MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Extremum seeking controller tuning for heat pump optimization using failure-robust Bayesian optimization

Chakrabarty, Ankush; Burns, Daniel J.; Guay, Martin; Laughman, Christopher R.

TR2022-144 November 15, 2022

Abstract

Extremum seeking controllers have been investigated for multivariable data-driven energy optimization in heat pumps. In particular, proportional-integral extremum seeking control (PI-ESC) has demonstrated potential for significant accel- eration compared to other ESC variants for nonlinear closed-loop control systems. A barrier to PI-ESC's utilization in self-optimizing control is the fact that the PI-ESC algorithm is fragile. That is, unless the PI-ESC gains (e.g., controller gains, estimator gains) are carefully tuned, small perturbations to these gains can render the closed-loop unstable. Since arbitrary combinations of PI-ESC gains can result in instabilities, we propose a failure-robust Bayesian optimization (FRBO) algorithm that computes PI-ESC gains that ensure the closed-loop system can be driven rapidly to the optimum, while identifying and avoiding regions in the space of PI-ESC gains that are likely to result in instabilities (i.e., failures). The FRBO-tuned PI-ESC is shown to result in rapid closed-loop convergence to optimal values both on benchmark examples and a production-level model of an air conditioning system.

Journal of Process Control 2022

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Extremum seeking controller tuning for heat pump optimization using failure-robust Bayesian optimization

Ankush Chakrabarty[a,1], Daniel J. Burns, Martin Guay[b], Christopher R. Laughman[a]

[a] *Mitsubishi Electric Research Laboratories, Cambridge, MA, USA*
[b] *Department of Chemical Engineering, Queens University, ON, Canada.*

Abstract

Extremum seeking controllers have been investigated for multivariable data-driven energy optimization in heat pumps. In particular, proportional-integral extremum seeking control (PI-ESC) has demonstrated potential for significant acceleration compared to other ESC variants for nonlinear closed-loop control systems. A barrier to PI-ESC's utilization in self-optimizing control is the fact that the PI-ESC algorithm is fragile. That is, unless the PI-ESC gains (e.g., controller gains, estimator gains) are carefully tuned, small perturbations to these gains can render the closed-loop unstable. Since arbitrary combinations of PI-ESC gains can result in instabilities, we propose a failure-robust Bayesian optimization (FRBO) algorithm that computes PI-ESC gains that ensure the closed-loop system can be driven rapidly to the optimum, while identifying and avoiding regions in the space of PI-ESC gains that are likely to result in instabilities (i.e., failures). The FRBO-tuned PI-ESC is shown to result in rapid closed-loop convergence to optimal values both on benchmark examples and a production-level model of an air conditioning system.

Keywords: Multivariable control, self-optimizing control, Gaussian processes, Bayesian optimization, extremum seeking, vapor compression system, refrigerant cycle.

1. Introduction

Vapor compression systems (VCSs) provide essential functionality in many energy transfer systems, such as heat pumps, refrigeration, and organic Rankine cycles, due to their cost-effectiveness and ability to operate reliably and efficiently over a long period of time [1]. While a wide variety of cycle architectures can be adopted for different applications, typical components of these architectures include: compressors for pressurizing and pumping a working fluid, valves for metering flow and regulating pressures in different parts of the system, and heat exchangers that enable the exchange of thermal energy between two or more media. The design of these VCSs have continued to evolve to meet increasingly stringent specifications, and key trends that characterize recent developments include the use of multiple interacting fluid circuits, environmentally friendly working fluids, and variable actuators to achieve better control and energy performance [2, 3].

Variable actuators play a particularly important role in contemporary VCSs because they enable the factory-produced equipment performance to adapt to real-world disturbances and uncertainties. However, such variable actuators introduce additional complexity into the system design because they require the use of feedback controllers to regulate the system's dynamic behavior to prescribed actuator setpoints. These setpoints represent an additional degree of design flexibility, and properly assigned setpoints for the VCS actuators can optimize closed-loop performance without further alterations to the feedback controller [4].

Since the VCS dynamics are coupled to the dynamics of the surrounding environment, controllers must be designed to ensure robust performance despite a wide array of disturbances that affect the system. Decentralized feedback loops with proportional-integral controllers [5] and model predictive controllers [6, 7] have both proven to be effective in these applications. These controllers comprise multiple tunable 'outer loop' parameters (e.g., setpoints) that can be adjusted to optimize performance after the 'inner-loop' parameters (e.g., PID gains) are fixed. For VCS applications, the compressor speed is often controlled to regulate the heating capacity of a heat pump in response to disturbances in ambient conditions or room occupancy, while the setpoints for the remaining actuators can be chosen to optimize the system performance. In comparison to the feedback control design problem, the task of obtaining this set of optimal setpoints as a function of the regulated inputs and the driving conditions is challenging due to the nonlinear, multi-variable, and often unmodeled dynamics of the closed-loop system.

Recent studies have documented the use of extremum-

Email addresses: achakrabarty@ieee.org (Ankush Chakrabarty), danburns@ieee.org (Daniel J. Burns), guaym@queensu.ca (Martin Guay), laughman@merl.com (Christopher R. Laughman)

[1] Corresponding author. *Phone*: +1 (617) 758-6175.

seeking control (ESC) design techniques to compute energy-efficient operating points of complex physical systems [8–12]. Extremum seeking algorithms are model-free techniques that perform a gradient descent on an unknown convex map representing the steady-state relationship between manipulated inputs and a performance output. Since ESC works without an *a priori* characterization of this map, the approach is inherently robust to disturbances and the wide variation of environments in which vapor compression systems are deployed. However, convergence to the optimizer using the common perturbation-based extremum seeking control occurs at a rate two timescales slower than the dominant plant dynamics—a severe limitation for thermal systems with time constants of tens or hundreds of minutes [9]. To address this, a proportional-integral extremum seeking control (PI-ESC) has been developed that converges to the optimizer at the same timescale as the dominant plant dynamics [13]. Unlike the prior work in [9, 11, 13] which relied on trial-and-error based tuning methods for PI-ESC using a simplified reduced-order dynamical model of the VCS, the main contribution of this paper is the development of an automated tuning procedure for the algorithm. The tuning procedure employs a Bayesian optimization approach applied to a high-fidelity black-box simulation model to assign every parameter of the PI-ESC used as decision variables of the closed-loop system. This provides a holistic approach to ESC tuning that considers simultaneously the gains of estimation and control as well as the parameters of the dither signal tuning strategy. The resulting optimal tuning guarantees closed-loop transient performance when implemented to physical system.

While ESC offers an efficient data-driven optimization technique for systems with unmodeled dynamics with objective functions whose analytical forms are difficult to obtain, applying ESC to practical problems remains challenging. In particular, the tuning of ESC techniques can be exceptionally difficult in the absence of adequate information about the dynamics of the system or the nonlinearity of the objective function. ESC requires a minimum number of underlying assumptions that should be met to guarantee its performance. Since these assumptions cannot be checked explicitly in the absence of accurate process models, many situations arise where the unknown system violates the assumptions. This problem can cause deterioration of the performance of the resulting ESC. The remedying strategy would then be to attempt to improve its performance by using some alternative tuning of the ESC parameters. This problem is shared by all ESC design techniques to varying degrees. The primary contribution of this study is a systematic formalism to improve the tuning of PI-ESC in vapor compression systems. Of primary interest is the estimation-based technique proposed in [9, 11, 13]. This class of techniques has been shown to perform well in VCS in practice. However, it is important to note that the proposed approach could be used for other newer dual-mode based techniques such as the perturbation-based approach [14] and Lie bracket averaging techniques [15]. Such techniques, while limited to continuous-time systems, can be implemented using a smaller number of tuning parameters. They also require specific choices of high frequency sinusoidal dither signals that may not be suitable for application in physical systems such as vapor compression systems.

PI-ESC (or dual-mode ESC) has been shown to offer significant improvements in convergence rates in VCS [9]. However, it remains difficult to tune because its multiple parameters interact in non-intuitive ways and, in some instances, certain combinations have been observed to render the closed-loop system unstable. In this paper, we present a Bayesian optimization (BO) approach that automatically discovers combinations of PI-ESC gains that achieve rapid convergence and outperforms gains obtained by extensive manual tuning. Since BO is a global, derivative-free optimization methodology that is designed to obtain solutions without a large number of function evaluations or iterations, it has been reported to perform well on controller tuning problems for many industrial applications [16, 17]. BO's data-driven nature implies that it is agnostic to the control architecture under consideration and immune to unmodeled dynamics of the closed-loop system [18–20], and additional constraints can be incorporated into the optimization process such as context, robustness, and safety [21–23]. In fact, BO has been successfully implemented on a simplified heat pump model in [18] for PID controller tuning. Our work differs significantly from this prior work, however, in two respects: (i) we tune PI-ESC control parameters in the outer-loop and not the inner-loop PID controllers, out of the practical concern that, post-production, inner-loop VCS controller gains are usually not easy to alter; and, (ii) prior work has not explicitly considered potential instabilities that arise during the tuning process: our proposed failure-robust BO (FRBO) algorithm [24] is designed specifically to learn from unstable combinations of ESC parameters, and avoid them in future iterations. The application of FRBO for the tuning of PI-ESC significantly improves the implementation potential of ESC for a large class of systems. While earlier studies [9, 11, 13] were primarily focused on the theoretical analysis of PI-ESC. The present study overcomes the practical limitations arising from the tuning of PI-ESC and provides optimally tuned ESC-based systems for general classes of unknown discrete-time nonlinear dynamical systems.

