Highlights

A Bootstrapped Automated Pipeline for Developing Model Predictive Controllers for Non-domestic Buildings

Prathamesh Manoj Khatavkar, Peter Rockett, Yuri Kaszubowski Lopes, Elizabeth A Hathway

- Addressed the costs associated with generating models for model predictive control (MPC) of buildings—that currently consume around 75% of an MPC project budget—by proposing an automated pipeline to generate predictive models for practical MPC implementations.
- Bootstrapped the automated pipeline with a cheap and fast-to-identify model compatible with building commissioning.
- Investigated the iterative improvement of the MPC controllers using closed-loop re-identification that maintains the the building's internal climate under control at all times.

A Bootstrapped Automated Pipeline for Developing Model Predictive Controllers for Non-domestic Buildings

Prathamesh Manoj Khatavkar[a], Peter Rockett[a], Yuri Kaszubowski Lopes[b], Elizabeth A Hathway[c]

[a] *School of Electrical and Electronic Engineering, University of Sheffield, Sheffield, S1 3JD, UK*
[b] *Department of Computer Science, Santa Catarina State University, Joinville, SC, 89219-710, Brazil*
[c] *School of Mechanical, Aerospace and Civil Engineering, University of Sheffield, Sheffield, S1 3JD, UK*

Abstract

In this paper, we motivate and investigate an alternative approach to the development of predictive models for the practical implementation of model predictive control in non-domestic buildings. We describe how the process can be 'bootstrapped' with a very simple model, the crude nature of which illustrates the robustness of our approach. A predictive model for the controller is refined/adapted to the building in operation while maintaining climate control throughout at all times using closed-loop system identification. To remove the necessity for human intervention, we have used genetic programming to learn the predictive models since this combines a number of what are traditionally sequential search operations into a single step. We report preliminary results of a series of simulation experiments that validate the basic approach, and identify further research needed to develop the proposed methodology. Our approach facilitates the adoption of model predictive control by using commissioning data and refinement of models with data from the occupied building, while maintaining thermal comfort.

Keywords:
Closed-loop re-identification of dynamic non-linear system, Model predictive control, Building energy management

---

1. Introduction

The need to reduce the energy consumption of buildings is very well rehearsed in the literature. Although improved thermal performance of the envelope is the obvious approach, operational performance can still be enhanced with better control systems [1]. Furthermore, the rate of building replacement is quite slow, and large numbers of existing buildings have poor insulation standards; upgrading their thermal performance is often a significant technical challenge, especially for historically-important buildings. While improved envelope characteristics are highly desirable, there is an overarching requirement to achieve the best possible energy performance for buildings as they exist. To this end, model predictive control (MPC) has received a great deal of attention—the application of MPC to building climate control has been recently reviewed in [2, 3].

In this paper, we are concerned solely with non-domestic (*i.e.* non-residential) buildings; throughout, we use the term "building" to mean a non-domestic building.

1.1. MPC background

The most challenging part of implementing MPC (for any system, not just buildings) is widely acknowledged to be generating a suitable predictive model of the system's dynamics [4], with this phase of the control project typically cited as consuming around 75% of the budget [5, 6].

Hitherto, most published work in buildings has developed MPC controllers using a white- or gray-box methodology—see [7] for a fuller discussion of model taxonomies. Such engineer-intensive creation/tuning of the dynamical model is, however, highly undesirable since the economics of the construction industry mean that the need for extended involvement of skilled control specialists is likely to seriously limit the uptake of MPC [4]. This point has also been made in a very practically-oriented paper [8] that remarks that "Buildings are complex systems, each is unique and therefore a detailed modeling of every building where MPC shall be applied is economically unjustifiable". These points have been further discussed in [3, 4].

Additionally, the characteristics of buildings inevitably change over time (*e.g.* changes to shading, partitioning, occupancy, *etc.*), which will likely require periodic recalibration, or indeed complete re-identification, of the dynamical model according to some hard-to-establish schedule, and will further compound the shortage of appropriately skilled control engineers.

To address the issue of MPC system implementation, Jorissen et al. [1] have proposed TACO, a toolchain based on the Modelica simulation language to automate the generation of MPC controllers. This work, however, assumes the *a priori* existence of a calibrated white-box model of the building envelope (as well as other simplifying assumptions about the HVAC plant). Accommodating subsequent changes to the building characteristics within the TACO framework would seem to need a complete repeat of the procedure given that TACO's starting point is a calibrated white-box model of the building envelope. Further, creating a calibrated white box model of an existing building (and updating it in response to retrofit) may prove problematic due to the uncertainties in the envelope characteristics.

Andriamamonjy et al. [9] explored a semi-automated toolchain to convert from Building Information Management (BIM) data to a gray-box resistor/capacitor (RC) model using a multiobjective genetic algorithm to search over the combinatorial space of possible RC circuit configurations. These authors were motivated by an insightful literature review that concluded that the RC models commonly used for buildings MPC implementations were case dependent and need to be specialized to each individual building. This paper reported the refinement of accurate predictive RC models, but did not carry the process through to demonstrating closed-loop control. More importantly, this BIM route assumes that the BIM data have been entered accurately, and that the building has been constructed exactly as it was designed, something that is often questionable. In fact, there is emerging anecdotal evidence that many of the deficiencies of building climate control that have traditionally been blamed on building services, in fact, have their root cause in construction defects[1].

All the above factors imply that machine learning—black-box (as opposed to white- or gray-box) approaches—are required in which the dynamical model is learned from measured data gathered from the building in operation as it physically exists, including the influence of any construction defects. In a wide-ranging comparison of different approaches for constructing dynamical models for building MPC, Stoffel et al. [10] concluded that a machine-learning model produced the best results.

A common criticism of black box models—see, for example, [9]—is that they lack *interpretability*, and, for this reason, gray-box models based on

[1]James Thomas – Private communication (2023)

electrical RC analogies are often deemed preferable. However, considering some of the complex, optimized RC circuit topologies generated by Andriamamonjy et al., we suggest this argument about interpretability is misplaced: many RC models are every bit as empirical as pure black-box models—just because the individual R and C components have a simple, physical interpretation does not mean that an arbitrarily complex assemblage of R's and C's is interpretable. Complex systems are likely to require complex models however they are formulated.

1.2. Machine learning approaches

In terms of machine learning (ML) models, many possible approaches are available. Since buildings are usually sufficiently nonlinear to require nonlinear predictive models, we restrict consideration to this class. The topic has been nicely reviewed by [11].

Deep neural networks (DNNs) have received a huge amount of recent attention in the machine learning (ML) literature, and some impressive results have been reported across wide range of applications. DNNs employ large—often very large—numbers of layers and parameters which implies the need for a correspondingly large training set. In the context of performing system identification (SID) measurements on a building, the need to acquire a large set of training examples is a serious practical limitation since this equates to a long data acquisition time during which control of an (occupied) building is uncertain or even impossible. In addition, DNNs typically have a large computational overhead for both training and recall. Finally, the architecture of a DNN—the *model selection*—typically needs to be matched to the particular application, a task that has traditionally been carried out by trial-and-error although automating this neural architecture search (NAS) process has received much attention [12], albeit at the cost of an even larger computational burden. AutoML, that attempts to automate the deployment of DNNs using a pipeline process, has been reviewed in [13] and by others.

Feedforward (FNN) and recurrent neural networks (RNNs) using much smaller neural architectures than DNNs—what are often termed *multi-layer perceptrons* (MLPs)—have been widely employed in MPC; see [14] for a comprehensive review. Nonetheless, the same requirement to search over the set of possible architectures (numbers of inputs, numbers of hidden neurons, *etc.*) remains, and so many of the previous comments on DNNs about trial-and-error model selection and neural architecture search (NAS) also hold for FNNs/RNNs. Appropriate selection of the lag set of inputs, the time-delayed, previous exogenous inputs and autoregressive outputs compounds the complexity of the search problem.

Reinforcement learning (RL) has also received attention for building controllers [15], but the very low data efficiency of this technique is a serious concern since it can lead to prolonged training times during which the building may be operating sub-optimally. The alternative to using real, measured data is to use simulated data [15], but this requires an accurately calibrated building model.

A range of other nonlinear predictive models have been used in buildings MPC. Hammerstein models [16, 17], which cascade a static nonlinearity with a linear dynamical model have been widely employed although data-driven approaches for constructing Hammerstein models share the same combinatorial characteristics as neural network search.

Finally, Gaussian processes that model data as a multivariate Gaussian distribution have been used for MPC [18] as well as buildings MPC [19, 20] specifically. Gaussian processes have the nice properties of providing not only a prediction, but also a Gaussian-distributed error on that prediction. Nonetheless, they are also technically challenging to deploy; selecting the input lag set again seems to require trial-and-error.

There have been several reports [21, 22, 23] of using various machine learning methods to learn the control laws of an existing optimization-based MPC system. While reducing the computational resources required for routine building operation, these imitation-learning approaches still require the commissioning of a full MPC system from which to learn, which—paradoxically—in turn requires an accurate dynamical model of the building. Further, since it is an imitative technique, such a controller is only likely to perform, at very best, as well as the full MPC system from which it learned. If the full MPC system supplying the training data is sub-optimal, the imitative system is also likely to be sub-optimal. System updating to accommodate changes in building characteristics is problematic.

Drgoňa et al. [11] have extended DNNs to embed physics-informed constraints although their report details only the successful learning of 10 days of *observational* data without demonstrating closed-loop control of the target building It is well-established that good observational models do not necessarily lead to good control performance [24, 25].

In terms of the requirements for an ML framework for the proposed automated pipeline:

1. The models should be as compact as possible to maximize data efficiency—larger datasets require longer acquisition periods, which is undesirable in buildings. Compactness is also important in order to minimize the computational requirements of the controller hardware.
2. Notwithstanding the above, the model architecture should have sufficient representational capacity to capture the system dynamics.
3. The search over the model architecture should be automated since if the building's characteristics change, the architecture of the most appropriate model may also change. Or alternatively, the architecture should adjust to the problem in hand during training.
4. Selection of the appropriate set of system inputs, including lagged variables, should ideally be integrated with selection of the model architecture. If a separate feature selection search is required for each potential architecture then this pairing constitutes an undesirable combinatorial search.

