

Kevin J. Kircher

Sibley School of Mechanical and Aerospace
Engineering,
Cornell University,
Ithaca, NY 14853
e-mail: kjk82@cornell.edu

Walter Schaefer

Sibley School of Mechanical and Aerospace
Engineering,
Cornell University,
Ithaca, NY 14853
e-mail: was87@cornell.edu

K. Max Zhang¹

Sibley School of Mechanical and Aerospace
Engineering,
Cornell University,
Ithaca, NY 14853
e-mail: kz33@cornell.edu

A Computationally Efficient, High-Fidelity Testbed for Building Climate Control

Advanced building climate control systems have the potential to significantly reduce greenhouse gas emissions and energy costs, but more research is needed to bring these systems to market. A key component of building control research is testing algorithms through simulation. Many high-fidelity simulation testbeds exist, but they tend to be complex and opaque to users. Simpler, more transparent testbeds also exist, but they tend to neglect important nonlinearities and disturbances encountered in practice. In this paper, we develop a simulation testbed that is computationally efficient, transparent and high fidelity. We validate the testbed empirically, then demonstrate its use through the examples of system identification, online state and parameter estimation, and model predictive control (MPC). The testbed is intended to enable rapid, reliable analysis of building control algorithms, thereby accelerating progress toward reducing greenhouse gas emissions at scale. We call the resulting testbed and supporting functions the `bdg` toolbox, which is free, open source, and available online. [DOI: 10.1115/1.4048895]

Keywords: buildings, heating, cooling, simulation, estimation, control, control systems, demand response, interactive buildings, optimal controls, thermodynamics

1 Introduction

The benefits of advanced heating and cooling controls are well established [1–4], but not much research has been done into the costs of developing and deploying them. A cost–benefit analysis available in the literature concludes that, at present, the required initial investment appears to outweigh the operating cost savings [4]. To realize the potential emission reductions of advanced heating and cooling controls at scale, their cost–benefit ratio will likely need to be improved. This could be accomplished, for example, by streamlining the process of building modeling or improving the performance of estimation and control algorithms.

Many methods have been proposed for these purposes. Since 2010, for example, researchers have tested artificial neural networks [5,6], subspace system identification [7], prediction error methods for system identification [7,8], online state or parameter estimation with linear, extended, and unscented Kalman filters [9–13], and model predictive control (MPC) in its certainty-equivalent [14–18], robust [19,20], and stochastic [21–24] forms.

It is not clear how different combinations of these methods balance performance against ease of implementation. Answering this question in general is difficult; buildings have a wide variety of geometries, construction types, mechanical systems, weather conditions, and control objectives. Therefore, building control algorithms are typically evaluated and compared through case studies.

Experimental case studies in occupied buildings and varying weather conditions, such as Refs. [1–4,14,16], allow the strongest conclusions to be drawn. They generally include the nonlinearities, disturbances, and other complications encountered in practice. Experiments have drawbacks, however. They tend to be costly and time consuming. Experimentally comparing algorithms requires either side-by-side testbeds or replicating weather conditions and occupant behavior. Some questions, such as sensitivity to construction parameters like thermal mass, or robustness to rare disturbances like extreme weather events, are difficult to investigate experimentally. For these reasons, most case studies are done in simulation testbeds.

We believe that two desirable attributes of a simulation testbed are *simplicity* and *fidelity*. Simple testbeds reduce modeling effort,

make it easy to vary parameters, and enable fast simulation and clear interpretation of results. The majority of the case studies mentioned earlier considered either one or two thermal zones.

Testbed fidelity is also important. Real buildings have nonlinear dynamics caused, for example, by longwave radiation and by convection with temperature-dependent film coefficients. They also have significant disturbances from weather and occupant behavior. Solar forcing, in particular, is a strong effect that is nontrivial to accurately model or predict. Simulation testbeds that neglect nonlinearities or oversimplify disturbances could bias case studies in unknown ways.

Existing simulation testbeds tend to be either simple or high fidelity, but not both. Simple testbeds, such as the low-order resistor-capacitor networks often used for MPC, usually neglect nonlinearities or simplify solar forcing. High-fidelity testbeds, such as ENERGYPLUS [25] and TRNSYS [26], usually require significant effort to learn the software, model a building, vary its parameters, apply randomly generated disturbances, simulate a control algorithm, or interpret results.

Developing a simple, high-fidelity simulation testbed is the main purpose and contribution of this article. To maximize simplicity, we restrict our attention to a family of single-zone buildings indexed by a small number of parameters. Varying these parameters generates buildings of varying size, thermal mass, insulation, draftiness, and susceptibility to solar forcing. To maximize fidelity, we model and simulate this family of buildings using the assumptions and methods that underpin state-of-the-art tools such as ENERGYPLUS and TRNSYS. In particular, we include nonlinear thermal radiation exchange between surfaces, nonlinear convection with temperature-dependent film coefficients, wall temperatures governed by partial differential equations, and solar radiation treated via spherical geometry and optical physics.