We are also aware of the 'safe' BO literature that has been applied to robotics and battery charging problems, c.f. [25, 26]. While these applications can make use of constrained BO formulations, these algorithms cannot be used to solve the problem considered in our work without major modifications. Specifically, the authors in [25, 26] assume that evaluating the constraint function will yield continuous values, which can subsequently be approximated by a kernelized regressor like a Gaussian process. This is not the case in the FR-BO setting, because our constraint function is an indicator function that yields discontinuous

values '+1' or '-1' depending on whether the final PI-ESC design is stable or not.

The first **contribution** of this work is a failure-robust Bayesian optimization algorithm that determines ESC controller parameters which render the ESC outer loop stable. The FRBO algorithms automatically learns combinations of ESC parameters that are likely to result in unstable closed-loop trajectories, and leverages probabilistic machine learning to learn and avoid these regions in the ESC parameter space. Second, by sampling on the admissible set of initial conditions of the control system, FRBO is also designed to seek ESC parameters that result in closed-loop stable trajectories from a wide range of initial conditions, which is a significant advantage over manual tuning efforts which typically yield ESC parameters that perform well for a few initial conditions, often just one initial condition. Such initial condition-specific manual tuning often results in ESC parameters that do not generalize well to other initial conditions, as we demonstrate in simulation. A final contribution is the validation of the proposed FRBO approach in the tuning of actuator reference inputs for optimizing energy consumption of production-level vapor compression systems. We report that hand-tuning yielded unstable PI-ESC trajectories despite tens of attempts, and while using BO to find good ESC parameters was somewhat successful, FRBO outperforms the best solution found by BO by a significant margin, yielding a coefficient of performance (CoP) over 6.5 compared to BO's CoP of below 5.5.

We review the PI-ESC formulation in Section 2 and highlight the parameters to be tuned. Section 3 formulates the Bayesian optimization gain tuning problems and explains how FRBO can improve the convergence rate of performance-driven PI-ESC parameter tuning. We validate on a benchmark 2D problem as well as a real-world application for optimizing energy in a production-level vapor compression system in Section 4.

2. Proportional-Integral Extremum Seeking Control

This section outlines the development of an extremum seeking controller based on a time-varying estimate of the gradient of the cost and a PI control law to drive the system to its optimum operating point; c.f. [13] for the full development and stability and convergence analysis in discrete time.

2.1. PI-ESC Development

The problem of energy minimization of the VCS described in the previous section can be abstracted as follows. The underlying VCS dynamical system can be modeled by the closed-loop system

$$x_{t+1} = f(x_t, r_t) \tag{1a}$$
$$y_t = h(x_t, r_t), \tag{1b}$$

where t is the time index, $x_t \in \mathbb{R}^{n_x}$ is the vector of state variables at time t, r_t is the setpoint variable at time t taking values in $\mathcal{R} \subset \mathbb{R}^{n_r}$ and $y_t \in \mathbb{R}_{\geq 0}$ is a scalar power output of the VCS at time t; our objective is to minimize y. We assume that the dynamics f and the setpoint-to-energy function h are both unmodeled (therefore, unknown at design time). We do assume, however, that h exhibits sufficient smoothness to warrant the use of data-driven gradient estimates for energy optimization. In practice, the energy function h is usually smooth, and often, locally strongly convex; c.f. [11]. More formally, there exists some scalar $\beta > 0$ such that $\nabla^2 h(r) > \beta I$ for any $r \in \mathcal{R}' \subset \mathcal{R}$. Furthermore, there exists some unique $r^\star \in \mathcal{R}'$ such that $\nabla h(r^\star) = 0$.

The principle of ESC is based on obtaining a sequence of setpoints r_t for $t \geq 0$ such that each r_t moves along a direction of negative gradient of the function h. That is, a first-order ESC control law has the form $r_{t+1} = -k_g g_t + d_t$, where g_t is an estimate of the gradient of h w.r.t. r, k_g is a control gain or step-size, and d_t is a persistently exciting dither signal [12]. Clearly, by the convexity properties of h, this control law will asymptotically yield a solution in the neighborhood of a local minimum, where the size of the neighborhood depends on the magnitude of d; this has been proven in [27]. By incorporating integral action to the afore-mentioned control law, its convergence rate has been significantly accelerated, as demonstrated in [13]. The proportional-integral ESC (PI-ESC) law is given, in velocity form, by

$$r_{t+1} = r_t - k_g(g_{t+1} - g_t) - \frac{1}{\tau_I} g_t + d_t, \tag{2}$$

where τ_I is a time constant of the integral term. The question that remains is how to obtain the gradient estimate g_t and its update g_{t+1} despite the unmodeled dynamics and cost.

To this end, we exploit the smoothness of h to write a Taylor series approximation of the difference in cost values, ignoring second- and higher-order terms, as

$$\Delta y_t := y_{t+1} - y_t = \frac{\partial h}{\partial x} \Delta x_t + \frac{\partial h}{\partial r} \Delta r_t. \tag{3}$$

Since f and h are unmodeled functions, we cannot obtain a numerical or analytical derivative of $\partial h/\partial x$ or $\partial h/\partial r$. Instead, we estimate the unknown quantities in (3) directly from data $\{(r_t, y_t)\}$ obtained during experiments with the closed-loop system. Concretely, we can rewrite (3) as

$$\Delta y_t = g_t^x + g_t^r \Delta r_t. \tag{4}$$

One can then use the data to formulate a linear regression problem that involves solving the matrix equation

$$\begin{bmatrix} \Delta y_{t-N_\ell+1} \\ \vdots \\ \Delta y_t \end{bmatrix} = \begin{bmatrix} 1 & \Delta r_{t-N_\ell+1}^\top \\ \vdots & \vdots \\ 1 & \Delta r_t^\top \end{bmatrix} \underbrace{\begin{bmatrix} g_t^x \\ g_t^r \end{bmatrix}}_{g_t},$$

from which estimates of the gradient g_t can be obtained efficiently, for example, by using recursive least-squares (RLS) estimators.

2.2. Data-Driven Gradient Estimator for PI-ESC

We denote $\hat{g}_t^r$ and $\hat{g}_t^x$ as the estimates of g_t^x and g_t^r, respectively, and denote $\hat{g}_t$ to be the estimate of g_t. We use the following RLS estimator at time $t-1$ to estimate the gradient at time t:

$$w_t = w_{t-1} - Fw_{t-1} + \phi_{t-1}, \tag{5a}$$

$$P_t = \alpha P_{t-1} + w_{t-1}w_{t-1}^\top + \epsilon I, \tag{5b}$$

$$e_t = y_t - \hat{y}_t, \tag{5c}$$

$$K_t = \frac{P_t^{-1}w_{t-1}}{\alpha + w_{t-1}^\top P_t^{-1} w_{t-1}}, \tag{5d}$$

$$\hat{g}_t = \hat{g}_{t-1} + K_t e_t. \tag{5e}$$

Here, $\phi_{t-1} \triangleq \begin{bmatrix}1 & r_{t-1}\end{bmatrix}$, ϵ is a small scalar-valued term that seeks to ensure good numerical conditioning of P_t, and F is a scalar filter gain coefficient. Subsequently, the predictor

$$\hat{y}_{t+1} = \hat{y}_t + Fe_t + \phi_t^\top \hat{g}_{t-1} + w_t^\top(\hat{g}_t - \hat{g}_{t-1}).$$

is used to estimate the cost $\hat{y}_t$ at time t based on the estimated gradient (5e).

As long as the system is persistently excited, that is, there exist constants $\beta_T > 0$ and $T \in \mathbb{N}$, such that

$$\frac{1}{T}\sum_{t=t_0}^{t_0+T-1} w_t w_t^\top \geq \beta_T I, \ \ \forall t > T, \tag{6}$$

we know from [13, Theorem 4.1] that the cost associated with closed-loop system with the gradient estimator (5) and control law (2) asymptotically enters a neighborhood of the optimal cost. In order to maintain the persistence of excitation condition (6), the dither signal is chosen to be non-zero and small at all time. To this end, we employ a dither signal of the form

$$d_t = D\sin(\omega t) + \varphi_0,$$

where ω is a vector of unique frequencies of the sinusoidal dither, φ_0 is a vector of unique initial phases of the sinusoidal dither, and D is a user-defined small amplitude.