While shallow NNs, DNNs and Gaussian processes can meet Requirement #2, Requirement #3 is difficult to satisfy other than by NAS over neural network architectures. Identifying the input variables (Requirement #4) needs combinatorial search. Moreover, DNNs and RL especially do not satisfy Requirement #1.

1.3. Proposed pipeline approach

In light of the all above, our preferred approach to addressing the model selection problem is *genetic programming* (GP) that tackles the combinatorial search problem simultaneously over architecture *and* input variables using a single evolutionary search [26]. GP models tend to be much more compact than, for example, DNN models.

We have previously reported successful implementation of MPC on a simulated building using GP [27] in which both the predictive model form as well as the set of lagged input and autoregressive terms were determined in a single, unified search process. Previously, however, we obtained the predictive model using open-loop system identification under which the internal conditions may reach extreme values. Indeed, in an open-loop identification of an MPC controller for a naturally ventilated building, Sykes [28] observed that the zone temperatures could rise as high as 35℃ during system identification, which is, of course, completely unacceptable in an occupied building.

A subsidiary issue is the amount of energy that may be consumed during the open-loop SID process.

Although the use of genetic programming to generate compact predictive models with little intervention from a control engineer has great attractions, two fundamental questions remain. Firstly, how to generate sufficient training data of the necessary quality to produce control-capable models, and especially how to potentially accommodate changes in building characteristics. Our answer to this question is to use *closed-loop identification* of the system in operation. The subject of *identification-for-control* (I4C) is long-established in control engineering [5].

Secondly, although closed-loop estimation is a credible path to updating an *existing* controller, it does not address the question of how to implement the very first controller. Our initial answer is to use a simple, first-order plus time delay (FOPTD) model obtained from a step-response excitation of the building. This estimation procedure is very fast, but can be anticipated to produce a poor/deficient model. Providing it gives some sort of control, however, it will form a credible starting point for subsequent closed-loop (re-)estimations.

In the present paper, we extend our previous results [27] to address the problem of 'bootstrapping' a dynamical predictive model for MPC without the need for expert input. We start with a crude but simple and cheap-to-acquire model that does not provide especially good control performance, and then progressively generate refined models using GP search with system identification data obtained under closed loop such that the building remains under control at all times [5].

It has long been known that the best controller performance is obtained by identifying the plant dynamics in closed-loop under the control of the optimal controller [29]. This, of course, is a paradox since if an optimal controller is available, why would one need to identify a plant model? In practice, an iterative approach can be used whereby the system is identified in closed loop using some initial controller, and the new, improved controller used to control the system during a second closed loop re-identification. And so on until satisfactory performance is obtained. As Gevers [30] points out, this is common industrial practice anyway, except it is invariably carried out manually.

1.4. Contributions of this paper

The contributions of this paper are to demonstrate the feasibility of:

1. Bootstrapping the very first predictive model using data acquired directly from the building
2. Developing a *fully automated* pipeline to generate MPC controllers for building climate control without the significant involvement of control specialists, or the creation of handcrafted white- or gray-box models. Our use of genetic programming that automatically evolves the *structure* of the predictive model is key here.
3. Refining models using data collected from occupied buildings through controlled experiments with minimal disturbance to thermal comfort.

We are not aware of any previous work on using closed-loop re-identification to derive/refine dynamic models for buildings MPC.

The above objectives are motivated by the arguments in [3, 4] that the specialists necessary to commission advanced control methods, such as MPC, are in short supply already. Additional demand caused by the widespread adoption of MPC in non-domestic buildings is, therefore, likely to be unsustainable.

This paper comprises a first report of our proposed automated pipeline for MPC deployment together with an initial investigation of the key process of bootstrapping the starting predictive model. Section 2 sets out our methodology, implementation and training of our dynamical genetic programming models together with the EnergyPlus [31] simulation environment used in our experiments. Section 3 reports the results obtained, which are discussed in Section 4. Section 4 also suggests future work to improve on the initial pipeline methodology. Section 5 concludes the paper.

2. Methodology

We first outline the relevant details of MPC and the genetic programming (GP) method used in this work before setting out details of the simulation environment employed. We then describe the method for constructing—*bootstrapping*—the very first MPC model, followed by a description of how we progressively refine/improve the predictive model using closed-loop estimation.

2.1. Model predictive control

The basic ideas behind setpoint regulation with MPC have been well reviewed in a number of places [2, 3, 4]. The fundamental idea is to use a mathematical model of the plant dynamics to estimate a sequence of future predictions $\left\langle \widehat{T}_{k+1}, \widehat{T}_{k+2}, \ldots, \widehat{T}_{k+p} \right\rangle$ at current time index k, and where p is the length of the prediction horizon. The sequence of current and future inputs $\mathcal{U}_k = \langle u_k, u_{k+1}, \ldots, u_{k+p-1} \rangle$ is then optimized to minimize the measure:

$$J = \min_{\mathcal{U}_k} \sum_{i=k+1}^{k+p} \left(\widehat{T}_i - T_i^S \right)^2 + \lambda_1 \sum_{i=k}^{k+p-1} (\Delta u_i)^2 + \lambda_2 \sum_{i=k}^{k+p-1} u_i^2 \tag{1}$$

where T_i^S is the desired setpoint at the i-th time. The second term on the right of (1) is designed to penalize control effort Δu_i^2, where $\Delta u_i = u_i - u_{i-1}$, and is usually included to reduce actuator wear. The third term in (1) is a proxy for control energy. $\lambda_{1,2}$ are non-negative, user-defined weightings.

Although the values of $\lambda_{1,2}$ can be tuned by explicit optimization, a clear focus in this work has been to ensure the robustness of our proposed pipeline. Consequently, rather than fine-tuning the weighting factors, we have followed the general guidelines in [32] and set $\lambda_1 = 10$ and $\lambda_2 = 0$; the same value of λ_1 was also used by [27].

Again in the spirit of ensuring robustness, we have also followed general guidelines in the MPC literature for setting the other parameters. Specifically, we have set the prediction horizon length p to be ≈80% of the system's settling time [32], and the sampling period such that the prediction interval contains 10-20 samples [25].

(In Section 3.1, the system settling time was observed to be about 85 minutes, and so we have set the prediction horizon to be around 80% of this figure at 60 minutes. As a consequence, we set the sampling period to be 5 minutes so that we have $p = 12$.)

2.2. Genetic programming environment

In general, the machine learning problem is to identify some mapping from a set of inputs to desired outputs. Here we take the current and previous: zone temperatures, ambient temperatures, and sum of the direct and diffuse solar radiation to predict the zone temperature one time step into the future—the so-called *one-step ahead* (OSA) prediction. We achieve this by minimizing the sum of the ℓ_2-norm of the prediction errors (PEs) over a time series of training data. By applying the model recursively, we can predict the future zone-temperatures over the prediction horizon that are necessary to implement MPC.

Genetic programming (GP) synthesizes a mapping using a composition of elementary functions, typically in the form of an expression tree. Starting from a population of randomly-initialized trees, we employ evolutionary search [33] to improve the performance of the population of trees on an objective function, here the ℓ_2-norm of OSA predictive errors. The leaf nodes in our trees are the inputs listed above plus real constants. Our internal tree nodes comprise: $+$, $-$, $\times$, AQ, unary minus and time-delay operators of one, two or four time units. The analytic quotient (AQ) operation [34] is a replacement for the arithmetic division ($\div$) operator which robustly eliminates the numerical issues when the denominator of a division operator is very small/zero. The time delay operators [35] facilitate the automatic search for the appropriate set of lagged inputs since the GP system is able to synthesize delays of arbitrary length by concatenating the basic delay operators.

Evolutionary GP search is conducted by selecting a pair of trees from the population biased in their fitness [33]. These so-called *parents* are then split randomly into two sub-trees and the two partitions of each parent crossed-over to form two new child trees. A random mutation is also applied to each child tree. In this work, we have used standard sub-tree crossover, and sub-tree mutation in which a randomly selected sub-tree is replaced with a new, randomly generated tree [33].

The basic GP framework used here is almost identical to that employed in [27]; in order to counter the tendency of GP trees to grow without limit ('bloat'), we actually employ a Pareto-based multiobjective fitness comprising i) the PE measure and ii) tree node count as a simple measure of tree complexity (which we wish to minimize subject to also minimizing the PE measure).

Additional to the basic GP framework in [27], we also standardize the input variables as well as minimizing the PE loss measure by numerically tuning the values of the constants in each of the offspring trees [36]. Both these steps have been shown to produce significantly improved performance compared to allowing the constant values to evolve only via crossover/mutation [36]. For further details of the GP framework employed, see [27] and [36].

GP training was performed offline with the best evolved model from the GP population selected using the smallest PE measure over an independent validation set, in keeping with standard ML practice. As is customary in GP (and indeed most non-convex optimizations), we have repeated the training process 30 times, each with an independently-initialized population, and taken the tree with the smallest overall validation error as the final model.

The final, optimized tree was imported into the separate control algorithm via the human-readable Genetic Programming Markup Language (GPML) [37]. This separation between training and MPC deployment greatly contributes to ease-of-integration in our pipeline since the training and control phases can run as separate, indeed concurrent, computational processes.

2.3. Simulation environment

As is common for studies in this area, we have conducted this work using simulation for convenience, precise regulation of experimental conditions and flexibility. We have used the well-known EnergyPlus building simulator [31] interfaced to a series of external programs via a Functional Mockup Interface (FMI) [38]. The FMI allows the passing of data between the simulated building and the external control program, as well as the setting of manipulatable control inputs in the (simulated) building by the external controller program.

For this first report, we have used the same single-zone building as [27], with a fixed zone occupancy of 50% of the nominal occupancy. Our rationale here was that, in practice, the control model would be acquired during the commissioning of the building services equipment, before handover, when the building would not be fully occupied. At the end of the model identification process, however, we have tested the evolved controller with a more realistic office occupancy schedule—see Section 3 for details.

The heating to the space is provided by a water baseboard heater that includes both convective and radiative heat transfer. The boiler setpoint is a constant 67℃ resulting in no time delays between the boiler and baseboard; the model outputs show that the boiler capacity is sufficient to achieve the setpoint temperature. The dynamics of the boiler do not account for variable plant efficiencies. However, the heat transfer to the room accounts for both time lags, due to thermal capacity of the fabric, and the nonlinear relationship between mass flow rate and the heat output of the baseboard. The ventilation rate is 9.4 l/s/person, and the total air flow rate adjusted according to occupancy. For further details of the (simulated) building used in this study, see [27].