We call the resulting testbed and supporting functions the `bdg` toolbox (or simply `bdg`). It is implemented in MATLAB, a common environment for control design and analysis, with no links to external software. At its core, `bdg` represents a building as a discrete-time system with nonlinear, time-varying dynamics. The building's thermal behavior is determined by the difference equations

$$x_{k+1} = f_k(x_k, u_k, w_k), \quad k = 0, 1, 2, \dots$$

Here, k indexes discrete time, x_k is the building state (a high-dimensional vector of temperatures of the indoor air and

¹Corresponding author.

Manuscript received July 26, 2020; final manuscript received October 19, 2020; published online November 23, 2020. Assoc. Editor: Monem Beitelmal.

components of the building envelope), u_k is the control input (either a heat flow, or a supply air temperature and mass flow), and w_k is the disturbance (the outdoor air temperature, solar irradiance, and internal heat flows from bodies, lights, and equipment). The main purpose of `bldg` is to provide the dynamics functions f_k . This allows users to decide u_k by any method (e.g., by MPC using a low-order, linear, time-invariant model that approximates the f_k) and then simulate its effect in a high-order, nonlinear, time-varying testbed.

This article is organized as follows. We review related work in Sec. 2. In Sec. 3, we discuss the mathematical model underlying the testbed. We describe the numerical solution scheme in Sec. 4 and the required input data in Sec. 5. We empirically validate some aspects of the testbed in Sec. 6. Algorithm testing examples, including system identification, online state and parameter estimation, and MPC, are presented in Sec. 7. We conclude in Sec. 8.

2 Related Work

The `bldg` toolbox is not a general-purpose building simulator. Although it has a similar mathematical structure to tools such as `ENERGYPLUS` and `TRNSYS`, it does not compete with them. These tools can simulate wide varieties of building geometries, construction types, and mechanical systems. They were designed for diverse purposes: comparing the energy efficiency of architectural designs, predicting the energy savings of building retrofits, sizing heating and cooling equipment, evaluating solar photovoltaic and solar thermal systems, and many more. This design philosophy, while undoubtedly powerful, necessitates a level of complexity that renders these tools opaque to most users. By contrast, `bldg` was designed for the narrow purpose of simple, high-fidelity building control testing. Its scope is one family of single-zone buildings indexed by a small number of parameters. Its intended audience is building control researchers. Its focus is on accessibility to this community.

Two other `MATLAB` toolboxes have been recently developed for the building control community. These are the Building Resistance-Capacitance Modeling (BRCM) [27] and `OPENBUILD` [28] toolboxes. Both are built on the Building Controls Virtual testbed [29] and `MLE+` [30], which enable communication between `MATLAB` and `ENERGYPLUS`. Unlike `bldg`, which was designed for control *testing*, BRCM and `OPENBUILD` were designed primarily for control *synthesis*. BRCM and `OPENBUILD` map `ENERGYPLUS` building models into state space models and facilitate their use in MPC of real buildings. BRCM has been used in MPC of an occupied office building for 7 months [4] and for numerous simulation studies [31–33]. `OPENBUILD` has been used to simulate demand response [34] and ancillary service provision [35] from commercial buildings and to investigate the influence of controller model order on MPC performance [36].

Unlike BRCM, `OPENBUILD` allows controls decided in `MATLAB` to be simulated in `ENERGYPLUS`. Therefore, `OPENBUILD` can be a very high-fidelity, general testbed. In principle, `OPENBUILD` can simulate as complex a building envelope as `ENERGYPLUS` can model. `OPENBUILD` also benefits from the extensive vetting of `ENERGYPLUS` over the last two decades. Although we have validated core aspects of `bldg` empirically (see Sec. 6) and verified others against BESTEST [37] and ASHRAE Standard 140P [38] test cases, these steps are nowhere near as comprehensive as `ENERGYPLUS`' validation.

The main feature that distinguishes `bldg` from `OPENBUILD` is that `bldg` is implemented entirely in `MATLAB` with no connection to `ENERGYPLUS`. This may make `bldg` somewhat easier for control researchers to use; `bldg` models can be created, modified, and simulated from the `MATLAB` command line in a few lines of code. It also has the advantage of transparency. The inner workings of `bldg` can be understood with this article and a familiarity with `MATLAB` differential equation solvers. By contrast, `OPENBUILD`'s connection to `ENERGYPLUS` introduces the complexities described in its engineering reference [39] and input–output reference [40]

and each document contains thousands of pages. Finally, `bldg` has some computational advantages. It requires no middleware or cosimulation. Its governing equations are solved by vectorized `MATLAB` methods that exploit the structure and sparsity. This facilitates fast simulations over long time spans. For example, simulating a default building for 1 year with 15-min time-steps takes about 6 s on a 1.2 GHz Intel Core M processor.

In summary, BRCM and `OPENBUILD` are powerful tools for synthesizing control-oriented building models. The `bldg` toolbox complements them by enabling algorithm evaluation in a transparent, native `MATLAB` testbed. While `OPENBUILD` is another excellent testbed option, `bldg` has some advantages that may be appealing for research such as sensitivity analysis and Monte Carlo simulation. The tools could be used in series, with `bldg` facilitating rapid prototyping and comparison of algorithms, and BRCM or `OPENBUILD` could be used to investigate scalability to large multi-zone buildings.

