [20,21].

Some applications of VRFT include motion controls, power electronics, or process control. For example, [22] employs a dual VRFT controller tuning method for velocity ripple compensation on a motor system. In [23], VRFT is employed to find the compensator for several DC–DC power converters with non-minimum phase (NMP) behavior as buck,

boost, and buck–boost configurations with a reference model that compensates for the NMP zero. Likewise, [24] presents the use of VRFT on a cooperative control system with Neuroendocrine Ultrashort Feedback Controllers. Moreover, VRFT can be employed on multiple input multiple output (MIMO) configurations by defining a multivariable reference model [25] or combined with the internal model control (IMC) design method for dual temperature control [26].

In these applications, the synthesis process of VRFT follows the standard framework proposed by [20] following standard controller architectures such as PI and PID, and the model reference selection is based on prior knowledge of the system response. However, sometimes, the controller response has differences regarding its desired response. This can be explained by the fact that there are no criteria for reference model selection within the controller design method that ensure adequate closed-loop system performance.

This paper presents the design of a data-driven controller based on the Virtual Reference Feedback Tuning (VRFT) framework [3] to improve the setpoint tracking response of uniform temperature control systems.

Unlike other VRFT approaches, this work first generates the virtual error signal using the system's data-driven response with the reference model. Then, a compensator transfer function is selected (e.g., first-order, second-order, high-order), and parameters are determined through an optimization algorithm using the virtual error as input and control action as output signals. The controller synthesis process evaluates several reference model structures to select a suitable one that minimizes the system's energy consumption and steady-state error and allows a lower-order system implementation. Likewise, the proposed controller uses adaptive feedforward compensation based on the desired reference system model to improve the closed-loop system response based on the error between the reference model and system output. The proposed VRFT controller is tested for the temperature uniformity control of a Peltier thermoelectric system in simulation and experimentally using hardware in the loop configuration. The VRFT controller is evaluated against the classic PI controller tuned via pole IMC and Skogestad IMC (SIMC) [27].

The contributions of this paper are:

- Extending the VRFT framework for synthesizing different control structures through an optimization process using the virtual error signal obtained only from the system data.
- Present a method to assess reference models for VRFT controller design and different compensator architectures.
- Add a feedforward compensation to the system using the VRFT reference model to improve closed-loop response in setpoint tracking tasks.

The main benefit of the proposed algorithm is the recursive solution of the VRFT control problem, which can be extended into controllers with fixed complex structures, such as high-order compensators or non-linear controllers in SISO and MIMO configurations that include feedforward capabilities. Thus, the controller parameters are synthesized using an optimization algorithm using the virtual error that minimizes the control action and tracking error based on a desired reference model.

The manuscript is structured as follows. Section 2 introduces the foundations of VRFT controller design. Section 3 shows the recursive framework used to obtain the virtual error signal, the optimization process for VRFT synthesis, and the feedforward compensation scheme based on the reference model. Section 4 shows a case study corresponding to the temperature uniformity control system and its modeling using multiphysics simulation tools, the VRFT controller synthesis, feedforward design, and its validation in the simulation environment. Section 5 shows the experimental validation of the VRFT controller on the temperature uniformity control system. Finally, our conclusions and possibilities for future work are presented.

2. VRFT Control Framework

The VRFT framework is shown in Figure 1. Initially, an input–output $u(t)$, $y(t)$ open-loop response dataset of the system P is obtained, and a reference model M is defined to establish the system's desired behavior. So, using the inverse reference model M^{-1}, the virtual reference $r_v(r)$ is created to compute the virtual error $e_v(t)$. Thus, after selecting a structure for the controller C_ρ, its coefficients are determined using the virtual error signal $e_v(t)$ to make the virtual control action $u_v(t)$ closer to the real system's control action $u(t)$ [28,29]. During the data acquisition process, it is important that the input signal $u(t)$ satisfies the persistent excitation conditions to ensure a representative output $y(t)$ in terms of system identification. Likewise, filtering $y(t)$ is crucial for calculation of the virtual error $e_v(t)$ and the controller's synthesis quality [21,28]. Thus, using the filtered version of the input–output signals $u_f(t) = u(t) * F(t)$ y $y_f(t) = y(t) * F(t)$, where $F(t)$ is the filter transfer function, the virtual reference signals r_{vf} and e_{vf} can be computed as (1)

$$\begin{aligned} r_{vf}(t) &= y_f(t) * M^{-1}, \\ e_{vf}(t) &= r_{vf}(t) - y_f(t). \end{aligned} \tag{1}$$

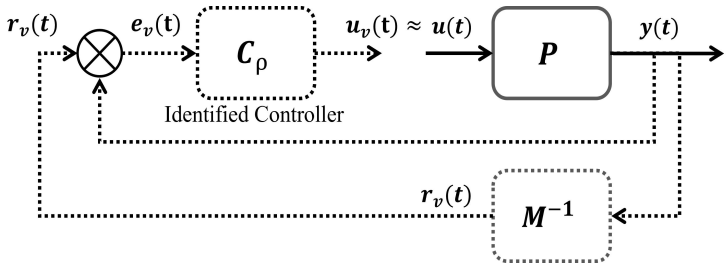


Figure 1. Classic VRFT framework [28].

So, using the filtered signals $[e_{vf}(t), u_f(t)]$, the parameters of the selected C_ρ can be obtained by minimizing (2), where N is the total amount of data samples and θ is the vector of controller parameters.

$$J_{VR} = \lim_{N\to\infty} \sum_{t=1}^{N} (u_f(t) - C_\rho(z;\theta) e_{vf}(t))^2. \tag{2}$$

Assuming that $C_\rho(z;\theta)$ is in the discrete domain z and is in the form $C_\rho(z;\theta) = \beta(z)\theta$, (2) can be rewritten as (3)

$$\begin{aligned} J_{VR} &= \lim_{N\to\infty} \textstyle\sum_{t=1}^{N} (u_f(t) - \varphi_f^T \theta))^2 \\ \varphi_f &= \beta(z) e_{vf}(t). \end{aligned} \tag{3}$$

Thus, (3) can be rewritten as a least squares problem, defined as (4), where $\widehat{\theta}_N$ is the estimated controller's parameters' vector.

$$\widehat{\theta}_N = \left(\sum_{t=1}^{N} \varphi_f(t) \varphi_f^T(t) \right)^{-1} \sum_{t=1}^{N} \varphi_f(t) u_f(t). \tag{4}$$

It is important to notice that for standard controller structures, such as PI, PID, or PD, the VRFT problem can be straightforwardly expressed using (3) and (4). However, for more complex architectures, such as high-order transfer function-based or non-linear controllers, expressing the VRFT problem is harder. Therefore, this paper proposes solving (3) by using standard optimization procedures, leveraging the virtual error and reference signals for controller synthesis.

3. Recursive VRFT Controller Synthesis

The procedure to perform a recursive VRFT controller synthesis is shown in Figure 2. The first step defines the reference model M and the controller structure C_ρ and collects the input–output data of the open-loop stable process.

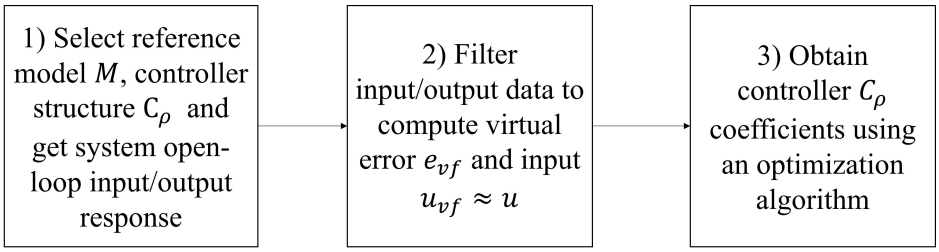


Figure 2. Recursive VRFT controller synthesis framework for a general controller architecture design based on an optimization algorithm to find controller parameters according to (4).

In the second step, the input–output data are filtered to compute the reference signal's E_v using M. This procedure is illustrated on Figure 3. According to [21,28], the virtual error E_v can be computed using $Y(s)$ and the inverse reference model M^{-1}, as shown in Figure 3a, which can be simplified using the equivalent reference system G^{-1} shown in Figure 3b. Thus, M and G can be related in the frequency domain as (5).

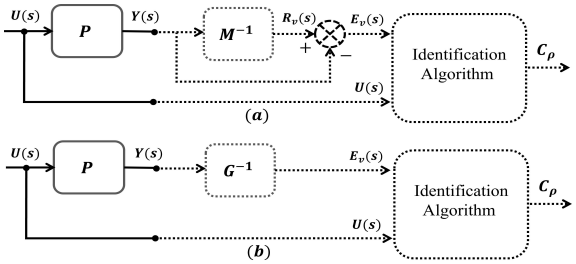


Figure 3. Recursive VRFT signal architecture. (**a**) shows the the virtual error $E_v(s)$ calculation using the closed-loop model $M(s)$, which feeds the identification algorithm to find the controller parameters. Likewise, (**b**) simplifies the computation of $E_v(s)$ by using the open-loop $G(s)$ reference model.

$$M(s) = \frac{G(s)}{1 + G(s)}. \tag{5}$$

Likewise , (1) can be rewritten in the frequency domain as (6) and (7)

$$R_v(s) = Y(s)M^{-1}(s), \tag{6}$$

$$E_v(s) = R_v(s) - Y(s). \tag{7}$$

Replacing (6) with (7), the virtual error in terms of M is given by (8)

$$E_v(s) = Y(s)M^{-1}(s) - Y(s) = Y(s)[M^{-1}(s) - 1]. \tag{8}$$

Now, substituting (5) in (8), the virtual error E_v is expressed in terms of $G(s)$ by (9), as shown in Figure 3b.

$$E_v(s) = Y(s)\left(\frac{1 + G(s)}{G(s)} - 1\right) = Y(s)G^{-1}(s). \tag{9}$$

Finally, in the third step, the coefficients of the controller C_ρ can be calculated using an optimization algorithm using e_{vf} and u_{vf} as input/output signals, respectively. In this manuscript, the optimization stage is performed using the Matlab system identification toolbox in order to synthesize complex high-order controllers rather than classic PI or PID regulators.