We have used 2016 weather data from Manchester, UK as the training data, and the 2017 weather year from the same location as an independent test set. For all MPC runs reported here, we have assumed a zone setpoint of 20℃ during working hours (Monday to Friday, 9am to 5pm), and a setpoint of >6℃ otherwise to provide frost protection.

2.4. The initial bootstrapped model

The fundamental requirement for an initial bootstrapped model is that it is straightforward to acquire and needs fairly little input from control engineers. For this initial report, we have used a simple, first-order plus time delay (FOPTD) model (2) with parameters estimated from a single step response.

$$T(t) = K\Delta u \left(1 - \exp^{\frac{-(t-d)}{\tau}}\right) + T_0 \quad (2)$$

where $T = T(t)$ is the zone temperature as a function of time t, τ is the time constant, K is the process gain, $\Delta u \in [0, 1]$ the radiator flow fraction, d the time delay, and T_0 the baseline temperature.

We have employed this model for two reasons: Firstly, it meets the simplicity criterion as it only requires switching the water flow rate in the hydronic radiator from 0% to 100% output once, and logging the zone temperature. The acquisition of such data could easily be integrated with the 'witnessing tests' that are commonly used to assure proper installation of the building services plant although ultimately, we foresee using a more sophisticated initial model in more complex buildings—see future work below. In addition, FOPTD models are very commonly used in control engineering more generally [25].

Our second reason for adopting an FOPTD approach is that it can be anticipated to produce a very crude model of the building's dynamics since we are: i) incorrectly assuming the system is linear, and ii) ignoring important environmental factors such as ambient temperature and incident solar radiation. One could therefore expect that the control performance using such a rudimentary model would be poor. Our second motivation, therefore, is whether the subsequent re-identification stages—see below—are able to eventually produce acceptable MPC performance from what is a poor starting point.

2.5. Closed-loop re-identification

In terms of the excitations employed for closed-loop system identification, we have used the amplitude-modulated pseudo-random binary sequence (APRBS) [39]. Rather than directly 'dithering' the control input, we have perturbed the setpoint with an APRBS signal of amplitude in the range ±2℃, and allowed the controller to determine any changes to the radiator mass flow rate (MFR). This approach is straightforward to implement and appears to perform satisfactorily. The ±2℃ range was chosen because it "could result in mild discomfort among a proportion of [the] occupants" [40] and is therefore unlikely to cause significant problem to a building's occupants. We have verified that during all the closed-loop SID experiments, the zone temperature did not deviate outside the 20 ± 2℃ band.

We have adopted a 10-bit pseudo-random binary sequence (PRBS) to generate the both the training and validation data. The *minimum* hold time of the APRBS identification sequences was set to 15 minutes (or 3 sample periods), around the 'time constant' of the system leading to data sequences of 2049 samples and lengths of $\sim$7.1 days. Clearly there is a compromise between model quality and the length of the training sequence—the longer the data sequence, generally the better the model quality. An example two-day window of the APRBS signal is shown in Figure 1.

We have standardized our experiments on an excitation duration of about 7 days as a somewhat subjective compromise between the desire for a better-trained model ($\implies$ more data) and how long is acceptable during the commissioning of a building ($\implies$ fewer data). The length of the validation sequence used to select which single model was chosen from the GP training procedure was also set to be the same $\sim$7 day duration as the training set length. (See Section 2.2).

Starting with the bootstrapped FOPTD controller, we have performed a sequence of SID experiments in closed-loop such that the system remained under feedback control and within the comfort tolerance band.

In order to keep the experiments as close to reality as possible, the first re-estimations acquired SID data over the first 14 days (7 days for training, 7 days for validation) of the 2016 training year. The second re-estimation used the GP model that resulted from the first closed-loop SID experiment, and the next sequence of closed-loop SID data were acquired over days 15 to 28 of the year, and so on. Note that this procedure inevitably produces some variability in the training/validation sets due to the naturally varying weather conditions. Our experiments thus properly take this factor into account.

This sequence of closed-loop re-estimations can, in principle, be repeated until no worthwhile further improvement in controller performance is observed.

2.6. The overall re-identification pipeline

An overview of the pipeline and how the various processes described above

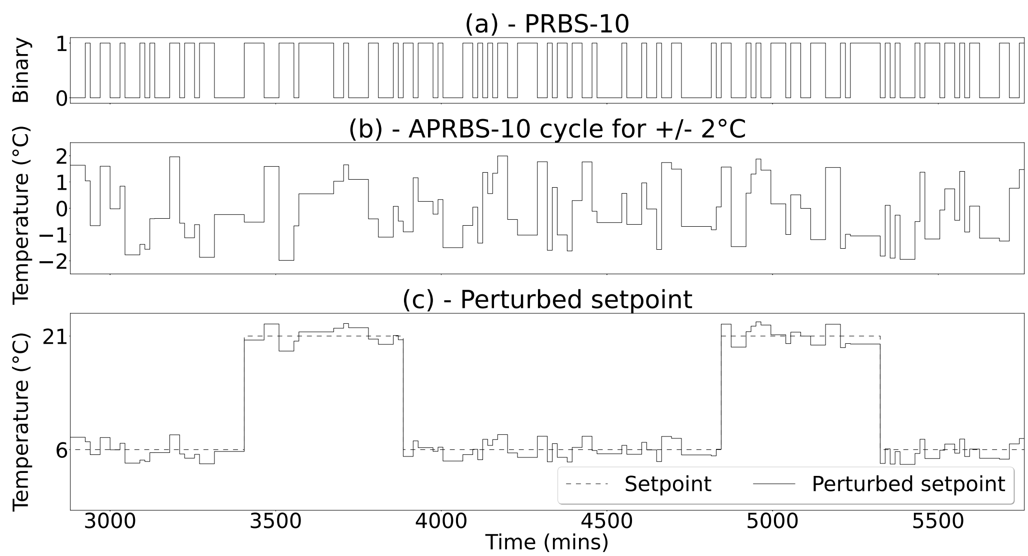


Figure 1: Illustration of the APRBS perturbation signal over a 2-day window. (a) shows the binary PRBS, (b) shows the amplitude modulated PRBS in the range ±2℃, and (c) shows the variation of the tracking setpoint.

combine is shown in Figure 2. The first (re-)identification commences with MPC using the bootstrap model and perturbations to the building setpoints injected with the APRBS signal. The SID data accumulated over a period are used to train a GP predictive model. This model is then used in the second re-identification, and the process repeated. Note that the model 'GPML_1' is both the output of '(Re-)identification #1' as well as effectively the 'input' to 'Re-identification #2'. The sequence of re-identification phases can be repeated as many times as desired, or indeed initiated if the building's characteristics change.

In the present report, we are, of course, using the EnergyPlus simulator as a surrogate for the real building in order to rapidly progress the experiments.

2.7. Control benchmarks

In addition to the bootstrapping/re-identification pipeline, for comparison, we have also examined two other common control strategies: bang-bang control, and MPC using a model estimated in open loop [27].

Bang-bang control. Bang-bang, also called on/off, control is a very simple reactive strategy whereby for a nominal setpoint of 20℃, if the zone temperature falls below 19℃, the MFR is set to 100%. If the zone rises above 21℃,

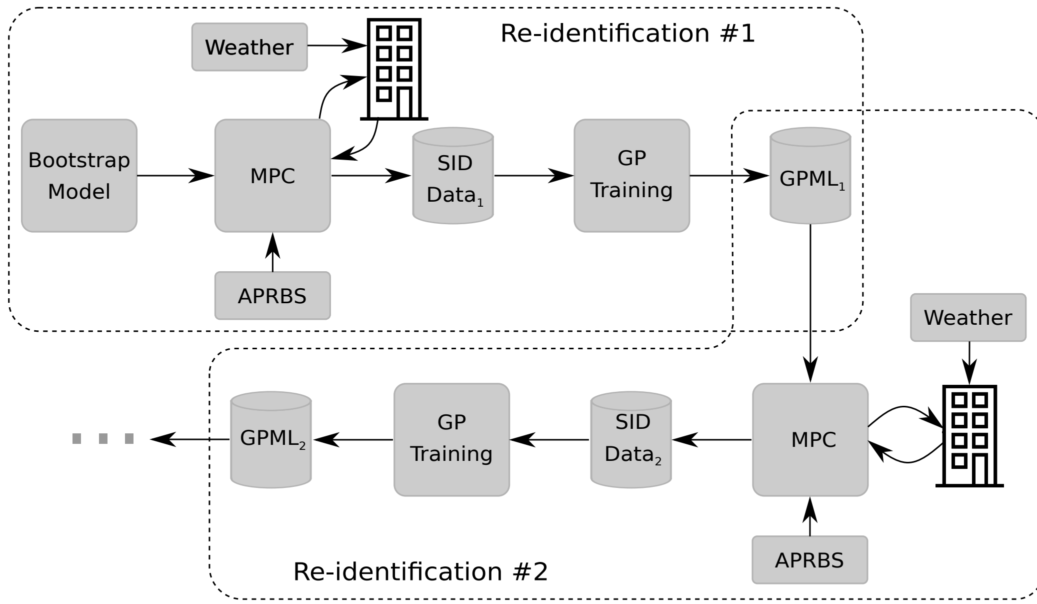


Figure 2: Illustration of the overall pipeline flow. The scope of each individual re-identification phase is delineated by a dashed box.

the MFR is set to zero. The control input thus switches between 0 and 100% at a rate that depends on the heat losses. Bang-bang control is commonly used in domestic heating, and often in simpler non-domestic situations, and is therefore a realistic baseline. Indeed, in some situations, it is an *optimal* controller [41], and in spite of its simplicity, actually performs quite well as a regulator of temperature in a single zone. In practice, however, it is far from ideal due to the excessive number of on-off cycling events, and would typically be replaced by a customized rules-based controller (RBC). We have deliberately avoided using a custom RBC here due to the obvious risk of being seen to create a 'man of straw'—a poor controller that is an unfair baseline.

Open-loop identification. For the open-loop baseline model, we distributed the MFR amplitudes of the APRBS in the range $0-100\%$, and used the same training and validation acquisition periods ($\sim$7.1 days each) as for closed-loop SID. The predictive model was generated by GP under the same conditions as the closed-loop identified models. As pointed out above, open-loop SID is not desirable as a model-updating method since the plant is, by definition, not under control, and the internal conditions in an occupied building can vary over an unacceptable range for the full duration of the SID experiment [28]. Our previous work used an open-loop identified model [27]. Nonetheless, we have included an open-loop identified controller as a benchmark.