3 Mathematical Model

3.1 Geometry and Heat Transfer Mechanisms. The buildings simulated by `bldg` are a family of single-zone rooms. They have one single-pane window and walls of uniform composition. The building family is indexed by the dimensions of the room and the material properties of the window and walls. Figure 1 illustrates an example building and its environment.

The building is coupled to the sky and ground by longwave radiation and to the outdoor air by convection and infiltration through gaps in the building envelope. The walls and window receive beam and diffuse sunlight depending on the cloud cover and time of day. The walls and window exchange heat with each other through radiation and with the indoor air through convection. Heat is generated internally by occupants' bodies, lights, heating and cooling systems, and other equipment.

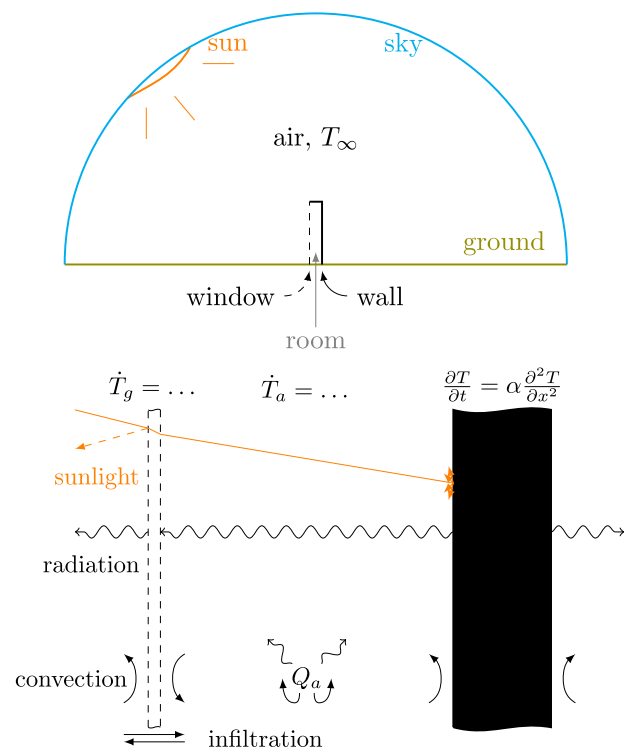


Fig. 1 Geometry of the environment (left) and a cross section of an example building (right). Heat is transferred through convection, radiation, conduction, infiltration, and the transmission, absorption, and reflection of visible light. The internal heat flow Q_a is from bodies, mechanical systems, lights, and other equipment.

3.2 Assumptions. A standard set of assumptions underlies the modeling techniques used by state-of-the-art building simulators. [25] Under these assumptions, all material properties are spatially uniform and independent of temperature. The air in each thermal zone is well mixed and radiatively nonparticipating. Conduction through surfaces is one dimensional. Each surface is modeled as isothermal and, with respect to longwave radiation, as opaque, diffuse, and gray. Optical properties are assumed to vary between the shortwave and longwave bands of the electromagnetic spectrum, but they are independent of wavelength within each band. We make all of the aforementioned assumptions in the remainder of this article. In addition, we assume that the outdoor air is dry, still, and isothermal with the sky and ground. Furthermore, we assume that heat transfer is dominated by flows in the direction normal to the window surface, so that it can be treated as one dimensional.

3.3 Notation. We denote temperature by T (°K), heat flow by Q (W), heat flux by q (W/m²), convection coefficients by h (W/m² · °K), and solar irradiance by I (W/m²). The subscripts g , a , and ∞ denote window glass, indoor air, and the outdoor environment, respectively, far from the building; quantities absent these subscripts refer to the wall opposite the window. The subscripts $-$ and $+$ indicate left and right boundaries given the orientation depicted in Fig. 1. Initial temperatures have the superscript 0. We denote the beam component of solar irradiance by the superscript b and the diffuse component by d ; the subscripts h and $\perp$ indicate incidence on a horizontal surface and a surface normal to the sun. For example, h_{g-} is the convection coefficient at the left glass surface, T_a^0 is the initial air temperature, and I_+^b is the beam irradiance on the right wall surface. For internal heat sources, the subscripts p , c , l , and e indicate people, control systems, lights, and other equipment, respectively.

3.4 Governing Equations. Under the assumptions in Sec. 3.2, the temperature distribution $T(x, t)$ in the wall opposite the window satisfies the heat equation with initial and boundary conditions:

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2}, \quad T(x, 0) = T^0(x) - k \frac{\partial T}{\partial x} \Big|_- = q_-, \quad -k \frac{\partial T}{\partial x} \Big|_+ = q_+ \quad (1)$$

where α (m²/s) and k (W/m · °K) are the wall's thermal diffusivity and conductivity. The left and right surface fluxes q_- and q_+ (W/m²) are discussed in Secs. 3.5 and 3.6. Heat fluxes and flows to the right are taken to be positive.

As demonstrated in Ref. [41], the window temperature dynamics can be accurately modeled by

$$C_g \dot{T}_g = A(q_{g-} - q_{g+}), \quad T_g(0) = T_g^0 \quad (2)$$

where C_g (J/°K) is the thermal capacitance of the window and A (m²) is the window surface area. The fluxes q_{g-} and q_{g+} (W/m² · °K) at the left and right window surfaces, respectively, are discussed in Secs. 3.5 and 3.6.