We deduce that the following quantities parameterize a PI-ESC: (i) the integral time constant τ_I, (ii) the control gain k_g, (iii) the forgetting factor α, (iv) the filter coefficient F, and (v–vii) the dither variables D, ω, and φ. Unfortunately, the question of how to select these variables remains unclear. Empirically, we have observed that the PI-ESC performance is strongly correlated with the selection of each of these variables. In fact, if these variables are not carefully selected, the PI-ESC can result in numerical instability and the closed-loop system can exhibit unstable dynamics. An additional complication is that seemingly arbitrary combinations of the parameters (i)–(vii) can render the closed-loop unstable; indeed, we have noticed that there is no easily identifiable region in the space of these PI-ESC parameters where any combination of parameters results in stable behavior. Identifying such a region is problem-specific, and therefore, using problem-specific data provides a suitable option to attack this problem. Consequently, we propose a performance-(data)-driven Bayesian optimization algorithm for automatically identifying regions of safe combinations of PI-ESC parameters, and search efficiently within such safe regions to tuning PI-ESC algorithms. We demonstrate that our failure-robust BO (FRBO) framework outperforms hand-tuning in both benchmark and real-world examples, while offering a systematic tuning procedure that can be integrated seamlessly to a wide range of tuning problems.

3. Performance-Driven Controller Tuning with Failure Robustness

3.1. Performance-Driven Bayesian Optimization

In this section, we describe how to use Bayesian optimization (BO) for tuning the set of parameters listed in the previous paragraph, so that the PI-ESC algorithm demonstrates good closed-loop performance. We do so by iteratively performing experiments with different PI-ESC parameters, and for each experiment, we assign a value to the closed-loop performance (e.g. integral-time absolute error) to be optimized. Tuples of PI-ESC parameters and their corresponding performance values are subsequently optimized by FRBO.

For the ensuing discussion, we will search for the following PI-ESC parameters:

$$\theta \triangleq \begin{bmatrix}\tau_I & k_g & \alpha & F & D^\top & \omega^\top & \varphi^\top\end{bmatrix}^\top,$$

where the dimension of $\theta = 1+n_r+1+1+n_r+n_r+n_r = 3+4n_r$. We assume that the set of admissible parameters, which forms the search space for BO, Θ is known to the designer. Clearly, even for small n_r, the search space is not low-dimensional enough to justify trying to tune the PI-ESC parameters manually. In fact, for VCS applications, we have observed that attempts made to tune by hand often results in catastrophic failures i.e. unstable closed-loop dynamics or frequent triggering of failsafe mechanisms due to impractically large PI-ESC control actions.

For the j-th BO iteration, let θ_j denote the candidate set of PI-ESC parameters. With these parameters, an experiment is performed, that is, the PI-ESC closed-loop system parameterized by θ_j is observed on a time interval of interest $[\mathcal{T}_0, \mathcal{T}_f]$, and measurements

$$y_{\mathcal{T}_0:\mathcal{T}_f} := \{y_{\mathcal{T}_0}, y_{\mathcal{T}_0+1}, \cdots, y_{\mathcal{T}_f}\}$$

obtained from the system over this time interval are used to compute a performance value J_j. By learning a probabilistic surrogate model from θ to J and exploiting the

statistics of the learned surrogate model, BO generates a sequence of θ candidates that converge[2] to the optimal (in the sense of the performance metric chosen) PI-ESC parameters $\theta^\star$.

For performance-driven BO, the objective function (to be minimized) we have found to be useful has the form

$$J(x) = \eta_1 J_{\text{avg}}(x) + \eta_2 J_{\text{osc}}(x),$$

where x is an initial condition of the system and η_1 and η_2 are positive weights on each component. Here, J_{avg} is designed to filter out the dithering effect in steady state and promote lower steady state J values. This component of the cost is obtained by computing the mean of the final $\mathcal{T}'$ cost values, that is

$$J_{\text{avg}} \triangleq \mathbb{E}[J_{\mathcal{T}_f - \mathcal{T}':\mathcal{T}_f}],$$

where $\mathbb{E}$ is the expectation operator. The second component of the cost

$$J_{\text{osc}} \triangleq \frac{1}{(\mathcal{T}_f - \mathcal{T}_0)} \sum_{t=\mathcal{T}_0+1}^{\mathcal{T}_f} |J_t|$$

is designed to penalize oscillations in the closed-loop PI-ESC response. Since we aim to evaluate performance over multiple initial conditions within an admissible set of initial conditions, $\mathbb{X}$, we repeat these simulations for a set of $N_x \in \mathbb{N}$ initial conditions $\{x_i\}_{i=1}^{N_x}$ and compute the total cost over this set. That is, the cost assigned to the parameter θ_j is given by

$$J_j = \sum_{i=1}^{N_x} J_{\text{avg}}(x_i) + J_{\text{osc}}(x_i), \tag{7}$$

where the two components are defined above.

Classical BO methods assume the presence of a single global optimum, and smoothness of the θ to J map. Since J is typically assumed to be continuous, one can leverage the data at the j-th iteration to construct a surrogate GP model of the cost, given by

$$\hat{J}_j := \mathsf{GP}\left(\mu(\theta; \mathcal{D}_j), \sigma(\theta, \theta'; \mathcal{D}_j)\right), \tag{8}$$

where $\mu(\cdot)$ is the predictive mean function, and $\sigma(\cdot,\cdot)$ is the predictive variance function, and $\mathcal{D}_j$ containing $\{\theta_{[0:j]}, J_{[0:j]}\}$ is the dataset collected thus far. Typically, the variance is expressed through the use of kernel functions [29].

At the j-th learning iteration, for a new query sample $\theta \in \Theta$, the GP model predicts the mean and variance of the reward to be

$$\mu(\theta) = k_j(\theta)^\top \mathfrak{K}_j^{-1} J_{0:j}$$

and

$$\sigma(\theta) = \mathcal{K}(\theta,\theta) - k_j(\theta)\mathfrak{K}_j^{-1} k_j(\theta)^\top,$$

where

$$k_j(\theta) = \begin{bmatrix}\mathcal{K}(\theta_0,\theta) & \mathcal{K}(\theta_1,\theta) & \cdots & \mathcal{K}(\theta_j,\theta)\end{bmatrix},$$

and $\mathfrak{K}_j$ is given by

$$\mathfrak{K}_j = \begin{bmatrix}\mathcal{K}(\theta_0,\theta_0) & \cdots & \mathcal{K}(\theta_0,\theta_j) \\ \vdots & \ddots & \vdots \\ \mathcal{K}(\theta_j,\theta_0) & \cdots & \mathcal{K}(\theta_j,\theta_j)\end{bmatrix}, \tag{9}$$

The accuracy of predicted mean and variance is strongly linked to the selection of the kernel and the best (in some sense) set of hyperparameters such as length scales and variance parameters of the kernels and estimated noise. We obtain these hyperparameters by maximizing the log marginal likelihood function (MLL)

$$-\frac{1}{2}\log|\mathfrak{K}_j| - \frac{1}{2}J(\theta)^\top \mathfrak{K}_j^{-1} J(\theta) - \frac{n_\theta}{2}\log 2\pi.$$

This problem is non-convex but can be solved using quasi-Newton methods or adaptive gradient methods [30].

In Bayesian optimization, we use the mean and variance of the surrogate model $\hat{J}_j$ in (8) to construct an acquisition function to inform the selection of a θ_j that increases the likelihood of minimizing the current best cost. To this end, we compute the incumbent $\hat{J}_j^\star := \min_{\theta\in\Theta} \mu(\theta; \mathcal{D}_j)$ and use it to define an expected improvement (EI) acquisition function that has the form

$$\mathsf{EI}(\theta, j) = \begin{cases}\sigma(\theta)\gamma(z) + (\hat{J}_j^\star - \mu(\theta))\Gamma(z), & \text{if } \sigma(\theta) > 0, \\ 0 & \text{if } \sigma(\theta) = 0.\end{cases}$$

where $z = \frac{\hat{J}_j^\star - \mu(\theta)}{\sigma(\theta)}$, and $\gamma(\cdot)$, $\Gamma(\cdot)$ are the PDF and the CDF of the zero-mean unit-variance normal distribution, respectively.

In the j-th iteration of learning, we use the data $\mathcal{D}_j$ to construct the EI acquisition function using the surrogate $\hat{J}_j$. Subsequently, we compute the optimizer candidate

$$\theta_{j+1} = \arg\max_{\theta\in\Theta} \mathsf{EI}(\theta, j), \tag{10}$$

which serves as the parameter estimate θ in the next BO iteration.

Remark 1. *Other acquisition functions such as the lower confidence bound or knowledge gradients could also be used instead of EI [29].*

3.2. Failure Robust Bayesian Optimization

While one could, in theory, apply classical Bayesian optimization directly in order to find an optimal set of PI-ESC parameters $\theta^\star$, we have (empirically) found that this approach has severe limitations. Frequently, the classical BO procedure tends to select θ candidates that make the PI-ESC closed-loop system unstable. These instabilities culminate in simulation failures, that is, the closed-loop system states attain arbitrarily large magnitudes and the

[2] Typically, the convergence proofs are for simple, or cumulative, regret; c.f. [28].