2.8. Performance measures

Quantitative performance measures of building control systems have been considered in the International Building Performance Simulation Association (IBPSA) Project 1[2] Building Optimization Test (BOPTEST) framework [42], which identified six core Key Performance Indicators (KPIs): thermal (dis)comfort, indoor air quality (IAQ), energy use, energy cost, CO_2 emissions, and computational time ratio that calculates the average fraction of a controller time step that is required to determine the next control input. Of these, IAQ is not relevant in our test building which assumes a fixed ventilation rate based on occupancy. Similarly, our test building uses a single primary energy source (natural gas) and thus energy cost and CO_2 emissions are effectively proxies for energy use so we omit them.

Regarding the computational time ratio KPI, Blum et al. [42] define this as the average fraction of time taken to compute the next controller update divided by the system sampling time. We have computed this for the typical case of MPC with the seventh GP model over the whole test year, and we obtain an averaged KPI value of 3.5×10^{-5}. This compares with values of $\leq 1 \times 10^{-3}$ reported by Blum et al. for a similar single zone test case, but for a sampling time of 15 minutes. Insofar as an MPC system is a *firm* realtime system—one where occasional failures to meet the deadline can be tolerated but will lead to reduced performance—we believe a practically more useful measure would be the percentage of sampling intervals where the controller *fails* to calculate the next control action by the deadline. In reality, these computation times will form a distribution. In the present case, we observed a zero probability of failing to meet the deadline. Indeed the largest individual value in the distribution is 8.2×10^{-5} so the constraint is met by a very wide margin. We therefore omit any further consideration of computational issues.

We have quantified the performance of the various control schemes examined using three complementary measures over the test year since the control problem involves a number of conflicting trade-offs. For instance, we could

[2] https://ibpsa.github.io/project1/

obtain an energy usage of zero by not heating the building at all, but this, of course, would result in unacceptable occupant discomfort.

Firstly, we have considered cumulative energy use over the whole test year obtained directly from EnergyPlus[3]. Clearly the lowest possible energy usage is desirable subject to meeting comfort constraints.

Secondly, we have calculated (an approximation to) the integral over time of the temperature excursions outside the 20±1℃ tolerance band *during working hours* since variability within a 20±1℃ band "would attract little notice" [40]. This is equivalent to the BOPTEST thermal discomfort measure. Note, this normal-operation tolerance band is tighter than the ±2℃ band used for short periods during SID experiments.

In practice, we have summed the out-of-tolerance (±1℃) zone temperature deviations every minute over the test year. Rather than simply counting the numbers of minutes of departure from the 19-21℃ band, we consider the *magnitude* of the departure should be taken into account: a temperature of 16℃ is more uncomfortable than a temperature of 17℃. (Arguably, larger temperature deviations should receive a larger weighting, but this would seem to involve a degree of subjectivity hence we have opted for a simpler discomfort measure.) The calculation of this discomfort is complicated, however, by the fact that our simulated building does not contain any cooling, only heating, a situation that is common in the UK. Therefore, in the summer months, it is entirely possible that the zone temperature could exceed 21℃ when the ambient temperature is high, something that is outside the influence of the (heating-only) controller. Consequently, we have adopted a scheme of accumulating out-of-band temperatures only where the MFR has exceeded a small threshold value of 0.01 kg/s (compared to a maximum flow rate of 0.13 kg/s) on at least one occasion in the preceding 60 minutes, this being the duration of the prediction horizon. If the controller has exerted any active control over the preceding 60 mins and the temperature is out-of-band, this implies that control action is deficient. Obviously, a lower value of this discomfort measure is better; a similar type of discomfort measure was employed in [10]. To provide further insights, we have separated the deviations outside the comfort band into positive-going excursions (zone temperature >21℃) and negative going (<19℃) as well as providing the sum of the two.

[3]This is reported to only 2-decimal places, the default precision available from EnergyPlus.

Finally, we have calculated the cumulative control moves over the test year, namely the sum of the *first* of the $|\Delta u|$ terms in (1) that were actually *applied* to the plant, one per MPC update, and ignoring the remaining $(p-1)$ computed values that, by convention, are never used. We treat this cumulative measure as a proxy for actuator wear associated with a given control scheme. Again, lower is better. As of 2021, this measure was anticipated as a future extension to the BOPTEST framework.

2.9. Summary of parameters from present and previous MPC using GP

For convenient comparison, Table 1 presents a summary of the main parameters used in the work presented in [27] and this work.

Table 1: Summary of main parameters from previous work presented in [27] and this work.

Category	**Description**	**Previous work** [27]	**This work**
System Identification	Type	open-loop	closed-loop
	APRBS perturbation	0% to 100% MFR	setpoint temperature $\pm$ 2℃
	Sampling period	15 minutes	5 minutes
	APRBS length	2976 steps (31 days)	2049 steps (7.1 days)
	Input scaling	Min-max normalization	Standardization
	Constants leaf nodes	range [0.1, ..., 2.0]	range [0.1, ..., 1.0] and optimized (see [36])
	GP internal nodes	$+, -, \times, AQ$, unary minus, and time-delays	$+, -, \times, AQ$, unary minus, and time-delays
	Time-delay operators	1, 2, 3	1, 2, 4
MPC	Cost function parameters (see (1))	$\lambda_{1,2} = 10$	$\lambda_1 = 10, \lambda_2 = 0$
	Sampling period	15 minutes	5 minutes
	Prediction horizon	12 steps (180 minutes)	12 steps (60 minutes)

3. Results

3.1. Initial bootstrapped model identification

The step response in zone temperature obtained by applying the 0-to-100% step in radiator mass flow rate (MFR) using the 2016 training year is shown in Figure 3. The data were recorded until the zone temperature (approximately) settled, and the model parameters estimated by minimizing the maximum absolute deviation between the model's predictions and the actual temperature transient.

We found that the best fit to the data gives: $K = 7.34$ ℃, $d = 7$ minutes, $\tau = 17.0$ minutes, and $T_0 = 15.60$ ℃. From Figure 3 , it can be seen that the

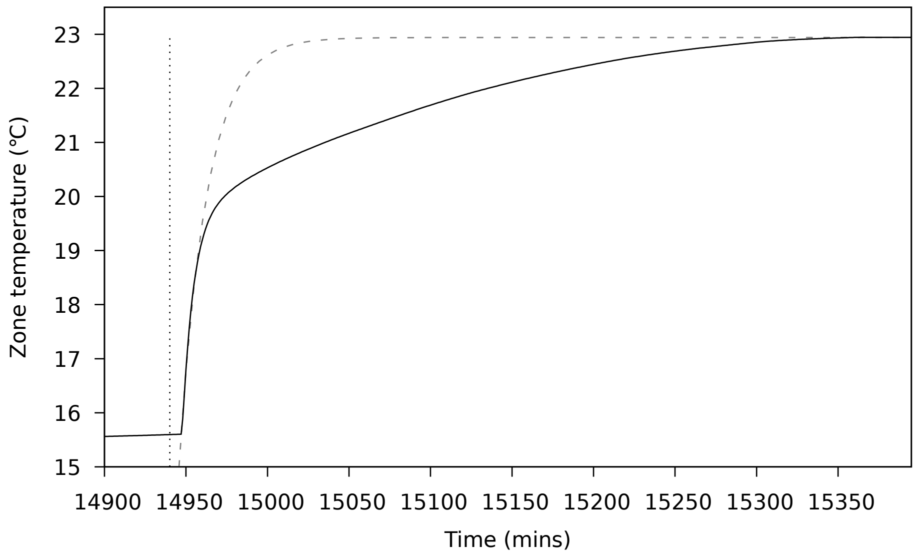


Figure 3: Step response of the simulated building. The vertical dotted line shows the time at which the step excitation was applied, and the dashed gray line shows the best-fitting exponential function as in (2). The measured data are shown with the solid black line.

difference between the observed and model characteristics is significant, and that the system is far from linear. The delay between applying the step and system's response is also evident from Figure 3. Nonetheless, we have used this FOPTD predictive model in an MPC controller, and the temperature regulation results for the first 60 days of the test year are shown in Figure 4a. For clarity, in this and all similar figures, we have displayed the zone setpoint temperatures—the top plot—scaled and offset from the other plots; as noted elsewhere, the zone setpoint switches between 20℃ and 6℃.

As might be anticipated from Figure 3, the zone temperature regulation shown in Figure 4a is very poor, and typically fails to even reach the 20±1℃ band although some control action is apparent from the bottom plot showing the mass flow rate. To reiterate: the reason for using the FOPTD model is that it is very easy to identify. Further, such an obviously poor initial model poses a significant test for the subsequent re-estimation procedure we describe in the next section.

3.2. Closed-loop re-identification

Starting with the FOPTD model, we have performed a series of closed-loop re-estimations. The first re-estimation used the FOPTD model as the predictive model for MPC and yielded a new, GP-based model. The second re-estimation used the first re-estimated GP model. The third re-estimation used the second model produced. And so on in a continual pipeline that can be easily automated. A typical training time for a single GP run for the seventh re-estimation is 68 ± 25 minutes[4], which suggests that this phase of the process does not represent a computational bottleneck.

The performance over the first 60 days of the test year for the first GP model identified from the FOPTD model is shown in Figure 4b. Compared with Figure 4a, the regulation performance is much improved with the zone temperature falling consistently within the 20±1℃ comfort band.

Taking the controller shown in Figure 4b, we have used this to re-estimate a second GP model, and then a third, and so on. The performance of the second re-estimated model for this sequence over the first 60 days of the test year is shown in Figure 4c. Zone temperature regulation is qualitatively similar to Figure 4b as indeed are the family of plots for all re-estimations in this sequence. As a consequence, and for brevity, we omit most of these plots.

Figure 4d shows the outcome for the seventh re-estimated model; note the small temperature overshoot at the start of each day. These small overshoots in zone temperature emerge after the second re-estimation, and are, of course, usually associated with a faster transient response of the controller. Qualitatively, the regulation looks similar for all these 'first-60-day' plots with no obvious 'winner' although arguably somewhat less 'chattering' of the control input is apparent from the second re-estimation onward.