Feedforward VRFT-SISO Controller Recursive Synthesis

As can be observed in Figure 4a, if the controller is ideal, it means $C_\rho = C_{ideal} = G/P$, and the error between the desired response of the system $y_d(t)$ and the current one $y(t)$ is zero. However, the obtained controller C_ρ is not ideal in almost all the cases, making $e_a \neq 0$, showing a difference between the desired and actual system response. For this reason, the Feedforward VRFT-SISO controller shown in Figure 4b can be implemented to account for this mismatching. As can be observed, it uses the error signal $E_a(s) = Y_d(s) - Y(s)$ multiplied by $C_a(s)$ to produce an additional correction $U_a(s)$ to the control action $U(s)$ to minimize the difference between the desired and actual process response. To analyze the controller response, the system response is computed as (10):

$$Y(s) = [E(s)C_\rho(s) + E_a(s)C_a(s)]P(s) = [[R(s) - Y(s)]C_\rho(s)[Y_d(s) - Y(s)]C_a(s)]P(s), \quad (10)$$

replacing $Y_d(s) = R(s)M$ in (10),

$$Y(s)[1 + P(s)[C_\rho(s) + C_a(s)]] = R(s)[C_\rho(s) + M(s)C_a(s)]P(s). \quad (11)$$

So, the system's transfer function is given by (12)

$$\frac{Y(s)}{R(s)} = \frac{P(s)[C_\rho(s) + M(s)C_a(s)]}{1 + P(s)[C_\rho(s) + C_a(s)]}. \quad (12)$$

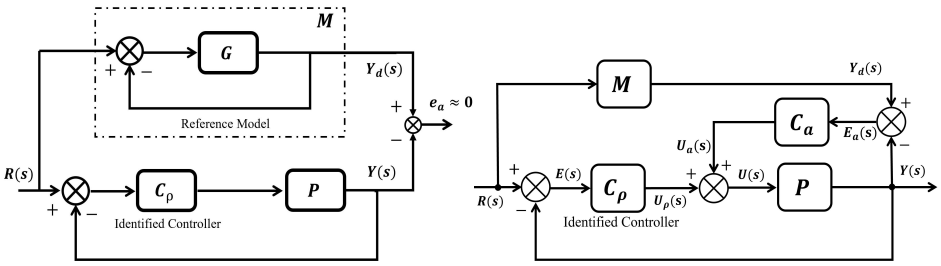


Figure 4. VRFT feedforward controller proposal. (**a**) Model reference scheme of VRFT; (**b**) feedback feedforward VRFT control system.

Replacing $C_a(s) = k \cdot C_\rho(s)$ and (5) in (12), where k is a proportional gain, the closed-loop response is given as (13):

$$\frac{Y(s)}{R(s)} = \frac{P(s)C_\rho(s)[1 + (k+1)G(s)]}{[1 + G(s)][1 + P(s)C_\rho(s)(k+1)]}. \quad (13)$$

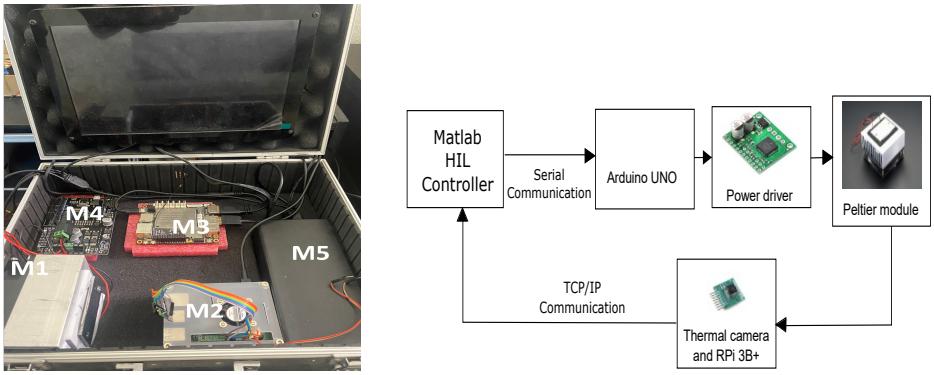
If the controller is ideal, $C_\rho(s) = C_{ideal}(s) = G(s)/P(s)$, and the feedforward will not affect the controller performance. However, if C_ρ(s) is not ideal, the feedforward improves the process response. The effect will be more noticiable for a value of k that makes $G(s)[k+1] \gg 1$, so (13) can be approximated as (14).

$$\frac{Y(s)}{R(s)} \approx \frac{P(s)C_\rho(s)[G(s)(k+1)]}{[1 + G(s)][P(s)C_\rho(s)(k+1)]} \approx \frac{G(s)}{1 + G(s)}. \quad (14)$$

4. A Case Study: Temperature Uniformity Control System

The temperature uniformity control system using real-time thermal infrared (TIR) vision feedback presented in Figure 5 is employed as a case study for VRFT controller design and implementation. The system is composed of a Peltier thermoelectric module (M1) acting as a heating/cooling element, a thermal infrared camera (M2) with TCP/IP communication protocol as a temperature feedback sensor running on a Raspberry Pi and communicated using TCP/IP communication protocol, a LattePanda Embedded board

(M3) running Windows 10 that executes Matlab in a hardware in the loop configuration (HIL). The power applied to the Peltier module is controlled using the power driver (M4) via PWM. The platform is equipped with a battery (M5) that provides the power for all of the components in the box with four hours of autonomy. A detailed description of the components of the temperature uniformity control system is presented in Table 1. Likewise, a guide for the system implementation can be found in [30,31].



(**a**) (**b**)

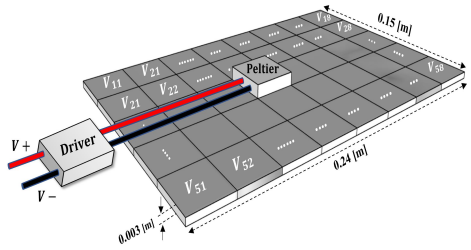
Figure 5. Case study: real-time vision feedback infrared temperature uniformity control system. (**a**) Physical asset; (**b**) HIL closed-loop controller implementation using Simulink/Arduino.

Table 1. Temperature Uniformity Control System Specifications.

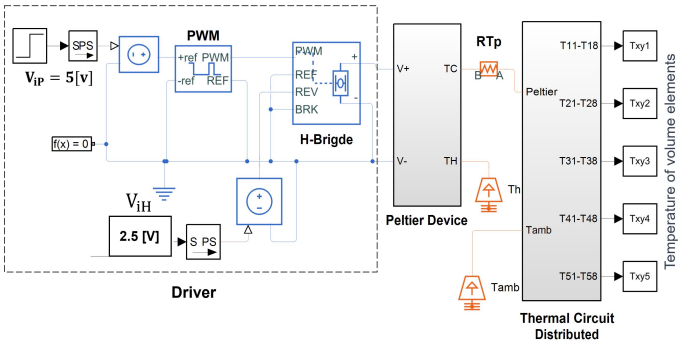
Component	Features
FLIR lepton thread Infrared thermal camera	Wavelength: 8 to 14 µm Resolution: 80 × 60 pixels Accuracy: ±0.5 °C
TEC1-12706 Peltier Module	$Q_{max} = 50$ W $\Delta_{Tmax} = 75$ °C $I_{Max} = 6.4$ A $V_{max} = 16.4$ V
MC33926 DC Power Driver	Input: 0–5 V Output: 0–12 V Peak Current: 5 A
LattePanda board	5 inch Windows 10 64 bits PC Intel Atom μp 4 GB of RAM Built-in Arduino Leonardo board

The main components of the system are the Peltier thermoelectric module and the power driver, whose detailed representations are shown in Figure 6a. They consist of the cascade connection of the driver, the Peltier cell, and an aluminum plate divided in V_{ij} volumetric elements sorted in an 8 × 5 array, where the Peltier and plate are thermally coupled through the volume element V_{34}, providing forty measurement points equally separated by 0.03 m.

The system is simulated using Matlab/Simscape, and the corresponding simulation model is shown in Figure 6b containing three main elements: the power driver, the Peltier device, and the plate thermal circuit.



(**a**)



(**b**)

Figure 6. Temperature uniformity system process. (**a**) The 3D schematic including driver, Peltier module, and the volumetric modeled thermal plate; (**b**) Simulink/Simscape process model representation.

The power driver is configured so that the Peltier cell cools the plate when $V_{iH} < 2.5$ [V]. For $V_{iH} \geq 2.5$ [V], the Peltier acts as a heater. The electrical power input to the Peltier cell is regulated by Pulse Width Modulation (PWM), modulating the input voltage in the range $0\text{ [V]} \leq V_{iP} \leq 5\text{ [V]}$.

Likewise, the Peltier equivalent circuit is implemented in Simscape using discrete thermal and electrical components as shown in Figure 7. It shows the implemented thermoelectric circuit for the Peltier cell, where the gray blocks represent the Seebeck, Joule, and Peltier effects given by (15)–(17), respectively, where α is the Seebeck coefficient, T_c, q_c and T_h, q_h are temperatures and heat on the cold and hot sides of the Peltier, and I, R_E are the Peltier current and resistance. The manufacturer's datasheet specifications for the Peltier coefficients can be found in [32]. Likewise, a complete and detailed description of each parameter of the Peltier and thermal plate modeling can be found in [33]. For this paper, the values employed to simulate the Peltier are $\alpha = 0.0421$ [V/K], $R_E = 2.65$ [Ω].

$$FEM = \alpha(T_h - T_c), \tag{15}$$

$$q_j = I^2 R_E, \tag{16}$$

$$q_c = \alpha T_c I \;\; q_h = \alpha T_h I. \tag{17}$$

Finally, the plate heat transfer is modeled by conduction and convection based on the finite element method in a volumetric grid as shown in Figure 8a, where each element is modeled as a thermal resistive–capacitive (RC) network, shown in Figure 8b, with a constant temperature per element volumetric element, as shown in [33]. The Matlab/Simscape implementation of the plate thermal circuit is shown in Figure 8b, where each volume element V_{nm} is represented with a thermal capacitance C_{Tnm} and resistance R_{Tnm}.