Under the well-mixed assumption, the indoor air temperature satisfies

$$C_a \dot{T}_a = A(q_{a-} - q_{a+}) + Q_a, \quad T_a(0) = T_a^0 \quad (3)$$

where C_a (J/°K) is the thermal capacitance of the air and any material that's isothermal with it. The fluxes q_{a-} and q_{a+} at the left and right air boundaries, respectively, are discussed in Sec. 3.6. The internal heat flow is expressed as follows:

$$Q_a = \dot{m}_\infty c_a (T_\infty - T_a) + \zeta_c Q_c + \zeta_p Q_p + \zeta_l Q_l + \zeta_e Q_e \quad (4)$$

where $\dot{m}_\infty$ (kg/s) is the mass flowrate of outdoor air entering the space by infiltration, c_a (J/kg · °K) is the specific heat of air at constant pressure, and ζ_c , ζ_p , ζ_l , and ζ_e (W) are the

convective heat flows from control systems, people, lighting, and other equipment, respectively. Each ζ_* is between zero and one; the remaining fraction $1 - \zeta_*$ of Q_* radiates to the wall and the window. The internal heat sources are discussed in Sec. 5.1.

3.5 Outdoor Heat Fluxes. The flux at the outdoor surface of the wall is expressed as follows:

$$q_+ = h_+(T_+ - T_\infty) + \sigma \varepsilon (T_+^4 - T_\infty^4) - \alpha_s (I^d + I_+^b) \quad (5)$$

where $\sigma = 5.67 \times 10^{-8}$ W/m² · °K⁴ is the Stefan–Boltzmann constant, ε is the wall's longwave emissivity, α_s is its shortwave absorptivity, I^d (W/m²) is the (isotropic) diffuse solar irradiance, and I_+^b (W/m²) is the beam component of the solar irradiance incident on the wall. The longwave term in Eq. (5) follows from the fact that the radiative transfer factor between a small, convex, gray object (in this case, the wall's outer surface) and the much larger, isothermal environment that contains it is simply the small object's emissivity; see Sec. 10.4 of Ref. [42] for details.

The flux at the outdoor surface of the window is expressed as follows:

$$q_{g-} = h_{g-}(T_\infty - T_g) + \sigma \varepsilon_g (T_\infty^4 - T_g^4) + \alpha_g(\theta_g) I_{g-}^b + \bar{\alpha}_g I^d \quad (6)$$

where ε_g is the longwave emissivity of the glass and I_{g-}^b (W/m²) is the beam component of the incident sunlight. The diffuse sunlight I^d is assumed to be isotropic, so it equally irradiates the wall and the window. Following Sec. 1.5 of Ref. [43], the absorptivity of beam sunlight $\alpha_g(\theta_g)$ and the absorptivity of diffuse sunlight $\bar{\alpha}_g$ are derived from the beam angle of incidence θ_g and the index of refraction n_g , attenuation coefficient μ_g (m⁻¹), and thickness l_g of the window glass. The derivation uses Snell's law, the Fresnel equation, and Beer's law.

The convection coefficients h_+ and h_{g-} (W/m² · °K) depend nontrivially on the difference between the indoor wall surface temperature (or window, respectively) and the outdoor air temperature T_∞ . The ENERGYPLUS engineering reference [39] catalogs 37 different empirical formulas for convection coefficients at surfaces of various orientations. For simplicity, we use the ASHRAE model for natural convection along a vertical surface,

$$h = 1.31 |\Delta T|^{1/3}$$

Here, ΔT is the temperature difference between the surface and the surrounding air. The dependence of h on ΔT makes this model of convection nonlinear in the temperatures of the indoor air and the wall and window surfaces.

3.6 Indoor Heat Fluxes. The flux at the indoor surface of the wall is expressed as follows:

$$q_- = q_{a+} + q_-^{\text{short}} + q_-^{\text{long}} \quad (7)$$

where

$$q_{a+} = h_-(T_a - T_-)$$

is the convective flux from the indoor air to the wall. Similarly, the indoor window surface flux is

$$q_{g+} = q_{a-} + q_{g+}^{\text{short}} + q_{g+}^{\text{long}} \quad (8)$$

where

$$q_{a-} = h_{g+}(T_g - T_a)$$

is the convective flux from the window to the indoor air. For the indoor surface film coefficients, we use a model for a vertical

surface near a heating/cooling source [44],

$$h = 1.98|\Delta T|^{0.32}$$

The shortwave radiation fluxes q_{-}^{short} and q_{g+}^{short} involve three sources: beam and diffuse sunlight transmitted through the window and artificial light from indoors. They can be computed by ray tracing; under the diffusivity and isotropy assumptions in Sec. 3.2, the results can be obtained in a closed form by summing geometric series.