Starting from the top of Table 2, the first three rows contain the results from the benchmark controllers starting with bang-bang control. The first row is for a bang-bang controller that starts operation at the exact 9am start of the working day whereas the second row is for a bang-bang controller that commences control a fixed 30 minutes before the start of the working day with the objective of providing improved thermal comfort when the occupants first arrive. Comparing these two bang-bang variants, the variant that starts 30 minutes early unsurprisingly uses more energy and performs worse on the

[4]On a 3.4 GHz Intel i7 PC running Linux Mint.

control moves metric, but produces slightly worse discomfort. It is notable that both bang-bang variants both exhibit small but consistent over- and under heating as a consequence of the purely reactive strategy they employ.

3.3. Performance measures

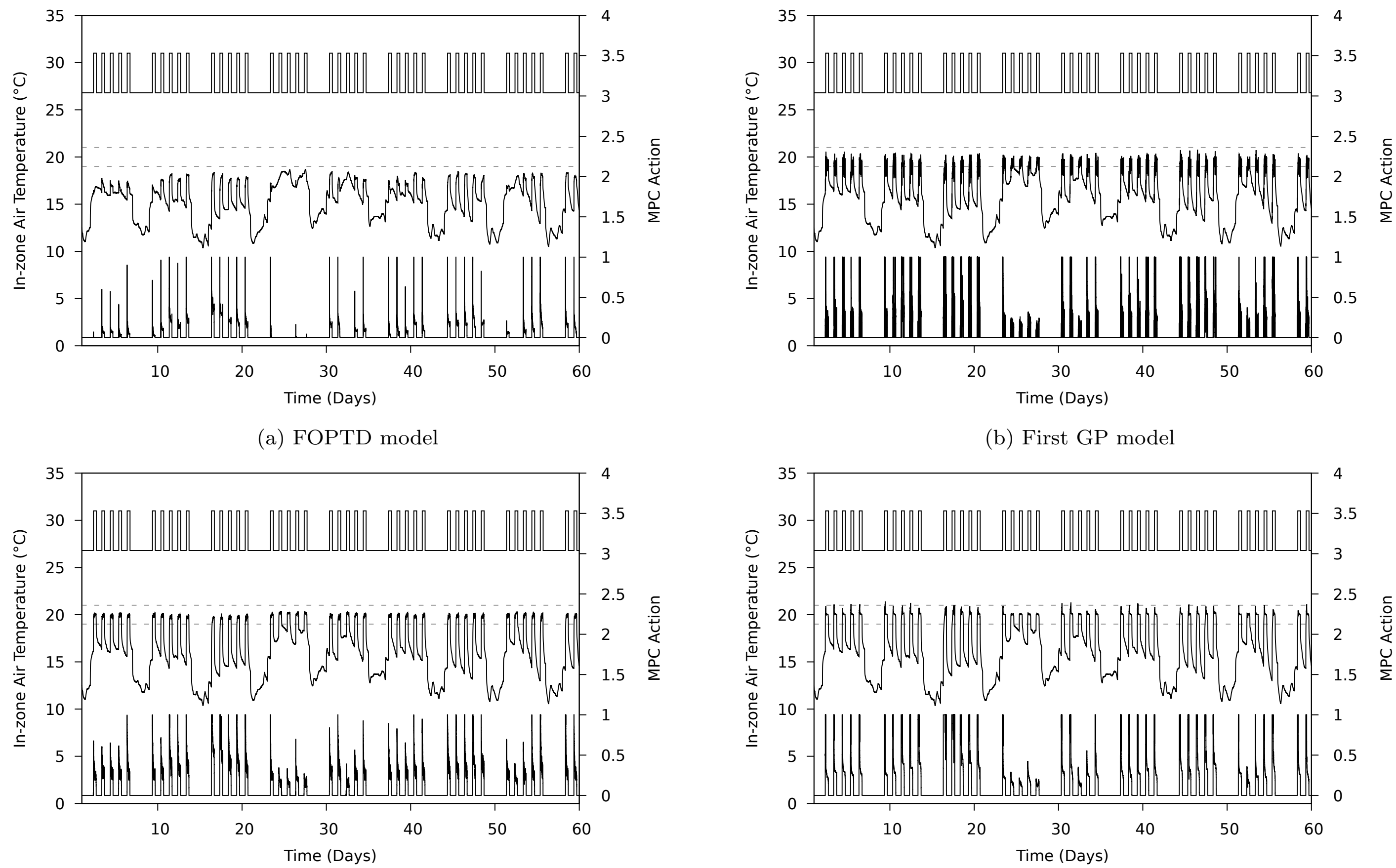
To gain a more global view of the performance of the set of controllers, Table 2 shows the various performance measures detailed in Section 2.8 taken over the independent 2017 test year.

The open-loop controller in the third row of Table 2 has much improved comfort compared to the bang-bang controllers as well as an improved control move measure although the annual heating energy usage is much higher at 13.09 GJ over the test year.

As would be surmised from Figure 4a, the initial FOPTD controller (fourth row) has very large negative discomfort deviations (too cold) and a correspondingly low annual energy usage; the cumulative control moves measure is also very low.

The fifth through twelfth rows of the table show the performance results for successive closed-loop re-estimations of GP-based controllers. The very first re-identification (5$^{\text{th}}$ row) produces an improvement on the initial FOPTD model in terms of zone temperature regulation—compare Figures 4a and 4b—but some of the annual performance measures are less favorable. Although the energy consumption is slightly lower than for bang-bang control, and the positive-going temperature deviations (column 3) are quite modest, the negative temperature excursions (column 4) are almost as large as the bang-bang controllers. In addition, the control effort (column 6) is also quite high, which might be inferred from Figure 4b where some 'chattering' of the control input is apparent.

In contrast to the first re-identification, the second re-identification uses the smallest amount of energy of any of the controllers (barring FOPTD), but contrarily, exhibits the worst positive-going temperature deviations of any of the controllers due to occasional overheating. By contrast, the control moves measure is quite modest.



(a) FOPTD model

(b) First GP model

(c) Second GP model.

(d) Seventh GP model.

Figure 4: Closed-loop performance over the first 60 days of the test year for the sequences of models starting with (a) the FOPTD model. (b)-(?) are for GP models identified under closed-loop SID from the previous model in the sequence. The middle plot shows the zone temperature (left scale); note the dashed ±1℃ lines denoting the tolerance region. The lower plot is the fractional mass flow rate (right scale). The top (offset and scaled) plot indicates the temperature setpoints switching between 6℃ and 20℃.

Table 2: Performance measures over the 2017 test year—see text for details. The temperature regulation measure is broken by positive temperate excursions out of the comfort band ("+ve"), and negative excursions ("-ve"). See also Section 2.8 for a full explanation of the various metrics.

Model	**Heating (GJ)**	**Temperature regulation (℃ mins)**			**Cumulative moves**
		+ve	**-ve**	**Total**	
Bang-bang	9.23	1239.1	4904.4	6143.5	3500.0
Bang-bang + 30 mins	9.83	1326.7	5070.0	6396.7	3734.0
Open-loop baseline	13.09	529.2	398.5	927.7	1108.8
FOPTD	3.17	0	51350.0	51350.0	207.2
First re-identification	9.14	118.1	5035.1	5153.2	2846.3
Second re-identification	7.41	4535.5	2433.5	6969.0	841.3
Third re-identification	9.22	25.8	823.5	849.3	958.7
Fourth re-identification	10.46	127.3	571.7	699.0	3350.3
Fifth re-identification	7.34	161.9	4999.9	5161.8	397.8
Sixth re-identification	10.44	248.2	563.3	811.5	1474.1
Seventh re-identification	9.55	0.0	3112.1	3112.1	504.0
Eighth re-identification	9.47	3.4	677.6	681.0	1932.4

Viewing the remaining sequence of re-identifications, there appear to be significant fluctuations in the various multiobjective performance measures. For example, annual energy usage over the test year varies between 6-13% larger and 20-22% smaller relative to the two bang-bang variants. Similarly, out-of-band temperature regulation is generally much better for MPC although not necessarily; in a few cases, the MPC controllers perform comparably or slightly worse than bang-bang control in terms of bounding the zone temperature. The fifth re-identification stands out in this regard although the energy consumption is the lowest of all the controllers considered (ignoring FOPTD).

The control moves measure is generally better for the MPC controllers although again not exclusively. The fourth re-identified controller, for example, has a cumulative control move measure comparable to bang-bang control, has the lowest total discomfort measure of all the controllers considered yet one of the largest energy consumptions.

Since we are imposing an additional excitation phase during system re-identification, it is appropriate to ask how much extra energy is required to perform the SID experiments. The additional energy requirements are shown in Table 3 where we have taken the difference between the energy usage for MPC operation (using the training year) *with* the setpoint perturbations for re-identification and the energy usage *without* those perturbations over the same period.

In general, the extra energy usage for re-estimation is approximately 1-3% of the annual consumption for each re-identification so model re-identification clearly comes at *some* energy cost. An interesting feature of Table 3 is that the identification of the first GP model using the FOPTD-based controller uses *less* energy. It should be borne in mind, however, that the FOPTD controller performs poorly, and even without closed-loop setpoint perturbations is incapable to maintaining the zone temperature in the required band.

To assess the fitness-for-purpose of the MPC controllers produced here, we explored changing the occupancy of the simulated building to a more realistic office-worker schedule. We have modified the occupancy schedule used with EnergyPlus to be 100% occupancy Monday to Friday between 9am and 5pm but with a fall to 75% occupancy between 11.00am to 1.00pm to simulate workers fetching lunch; outside these times, the building occupancy was set to zero.

The annual performance figures with this revised occupancy schedule are

Table 3: Additional energy requirements for each of the model re-identifications reported in Table 2.

Re-identification	**Additional Energy (GJ)**
FOPTD $\rightarrow$ First model	-0.01
First $\rightarrow$ Second model	0.11
Second $\rightarrow$ Third model	0.25
Third $\rightarrow$ Fourth model	0.10
Fourth $\rightarrow$ Fifth model	0.15
Fifth $\rightarrow$ Sixth model	0.07
Sixth $\rightarrow$ Seventh model	0.26
Seventh $\rightarrow$ Eighth model	0.01

shown in Table 4 using the seventh re-identified model in the MPC controller; we have selected this model somewhat arbitrarily as a 'middling' performer as judged from Table 2, and towards the end of the SID sequence. The first row in Table 4 is reproduced directly from Table 2 for convenience.