Algorithm 1 PI-ESC Tuning with FRBO

Require: Initial and total FRBO iterations, N_0, N_∞
Require: Admissible initial conditions, $\mathbb{X}$
Require: Search space for PI-ESC parameters, Θ
Require: Cost function surrogate GP model, GP
Require: Failure region learning algorithm, $\mathcal{F}$
Require: Time interval of experiment, $[\mathcal{T}_0, \mathcal{T}_f]$
Require: Frequency of retraining $\mathcal{F}$, $N_{\mathcal{F}}$
Generate fixed set of initial conditions, $\mathbb{X}_{N_x}$

Generate initial dataset $\{\theta_t, \ell_t, J_t\}_{t=0}^{N_0}$

for $t = 1$ to N_0 **do**
  Randomly select $\theta_t \in \Theta$
  procedure CollectData(θ_t)
    Perform experiment with PI-ESC parameter θ_t for all initial conditions in $\mathbb{X}_{N_x}$
    if experiment fails **then**
      Label failure, $\ell_t = +1$
      Assign failure cost, $J_t = \texttt{NaN}$
    else
      Label success, $\ell_t = -1$
      Compute cost, J_t using (7)
    end if
  end procedure
end for

Run FRBO

Train $\mathcal{F}$ with $\{\theta_t, \ell_t\}_{t=0}^{N_0}$
Train GP with $\{\theta_t, J_t\}$ for real-valued costs only
for $t = N_0$ to N_∞ **do**
  Compute candidate using FREI (12), θ_{t+1}
  Collect data, $\ell_{t+1}, J_{t+1} \leftarrow$ CollectData(θ_{t+1})
  Append dataset, $\{\theta_t, \ell_t, J_t\}_{t=0}^{N_0+t+1}$
  Train GP with $\{\theta_t, J_t\}$ for real-valued costs so far
  if $\texttt{modulo}(t, N_{\mathcal{F}}) = 0$ **then** ▷ Infrequent retraining
    Train $\mathcal{F}$ with all $\{\theta_t, \ell_t\}$ data so far
  end if
end for
Best cost index, $t^\star = \arg\min_t \{J_t\}$
Return Optimal PI-ESC parameters, $\theta_{t^\star}$

experiments trigger failsafe mechanisms before the termination time $\mathcal{T}_f$. To reiterate, the failures described are not only numerical intractabilities in simulation experiments, but originate from a common physical phenomenon that has been observed experimentally. Poor combinations of PI-ESC gains will destabilize the operation of the VCS, causing loss of cooling capacity. In hardware, this triggers failsafe operation, or in simulated systems this results in non-physical states that violate component model assumptions. Additionally, the failed simulations result in arbitrarily large, and often, nonsensical e.g. `NaN` cost values that render the (θ, J) datapoint useless for consequent BO iterations. Since seemingly arbitrary combinations of the components of θ result in these instabilities, classical BO wastes a lot of iterations evaluating candidates that cause instability, resulting in very slow convergence rates.

Rather than employing heuristics to avoid the regions in parameter space that are likely to result in simulation failures, we adopt a data-driven approach to estimate these regions and avoid them by means of a modified acquisition function tailored to promote 'failure robustness'. We refer to this algorithm as failure-robust BO (FRBO) and provide a few key details about the algorithm herein. A more detailed description is available in [24].

The first step of the FRBO algorithm involves constructing a dataset for failure region estimation, which we will pose as a supervised learning problem. We begin by randomly sampling a set of initial conditions $\mathbb{X}_{N_x} := \{x_i\}_{i=1}^{N_x}$ within the space of admissible initial conditions $\mathbb{X}$; this set is then kept fixed. At the j-th FRBO iteration, we simulate the PI-ESC closed-loop system on $[\mathcal{T}_0, \mathcal{T}_f]$ where the PI-ESC is parameterized by the candidate parameter θ_j. If the simulation fails, we assign a label $\ell_j = 1$ and store a nonsense value in the cost $J_j = \texttt{NaN}$. If the simulation is successful, we assign a label $\ell_j = -1$ and store the real-valued cost J_j described in (7). Thus, at the end of the j-th FRBO iteration, we have a dataset $\{(\theta_k, \ell_k, J_k)\}_{k=0}^{j}$.

We can use the (θ, ℓ) components of this dataset to estimate the failure region boundary by casting the estimation problem as a supervised learning problem. In particular, we need to construct a probabilistic learning machine $\mathcal{F} : \Theta \to [0, 1]$, where the output of the learner is the probability that $\theta \in \Theta$ is inside the failure region. That is, $\mathcal{F}(\theta)$ is the learned probability that the closed-loop PI-ESC parameterized by θ will be unstable (and therefore, fail). A few considerations go into the selection of a learning algorithm suited for the task of failure region estimation. First, $\mathcal{F}$ should be able to generate meaningful estimates of the failure region despite being trained on limited (θ, ℓ) data, since the FRBO algorithm is designed to converge without requiring a large number of iteration. Second, $\mathcal{F}$ needs to be retrained often, so a learner that requires a large number of training iterations before yielding accurate predictions is not ideal for FRBO. Third, $\mathcal{F}$ needs to be able to generate decision boundaries exhibiting complex geometries, since the failure region may have irregular contours. One or more of these pre-requisites restrict the utility of deep neural networks (needs large datasets) or linear classifiers (cannot generate nonlinear decision boundaries). Instead, we have found that a nonparametric kernel-based probabilistic classifier such as a variational Gaussian process classifier (VGPCs) [24, 31] works well in practice for failure region estimation in the context of FRBO. Over comparable classifiers like a probabilistic support vector machine, the VGPC has the advantage that its outputs—a mean and variance—is easily interpretable since the variational proxy distribution is taken to be Gaussian.

Once the failure region estimator is trained, the FRBO algorithm utilizes a failure-robust expected-improvement

(FREI) acquisition function that has the form

$$\mathsf{FREI}(\theta, j) = \mathsf{EI}(\theta, j) \cdot (1 - \mathsf{PF}(\theta, j)), \tag{11}$$

where $\mathsf{PF}(\theta, j) = \mathcal{F}(\theta_j)$ is the probability of failure calculated by training $\mathcal{F}$ using (θ, ℓ) data up to the j-th iteration, and EI is described in (10). Note that the GP surrogate of the cost (8) is trained on (θ, J) data which resulted in stable closed-loop trajectories, that is, excluding those data points where $\ell_j = +1$. Upon maximizing this acquisition function, FR-BO selects the next optimizer candidate

$$\theta_{j+1} = \arg\max_{\theta \in \Theta} \ \mathsf{FREI}(\theta, j). \tag{12}$$

This maximization problem is often solved in low dimensions by sampling on Θ, evaluating the acquisition function on those samples, and choosing the maximizer on that finite set of samples. In such an approach, the samples on Θ are selected randomly at each FRBO iteration to encourage exploration.

We note that maximizing FREI indicates that the component EI should be large, and that the component PF should be small, i.e., near zero. The former indicates that the considered θ is likely to minimize the cost function (7), and the latter implies that such a θ is likely to result in a successful simulation. By combining both these beneficial qualities into the selection of the next candidate θ, FRBO automatically increases the likelihood of choosing θ values that do not result in instabilities of the closed-loop system, while remaining likely to optimize the cost function. A full pseudocode is provided in Algorithm (1).

3.3. A heuristic for retraining $\mathcal{F}$

It is not necessary to retrain $\mathcal{F}$ at every FRBO iteration. In fact, we have found that a combination of active learning and retraining infrequently results in significant speedup of the FRBO algorithm. Active learning involves selecting θ values that the learner is most uncertain about, and therefore these θ values are most useful for determining the failure region boundary. If $\mathcal{F}$ is a VGPC, then the most informative sample obtained by active learning is

$$\theta_j^\star = \arg\max_{\theta \in \Theta} \ \frac{1}{2} \log\left(2\pi \, \mathsf{Var}(\theta)\right), \tag{13}$$

which is the sample that exhibits highest variance (i.e. uncertainty). The active learning step does not directly improve convergence speed of FRBO as the most informative sample for determining the failure region is unlikely to be a sample that optimizes the cost function. However, a few actively learned samples can quickly lead to a good estimate of the failure region, which consequently reduces the amount of time wasted evaluating parameters that result in instability. One effective heuristic for every 100 FRBO iterations is to: (i) generate 80–90 θ candidates via the FREI acquisition function (12); (ii) generate the rest of the θ candidates via active learning (13); (iii) retrain the surrogate cost GP at every iteration, and (iv) retrain $\mathcal{F}$ once at the end of the 100 iterations.

4. Simulation Results

4.1. Nonlinear System

To demonstrate the effect of the proposed method on a multi-variable tuning problem, we consider the dynamical system

$$\begin{aligned} x_{1,t+1} &= x_{3,t}^2 + r_{1,t}, \\ x_{2,t+1} &= x_{2,t} + r_{1,t}, \\ x_{3,t+1} &= 2x_{3,t}\big(r_{1,t} + x_{1,t}x_{2,t}r_{2,t}\big), \\ y_t &= \frac{1}{2} x_t^\top x_t \end{aligned}$$

studied in [13, 32]. The optimal solution of this system is $y^\star = 0$ at $r^* = 0$, but the quadratic terms in both the state dynamics and cost function imply that without careful tuning of the PI-ESC algorithm, the state and cost can become arbitrarily large very quickly.