The longwave radiation fluxes q_{-}^{long} and q_{g+}^{long} involve two effects: gray body radiation exchange between the window wall surfaces, and the radiative components of indoor heat flows from bodies, control systems, lights, and other equipment. The longwave fluxes from both effects can be computed using the radiosity method [45]. For a generic enclosure with n surfaces, this involves computing the n^2 view factors between the surfaces and then numerically solving a system of n linear equations. For our simple case with two dominant indoor surfaces and view factors close to one, the fluxes can be computed in a closed form.

4 Numerical Solution

Simulating the building requires solving the governing equations (1), (2), and (3). The complexity of the boundary conditions and disturbances makes analytical solution intractable, so we use a finite difference approach called the numerical method of lines. [46–48] The method of lines involves approximating spatial derivatives with algebraic expressions, reducing the PDE (1) to a system of ODEs that can be combined with Eqs. (2) and (3) and then fed to an ODE solver. This is accomplished through the central function in the `bldg` toolbox, `bsim`.

4.1 Space Discretization. We define the space coordinate x , such that $x=0$ at the left-hand side of the wall and $x=l$ at the right. To discretize space, we introduce N nodes on $[0, l]$, uniformly spaced by $\Delta x = l/(N-1)$ and indexed by $i \in \{1, \dots, N\}$ such that $x_i = (i-1)\Delta x$. We approximate each temperature $T(x_i, t)$ by a function $T_i(t)$, chosen such that $T_i(t) \rightarrow T(x_i, t)$ as $N \rightarrow \infty$.

The central difference approximation to the spatial derivative in the heat equation, evaluated at x_i , is given as follows:

$$\frac{\partial}{\partial t} T(x_i, t) = \alpha \left(\frac{T(x_{i-1}, t) - 2T(x_i, t) + T(x_{i+1}, t))}{(\Delta x)^2} \right) + \mathcal{O}((\Delta x)^2)$$

Dropping the truncation error term and replacing the true temperatures by their approximations gives

$$\dot{T}_i = r(T_{i-1} - 2T_i + T_{i+1}) \quad (9)$$

where we have suppressed the time arguments and defined $r = \alpha/(\Delta x)^2$.

Evaluating Eq. (9) at each $i \in \{1, \dots, N\}$ defines N ODEs in the $N+2$ temperatures $T_0, \dots, T_{N+1}$. Since $x_0 = -\Delta x$ and $x_{N+1} = l + \Delta x$, however, T_0 and T_{N+1} represent temperatures outside of the wall. They can be eliminated using the boundary conditions

$$-k \frac{\partial T}{\partial x} \Big|_{0,t} = q_{-}, \quad -k \frac{\partial T}{\partial x} \Big|_{l,t} = q_{+}$$

To preserve the $\mathcal{O}((\Delta x)^2)$ truncation error of the method, we replace the spatial derivatives by their central difference approximations,

$$\begin{aligned} \frac{\partial T}{\partial x} \Big|_{0,t} &= \frac{T(x_2, t) - T(x_0, t)}{2\Delta x} + \mathcal{O}((\Delta x)^2) \\ \frac{\partial T}{\partial x} \Big|_{l,t} &= \frac{T(x_{N+1}, t) - T(x_{N-1}, t)}{2\Delta x} + \mathcal{O}((\Delta x)^2) \end{aligned}$$

Dropping the truncation errors and replacing true by approximate temperatures, we have

$$\begin{aligned} T_0 &= T_2 + \left(\frac{2\Delta x}{k} \right) q_{-} \\ T_{N+1} &= T_{N-1} - \left(\frac{2\Delta x}{k} \right) q_{+} \end{aligned}$$

Substituting these expressions into the approximate heat equation at x_1 and x_N gives

$$\begin{aligned} \dot{T}_1 &= 2r(-T_1 + T_2) + \left(\frac{2\alpha}{k\Delta x} \right) q_{-} \\ \dot{T}_N &= 2r(T_{N-1} - T_N) - \left(\frac{2\alpha}{k\Delta x} \right) q_{+} \end{aligned} \quad (10)$$

4.2 Initial Value Problem. Equations (10), along with Eq. (9) evaluated at each $i \in \{2, \dots, N-1\}$, defines a system of N ODEs in $T_1, \dots, T_N$. These ODEs involve the internal surface flux q_{-} , which couples the system to Eqs. (2) and (3) for T_g and T_a . The full problem is therefore

$$\begin{bmatrix} \dot{T}_1 \\ \dot{T}_2 \\ \vdots \\ \dot{T}_{N-1} \\ \dot{T}_N \\ \dot{T}_g \\ \dot{T}_a \end{bmatrix} = \begin{bmatrix} 2r(-T_1 + T_2) + 2\alpha q_{-}/k\Delta x \\ r(T_1 - 2T_2 + T_3) \\ \vdots \\ r(T_{N-2} - 2T_{N-1} + T_N) \\ 2r(T_{N-1} - T_N) - 2\alpha q_{+}/k\Delta x \\ A(q_{g-} - q_{g+})/C_g \\ (A(q_{a-} - q_{a+}) + Q_a)/C_a \end{bmatrix} \quad (11)$$

with the corresponding initial condition. The fluxes q_{-} , q_{+} , q_{g-} , q_{g+} , q_{a-} , and q_{a+} , defined in Secs. 3.5 and 3.6, are polynomials of degree up to four in the temperatures, so system (11) is nonlinear. The solar fluxes depend on the sun's position in the sky, so the system is also time varying.