The most notable feature of Table 4 is that the annual energy consumption has almost halved, presumably due to the increased metabolic heat gains from the increased number of occupants. This underlines the complex influences occupants exert on a building [43]. Otherwise, the comfort metrics are actually improved over the base case principally due to the reduction in negative-going temperature deviations while the control effort remains essentially unchanged. The identified controller thus appears able to accommodate a realistic occupancy schedule without modification.

4. Discussion and further work

The principal contribution of this paper has been to demonstrate a path to an automated pipeline for the deployment of MPC in buildings. Critically, this does not require the extended involvement of specialist control engineers or access to calibrated building-physics models. Since this is a first report, a number of observations can be made, and a number of avenues for future work identified.

Firstly, we have used a very crude bootstrapped model—a first-order plus time delay model (FOPTD)—in part because it could be anticipated to produce a poor predictions. The fact that significantly better-performing predictive models can be produced validates this phase of the concept. Nonetheless, the building will be under the control of the bootstrap model for some period so a more sophisticated starting point for the sequence of re-identifications is highly desirable. Potentially, a bootstrapped model based on the BIM data extracted from the architectural plans, as demonstrated by Andriamamonjy et al. [9] is a very promising solution. While progress on BIM-to-simulator research is being made, here too there is significant work to do.

We have already pointed out that a BIM-derived model is questionable as a final MPC model due to: i) the possibility of construction defects causing significant plant-model mismatch, and ii) the difficulty of accommodating changes in the building characteristics on an ongoing basis. A BIM-derived model should, however, constitute an excellent bootstrap model that could be refined by closed-loop re-identification. For example, given the BIM-to-Modelica process in [9], the generated Modelica model could be used as a surrogate building to generate a comprehensive set of training data with which to initiate our GP pipeline. In this way, we could avoid using the FOPTD bootstrap model completely. Moreover, any significant discrepancy between the performance a BIM-derived model, which embodies the designer's intentions, and the dynamics of the as-constructed building might indicate construction defects that require further investigation. Quality control in construction is arguably an under-addressed topic [44].

A subsidiary justification for using the poor FOPTD model in the present work has been to verify that, for example, a BIM-derived bootstrap model that performs poorly due to construction defects can still act as a successful starting point for our toolchain. Nonetheless, further studies of the pipeline's robustness are needed, especially its resilience to sensor noise and to missing data, an all-to-common common feature in building services systems.

It is instructive to examine the shortcomings of a representative selection of some of the different models. Figure 5a shows the tracking errors for the first-identified model over the test year. The light gray plot shows the zone temperature, which can be seen to lie mostly within the 20±1℃ band aside from the middle of the year which is the summer period. (Recall our test building has no cooling.) The out-of-band temperature deviations (during working days, Monday to Friday, 9am to 5pm) are shown in black, and are overwhelmingly negative-going (*i.e.* too cold) during the heating months of the year.

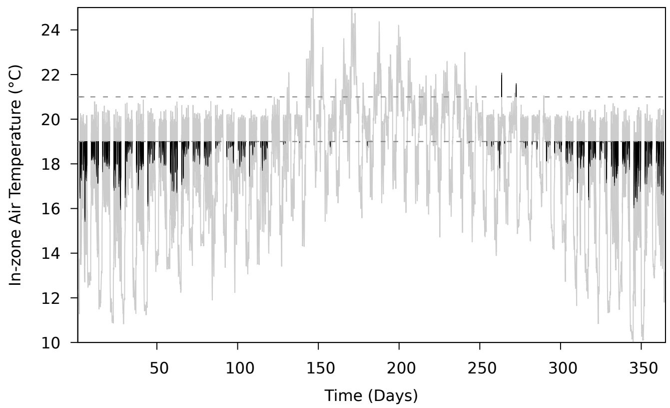
Figure 5b shows the first week of the data in Figure 5a, which is typical of the year. It is apparent that the control deficiencies are quite general with the controller struggling to maintain the target setpoint throughout the working day. We infer that the predictive model is not yet sufficiently well tuned at this first stage.

Table 4: Annual performance summary measures of the original, and after changes in the occupancy profile for the seventh re-identified model.

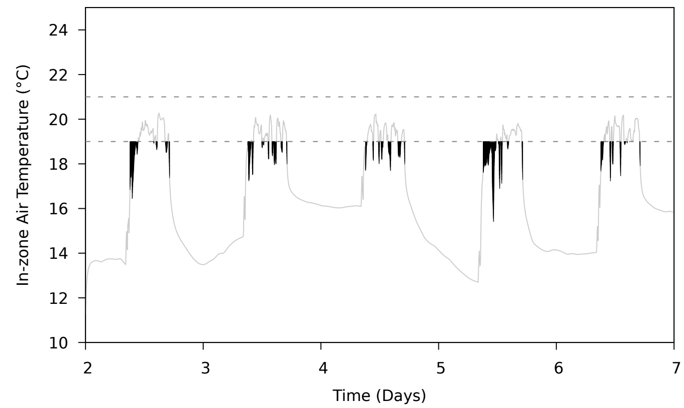
	Heating (GJ)	**Temperature regulation (℃ mins)**			**Cumulative moves**
		+ve	**-ve**	**Total**	
Original	9.55	0.0	3112.1	3122.1	504.0
Profile #2	5.56	206.0	1253.1	1459.1	496.8

On the other hand, the corresponding plots for the seventh-identified model are shown in Figures 5c and 5d. Comparing Figure 5a (first model) and Figure 5c (seventh model), it is clear that the 'envelope' of negative-going deviations is smaller. The (again typical) first week of the test year for the seventh-identified model is shown in Figure 5d from which we can see that almost all the contributions to the thermal discomfort measure are due to the controller ceasing heating a little too soon resulting in the zone temperature falling just before the end of the working day; in practice, this is likely to be a minor source of discomfort since the deviations are limited to about *leq*-2℃ and at a time when occupants will be preparing to depart. The shortcoming of this predictive model therefore appears to be its difficulty in accurately predicting the cooling transient. In overview, the interpretation of global measures, such as our discomfort index thus needs to be nuanced. This work also highlights the multiobjective nature of MPC.

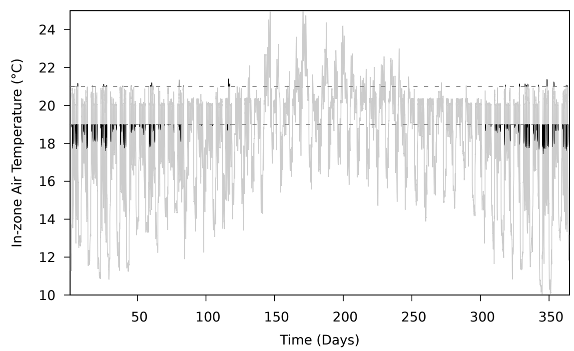
Figure 6 compares all the models over the 2017 test year in terms of heating energy, discomfort measure, and cumulative moves, where lower values are better for all three metrics. Except for the last column, a fixed occupancy rate of 50% is considered. Both reactive bang-bang controllers—the first column starting at the beginning of the working day and the second column starting 30 minutes prior—result in high discomfort and a large number



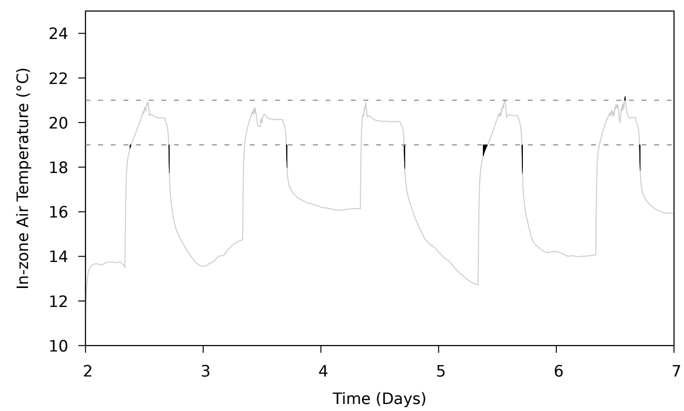
(a) First identified model.



(b) First identified model—first week.



(c) Seventh identified model.



(d) Seventh identified model—first week.

Figure 5: Tracking errors for the whole of the 2017 test year for MPC using the first and seventh identified models. The zone temperatures are shown in light gray. The black vertical lines extending down from 19℃ indicate where and by how much the zone temperatures are below the minimum tolerance during working hours whereas vertical black lines extending up from 21℃ indicate where and by how much the zone temperature exceed the upper tolerance level during working hours.

of control moves, despite using a significant amount of heating energy. The open-loop model achieves low discomfort and fewer control moves, but at the cost of high energy consumption, the highest among all the models tested. The re-identified models offer various trade-offs, excelling in one or two aspects but performing less well in others. For instance, the third re-identified model performs well in terms of discomfort and cumulative moves but has higher energy usage. On the other hand, the fifth re-identified model has the lowest energy usage and the second lowest number of control moves, but performs less well in discomfort reduction. In essence, as Blum et al. [42] remark, "which controller is 'better' may still be the subject of some subjectivity, especially when a controller performs better in one KPI and worse in another".

The last column of Figure 6 (Profile #2) shows the performance of the seventh re-identified model under a different occupancy schedule, with 100% occupancy Monday to Friday between 9 am and 5 pm, dropping to 75% between 11 am and 1 pm. A reduction in energy usage is observed due to the additional "free" heating from occupants (metabolic and equipment heat), as well as a decrease in the discomfort measure; the control moves measure is essentially unchanged.