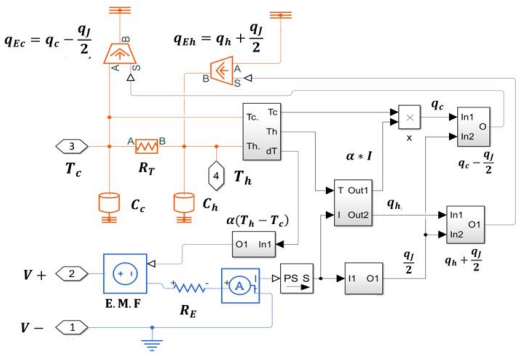
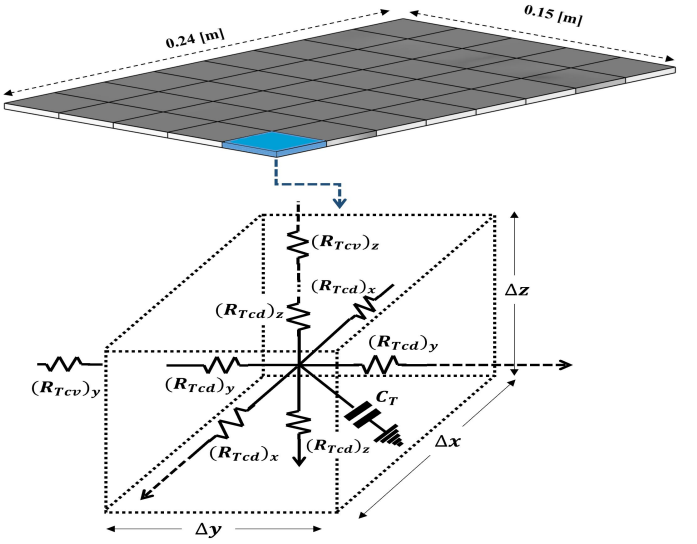
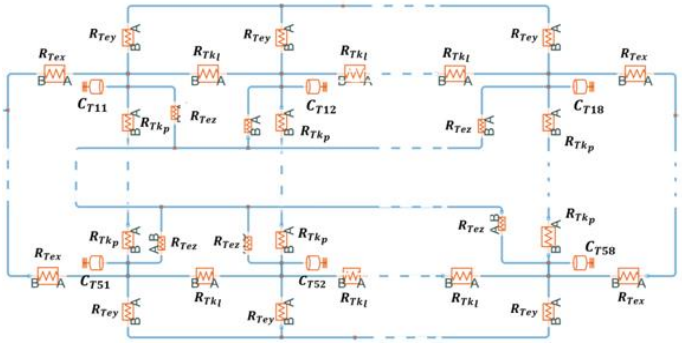


Figure 7. Peltier RC thermal circuit equivalent assembly in Simulink/Simscape including Peltier, Seebeck, and Joule effects.



(**a**)



(**b**)

Figure 8. Thermal plate interaction modeling: (**a**) based on RC network for individual volumetric elements and (**b**) Simulink/Simscape model.

4.1. Feedforward VRFT Controller Design

The VRFT controller synthesis is performed using the procedure shown in Figure 2. Initially, the reference models are defined, which are given by (18):

$$M_1(s) = \frac{w_n}{s + w_n}, \quad M_2(s) = \frac{w_n^2}{s^2 + 2w_n s + w_n^2}, \quad M_3(s) = \frac{w_n^2(T_z s + 1)}{s^2 + 2w_n s + w_n^2}. \tag{18}$$

By replacing (18) in (8) and (9), the open-loop transfer functions $G_{1,2,3}$ for the reference models $M_{1,2,3}$ are given by (19):

$$G_1(s) = \frac{w_n}{s}, \; G_2(s) = \frac{w_n^2}{s(2w_n + s)}, \; G_3(s) = \frac{w_n^2(T_z s + 1)}{s^2 + w_n(2 - w_n T_z)s}. \tag{19}$$

Once the models are defined, the open-loop response of the system is simulated by applying the PRBS signal shown in Figure 9a that sets a $\pm 100\%$ duty cycle in random periods to the Peltier system via the power driver to produce the output signal $y(t)$ shown in Figure 9b.

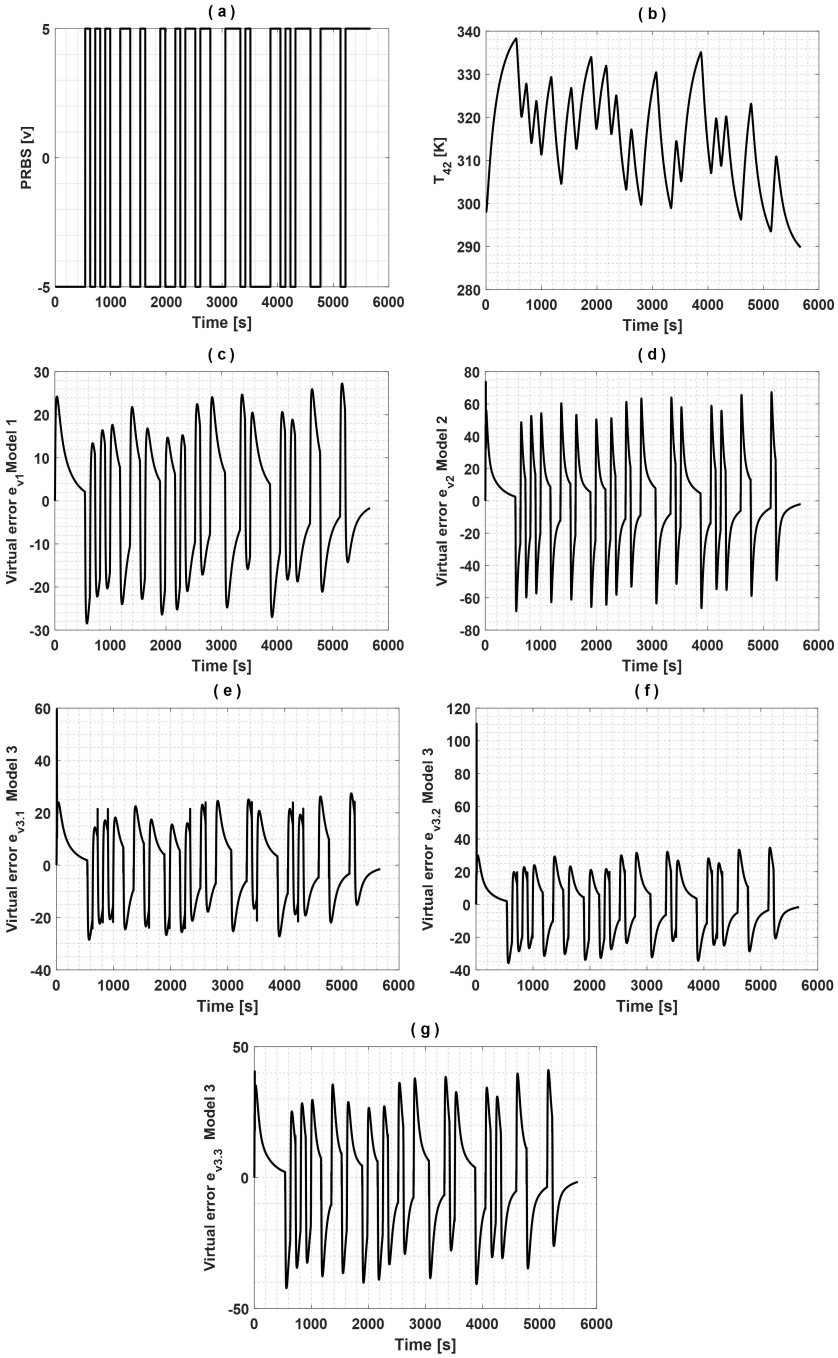


Figure 9. Peltier open-loop system identification. (**a**) Input (*PRBS* signal with 1/10 Hz periods), (**b**) Peltier temperature output $T(t)$, and (**c**–**g**) virtual errors for the reference models $G_{1-5}(s)$.