Depending on the physical parameters, control scheme, and choice of N , system (11) may be stiff. An implicit solver should therefore be used [46]. In Ref. [49], Crowley compared 15 numerical methods for a similar problem and recommended combining the trapezoidal rule and the second-order backward difference formula, as implemented in the MATLAB solver `ode23tb`. This scheme has second-order accuracy in both time and space. It is also possible to simulate perfect control, where the indoor temperature is maintained exactly at the setpoint. In this case, the last ODE reduces to the algebraic equation $A(q_{a-} - q_{a+}) + Q_a = 0$, and problem (11) becomes a system of differential-algebraic equations that can be solved, e.g., by MATLAB's `ode15s` solver.

Solving system (11) numerically requires specifying the initial temperatures in the wall, window, and air. Reasonable initial conditions can be produced by preconditioning the building as follows. For a simulation starting at time t_0 , the building state can be initialized isothermal with the environment some time earlier, for example, 1 or 2 weeks, then integrated forward to t_0 under the appropriate weather conditions and internal heat sources.

5 Input Data

Simulating the building requires specifying the model parameters, an initial state, the solar time span, and values of the input signals in Table 1 at every time-step. The input signals can be divided into internal heat sources (Sec. 5.1) and weather (Sec. 5.2). We discuss the model parameters in Sec. 5.3.

5.1 Internal Heat Sources. The internal sources are the heat flows Q_c , Q_p , Q_l , and Q_e from control systems, people, lighting, and other equipment. Building simulators typically require

Table 1 The input signals to the building model include the internal heat flows and weather

Signal	Symbol	Unit
Control heat flow	Q_c	W
Heat flow from people	Q_p	W
Heat flow from lighting	Q_l	W
Heat flow from equipment	Q_e	W
Outdoor air temperature	T_∞	$^{\circ}\text{K}$
Total horizontal irradiance	I_h	W/m^2
Beam normal irradiance	$I_\perp^b$	W/m^2

user-specified schedules for occupancy, lighting, and equipment, from which Q_p , Q_l , and Q_e are determined.

In general, the control heat flow Q_c may depend on the control objectives, the building state, and predictions of weather and internal heat sources over the control horizon. We assume that Q_c is either provided by the user or computed internally to perfectly regulate T_a at a specified setpoint. In the latter case, $\dot{T}_a = 0$, and Eqs. (3) and (4) give the heating load of the building:

$$Q_c = \frac{1}{\zeta_c} (A(q_{a+} - q_{a-}) - (\dot{m}_\infty c_a (T_\infty - T_a) + \zeta_p Q_p + \zeta_l Q_l + \zeta_e Q_e)) \quad (12)$$

Perfect regulation gives useful estimates of peak heating and cooling loads.

5.2 Weather. We take as basic weather inputs only quantities that are easily measured: the outdoor air temperature T_∞ , the total solar irradiance I_h on a horizontal surface (measured with a pyranometer), and the beam solar irradiance $I_\perp^b$ on a surface normal to the sun (measured with a pyrheliometer). These data are also available in typical meteorological year files for various locations [50]. As discussed in Sec. 1.6 of Ref. [51], the beam angles of incidence on the wall and window, θ and θ_g , can be computed from the day, time, latitude, longitude, and building orientation. Similarly, the diffuse irradiance I^d and the beam components I_+^b and I_{g-}^b can be inferred from I_h and $I_\perp^b$.

5.3 Parameters. The building is defined by the 21 parameters in Table 2; all other parameters are derived from them. Six of these

Table 2 The input parameters to the building model include the building geometry and material properties

Parameter	Units
Wall thickness, l	m
Glass thickness, l_g	m
Room width, l_a	m
Wall surface area, A	m^2
Wall azimuth angle, γ	rad
Latitude, ϕ	rad
Wall thermal diffusivity, α	m^2/s
Wall thermal conductivity, k	$\text{W/m} \cdot ^{\circ}\text{K}$
Wall longwave emissivity, ϵ	—
Wall shortwave absorptivity, α_s	—
Glass thermal capacitance, C_g	$\text{J}/^{\circ}\text{K}$
Glass longwave emissivity, ϵ_g	—
Glass index of refraction, n_g	—
Glass attenuation coefficient, μ_g	m^{-1}
Room thermal capacitance, C_a	$\text{J}/^{\circ}\text{K}$
Infiltration mass flowrate, $\dot{m}_\infty$	kg/s
Control convective fraction, ζ_c	—
People convective fraction, ζ_p	—
Lighting convective fraction, ζ_l	—
Equipment convective fraction, ζ_e	—
Lighting efficiency, η	—

parameters define the orientation and dimensions of the building. Nine are material properties: four for the walls, four for the window glass, and one for the room. The remaining six parameters are associated with the internal heat sources.

6 Empirical Validation

In this section, we empirically validate the model and simulator through a series of frequency response experiments. In each experiment, known input signals are sent to (1) a scaled laboratory testbed and (2) bldg with parameters reflecting the laboratory testbed's geometry and material properties. Temperature responses within the testbed are measured and compared to the corresponding simulator outputs. The experiments validate the numerical solution scheme and the conduction, convection, and longwave radiation models.