Since we do not have an accurate estimate of Θ, we search over a wide range of θ values, namely

$$\Theta = [1, 10^3] \times [0.01, 10] \times [0, 1] \times [0, 1] \times [0.01, 10]^2 \times [1, 10^3]^2.$$

Since $n_r = 2$ and we force k_g to be a scalar, so θ is 8-dimensional, since φ is fixed to be zero. We initially construct a dataset with $N_0 = 100$ samples, and our set of initial conditions is extracted from $\mathbb{X} = [-1.5, 1.5]^3$ using $N_x = 200$ low-discrepancy points in order to attain good coverage over the entire space; see [33, 34] for more details about low-discrepancy sequences and their advantages.

Once this initial sampling is complete, we train a VGPC and a GP regressor to obtain an estimate of the failure region and surrogate cost, respectively. Our specific implementation of the failure region classifier uses a VGPC with a squared exponential kernel, 200 inducing points, a Bernoulli likelihood function, and a variational ELBO loss function. The variational ELBO is optimized with an Adam optimizer for 2000 iterations initially (before FRBO is run) and then 500 iterations every $N_{\mathcal{F}} = 100$ FRBO iterations; the learning rate is fixed at 0.01 initially, and 0.005 thereafter. For the GP regressor, we use a Matern-3/2 kernel with Gaussian likelihood, a marginal log likelihood loss function that is maximized using stochastic gradient descent (SGD) with a learning rate of 0.05 throughout the FRBO algorithm, for 500 training iterations. While hyperparameter optimization could prove useful in improving performance even further, we have noticed empirically that manual selection of these hyperparameters, especially kernels, training iterations, and learning rates, is possible, as the FRBO algorithm is not sensitive within reasonable ranges of these hyperparameters. The FRBO is run for 900 iterations, with 1000 samples drawn randomly at each iteration to maximize the FREI acquisition function on Θ. The closed-loop system is run for $\mathcal{T}_0 = 0$ to $\mathcal{T}_f = 5000$ time instants, and the first component of the cost (7) is evaluated on the final $\mathcal{T}' = 50$ time indices.

The optimal parameters obtained by the FR-BO algorithm are $\tau_I = 109.51$, $k_g = 0.6791$, $\alpha = 0.21$, $F = 0.13$,

$D = [0.10, 0.11]$, $\omega = [185.49, 181.11]$. Note that if we restrict the elements of D and ω to be equal, then the FR-BO finds $D = 0.11$ and $\omega = 185.59$, which yields identical results to the full 8-dimensional version. The closed-loop simulation results using the PI-ESC that is parametrized by $\theta^\star$ is shown in Figure 1 and Figure 2; furthermore, we compare our results with a PI-ESC whose gains have been reported in the prior art by carefully hand-tuning [13], which we refer to as 'Manual'. We also compare with BO employing a constrained expected improvement (ConEI) acquisition function [35], and a scalable trust-region approach called TuRBO that has recently demonstrated excellent performance on constrained global optimization problems [36]. Both the ConEI and TuRBO algorithms were implemented with hyperparameters as close to FRBO's hyperparameters as much as possible, e.g. kernel type, method of optimizing the acquisition function. However, we could not make the implementations completely identical in terms of algorithm-specific hyperparameters, so for fairness, the default values in the `BoTorch` library [37] were used. The comparison of the states and outputs is made over 1000 runs, where in each run, we randomly select initial conditions x_0 of the system within the box $[-1.5, 1.5]^3$. The median and 95% confidence intervals are shown for the FRBO tuned PI-ESC using blue, and the hand-tuned PI-ESC using orange. We observe from Figure 1 that the FRBO tuning results in greatly improved closed-loop performance for two reasons. First, the output y of the system always converges to a small neighborhood of the optimum (the origin), and mostly does so within 50 time steps. Conversely, the gains reported by hand-tuning in the prior art converges for 757 out of 1000 initial states, and on those cases, the output decays to a larger neighborhood of the optima more slowly, in 250 time steps. Second, from the wide confidence intervals, we deduce that the closed-loop output response of the hand-tuned PI-ESC is much more oscillatory than the FRBO-tuned PI-ESC. This indicates that the FRBO version is less sensitive to initial condition variation, which is expected since the gains are tuned with respect to a finite, but wide, range of initial conditions in our proposed approach. While the state-of-the-art constrained BO algorithms ConEI and TuRBO perform better than the hand-tuned PI-ESC, it is evident from the figure that ConEI and FRBO achieve the lowest y values, with the median performance of FRBO outperforming ConEI. It is also noteworthy that the convergence rate of the FRBO tuning is faster than the others. These findings are corroborated in Figure 2, which illustrates the 2-norm of the states x and setpoints r. Clearly, the states and setpoints of the FRBO tuned PI-ESC converge significantly faster, and to tighter neighborhood than their hand-tuned counterpart.

The overall FRBO training performance is also compared against classical BO (with an EI acquisition function) in Figure 3. Both algorithms are run with the same initial dataset, and the costs J are scaled to lie on the unit line. The red vertical lines indicate iterations where the

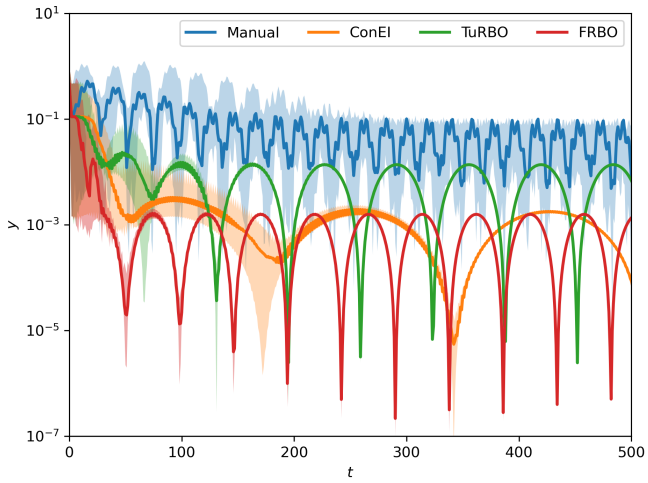

Figure 1: Comparison of output optimized by the PI-ESC tuned by FRBO and the other competitor algorithms for the 2D system, over 1000 runs. Note that the vertical axis is in log-scale.

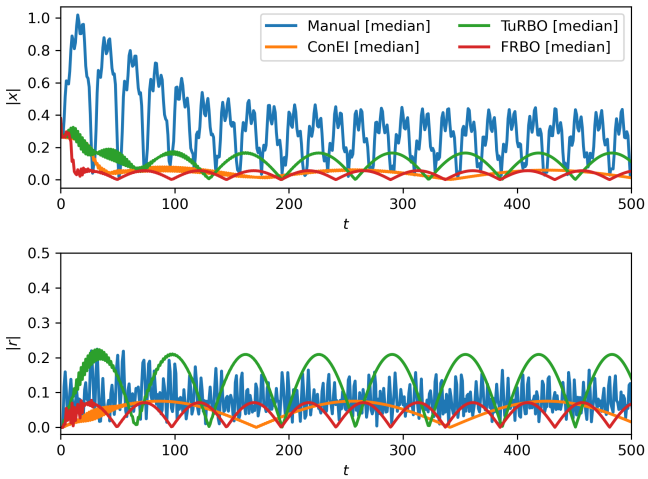

Figure 2: Comparison of closed-loop states and inputs of PI-ESC tuned by FRBO and the other competitor algorithms over 1000 runs.

θ candidate resulted in a failed simulation. The FRBO algorithm (left subplot), due to repeated learning of the failure region, shows a gradual reduction of the number of failed simulations as the number of FRBO iterations increase. After 750 iterations, there are no further failures because the BO algorithm searches in a subregion that is not failure-prone, as directed by the failure classifier $\mathcal{F}$. Contrarily, the right subplot shows that for classical BO, since there is no estimate of the failure region, the frequency of failures occurring throughout the optimization is fairly constant, and since a lot of those iterations are wasted from an optimization perspective, there are fewer improvements in the cost function J. Consequently, the final cost value for the FRBO algorithm is much better (0.03) compared to classical BO (0.42), for the same number of iterations. This demonstrates the potential of

FRBO on practical problems where the cost cannot always be evaluated and failures occur.

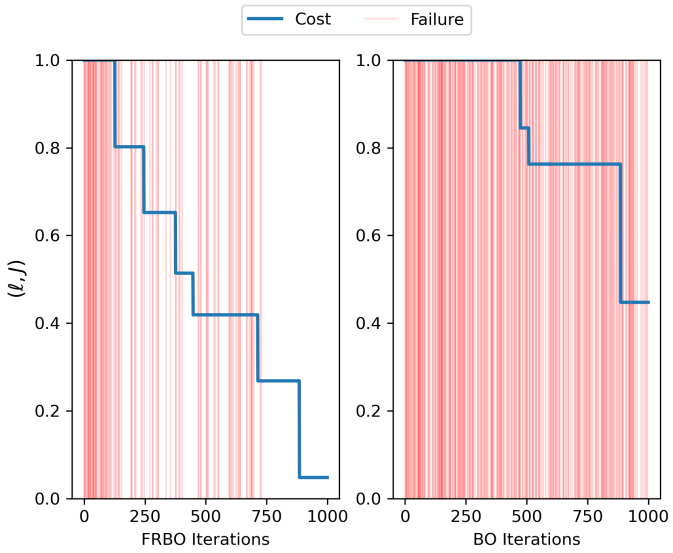

Figure 3: Comparison of cost decay curves of the FR-BO and BO tuning procedures (costs have been scaled to $[0, 1]$ by the same scaling factor).