So, the virtual error signal e_{vr} is computed by evaluating the response of $G_{1,2,3}$ using $y(t)$ as input for $w_n = 11.66 \times 10^{-3}$ and $T_z = [\frac{1}{1.5w_n}, \frac{1}{2w_n}, \frac{1}{2.5w_n}]$, resulting in five virtual error signals $e_{vr_{1,\ldots,5}}$, shown in Figure 9c–g. In this case, a first-order compensator is selected as controller $C_{\rho n}$'s structure, given by (20), where α, β are the controller coefficients. It is important to notice that for the proposed application, a lag-type compensator is selected ($\alpha_0 < \beta_0$) to achieve a smooth response.

$$C_{\rho n} = \frac{\alpha_1 s + \alpha_0}{\beta_1 s + \beta_0}. \tag{20}$$

Finally, based on the virtual error signals $e_{vr_{1-5}}$, the VRFT controller can be determined via optimization, which is performed using Matlab's system identification toolbox, using as the input signal the virtual error $e_{vr_{1,\ldots,5}}$ and the input signal $u(t)$ as output following the logic shown in Figure 3. Thus, five controllers are obtained which are defined in (21) with an adjustment of 59.51%, 64.42%, 60.48%, 62.13%, and 59.95%, respectively.

$$\begin{aligned} C_{\rho 1} &= \frac{-278.61 \cdot 10^{-3} s - 1.181 \cdot 10^{-3}}{s + 150.48 \cdot 10^{-6}}, \quad C_{\rho 2} = \frac{-136.39 \cdot 10^{-3} s - 1.17 \cdot 10^{-3}}{s + 260.29 \cdot 10^{-6}}, \\ C_{\rho 3.1} &= \frac{-277.86 \cdot 10^{-3} s - 1.35 \cdot 10^{-3}}{s + 175.89 \cdot 10^{-6}}, \quad C_{\rho 3.2} = \frac{-225.23 \cdot 10^{-3} s - 1.281 \cdot 10^{-3}}{s + 172.61 \cdot 10^{-6}}, \\ C_{\rho 3.3} &= \frac{-196.79 \cdot 10^{-3} s - 1.27 \cdot 10^{-3}}{s + 190.58 \cdot 10^{-6}}. \end{aligned} \tag{21}$$

The controllers (21) are evaluated using the simulation model shown in Figure 10, which incorporates the feedforward compensation that uses the reference model and the controller transfer function with a tuning gain k as shown in Figure 4b. The closed-loop responses of the Peltier system and the control actions for the VRFT controllers can be observed in Figure 11. When $k = 0$, the feedforward is not active on the system, indicating an error regarding the desired trajectory y_d. Likewise, as k increases for $k \in [0; 2; 10]$, the error is minimized and becomes closer to the reference temperature.

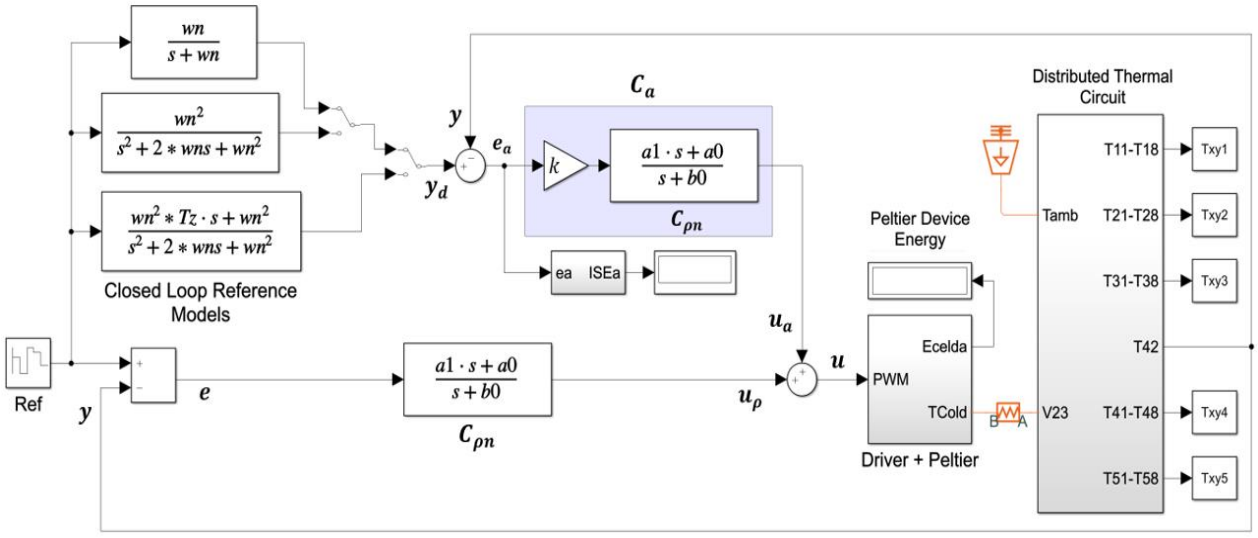


Figure 10. Feedforward VRFT controller implemented in Simulink composed by the reference models $M_{1,\ldots,5}$, the feedforward error compensation C_a, and the temperature uniformity control system.

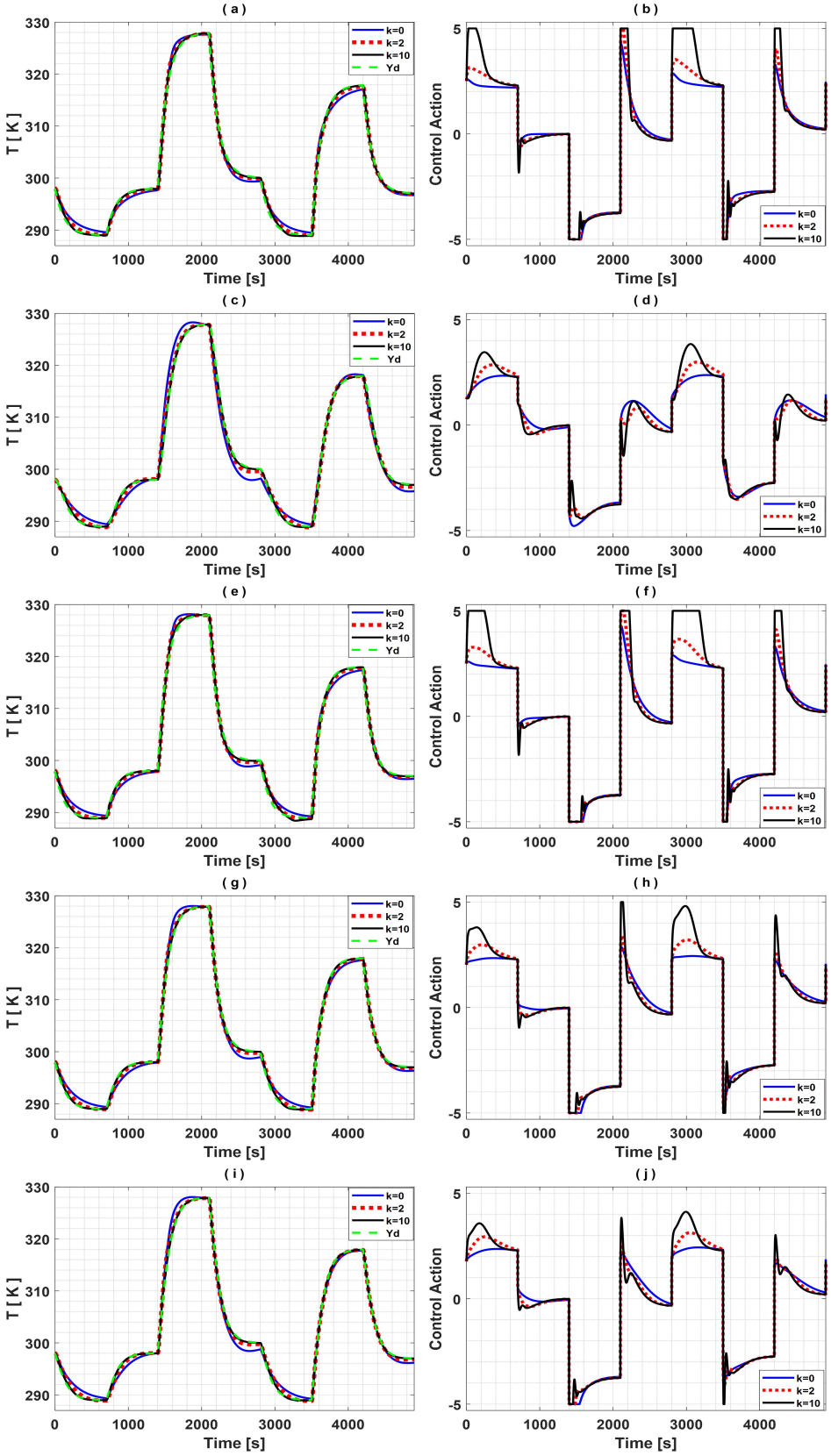


Figure 11. Temperature performance of controllers $C_{\rho_{1,\ldots,5}}$ (**a**–**e**) and their control actions (**f**–**j**) for different feedforward gains k. The feedforward compensates the tracking error but induces control action saturation as k increases.

The Integral Square Error ISE for the adaption error e_a and the Peltier cumulative energy u indices E_p given by (22) and (23) are used as key performance indicators (KPI) for the VRFT controller's quantitative analysis, where m is the number of samples. Likewise, the performance quantification against the feedforward gain is defined with the indices PD_{ISE} and PI_{EP}, defined in (24)–(25). These indices compute the percentage ratio between the ISE and E_P without feedforward $ISE(0)$, $E_P(0)$ and those related to the current feedforward employed $ISE(k)$ $E_P(k)$. The ISE and ITAE performance indices are standard KPIs to evaluate the transient response of the temperature in the Peltier cell given by the proposed VRFT compensators' C_ρ and the PI-IMC and PID-IMC controllers used as benchmark strategies. They also measure the tracking of the system to the desired response given by the reference model. Likewise, E_p is a performance index used to quantify the energy consumption of the Peltier Cell. Thus, the proposed indices can be used to provide a metric to determine a reasonable trade-off between the desired model's reference tracking and the energy consumption to find a suitable tuning rule for the feedforward gain k. Table 2 presents the performance indices for the synthesized VRFT controllers $C_{\rho_{1,\ldots,5}}$. For example, in the controller $C_{\rho_{3.3}}$, for $k = 0$, the adaptation error ISE $ISE_a(0) = 6008.65$ is bigger for any k. For values of k other than zero, for example, with $k = 2$, the adjustment to the desired response is improved by 75%, but the energy consumption increases by 8.7%.