The empirical testbed is an enclosure with five plywood surfaces and one clear acrylic window. The side, top, and bottom surfaces are insulated with rigid polystyrene and radiation shielded with low-emissivity aluminum foil. This directs heat transfer primarily through the window and the wall opposite. Joints are tightly sealed, and the ambient airflow into the box is negligible.

The bldg parameters are determined as follows. The lengths l , l_g , and l_a and area A are measured directly. The material properties α , k , ϵ , ϵ_g , and C_g are drawn from engineering tables. Nominally, the indoor air's thermal capacitance C_a is the product of its volume, density, and specific heat at constant pressure. As discussed in the section titled *Zone Sensible Heat Capacity Multiplier* of the ENERGY-PLUS engineering reference [39], it is a common practice to increase C_a over this nominal value to account for unmodeled thermal mass that is nearly isothermal with the air. In keeping with this practice, we increase C_a by an order of magnitude.

Controlled heat flows are supplied to the box through two resistive heating elements, each thermally glued to an aluminum heat sink. Temperatures are measured by 15 thermistors accurate to $\pm 0.1^{\circ}\text{C}$. Twelve thermistors are arranged in two lines normal to the window surface, with one line of six at one-quarter of the wall height and another at three quarters. Each line measures the inner and outer window surface temperature, the indoor air temperature, the inner and outer wall surface temperature, and the temperature at the midpoint of the wall. The remaining three thermistors measure the ambient temperature outside the box at different locations. Control and measurement signals are exchanged between the

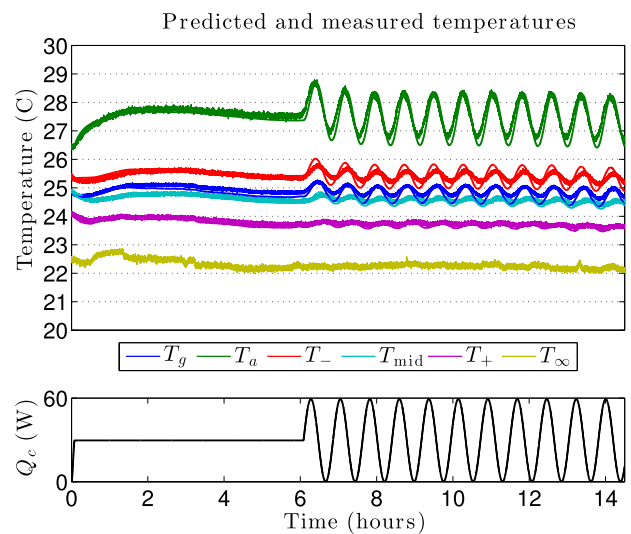


Fig. 2 The temperatures (top plot) and control heat flow (bottom) in an experiment with oscillation period of 3.5 h. Thick, fuzzy lines are measurements. Thin lines are predictions. Nine similar experiments were conducted to build the frequency response plots in Fig. 3.