4.2. Case Study: Energy Optimization of Production-Level Heat Pump

4.2.1. The Energy Optimization Problem

We demonstrate the practical application of the FRBO algorithm for the parameter tuning of PI-ESC controllers to reduce the energy consumption of a vapor-compression cycle (VCS) under standard operating conditions. Contemporary vapor-compression cycles often have many variable actuators, such as compressor speed, expansion valve position, and fan speeds; since the number of actuators often exceeds the number of variables regulated by feedback controllers, the remainder of these actuators can be used to optimize the system performance according to a given metric. In this case study, we regulate the cooling capacity provided by the equipment to a user-provided setpoint by controlling the compressor speed, and seek to identify the values of the other actuators that minimize the energy consumption.

The architecture of a conventional variable-capacity vapor-compression cycle is provided in Figure 4-A. A variable speed compressor compresses low-pressure refrigerant vapor to a high pressure, at which the saturation temperature for the refrigerant is higher than the temperature of the air entering the heat exchanger. As the refrigerant leaves the compressor and passes through the tubes of the outdoor heat exchanger, it condenses into a liquid as thermal energy is transferred to the ambient air, which is driven by an outdoor fan as it passes over the tubes. This liquid refrigerant is then adiabatically expanded through a variable-orifice expansion valve (EEV) to a pressure at which the saturation temperature is below the temperature of the air entering the indoor heat exchanger from the room. A variable-speed indoor fan is used to circulate the air over this indoor heat exchanger. The difference in between the inlet air temperature and the refrigerant saturation temperature causes the refrigerant to absorb thermal energy from the room air through the process of evaporation. The refrigerant then exits the heat exchanger in a low-pressure vapor state, after which it returns to the compressor.

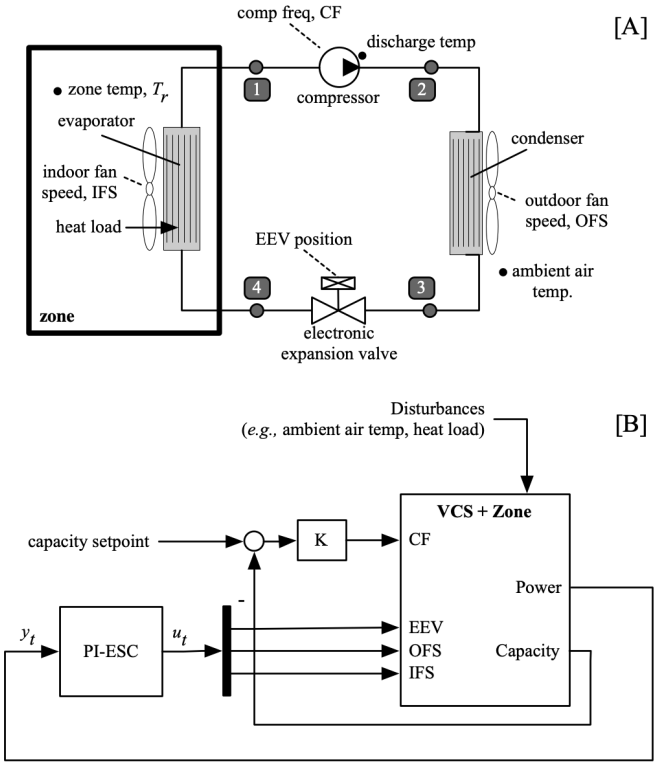

Figure 4: [A] Principal components of the vapor compression cycle. [B] Control architecture under study.

4.2.2. Experimental Setup

A high-fidelity model for the dynamics of a prototype VCS with the above architecture was developed in the Modelica language [38]. Equation-based models of the compressor, heat exchangers, valves, and fans were constructed using an object-oriented approach and interconnected to form a complete cycle model. Geometric and empirical parameters for these components, such as heat exchanger geometries, heat transfer coefficients, or performance curves, were obtained from experimental data and adapted to this use case for the purposes of generalization. These models were formulated using the governing physics of the system, and implemented as a set of differential algebraic equations (DAEs) that describes a spatial discretization of the conservation equations for mass, momentum, and energy. The dynamic behavior of the system, represented by f in (1a), is dominated by that of the

heat exchangers, as the transients moving mass between these components are much slower than the other time constants of the system. Accordingly, nonlinear algebraic models of the compressor and expansion valve were used to manage the size of the set of differential equations being solved. The heat exchanger models were discretized along the direction of refrigerant flow using finite volume methods, while the three-dimensional variations on the air-side of the heat exchangers were represented by a complex set of boundary conditions that characterized the conjugate heat and mass transfer processes essential to the system operation. 27 volumes were used in the model of the condensing heat exchanger to its long pipe length, while 16 volumes were used for the evaporating heat exchanger. A Helmholtz fundamental equation-of-state models was used to describe the algebraic constraints between the myriad refrigerant properties (e.g., pressure, temperature, density, specific enthalpy), while an ideal gas mixture model was used to describe the air properties. A nonlinear algebraic model was also used for y_t in the cost (1b). Compilation of these models and their interconnections resulted in a total cycle model that consists of 12,114 state equations. More information on the structure of this model and its components can be found in [39].

Because multiple combinations of actuator positions produce the same cooling capacity but differing values of electrical power consumption [40], we configured the vapor-compression cycle for realtime optimization via extremum seeking control to identify actuator values that minimize the energy consumption as shown in Figure 4-B. The value of the compressor frequency is computed by a proportional-integral (PI) controller acting on the difference between a setpoint of 2 kW and the measured value of the cooling capacity; the PI controller gains are designed and fixed offline and demonstrate good regulatory performance under regular operating conditions. This controller was also implemented in Modelica, as the vapor compression system and compressor feedback are thus treated as the optimization target for a proportional-integral extremum seeking controller (PI-ESC) [11]. The PI-ESC algorithm assigns EEV position, outdoor fan speed (OFS) and indoor fan speed (IFS) such that a measurement of equipment power is minimized. Assuming that zone setpoint and system disturbances (heat load and ambient temperature) are held constant, the combination of EEV, IFS, and OFS values at steady state are energy-optimal.

The closed-loop Modelica system model was interfaced to the PI-ESC code using the Functional Mockup Interface (FMI) standard [41]. The Dymola [42] environment was used for the initial development of this model, which was then exported as a Functional Mockup Unit (FMU) containing executable simulation code as well as a DAE solver. An advantage of the FMU-based approach is that the original model can be designed in Modelica, which can efficiently solve large sets of stiff nonlinear differential equations and preserves physics-informed dynamics, while the PI-ESC and FRBO code could be written in Python to take advantage of existing machine learning tools.

For this example, $n_r = 3$ as there are three setpoints (EEV, IFS, and OFS) to be ascertained by the PI-ESC. Thus, θ is 12-dimensional, as the phase φ is fixed at zero. The search space Θ is identical to that in Example 1, with the exception that the final two intervals for D and ω are $[0.01, 10]^3$ and $[1, 10^3]^3$ because $n_r = 3$, and the search space for k_g is $[0.1, 10^2]^3$. The initial set of states is fixed for this system in order to start the experiment on the RAC at an equilibrium point, so all FRBO costs are calculated for the same initial condition. From this initial time $\mathcal{T}_0 = 0$, we simulate forward to $\mathcal{T}_f = 120$ min to allow the system to enter a 95% settling zone. Each run of 120 min simulation time requires a wall time of 10–15 min due to the large number of internal states in the Modelica model. We select $\mathcal{T}'$ to be the final 20 min and power measurements are assumed to be obtained every 60 s. We use the same hyperparameters for the VGPC failure classifier and GP regressor as in Example 1, except that both are trained for 2000 iterations with a learning rate of 0.01. We allow the FRBO to terminate after 500 iterations. At termination, the parameter set generated by FRBO is given by $\tau_I = 61.3$, $k_g = [4.15, 10.55, 25.42]$, $\alpha = 0.92$, $F = 0.99$, $D = [1.31, 2.15, 4.85]$, $\omega = [10.21, 13.55, 17.96]$.

Perf-Index	BO-Tuning	FRBO-Tuning
s.s. Power [kW]	0.37	0.30
95% Settling [hr]	4.42	1.01
IAE ($\int \lvert e\rvert \, \mathrm{d}\tau$)	1.79×10^3	0.89×10^3
ITAE ($\int \tau \lvert e\rvert \, \mathrm{d}\tau$)	3.40×10^3	0.49×10^3
ISE ($\int \lvert e\rvert^2 \, \mathrm{d}\tau$)	2.76×10^5	2.20×10^5
ITSE ($\int \tau \lvert e\rvert^2 \, \mathrm{d}\tau$)	3.09×10^5	0.99×10^5
RMSE (EEV)	1.18	1.31
RMSE (OFS)	2.59	2.03
RMSE (IFS)	5.41	4.59

Table 1: Performance indices reported in the comparative study. Here, $e :=$ (Power − Optimal Power), where the optimal power is calculated by exhaustive sampling of the setpoint space for the purposes of reporting these values. RMSE = root mean squared error.