The effect of the feedforward gain k is further analyzed using the ISE and E_P indices as shown in Figure 12. As can be observed, the sweeping of k for $1 \leq k \leq 20$ for the controller $C_{\rho 3.3}$ shows that more energy is required as the adjustment error is minimized. Moreover, for the five controllers $C_{\rho_{1,\ldots,5}}$, it is noticeable that controller $C_{\rho 1}$ provides the best fit to the reference model without a feedforward. On the other hand, the controller $C_{\rho 2}$ presents the greatest deviation compared to the desired response, although it should be noted that it has the lowest energy consumption. Likewise, it is noticeable that the most significant changes in the adjustment to the desired response as k increases occur in the interval $0 < k \leqslant 6$. However, the performance can be affected for considerably large values of k, which can generate problems with noise and saturation of the control action. Additionally, it can be observed that for $k = 4$, the controller $C_{\rho 3.3}$ achieves the best performance.

On the other hand, the VRTF controller's robustness is evaluated in the presence of external disturbance in the control action at $t = [1500, 2500, 3500, 4500]$ s, whose response is shown in Figure 13, and its quantitative evaluation is presented in Table 3. It can be observed that increasing k improves the tracking response and disturbance rejection. However, by increasing k, the control action becomes more aggressive, which, in turn, results in higher energy consumption.

$$ISE = \frac{1}{m}\sum_{j=0}^{m} e_a^2(j), \tag{22}$$

$$E_P = \frac{1}{m}\sum_{j=0}^{m} u^2(j), \tag{23}$$

$$PD_{ISE} = \left|\frac{ISE(0) - ISE(k)}{ISE(0)}\right| 100\%, \tag{24}$$

$$PI_{EP} = \left|\frac{E_p(0) - E_p(k)}{E_p(0)}\right| 100\%. \tag{25}$$

Table 2. VRFT controllers quantitative performance for different feedforward gain k on setpoint tracking tasks.

Controller	*Gain* k	$ISEa$	PD_{ISEa} [%]	E_P [J]	PI_{Ep} [%]
$C_{\rho 1}$	0	5227.24	0.0	46,981.76	0.0
	2	1997.81	61.8	54,220.91	15.4
	5	1391.21	73.4	61,055.75	30.0
	10	1328.20	74.6	70,605.25	50.3
$C_{\rho 2}$	0	12,341.97	0.0	41,406.34	0.0
	2	2745.85	77.8	43,332.89	4.7
	5	1028.79	91.7	45,161.89	9.1
	10	406.08	96.7	46,495.30	12.3
$C_{\rho 3.1}$	0	5440.11	0.0	49,806.09	0.0
	2	2250.38	58.6	58,327.48	17.1
	5	1733.08	68.1	68,945.06	38.4
	10	1724.12	68.3	78,772.28	58.2
$C_{\rho 3.2}$	0	5225.76	0.0	45,982.25	0.0
	2	1466.74	71.9	50,930.10	10.8
	5	666.45	87.2	54,343.65	18.2
	10	409.33	92.2	59,674.20	29.8
$C_{\rho 3.3}$	0	6008.65	0.0	44,689.47	0.0
	2	1503.10	75.0	48,560.12	8.7
	5	578.69	90.4	50,752.74	13.6
	10	264.06	95.6	52,740.13	18.0

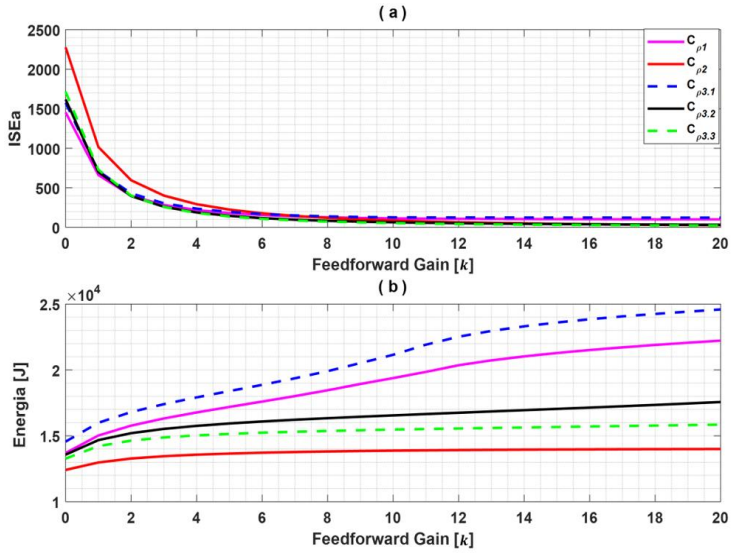


Figure 12. Sweeping of the adjustment error gain k vs. ISE and E_p for the synthesized controllers C_ρ. (**a**) explains the variation of ISE against k showing a error reduction for higher values of k. Likewise, (**b**) describes the variation of E_p against k showing an evident increment of energy consumption for higher k. Therefore, an acceptable performance is reached for $6 \leq k \leq 8$ with a good trade-off between error minimization and energy consumption.

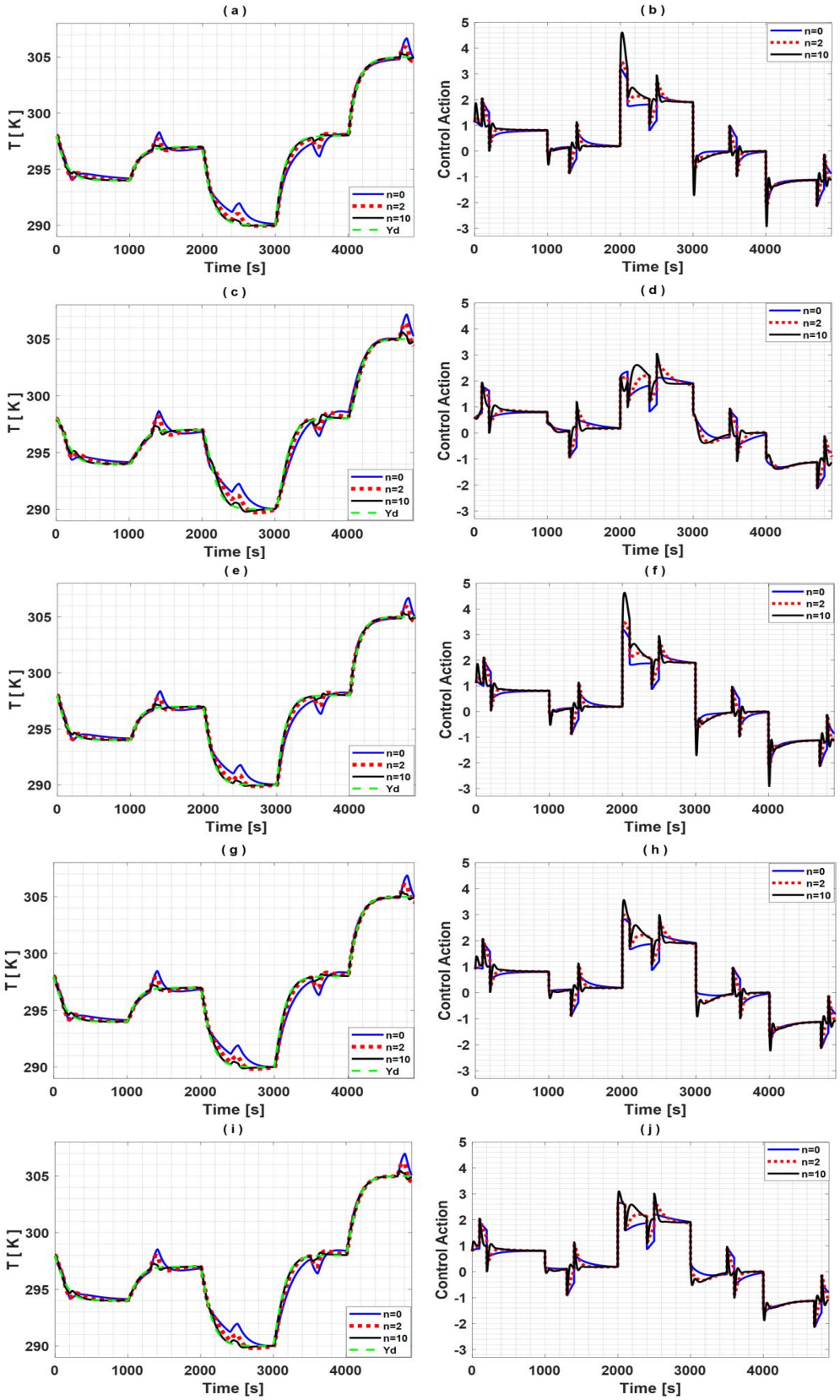


Figure 13. Temperature performance of controllers $C_{\rho_{1,\ldots,5}}$ (**a**–**e**) and their control actions (**f**–**j**) for different feedforward gains k in the presence of external disturbance in the control action at $t = [1500, 2500, 3500, 4500]$ s. The feedforward provides a disturbance rejection but increases the rejection energy required as k is bigger.

Table 3. VRFT controllers quantitative performance for different feedforward gain k on setpoint tracking tasks in the presence of thermal load disturbance.