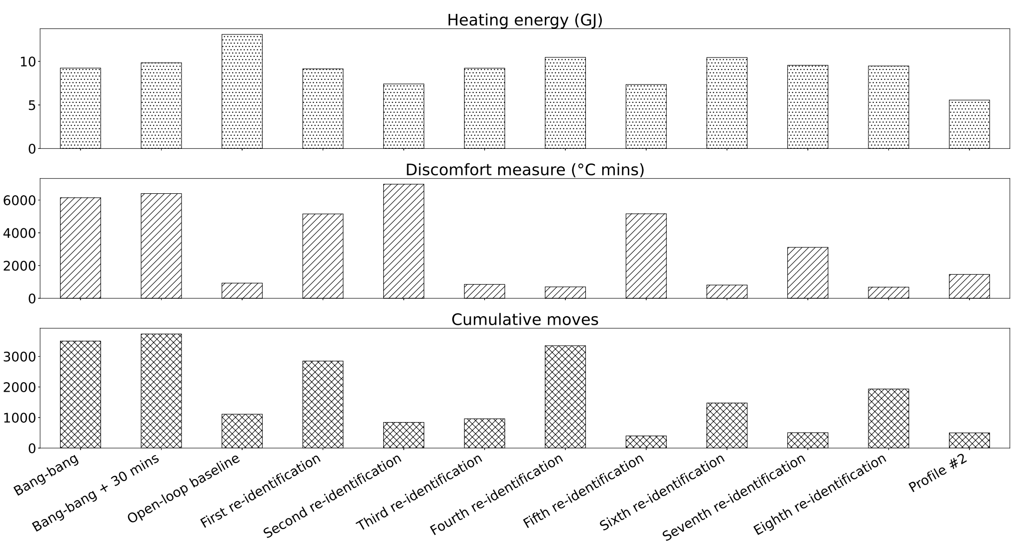


Figure 6: Performance comparison over the 2017 test year.

The other notable feature of this work is that although we have generally observed improved comfort relative to the baseline bang-bang controllers, we have also observed variable energy usage (relative to bang-bang control). This is in contrast to many reports in the buildings MPC literature, some of which have reported quite spectacular energy savings [3] (although we would question the reproducibility of some of those studies since the baseline "rules-based controllers" are often insufficiently specified). In fact, there is absolutely no reason why a model predictive controller that regulates zone temperature should save energy other than as a side-effect of eliminating any overheating produced by a sub-optimal rules-based controller. Nonetheless, there is clearly scope for improving the performance of the re-identified predictive models in the present work, and translating this to an industrial-grade toolchain.

In terms of the sequence of model re-identifications, an unsatisfying aspect is that while all the GP models produced credible, well-functioning MPC, the sequence of re-identifications does not appear to converge to a stationary endpoint, and therefore when to stop the sequence is unclear. In some senses, this phenomenon could have been anticipated [5] as this behavior has been frequently observed in the control literature where it is regarded as "well-known".

Gevers [30] has carried out a detailed theoretical analysis of this interplay between open-loop models trained to minimize PE and their resulting closed-loop performance. In outline, all models are subject to *some* modeling errors. In well-performing PE models these modeling errors have, by definition, a small effect, but in the absence of knowing the true system are difficult to quantify and predict. The characteristics of the *closed-loop* system, however, are usually significantly different from that of the open-loop. It is often the case that modeling errors in the open-loop model that have minimal effect on its open-loop performance are greatly amplified when that model is used in closed loop, resulting in worse than expected closed loop performance. In linear models this has motivated the use of filtering of the predictive errors to 'shape' the modeling errors and 'shift' them to regions of the spectrum that do not degrade the subsequent closed-loop performance. The use of data filters in nonlinear models, as used here however, is problematic. A further complicating factor [30] is that the characteristics of the model used to maintain control in the closed-loop SID experiment affect the data obtained and therefore the properties of the subsequently re-identified model. The influence of the open-loop modeling errors on the closed loop performance thus seems hard to predict and ameliorate. Sometimes it is minimal, other times not. For example, in a recent study of MPC of a nonlinear continuously stirred tank reactor (CSTR), Sorourifar et al. [45] observed that a PE-trained model with good performance produced disastrously poor closed loop control. Overall, it therefore seems reasonable that the widely observed lack of convergence to a stationary endpoint in a sequence of close-loop re-identifications [5] is due to the unpredictable propagation of modeling errors down the chain of re-identifications. A conclusion of the present work is, therefore, that modeling errors in the PE-trained models for buildings MPC have a noticeable effect (in that the sequence of re-identifications does not seem to converge), but that their influence is not so serious as to prevent effective control. An additional contributor to the variability in this work may be that each GP model has been trained with data collected over different time periods and therefore under different weather disturbances, as would be unavoidable in a practical MPC deployment.

The observation in this initial report of the sequence of re-identifications not converging to a stationary point is undesirable since it introduces an element of subjectivity (*i.e.* when to stop re-identifying) into what is intended to be a fully automated toolchain. Very recent work in the identification-for-control (I4C) community, however, has focused on calibrating the re-identified model, not to optimize predictive accuracy, but rather to optimize some measure of its *control performance* when the model is used in closed-loop [46]. In the buildings arena, setpoint regulation is only one of the criteria of interest—energy consumption is also important leading to a multiobjective optimization with trade-offs. Multiobjective performance-oriented model calibration has recently been reported by [47] for a biomanufacturing application. Indeed, the previously cited work of Sorourifar et al. [45] is also a contribution to this area. It is thus a key area of future work to calibrate our re-identified models to explicitly optimize closed-loop control performance rather than predictive accuracy in order to produce a convergent sequence of re-identifications. Such a refined scheme would proceed in two stages: firstly, training a surrogate model of the plant by minimizing the PEs on the system identification data. This model would then act in place of the real building in a second stage aimed at optimizing the closed-loop performance of a separate predictive model that will ultimately be deployed in the controller—see [47] for further details.

Finally, in terms of the training regimen, we have acquired both the training and validation data over a fixed 7-day period. An obvious area of future work is to examine the effects of the length of this acquisition window.

Furthermore, extension to economic MPC [48, 49] in which energy consumption is explicitly minimized subject to constraints on internal environmental quality (IEQ) is another avenue for future research. As we have pointed out, the approach of improving setpoint regulation alone in MPC cannot be guaranteed to reduce energy consumption except as a fortuitous side-effect of eliminating overheating due to an existing controller.

Finally, we noted from Table 3 that the re-identification process has a small but non-zero energy cost. Clearly, if the energy saving resulting from re-estimation is larger than the energy cost of that re-estimation then there is a positive benefit. Routine, periodic re-estimation, however, is likely to be questionable from an energy saving standpoint. The obvious solution is to monitor the controller's performance and initiate a re-identification if the plant-model mismatch becomes too great. There is a significant body of work in the general MPC literature on performance monitoring to draw on here [50], and this is another obvious area of future work, as is a thorough investigation of re-identification when the building's characteristics andigureor occupancy changes.

The other, equally important area of future work is the extension to buildings with multiple zones, as is practical demonstration in a range of real building forms as opposed to simulation. Here we have described the creation of multiple-input-single-output (MISO) models appropriate for a single zone. Nelles [25], for example, describes a number of options for assembling the multiple-input-multiple-out (MIMO) models needed for a multiple zone controller from MISO models.

5. Conclusions

In this paper, we have outlined a preliminary framework for the practical implementation of model predictive control (MPC) in buildings. Rather than rely on highly-specialized control engineers to iteratively 'hand tune' white- or gray-box predictive models, or relying on the availability of sufficiently accurate calibrated building-physics models, we have proposed obtaining a predictive model for MPC from a sequence of closed-loop system identification experiments of modest duration that can be carried out while the building is occupied. Our approach has the twin benefits of estimating the characteristics of the building as it is physically constructed (as opposed to what might have been intended by the designer), together with maintaining the building under control at all times during the identification procedure.

In addition, exactly the same re-identification procedure can be used to update the predictive model in the event of inevitable changes in the building's characteristics while in operation.

We have also demonstrated that this process can be initiated (or bootstrapped) with a very simple model obtained from a step excitation of the heating system. This quick path to bootstrapping can be carried out during the building services commissioning process although a more sophisticated approach using, for example, BIM models is desirable.

While proposing the framework of a readily automatable pipeline for MPC commissioning and ongoing operational tuning, we have identified a number of areas for development and future work. Overall, the results in this paper suggest a rather more nuanced interpretation of MPC performance data as it relates to buildings.

CRediT authorship contribution statement

Prathamesh Manoj Khatavkar: Methodology, Software, Validation, Formal analysis, Investigation, Visualization

Peter Rockett: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Supervision

Yuri Kaszubowski Lopes: Validation, Formal analysis, Writing - Review & Editing, Visualization

Elizabeth A Hathway: Conceptualization, Methodology, Writing - Review & Editing, Visualization, Supervision

References

[1] F. Jorissen, W. Boydens, L. Helsen, TACO, an automated toolchain for model predictive control of building systems: implementation and verification, Journal of Building Performance Simulation 12 (2) (2018) 180–192. doi:10.1080/19401493.2018.1498537.

[2] J. Drgoňa, J. Arroyo, I. Cupeiro Figueroa, D. Blum, K. Arendt, D. Kim, E. P. Ollé, J. Oravec, M. Wetter, D. L. Vrabie, L. Helsen, All you need to know about model predictive control for buildings, Annual Reviews in Control 50 (2020) 190–232. doi:10.1016/j.arcontrol.2020.09.001.

[3] M. Killian, M. Kozek, Ten questions concerning model predictive control for energy efficient buildings, Building and Environment 105 (2016) 403–412. doi:10.1016/j.buildenv.2016.05.034.

[4] P. Rockett, E. A. Hathway, Model-predictive control for non-domestic buildings: A critical review and prospects, Building Research & Information 45 (5) (2017) 556–571. doi:10.1080/09613218.2016.1139885.

[5] M. Gevers, Identification for control: From the early achievements to the revival of experiment design, European Journal of Control 11 (4-5) (2005) 335–352. doi:10.3166/ejc.11.335-352.

[6] M. A. Hussain, Review of the applications of neural networks in chemical process control - simulation and online implementation, Artificial Intelligence in Engineering 13 (1) (1999) 55–68. doi:10.1016/S0954-1810(98)00011-9.

[7] J. Schoukens, L. Ljung, Nonlinear system identification: A user-oriented road map, IEEE Control Systems Magazine 39 (6) (2019) 28–99. doi:10.1109/MCS.2019.2938121.

[8] J. Cigler, D. Gyalistras, J. Široký, V.-N. Tiet, L. Ferkl, Beyond theory: The challenge of implementing model predictive control in buildings, in: 11th REHVA World Congress (Clima 2013), Prague, Czech Republic, 2013, pp. 1008–1018.

[9] A. Andriamamonjy, R. Klein, D. Saelens, Automated grey box model implementation using BIM and Modelica, Energy and Buildings 188-189 (2019) 209–225. doi:10.1016/j.enbuild.2019.01.046.