4.2.3. Results and Discussion

In order to demonstrate the potential of the FRBO-tuned PI-ESC, we compare with a baseline, where the setpoints of EEV, IFS, and OFS are kept fixed at nominal values that have been chosen by a domain expert, and a PI-ESC whose parameters have been tuned by classical BO. Both classical BO and FRBO are allowed the same number of optimization iterations, and FRBO converges to a better solution within those iterations while BO wastes many iterations searching for candidate parameters that result in failure. We also tried to compare to a PI-ESC with hand-tuned parameters before acquiring the FRBO optimized parameters, but could not hand-tune parameters that did not result in closed-loop instabilities despite >30 attempts.

Figure 5 illustrates the comparison of closed-loop PI-ESC with FRBO tuning, BO tuning, and the baseline. We

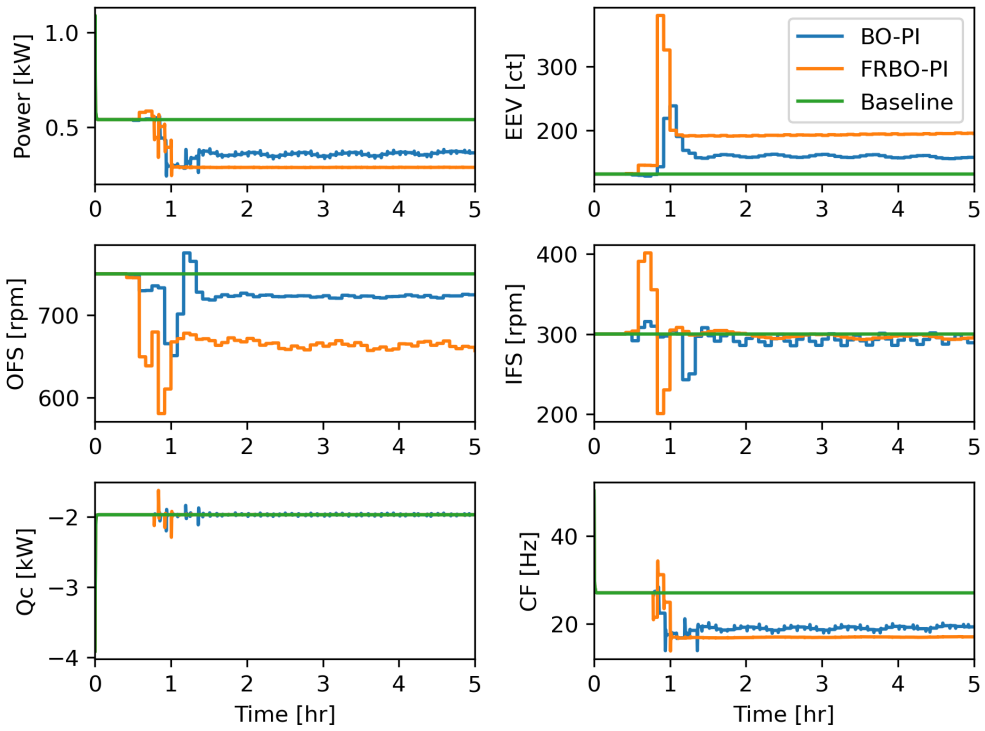

Figure 5: Comparison of closed-loop performance of PI-ESC and baseline for the multivariable space cooling application. (EEV = electronic expansion valve position; OFS = outdoor fan speed; IFS = indoor fan speed; Qc = capacity; CF = compressor frequency.)

also provide the numerical values of corresponding performance indices calculated from these simulations in Table 1. It is evident from the figure that the operating power is reduced significantly from the baseline power by the PI-ESC with FRBO tuning. In particular, the steady-state operating power of the FRBO-PI-ESC is 300 W, whereas the BO-PI-ESC power is oscillating around 370 W. This is further corroborated by calculating the coefficient of performance (CoP), defined as the instantaneous thermal capacity divided by electrical energy consumed by the system, in Figure 6. We observe that FRBO-PI-ESC converges to a higher COP compared to BO-PI-ESC (6.81 v. 5.44). Since the compressor frequency is directly under capacity control, the ESCs manipulate the fan speeds and valve position. The outdoor fan speed is reduced, the indoor fan speed largely settles near its initial setting, and the electronic expansion valve opens. The latter actuator in particular will reduce the pressure difference across the compressor, reducing its apparent load and enabling more capacity. Because the compressor is under capacity feedback control, the compressor speed is subsequently reduced, which accounts for the large increase in energy performance. Additionally, we checked by exhaustive sampling that the optimal power value is 285 W, which is much closer to that obtained by FRBO-PI-ESC than BO-PI-ESC, even though both algorithms were afforded the same number of optimization iterations, indicating the acceleration contributed by FRBO.

As we see from the settling time comparison in Table 1, the FRBO-PI-ESC actually converges to the optimal power value around 1 hour (both ESCs are switched on at the 1 hour mark after initial transient have disappeared), while the BO-PI-ESC has a strong oscillatory response, resulting in a far slower settling time of over 4 hours. In order to better quantify the power responses of the two ESC controllers, we provide some error metrics in Table 1, where the error e is computed by taking the difference from the measured power output and a theoretical minimum power obtained by exhaustive sampling. In particular, we see from the integral-time absolute (ITAE) and integral-time squared error (ITSE) that the FRBO-PI-ESC produces a power signal that converges quickly to the steady state value and remains there, whereas the larger ITSE/ITAE values for BO-PI-ESC implies that the signal exhibits sustained oscillations. Note that the capacity is consistently maintained at 2 kW despite setpoint changes in the ESC outer loop, except when the PI-ESC algorithms are first switched on around 0.5 hr.

The actuators manipulated by PI-ESC experience large amplitude swings initially, which is captured by the RMSE in each of the actuators in Table 1, where we note that the energy of the expansion valve and fan speed signals are higher for FRBO than BO. However, this excitation does not cause a loss in capacity in this case. If these swings were deemed to be too large for practical systems, they can be reduced with small adjustments to FRBO-PI-ESC parameters, or intercepted by VCS protection logic. Importantly, the improvement in convergence rate to about 1

hour overcomes important obstacles to wide-scale deployments of ESC in VCS applications. Typical disturbances acting on the system (see Fig. 4-B) are associated with building dynamics or diurnal weather patterns and therefore have timescales slower than 1 hour, implying that PI-ESC could be used to track optimal energy performance while rejecting disturbances in this frequency band, as previously reported in [11].

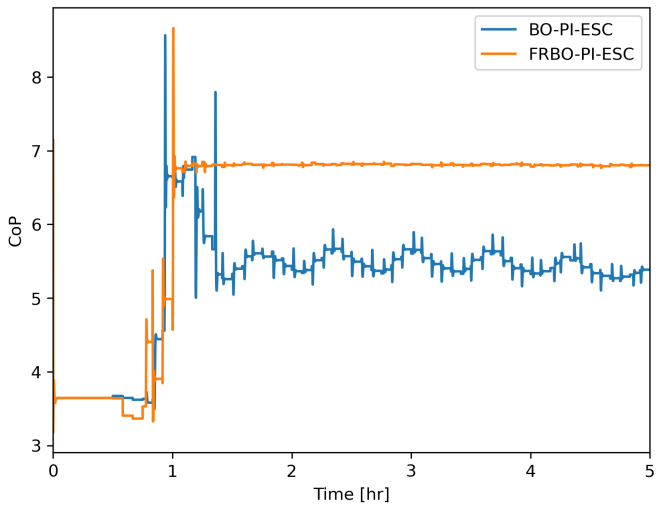

Figure 6: Comparison of coefficient of performance for PI-ESC with classical BO and FRBO tuning. Both classical BO and FRBO solutions have been obtained after allowing 1000 optimization iterations.

5. Conclusions and Future Work

In this study, we propose a failure-robust version of the Bayesian optimization (BO) algorithm to compute PI-ESC gains that ensure the closed-loop system can be driven to its optimum, despite lack of model knowledge. We focus on an estimation-based PI-ESC that has been proven to be effective for optimizing efficiency in vapor compression systems, and demonstrate that the FRBO algorithm can systematically tune PI-ESC parameters to outperform classical BO tuning. The proposed strategy was evaluated on a benchmark example, where we showed that the FRBO algorithms learns from unstable simulations and accelerates the tuning process compared to classical BO, even from a wide range of initial conditions. It was also tested on a real-world example using a Modelica model of a heat pump, where we reported that hand-tuning of PI-ESC is impractical, and a classical BO approach for tuning PI-ESC results in gains that are outperformed by FRBO tuning. In addition to discovering a more optimal operating point, PI-ESC converges in less time.

The tuning of ESC is a very important consideration in the application of ESC. ESC techniques are subject to difficulties associated with the tuning of the ESC system parameters. The tuning relies on properties of the system that cannot be directly assessed in practical situation. This is partially the issue faced by the application to VCS as reported in this study. The proposed BO technique overcomes these problems. It is a valuable tool in the broader context of model-free real-time optimization system design using ESC. Future work will be focussed on the application of the BO tuning approach over a wide range of competing ESC methodologies. This is particularly important in the application of fast ESC technique such as PI-ESC. Many different formulations of the PI-ESC approach have been proposed in the literature including a perturbation based and a Lie bracket approximation technique. While each technique has its own inherent limitations, the BO tuning approach can be used to provide a systematic comparison of competing approaches. It is planned to use the proposed technique to address tuning issues in ESC in general to provide a fair comparison of all existing approaches. This approach could provide some useful insights to aid in the development of general tuning strategies for ESC techniques in a general setting.