Controller	*Gain* k	$ISEa$	PD_{ISEa} [%]	E_P [J]	PI_{Ep} [%]
$C_{\rho 1}$	0	1982.44	0.0	10,466.43	0.0
	2	496.87	74.9	11,650.06	11.3
	5	176.86	91.1	12,217.27	16.7
	10	68.73	96.5	12,947.27	23.7
$C_{\rho 2}$	0	2534.21	0	9702.82	0.0
	2	918.82	63.7	10,496.29	8.2
	5	420.37	83.4	10,647.11	9.7
	10	179.47	92.9	10,708.07	10.4
$C_{\rho 3.1}$	0	1849.71	0.0	10,893.32	0.0
	2	467.95	74.7	12,094.35	11.0
	5	169.15	90.9	12,697.67	16.6
	10	66.78	96.4	13,523.16	24.1
$C_{\rho 3.2}$	0	2069.79	0.0	10,468.95	0.0
	2	562.13	72.8	11,443.26	9.3
	5	208.87	89.9	11,744.04	12.2
	10	77.59	96.3	11,944.28	14.1
$C_{\rho 3.3}$	0	2153.58	0.0	10,299.65	0.0
	2	625.07	71.0	11,186.99	8.6
	5	243.00	88.7	11,399.55	10.7
	10	91.99	95.7	11,485.10	11.5

4.2. VRFT against PID Controllers Comparison

The obtained VRFT controllers are compared against three standard PID controllers: one tuned using the VRFT methodology proposed by [34] and PI and PID controllers tuned using the Internal Model Control (IMC) Method. For the PID-VRFT controller tuning, the system input/output response, shown in Figure 9a,b, is used with the reference model (26) and the PID controller structure (27). After the synthesis process, the controller parameters are $K_p = -18.08 \cdot 10^{-3}$, $K_d = -2.2 \cdot 10^{-3}$, and $K_i = -365.03 \cdot 10^{-6}$.

$$M(z^{-1}) = \frac{2.34 \cdot 10^{-3} + 67.27 \cdot 10^{-6} z^{-1} - 2.27 \cdot 10^{-3} z^{-2}}{1 - 1.977 z^{-1} + 0.9769 z^{-2}}, \tag{26}$$

$$C_{VRFT-PID} = K_p + \frac{K_i}{s} + \frac{K_d s}{1 + T_s s}. \tag{27}$$

The design of the PI-IMC and PID-IMC controllers follows the method proposed in [35,36]. As shown in Figure 14a, the controller G_c is the inverse of the process model $\tilde{P}^{-1}$ followed for a nth-order filter to ensure a proper controller implementation, making λ the only tuning parameter, where $G(s)$ is given by (28) and $\tilde{P}$ is the approximated process model.

$$G_c(s) = \frac{\tilde{P}}{(\lambda s + 1)^n}. \tag{28}$$

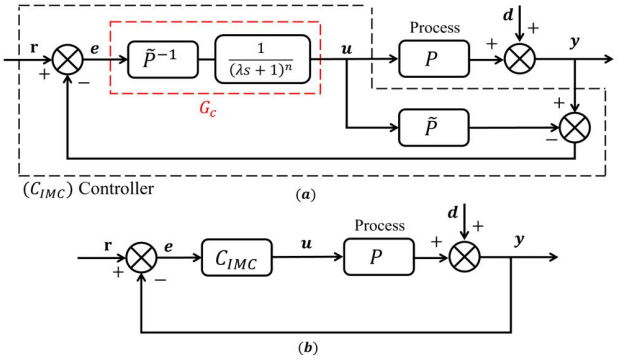


Figure 14. IMC controller framework design. (**a**) G_c is the inverse of the estimated process $\tilde{P}^{-1}$ plus a low-pass filter to obtain a proper and realizable controller. (**b**) shows the rearrangement of the IMC controller as a PI feedback regulator C_{IMC} given by (32).

So, (28) can be simplified, as shown in Figure 14b, where the resulting controller C_{IMC} can be represented as (29), according to [36].

$$C_{IMC}(s) = \frac{G_c}{(1 - \tilde{P}G_c)} = \frac{\tilde{P}^{-1}}{(\lambda s + 1)^n - 1}. \tag{29}$$

Thus, for the PI-IMC controller design, $\tilde{P}_1$ is selected as a first-order system calculated using the system response shown in Figure 9, resulting in (30),

$$\tilde{P}_1 = \frac{a_0}{b_1 s + 1} = \frac{-5.58}{211.51s + 1}, \tag{30}$$

which can be rewritten as (31)

$$C_{IMC_1}(s) = -\left(\frac{b_1}{a_0\lambda_1} + \frac{1}{a_0\lambda_1 s}\right). \tag{31}$$

and, considering the PI controller form in (32), its terms can be compared with (31) to obtain the PI controller's parameters as $K_{p1} = \frac{b_1}{a_0\lambda_1}$, $I_1 = \frac{1}{b_1}$, resulting in $k_{p1} = -261.21e^{-3}$, $k_{i1} = 4.727e^{-3}$ for $\lambda = 145$.

$$PI(s) = K_{p1}\left(1 + \frac{I_1}{s}\right). \tag{32}$$

For the PID-IMC, a second-order model of the process is obtained, which is given by (33)

$$\tilde{P}_2 = \frac{c_0}{s^2 + d_1 s + 1 + d_0} = \frac{-2.929 \cdot 10^{-3}}{s^2 + 108.97 \cdot 10^{-3} s + 545.65 \cdot 10^{-6}}, \tag{33}$$

and replacing (33) in (29) with $n = 1$, the IMC controller can be expressed as (34)

$$C_{IMC_2}(s) = -\frac{s^2 + d_1 s + d_0}{(c_1 s + c_0)\lambda_2 s} = -\left(\frac{s^2 + d_1 s + d_0}{c_0\lambda_2 s}\right)\left(\frac{1}{\frac{c_1}{c_0} + 1}\right). \tag{34}$$

Thus, comparing the PID controller form given by (35) with (34), the PID controller's gains can be calculated as $K_{p2} = \frac{d_1}{2c_0\lambda_2}$, $T_{d2} = \frac{1}{d_1}$; $I_2 = \frac{d_0}{d_1}$; $T_{f2} = \frac{\lambda_2}{2}$ resulting in $K_{p2} = -206e^{-3}$, $I_2 = 9.176$, $T_{d2} = 5e^{-3}$, and $T_{f2} = 45$ for $\lambda_2 = 45$.

$$PID(s) = K_{p2}\left(1 + \frac{I_2}{s} + T_{d2}s\right)\left(\frac{1}{T_{f2}s + 1}\right). \tag{35}$$

The performances of the PI-IMC and PID-IMC controllers are contrasted against the VRFT controller $C_{\rho 3.3}$ in three operating conditions: nominal, random noise, and external disturbance (adding external thermal load) using the ISE (22), $ITAE$ (36), and E_p (23) performance indices. The heat load disturbance used for the testing, shown in Figure 15c, modifies the thermal properties of the V_{ij} volumetric element. The thermal disturbance added to each V_{ij} element is given by $C_T = mc_p$, where m is a thermal mass of 1 kg and c_p is the specific heat of $c_p = 3671$ J/kg/K. The volumetric elements affected are $V_{ij} = [V_{24}, V_{25}, V_{36}, V_{33}]$, and the disturbance is set at each element at the times $t = [1500, 2500, 3500, 4500]$ s, respectively.

The controller's response to different temperature setpoints without disturbance (nominal) is shown in Figure 15a and Table 4. It is observed how, with the feedforward, the VRFT control performs better setpoint tracking compared to the PID-VRFT, PI-IMC and PID-IMC controllers. Likewise, the VRFT control action reacts faster to changes in process operation with an energy consumption similar to the other control systems. In the presence of random noise, the controllers response is shown in Figure 15b and Table 5. In this case, the feedforward VRFT is more sensitive to random noise compared with the PI-IMC, and PID-IMC controllers. Finally, when thermal load disturbance at $t = [1500, 2000, 3000, 4000]$ s is added to the system, the controller's response is given by Figure 15c and Table 6. It is observed how the feedforward VRFT controller rejects the disturbances more efficiently than the PI-IMC and PID-IMC controllers but requires a greater energy consumption to perform the disturbance rejection.

$$ITAE = \frac{t_s}{m} \sum_{j=0}^{m} e_a^2(j). \tag{36}$$

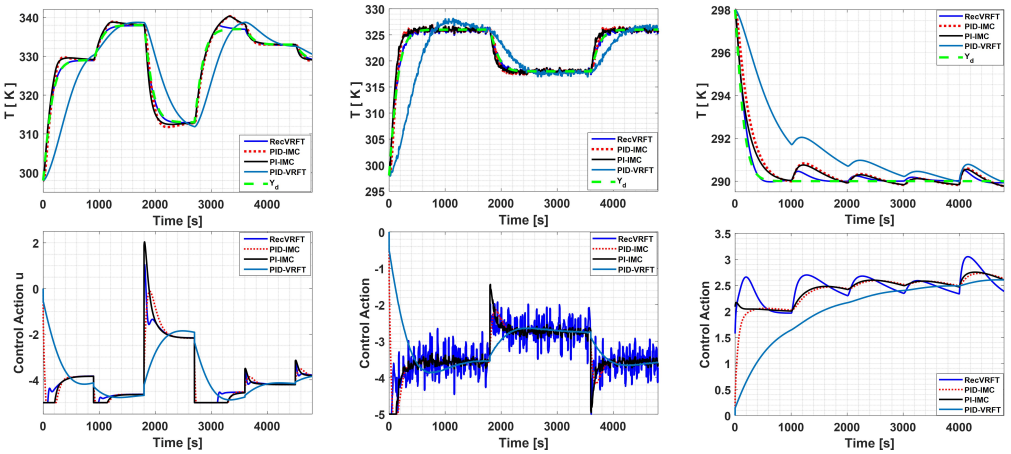


Figure 15. Recursive feedforward VRFT and PI/PID controllers' response: (**a**) temperature setpoint tracking; (**b**) random noise in the feedback loop; and (**c**) external disturbance produced by thermal load on the plate. The VRFT controller with a feedforward provides a more robust response for tracking and external disturbance.

Table 4. Process performance by changing the operating point.

Control System	ISE_a	$ITAE_a$	E_P
Recursive-VRFT	1009.65	3,543,789.93	120,879.63
IMC-PID	8796.40	10,322,383.31	122,416.94
IMC-PI	10,426.44	10,251,645.23	124,000.37
VRFT-PID	187,220.93	35,687,688.87	113,657.34

Table 5. Control system performance with feedback noise.