[10] P. Stoffel, L. Maier, A. Kümpel, T. Schreiber, D. Müller, Evaluation of advanced control strategies for building energy systems, Energy and Buildings 280 (2023) 112709. doi:10.1016/j.enbuild.2022.112709.

[11] J. Drgoňa, A. R. Tuor, V. Chandan, D. L. Vrabie, Physics-constrained deep learning of multi-zone building thermal dynamics, Energy and Buildings 243 (2021) 110992. doi:10.1016/j.enbuild.2021.110992.

[12] P. Ren, Y. Xiao, X. Chang, P.-y. Huang, Z. Li, X. Chen, X. Wang, A comprehensive survey of neural architecture search: Challenges and solutions, ACM Computing Surveys 54 (4) (2021). doi:10.1145/3447582.

[13] X. He, K. Zhao, X. Chu, AutoML: A survey of the state-of-the-art, Knowledge-based Systems 212 (2021) 106622. doi:10.1016/j.knosys.2020.106622.

[14] Y. M. Ren, M. S. Alhajeri, J. Luo, S. Chen, F. Abdullah, Z. Wu, P. D. Christofides, A tutorial review of neural network modeling approaches for model predictive control, Computers & Chemical Engineering 165 (2022) 107956. doi:10.1016/j.compchemeng.2022.107956.

[15] Z. Nagy, G. Henze, S. Dey, J. Arroyo, L. Helsen, X. Zhang, B. Chen, K. Amasyali, K. Kurte, A. Zamzam, H. Zandi, J. Drgoňa, M. Quintana, S. McCullogh, J. Y. Park, H. Li, T. Hong, S. Brandi, G. Pinto, A. Capozzoli, D. Vrabie, M. Bergés, K. Nweye, T. Marzullo, A. Bernstein, Ten questions concerning reinforcement learning for building energy management, Building and Environment 241 (2023) 110435. doi:10.1016/j.buildenv.2023.110435.

[16] I. Hazyuk, C. Ghiaus, D. Penhouet, Optimal temperature control of intermittently heated buildings using model predictive control: Part II – Control algorithm, Building and Environment 51 (2012) 388–394. doi:10.1016/j.buildenv.2011.11.008.

[17] Y. Yu, V. Loftness, D. Yu, Multi-structural fast nonlinear model-based predictive control of a hydronic heating system, Building and Environment 69 (2013) 131–148. doi:10.1016/j.buildenv.2013.07.018.

[18] J. Kocijan, Modelling and Control of Dynamic Systems Using Gaussian Process Models, Springer, Cham, 2016. doi:10.1007/978-3-319-21021-6.

[19] K.-U. Ahn, D.-W. Kim, Y.-J. Kim, C.-S. Park, I.-H. Kim, Gaussian process model for control of an existing building, Energy Procedia 78 (2015) 2136–2141. doi:10.1016/j.egypro.2015.11.295.

[20] G. D. Kontes, G. I. Giannakis, V. Sánchez, P. de Agustin-Camacho, A. Romero-Amorrortu, N. Panagiotidou, D. V. Rovas, S. Steiger, C. Mutschler, G. Gruen, Simulation-based evaluation and optimization of control strategies in buildings, Energies 11 (12) (2018) 3376. doi:10.3390/en11123376.

[21] M. Klaučo, J. Drgoňa, M. Kvasnica, S. Di Cairano, Building temperature control by simple MPC-like feedback laws learned from closed-loop

data, in: 19$^{\text{th}}$ IFAC World Congress, Cape Town, South Africa, 2014, pp. 581–586. doi:10.3182/20140824-6-ZA-1003.01633.

[22] J. Drgoňa, D. Picard, M. Kvasnica, L. Helsen, Approximate model predictive building control via machine learning, Applied Energy 218 (2018) 199–216. doi:10.1016/j.apenergy.2018.02.156.

[23] J. Drgoňa, L. Helsen, D. L. Vrabie, Cutting the deployment costs of physics-based MPC in buildings by simulation-based imitation learning, Vol. 1, Virtual online, 2020. doi:10.1115/DSCC2020-3118.

[24] J. S. Sykes, E. A. Hathway, P. Rockett, Model identification for the control of naturally ventilated buildings, in: Building Simulation, Hyderabad, India, 2015. doi:10.26868/25222708.2015.2887.

[25] O. Nelles, Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models, Springer, Berlin, 2001.

[26] R. Poli, W. B. Langdon, N. F. McPhee, A Field Guide to Genetic Programming, Published via `http://lulu.com` and freely available at `http://www.gp-field-guide.org.uk`, 2008.

[27] T. Dou, Y. Kaszubowski Lopes, P. Rockett, E. A. Hathway, E. Saber, Model predictive control of non-domestic heating using genetic programming dynamic models, Applied Soft Computing 97 (2020) 106695. doi:10.1016/j.asoc.2020.106695.

[28] J. S. Sykes, Control of naturally ventilated buildings: A model predictive control approach, Ph.D. thesis, University of Sheffield, UK (2017).

[29] H. Hjalmarsson, M. Gevers, F. de Bruyne, For model-based control design, closed-loop identification gives better performance, Automatica 32 (12) (1996) 1659–1673. doi:10.1016/S0005-1098(96)80003-3.

[30] M. Gevers, Towards a joint design of identification and control?, in: H. L. Trentelman, J. C. Willems (Eds.), Essays on Control: Perspectives in the Theory and its Applications, Birkhäuser, Boston, MA, 1993, pp. 111–151. doi:10.1007/978-1-4612-0313-1_5.

[31] D. B. Crawley, C. O. Pedersen, L. K. Lawrie, F. C. Winkelmann, EnergyPlus: Energy simulation program, ASHRAE Journal 42 (4) (2000) 49–56.

[32] M. Alhajeri, M. Soroush, Tuning guidelines for model-predictive control, Industrial & Engineering Chemistry Resarch 59 (10) (2020) 4177–4191. doi:10.1021/acs.iecr.9b05931.

[33] S. Luke, Essentials of Metaheuristics, 2nd Edition, Lulu, 2014, available at http://cs.gmu.edu/∼sean/book/metaheuristics/.

[34] J. Ni, R. H. Drieberg, P. I. Rockett, The use of an analytic quotient operator in genetic programming, IEEE Transactions on Evolutionary Computation 17 (1) (2013) 146–152. doi:10.1109/TEVC.2012.2195319.

[35] M. P. Hinchliffe, M. J. Willis, Dynamic systems modelling using genetic programming, Computers and Chemical Engineering 27 (12) (2003) 1841–1854. doi:10.1016/j.compchemeng.2003.06.001.

[36] P. Rockett, Constant optimization and feature standardization in multiobjective genetic programming, Genetic Programming and Evolvable Machines 23 (1) (2021) 37–69. doi:10.1007/s10710-021-09410-y.

[37] T. Dou, Y. Kaszubowski Lopes, P. Rockett, E. A. Hathway, E. Saber, GPML: An XML-based standard for the interchange of genetic programming trees, Genetic Programming and Evolvable Machines 21 (4) (2020) 605–627. doi:10.1007/s10710-019-09370-4.

[38] T. Nouidui, M. Wetter, W. Zuo, Functional mock-up unit for co-simulation import in EnergyPlus, Journal of Building Performance Simulation 7 (3) (2014) 192–202. doi:10.1080/19401493.2013.808265.

[39] O. Nelles, R. Isermann, Identification of nonlinear dynamic systems classical methods versus radial basis function networks, in: American Control Conference, Vol. 5, Seattle, WA, 1995, pp. 3786–3790 vol.5. doi:10.1109/ACC.1995.533847.

[40] K. Butcher, B. Craig, Environmental Design : CIBSE Guide A, 8th Edition, CIBSE Guide ; A, The Chartered Institution of Building Services Engineers, London, 2015.

[41] K. J. Åström, R. M. Murray, Feedback Systems : An Introduction for Scientists and Engineers, 2nd Edition, Princeton University Press, Princeton, NJ, 2021.

[42] D. Blum, J. Arroyo, S. Huang, J. Drgoňa, F. Jorissen, H. T. Walnum, Y. Chen, K. Benne, D. Vrabie, M. Wetter, L. Helsen, Building optimization testing framework (BOPTEST) for simulation-based benchmarking of control strategies in buildings, Journal of Building Performance Simulation 14 (5) (2021) 586–610. doi:10.1080/19401493.2021.1986574.

[43] E. Delzendeh, S. Wu, A. Lee, Y. Zhou, The impact of occupants' behaviours on building energy analysis: A research review, Renewable and Sustainable Energy Reviews 80 (2017) 1061–1071. doi:10.1016/j.rser.2017.05.264.

[44] E. Mills, Building commissioning: A golden opportunity for reducing energy costs and greenhouse gas emissions in the United States, Energy Efficiency 4 (2) (2011) 145–173. doi:10.1007/s12053-011-9116-8.

[45] F. Sorourifar, G. Makrygirgos, A. Mesbah, J. A. Paulson, A data-driven automatic tuning method for MPC under uncertainty using constrained bayesian optimization, IFAC-PapersOnLine 54 (3) (2021) 243–250. doi:10.1016/j.ifacol.2021.08.249.

[46] D. Piga, M. Forgione, S. Formentin, A. Bemporad, Performance-oriented model learning for data-driven MPC design, IEEE Control Systems Letters 3 (3) (2019) 577–582. doi:10.1109/LCSYS.2019.2913347.

[47] G. Makrygiorgos, A. D. Bonzanini, V. Miller, A. Mesbah, Performance-oriented model learning for control via multi-objective Bayesian optimization, Computers & Chemical Engineering 162 (2022) 107770. doi:10.1016/j.compchemeng.2022.107770.

[48] J. B. Rawlings, N. R. Patel, M. J. Risbeck, C. T. Maravelias, M. J. Wenzel, R. D. Turney, Economic MPC and real-time decision making with application to large-scale HVAC energy systems, Computers & Chemical Engineering 114 (2018) 89–98. doi:10.1016/j.compchemeng.2017.10.038.

[49] Y. Zong, G. M. Böning, R. M. Santos, S. You, J. Hu, X. Han, Challenges of implementing economic model predictive control strategy for buildings interacting with smart energy systems, Applied Thermal Engineering 114 (2017) 1476–1486. doi:10.1016/j.applthermaleng.2016.11.141.

[50] P. D. Domanski, Performance assessment of predictive control – A survey, Algorithms 13 (4) (2020). doi:10.3390/a13040097.