References

[1] X. She, L. Cong, B. Nie, G. Leng, H. Peng, Y. Chen, X. Zhang, T. Wen, H. Yang, Y. Luo, Energy-efficient and-economic technologies for air conditioning with vapor compression refrigeration: A comprehensive review, Applied Energy 232 (2018) 157–186.

[2] K. J. Chua, S. K. Chou, W. M. Yang, Advances in heat pump systems: A review, Applied Energy 87 (12) (2010) 3611–3624. doi:10.1016/j.apenergy.2010.06.014.

[3] S. A. Tassou, G. De-Lille, Y. T. Ge, Food transport refrigeration—Approaches to reduce energy consumption and environmental impacts of road transport, Applied Thermal Engineering 29 (8-9) (2009) 1467–1477. doi:10.1016/j.applthermaleng.2008.06.027.

[4] J. Jäschke, S. Skogestad, Nco tracking and self-optimizing control in the context of real-time optimization, Journal of Process Control 21 (10) (2011) 1407–1416.

[5] N. Jain, J. P. Koeln, S. Sundaram, A. G. Alleyne, Partially decentralized control of large-scale variable-refrigerant-flow systems in buildings, Journal of Process Control 24 (6) (2014) 798–819.

[6] M. Wallace, B. Das, P. Mhaskar, J. House, T. Salsbury, Offset-free model predictive control of a vapor compression cycle, Journal of Process Control 22 (7) (2012) 1374–1386.

[7] S. A. Bortoff, P. Schwerdtner, C. Danielson, S. Di Cairano, D. J. Burns, H-infinity loop-shaped model predictive control with hvac application, IEEE Transactions on Control Systems Technology (2022) 1–16doi:10.1109/TCST.2022.3141937.

[8] P. Li, Y. Li, J. E. Seem, Efficient Operation of Air-Side Economizer Using Extremum Seeking Control, Journal of Dynamic Systems, Measurement, and Control 132 (3) (May 2010). doi:10.1115/1.4001216.

[9] D. J. Burns, C. R. Laughman, M. Guay, Proportional-integral extremum seeking for vapor compression systems, in: 2016 American Control Conference (ACC), IEEE, 2016, pp. 2352–2357.

[10] W. Wang, Y. Li, Intermediate pressure optimization for two-stage air-source heat pump with flash tank cycle vapor injection via extremum seeking, Applied Energy 238 (2019) 612–626.

[11] D. J. Burns, C. R. Laughman, M. Guay, Proportional–integral extremum seeking for vapor compression systems, IEEE Transactions on Control Systems Technology 28 (2) (2020) 403–412. doi:10.1109/TCST.2018.2882772.

[12] A. Chakrabarty, C. Danielson, S. A. Bortoff, C. R. Laughman, Accelerating self-optimization control of refrigerant cycles with

bayesian optimization and adaptive moment estimation, Applied Thermal Engineering 197 (2021) 117335.
[13] M. Guay, D. J. Burns, A proportional integral extremum-seeking control approach for discrete-time nonlinear systems, International Journal of Control 90 (8) (2017) 1543–1554.
[14] M. Guay, A perturbation-based proportional integral extremum-seeking control approach, IEEE Transactions on Automatic Control 61 (11) (2016) 3370–3381. doi:10.1109/TAC.2016.2519840.
[15] M. Guay, K. Atta, Dual mode extremum-seeking control via lie-bracket averaging approximations, in: Proceedings of the 2018 American Control Conference, Milwaukee, WI, 2018.
[16] W. Xu, C. N. Jones, B. Svetozarevic, C. R. Laughman, A. Chakrabarty, Vabo: Violation-aware bayesian optimization for closed-loop control performance optimization with unmodeled constraints, in: 2022 American Control Conference (ACC), 2022, pp. 5288–5293. doi:10.23919/ACC53348.2022.9867298.
[17] M. Neumann-Brosig, A. Marco, D. Schwarzmann, S. Trimpe, Data-efficient autotuning with bayesian optimization: An industrial control study, IEEE Transactions on Control Systems Technology 28 (3) (2019) 730–740.
[18] M. Khosravi, A. Eichler, N. Schmid, R. S. Smith, P. Heer, Controller tuning by bayesian optimization an application to a heat pump, in: 2019 18th European Control Conference (ECC), IEEE, 2019, pp. 1467–1472.
[19] M. Khosravi, V. N. Behrunani, P. Myszkorowski, R. S. Smith, A. Rupenyan, J. Lygeros, Performance-driven cascade controller tuning with bayesian optimization, IEEE Transactions on Industrial Electronics 69 (1) (2021) 1032–1042.
[20] Q. Lu, R. Kumar, V. M. Zavala, Mpc controller tuning using bayesian optimization techniques, arXiv preprint arXiv:2009.14175 (2020).
[21] J. A. Paulson, K. Shao, A. Mesbah, Probabilistically robust bayesian optimization for data-driven design of arbitrary controllers with gaussian process emulators, in: Proceedings of the IEEE Conference on Decision and Control, 2021.
[22] M. Fiducioso, S. Curi, B. Schumacher, M. Gwerder, A. Krause, Safe contextual bayesian optimization for sustainable room temperature pid control tuning, arXiv preprint arXiv:1906.12086 (2019).
[23] J. A. Paulson, M. Martin-Casas, A. Mesbah, Optimal bayesian experiment design for nonlinear dynamic systems with chance constraints, Journal of Process Control 77 (2019) 155–171.
[24] A. Chakrabarty, S. A. Bortoff, C. R. Laughman, Simulation failure robust Bayesian optimization for estimating black-box model parameters, in: Proc. IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2021, pp. 1533–1538.
[25] F. Berkenkamp, A. Krause, A. P. Schoellig, Bayesian optimization with safety constraints: safe and automatic parameter tuning in robotics, Machine Learning (2021) 1–35.
[26] B. Jiang, X. Wang, Constrained bayesian optimization for minimum-time charging of lithium-ion batteries, IEEE Control Systems Letters 6 (2021) 1682–1687.
[27] M. Guay, A time-varying extremum-seeking control approach for discrete-time systems, Journal of Process Control 24 (3) (2014) 98–112.
[28] F. Berkenkamp, A. P. Schoellig, A. Krause, No-regret bayesian optimization with unknown hyperparameters, Journal of Machine Learning Research 20 (50) (2019) 1–24.
[29] J. Snoek, H. Larochelle, R. P. Adams, Practical Bayesian optimization of machine learning algorithms, in: Proc. NeurIPS, 2012, pp. 2951–2959.
[30] C. K. Williams, C. E. Rasmussen, Gaussian Processes For Machine Learning, Vol. 2, MIT press Cambridge, MA, 2006.
[31] J. Hensman, A. G. Matthews, Z. Ghahramani, Scalable variational Gaussian process classification, J. Mach. Learn. Res. 38 (2015) 351–360.
[32] W. Lin, Further results on global stabilization of discrete nonlinear systems, Systems & Control Letters 29 (1) (1996) 51–59.
[33] A. Chakrabarty, V. Dinh, M. J. Corless, A. E. Rundell, S. H. Żak, G. T. Buzzard, Support vector machine informed explicit nonlinear model predictive control using low-discrepancy sequences, IEEE Transactions on Automatic Control 62 (1) (2016) 135–148.
[34] A. Chakrabarty, C. Danielson, S. Di Cairano, A. Raghunathan, Active learning for estimating reachable sets for systems with unknown dynamics, IEEE Transactions on Cybernetics (2020) 1–12doi:10.1109/TCYB.2020.3000966.
[35] M. A. Gelbart, Constrained Bayesian Optimization and Applications, Ph.D. dissertation, Harvard University (2015).
[36] D. Eriksson, M. Pearce, J. Gardner, R. D. Turner, M. Poloczek, Scalable global optimization via local Bayesian optimization, Advances in neural information processing systems 32 (2019).
[37] M. Balandat, B. Karrer, D. Jiang, S. Daulton, B. Letham, A. G. Wilson, E. Bakshy, BoTorch: a framework for efficient Monte-Carlo Bayesian optimization, Advances in neural information processing systems 33 (2020) 21524–21538.
[38] Modelica Association, Modelica specification, Version 3.4 (2017).
URL `www.modelica.org`
[39] H. Qiao, V. Aute, R. Radermacher, Transient modeling of a flash tank vapor injection heat pump system - Part I: Model development, Int. J. Refrigeration 49 (2015) 169–182.
[40] D. J. Burns, C. R. Laughman, Extremum seeking control for energy optimization of vapor compression systems, in: International Refrigeration and Air Conditioning Conference at Purdue, 2012.
[41] Modelica Association, Functional Mockup Interface for Model Exchange and Co-Simulation, Version 2.0.1 (2019).
URL `www.fmi-standard.org`
[42] Dassault Systemes, Dymola 2020 (2019).