Control System	ISE_a	$ITAE_a$	E_P
Recursive-VRFT	718.74	3,275,236.08	83,427.47
IMC-PID	2644.18	4,284,296.65	83,738.90
IMC-PI	4079.30	4,801,885.60	84,462.43
VRFT-PID	91,064.21	14,019,821.75	77,352.45

Table 6. Process performance applying thermal load on the metallic plate.

Control System	ISE_a	$ITAE_a$	E_P
Recursive-VRFT	193.76	1,228,571.91	49,499.74
IMC-PID	1619.68	2,599,650.84	46,954.49
IMC-PI	736.25	2,306,393.53	47,797.99
VRFT-PID	16,352.03	7,674,062.47	36,716.94

5. Experimental Validation of Recursive VRFT

The experimental platform shown in Figure 5 is employed to perform experimental validation of the proposed recursive VRFT. Thus, the procedure described in Figure 2 is followed again to synthesize the controllers. First, second- and first-order reference models (37) are employed, which are based in G_1, G_2 defined in (19) with $w_n = 8 \times 10^{-3}$ for G_1 and $w_n = 11.66 \times 10^{-3}$ for G_2. Each model includes a low-pass filter $H(s) = \frac{1}{0.1371s+1}$ to compensate for the thermal camera feedback noise. In this case, the low-pass filter deals with the existing temperature randomness. However, it may not be accurate for more complex systems involving more variables with more robust stochastic behavior. Therefore, including a stochasticity awareness mechanism based on the randomness modeling that performs entropy minimization on the VRFT controller synthesis, such as [37,38], could lead to a more robust and stable compensator.

$$\begin{aligned} G_1(s) &= \frac{w_n^2}{s(2w_n+s)} H(s) \\ G_2(s) &= \frac{w_n}{s} H(s). \end{aligned} \tag{37}$$

Then, the open-loop response of the system is obtained by applying a PRBS signal, which is shown in Figure 16. So, the virtual error signals are computed using (37) along with the data in Figure 16. Thus, the VRFT controllers are synthesized using the compensator structures (38) and (39), along with the reference models (37), to obtain the first- and second-order controllers, respectively.

$$C_1(s) = \frac{A_1 s + A_0}{B_1 s + B_0} = \frac{2.75s + 0.1}{s + 640 \cdot 10^{-6}}, \tag{38}$$

$$C_2(s) = \frac{A_2 s^2 + A_1 s + A_0}{B_2 s^2 + B_1 s + B_0} = \frac{2.12s^2 + 0.11s + 543 \cdot 10^{-6}}{s^2 + 6.6 \cdot 10^{-3} s + 6.38 \cdot 10^{-12}}. \tag{39}$$

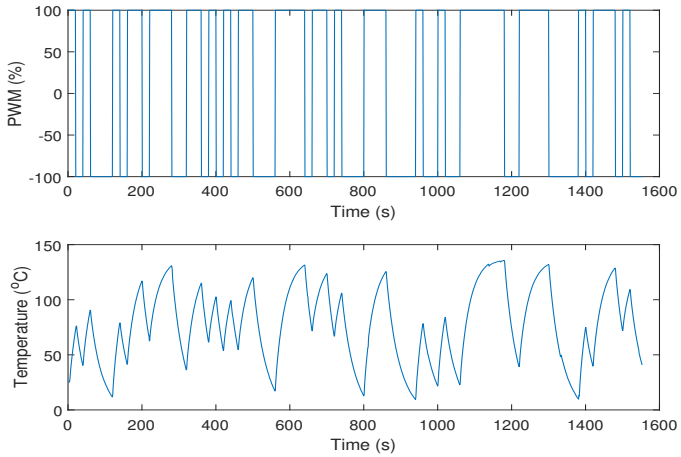


Figure 16. Temperature uniformity system open-loop response for a PRBS input signal.

Likewise, a PI controller is tuned using the Ziegler–Nichols method to compare the performance of the obtained VRFT controllers whose gains are $k_p = 25.05$ $k_i = 0.713$. The controller implementation is performed using a hardware in the loop configuration, shown in Figure 17, that can be downloaded from https://github.com/tartanus/Recursive-VRFT (accessed on 16 October 2023). As can be observed, the VRFT controllers are implemented in Simulink with real-time TCP/IP feedback for the infrared camera. Likewise, the PWM is applied using an Arduino board communicated via serial communication.

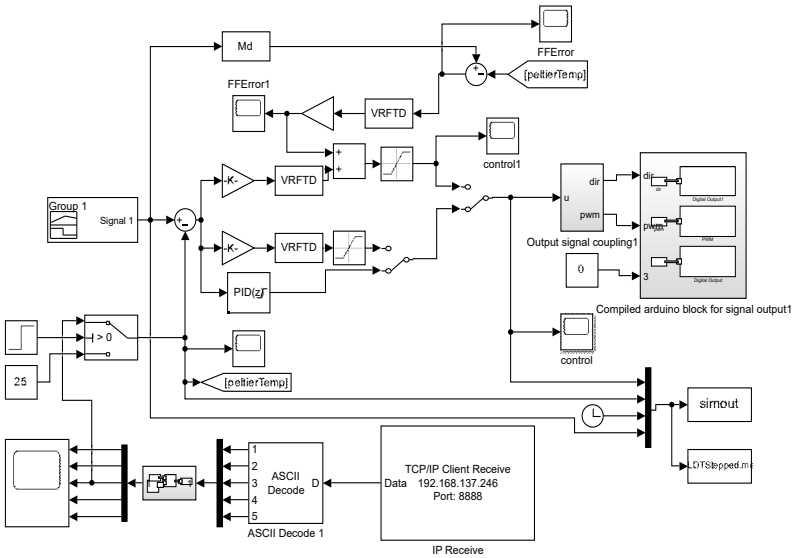
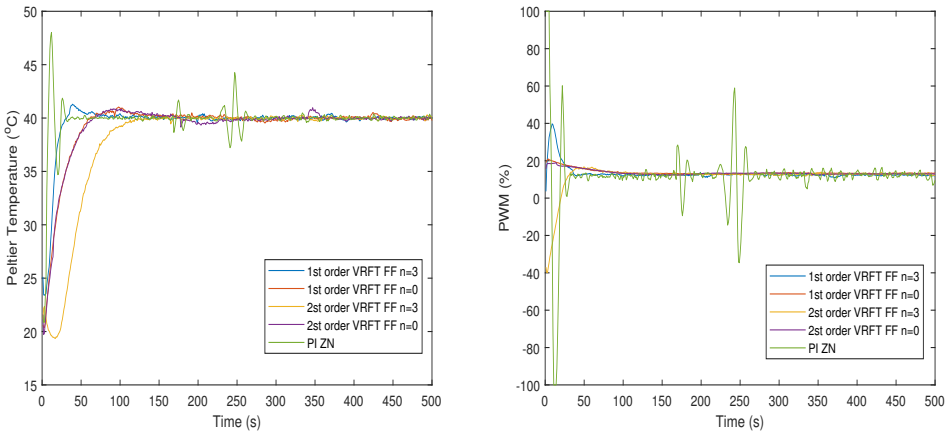


Figure 17. Matlab/Simulink feedforward VRFT HIL implementation. The implementation includes the TCP-IP feedback from thermal camera, VRFT controller, and feedback scheme with output interface using Arduino.

The response and control action for the VRFT controllers with a feedforward gain ($k = 3$), without a feedforward ($k = 0$), and the PI controller are shown in Figure 18a,b. As can be observed, the first-order VRFT controller with a feedforward has a better response, reaching a settling time of 100 s with a reasonable overshoot. Moreover, the second-order VRFT controller performance induces a slower response without feedforward compensation. Likewise, the PI controller has a faster settling time but a high overshoot and sensitivity in the presence of small feedback fluctuations. In this case, the presence of jitter during the time range [150–300] s is caused by the infrared thermal camera. It performs an automatic recalibration at 169 s to adjust the brightness and irradiance, which can be considered an external disturbance to the system. Thus, the high proportional and integral

gains obtained for the ZN-PI controller compared with the gains on the C1 and C2 controllers synthesized using the recursive VRFT method make the system more sensitive to external disturbances and the presence of random noise, causing the aggressive response on the ZN-PI controller. In terms of control action, the first-order VRFT requires slightly more energy in the beginning compared with the second-order controller but provides a fast correction of the setpoint. Therefore, we can say that for the Peltier system, the VRFT controller with a feedforward can improve the system response as well as improve its robustness in the presence of small disturbances.



(**a**) (**b**)

Figure 18. Temperature uniformity control response (**a**) for the VRFT and PI controllers computed control action (**b**) for a 40 °C setpoint. As can be observed, the time response (**a**) for the first-order VRFT controller with Feedforward gain $k = 3$ provides a faster response with less sensitivity to random noise compared with the other controllers. Also, the control action (**b**) of the VRFT first and second controller controllers provides an smooth and robust response compared with the PI-ZN in the presence of external disturbances as at $t = 250$ s caused by the thermal camera automatic recalibration.

6. Conclusions and Future Works

This paper presented a framework for synthesizing VRFT controllers with a feedforward using an optimization approach that allows the parameter calculation for different controller architectures based on a desired reference model and the system's open-loop response. It incorporates a feedforward strategy that minimizes the adjustment error between the desired reference model and system output. It employs a proportional gain k times the synthesized controller, providing an additional corrective control action that minimizes the adjustment error. The method was tested for the VRFT controller design of a temperature uniformity control system, using first and second-order reference models. The simulation and experimental results indicate that the VRFT controller with a feedforward improves the system's setpoint tracking response and external disturbance rejection, which increases as the feedforward gain k is increased. Due to the system's nature, the first-order compensator responds better than the high-order synthesized controllers. However, it is important to note that for more complex systems and controller architectures, the proposed VRFT synthesis can provide a reasonable and robust controller response based only on the available open-loop system's data. In future work, the proposed VRFT controller synthesis method can be modified for online controller tuning based on the open-loop or closed-loop system responses while ensuring its robustness and stability. Likewise, the evaluation of different reference models and controller architectures, such as non-linear or multivariable controls, is proposed.

Author Contributions: Conceptualization, J.G.A., L.A., J.V. and Y.C.; methodology, J.G.A. and L.A.; software, J.G.A. and L.A.; validation, J.G.A., L.A. and J.V.; formal analysis, L.A. and Y.C.; investigation, J.G.A. and L.A.; resources, J.G.A., L.A., J.V. and Y.C.; data curation, J.G.A., L.A. and J.V.; writing—original draft preparation, J.G.A., L.A. and J.V.; writing—review and editing, J.G.A., L.A., J.V. and Y.C.; visualization, J.G.A. and L.A.; supervision, L.A. and Y.C.; project administration, L.A.; funding acquisition, L.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hou, Z.S.; Wang, Z. From model-based control to data-driven control: Survey, classification and perspective. *Inf. Sci.* **2013**, *235*, 3–35. [CrossRef]
2. Guardabassi, G.O.; Savaresi, S.M. Virtual reference direct design method: An off-line approach to data-based control system design. *IEEE Trans. Autom. Control* **2000**, *45*, 954–959. [CrossRef]
3. David, J.; Fernández, R. Extensions and Applications of the Virtual Reference Feedback Tuning. Ph.D. Thesis, Universitat Autònoma de Barcelona, Barcelona, Spain, 2011.
4. Beninca, M.R. Virtual Reference Feedback Tuning of Controllers Parametrized Using Orthonormal Basis Functions. Ph.D. Thesis, Paraná Federal Univeristy, Parana, Brazil, 2015.
5. Imchen, S.; Das, D.K. Scheduling of distributed generators in an isolated microgrid using opposition based Kho-Kho optimization technique. *Expert Syst. Appl.* **2023**, *229*, 120452. [CrossRef]
6. Tian, J.; Zeng, Y.; Ji, L.; Zhu, H.; Guo, Z. Control Method of Cold and Hot Shock Test of Sensors in Medium. *Sensors* **2023**, *23*, 6536. . [CrossRef] [PubMed]
7. Wang, S.; Zhao, B.; Yi, S.; Zhou, Z.; Zhao, X. GAPSO-Optimized Fuzzy PID Controller for Electric-Driven Seeding. *Sensors* **2022**, *22*, 6678. . [CrossRef] [PubMed]
8. Pradhan, S.K.; Subudhi, B. Position control of a flexible manipulator using a new nonlinear self-tuning PID controller. *IEEE/CAA J. Autom. Sin.* **2020**, *7*, 136–149. [CrossRef]
9. Zhu, G.; Ma, Y.; Hu, S. Event-Triggered Adaptive PID Fault-Tolerant Control of Underactuated ASVs Under Saturation Constraint. *IEEE Trans. Syst. Man Cybern. Syst.* **2023**, *53*, 4922–4933. [CrossRef]
10. Chen, Q.; Wang, Y.; Song, Y. Tracking Control of Self-Restructuring Systems: A Low-Complexity Neuroadaptive PID Approach With Guaranteed Performance. *IEEE Trans. Cybern.* **2023**, *53*, 3176–3189. [CrossRef]
11. Sánchez-Palma, J.; Ordoñez-Ávila, J.L. A PID Control Algorithm With Adaptive Tuning Using Continuous Artificial Hydrocarbon Networks for a Two-Tank System. *IEEE Access* **2022**, *10*, 114694–114710. [CrossRef]
12. Shiro, M.; Matsui, Y. Data-Driven Control and Learning Systems. *IEEE Trans. Ind. Electron.* **2017**, *64*, 4070–4075. . [CrossRef]
13. Hjalmarsson, H.; Gevers, M.; Gunnarsson, S.; Lequin, O. Iterative feedback tuning: Theory and applications. *IEEE Control Syst. Mag.* **1998**, *18*, 26–41. [CrossRef]
14. Kutz, J.N.; Brunton, S.L.; Brunton, B.W.; Proctor, J.L. *Dynamic Mode Decomposition*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2016. [CrossRef]
15. Chen, Y.; Wen, C. *Iterative Learning Control: Convergence, Robustness and Applications*, 1st ed.; Springer: London, UK, 1999; p. 204. [CrossRef]
16. Ariyur, K.; Krstic, M. *Real-Time Optimization by Extremum-Seeking Control*; Wiley-Interscience: Hoboken, NJ, USA , 2003.
17. Hou, Z.; Jin, S. *Model Free Adaptive Control: Theory and Applications*; CRC Press: Boca Raton, FL, USA, 2013.
18. Safonov, M.; Tsao, T.C. The unfalsified control concept and learning. *IEEE Trans. Autom. Control* **1997**, *42*, 843–847. [CrossRef]
19. Spall, J.C. *Introduction to Stochastic Search and Optimization*, 1st ed.; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2003.
20. Lecchini, A.; Campi, M.C.; Savaresi, S.M. Sensitivity shaping via virtual reference feedback tuning. In Proceedings of the Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No.01CH37228), Orlando, FL, USA, 4–7 December 2001; Volume 1, pp. 750–755. [CrossRef]
21. Campi, M.C.; Lecchini, A.; Savaresi, S.M. Virtual reference feedback tuning: A direct method for the design of feedback controllers. *Automatica* **2002**, *38*, 1337–1346. [CrossRef]
22. Sato, T.; Sakai, Y.; Kawaguchi, N.; Arrieta, O. Dual-Rate Data-Driven Virtual Reference Feedback Tuning: Improvement in Fast-Tracking Performance and Ripple-Free Design. *IEEE Access* **2021**, *9*, 144426–144437. [CrossRef]
23. Remes, C.L.; Gomes, R.B.; Flores, J.V.; Líbano, F.B.; Campestrini, L. Virtual Reference Feedback Tuning Applied to DC–DC Converters. *IEEE Trans. Ind. Electron.* **2021**, *68*, 544–552. [CrossRef]

24. Ding, Y.; Xu, N.; Ren, L.; Hao, K. Data-Driven Neuroendocrine Ultrashort Feedback-Based Cooperative Control System. *IEEE Trans. Control. Syst. Technol.* **2015**, *23*, 1205–1212. [CrossRef]
25. Campestrini, L.; Eckhard, D.; Chía, L.A.; Boeira, E. Unbiased MIMO VRFT with application to process control. *J. Process Control* **2016**, *39*, 35–49. . [CrossRef]
26. Chiluka, S.K.; Ambati, S.R.; Sonawane, S.H.; Seepana, M.M.; Babu Gara, U.B. Robust IMC-PID Controller Design using VRFT: Theoretical and Experimental Investigation; IFAC-PapersOnLine . In Proceedings of the 7th International Conference on Advances in Control and Optimization of Dynamical Systems ACODS 2022, Silchar, India, 22–25 February 2022; Volume 55, pp. 241–246. [CrossRef]
27. Vilanova, R.; Visioli, A. *PID Control in the Third Millennium: Lessons Learned and New Approaches*; Springer: London, UK, 2012.
28. Formentin, S.; Campi, M.C.; Carè, A.; Savaresi, S.M. Deterministic Continuous-Time Virtual Reference Feedback Tuning (VRFT) with Application to PID Design. *Syst. Control Lett.* **2019**, *127*, 25–34. . [CrossRef]
29. Freddy F.; Valderrama Gutierrez, F.O.R. Novel Data-Driven Control Approaches. Ph.D. Thesis, Pontifical Javerian University, Bogota, Colombia, 2019.
30. Viola, J.; Radici, A.; Dehghan, S.; Chen, Y. Low-cost real-time vision platform for spatial temperature control research education developments. In Proceedings of the International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Anaheim, CA, USA, 18–21 August 2019; American Society of Mechanical Engineers: New York, NY, USA, 2019; Volume 59292, p. V009T12A030.
31. Viola, J.; Oziablo, P.; Chen, Y. A Portable and Affordable Networked Temperature Distribution Control Platform for Education and Research. *IFAC-PapersOnLine* **2020**, *53*, 17530–17535. [CrossRef]
32. Lineykin, S.; Ben-Yaakov, S. Modeling and analysis of thermoelectric modules. *IEEE Trans. Ind. Appl.* **2007**, *43*, 505–512. [CrossRef]
33. Araque–Mora, J.; Angel, L. Distributed Equivalent Circuit for Modeling Heat Transfer Process in a Thermoelectric System. In Proceedings of the 2021 IEEE 5th Colombian Conference on Automatic Control (CCAC), Ibague, Colombia, 19–22 October 2021; pp. 326–331. [CrossRef]
34. Carè, A.; Torricelli, F.; Campi, M.C.; Savaresi, S.M. A Toolbox for Virtual Reference Feedback Tuning (VRFT). In Proceedings of the 2019 18th European Control Conference (ECC), Naples, Italy, 25–28 June 2019; pp. 4252–4257. [CrossRef]
35. Angel, L.; Viola, J.; Paez, M. Speed control of a motor-generator system using internal model control techniques. In Proceedings of the 2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC), Cartagena, Colombia, 18–20 October 2017; pp. 1–6. [CrossRef]
36. Nath, U.M.; Dey, C.; Mudi, R.K. Review on IMC-based PID Controller Design Approach with Experimental Validations. *IETE J. Res.* **2021**, *69*, 1640–1660. [CrossRef]
37. Mifeng Ren, Q.Z.; Zhang, J. An introductory survey of probability density function control. *Syst. Sci. Control. Eng.* **2019**, *7*, 158–170. [CrossRef]
38. Zhang, Q.; Zhou, Y. Recent Advances in Non-Gaussian Stochastic Systems Control Theory and Its Applications. *Int. J. Netw. Dyn. Intell.* **2022**, *1*, 111–119. [CrossRef]