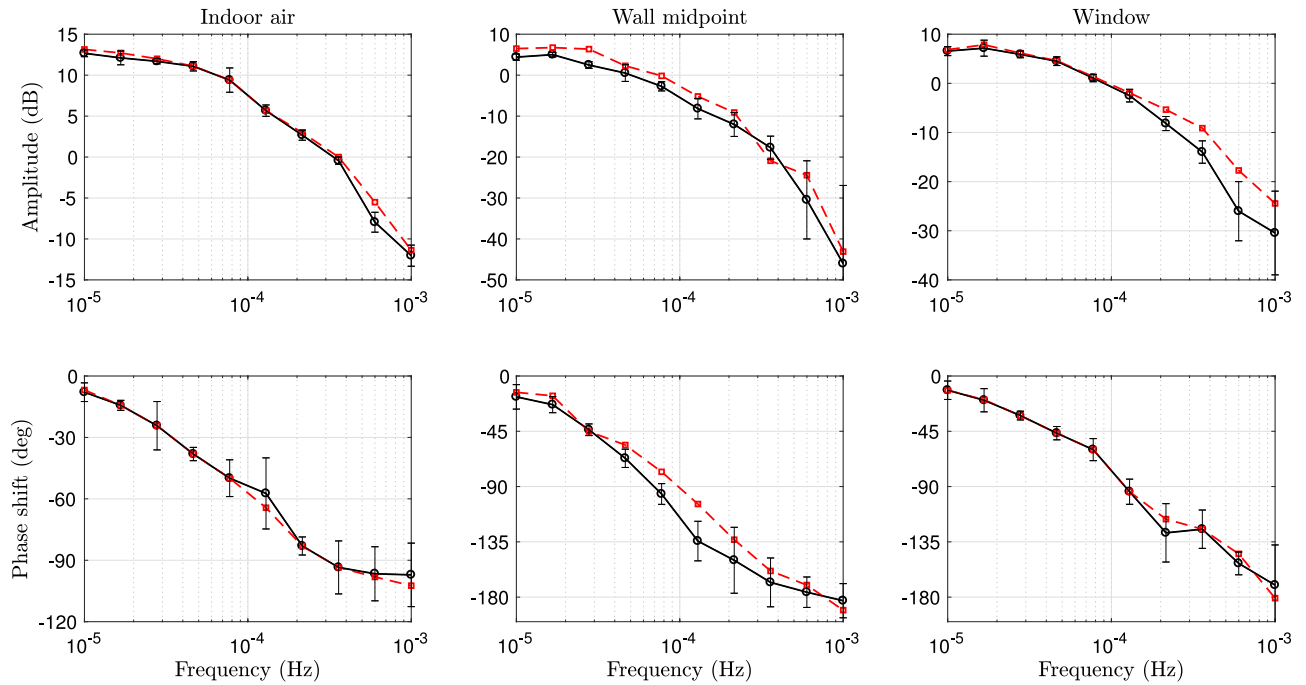


Fig. 3 Predicted (dashed) and measured (solid) frequency responses. The temperature amplitudes (top row) and phase shifts (bottom row) agree closely with measurements over the 31 days of total experiment time.

testbed and a laboratory computer running MATLAB and a National Instruments sensor toolbox.

Figure 2 shows the inputs and outputs of one experiment. Each experiment involves a 6-h warm-up phase: the heating system is turned on to half capacity and the box temperatures settle into a steady state. After the warm-up phase, the internal heat flow is varied sinusoidally between zero and full capacity for ten oscillations. Ten such experiments are conducted. Oscillation periods vary from 17 min (a 9-h experiment, including the 6 warm-up hours and ten oscillations) to 27 h (an 11-day experiment). The total experiment runtime, including ten oscillations at each of the ten frequencies, is about 31 days.

Figure 3 summarizes the results of the ten frequency response experiments. The top row of plots shows the amplitudes of the temperature waveforms, in units of decibels, caused by control heat flow sinusoids of varying frequencies. The bottom row of plots shows the phase shift between the temperature and heat flow waveforms. Each error bar shows one standard deviation from the mean of ten oscillations at each frequency in the experimental data. The model accurately predicts the full temperature waveforms (both amplitudes and phases) caused by oscillating heat flows at most frequencies. This empirical validation supports both the mathematical model and the numerical solution scheme.

7 Examples

In this section, we demonstrate the use of the `bldg` toolbox through the examples of system identification (Sec. 7.1), online state and parameter estimation (Sec. 7.2), and model predictive control (Sec. 7.3). Along the way, we highlight several open research questions that `bldg` could help investigate. Further examples are provided in the `bldg` documentation [52].

The context of the examples in this section is the building described in Sec. 3 during winter in New York. A building model with default parameter values can be created by instantiating an object of class `bldg`: `b = bldg`;

Under the default parameter values, the building is thermally massive and highly susceptible to solar forcing. The user can modify the building by adjusting parameters in the `bldg` object `b`. For example, the thermal mass can be decreased by increasing

the wall diffusivity α (`b.a`) or thickness l (`b.l`). The solar forcing can be weakened by decreasing the wall shortwave absorptivity α_s (`b.as`) or by changing the window orientation through the azimuth angle γ (`b.gam`).

The following code imports and interpolates a year of hourly weather data from a TMY3 file for New York, generates plausible internal gains for a home, and defines a 24-h simulation time span with step $\Delta t = 15$ min, starting at midnight ($t = 0$) on February 7 (day number $n_d = 38$). (The `bldg` toolbox uses SI units throughout, so time is given in seconds.) The last line packs the disturbances into the matrix form accepted by the building simulation function `bsim`.

```
dt = 15*60; [b, weather] = importWeather(b, 'NYC_TMY3.csv', dt);
gains = generateGains(b, weather.tw); nd = 38; t0 = 0; tf = 24*3600;
t = getTiming(weather.tw, nd, t0, tf); W = getDisturbances(weather,
gains, t);
```

Figure 4 shows the input signals generated by this code.

Simulating the building requires an initial state, which can be difficult to produce *a priori*. One is easily generated, however, using `x0 = precondition(b, t, weather, gains, N, Ts)`;

As discussed in Sec. 4.2, the `precondition` function starts the building isothermal with the outdoor air 2 weeks before t_0 . It then integrates the building forward with the indoor air temperature perfectly regulated at setpoint T_s . The building can then be simulated by `[X, Qc] = bsim(b, t, W, x0)`;

With $N = 50$ wall nodes, this 24-h simulation takes about a tenth of a second to run on a 2 GHz Intel Core 2 Duo processor. Figure 5 shows the output.

7.1 System Identification. In this example, we consider the problem of learning a low-order, linear model of the building dynamics from measurements of the indoor air temperature and weather signals. We specify the first-order ARX structure

$$T_a^{k+1} = \beta_1 T_a^k + \beta_2 Q_c^k + \beta_3 T_\infty^k + \beta_4 I_h^k + w_a^k \quad (13)$$

where the w_a^k are independently, identically $\mathcal{N}(0, \sigma_w^2)$ distributed. This model is naive since it neglects the internal gains and the dynamics of the building envelope and since the true solar

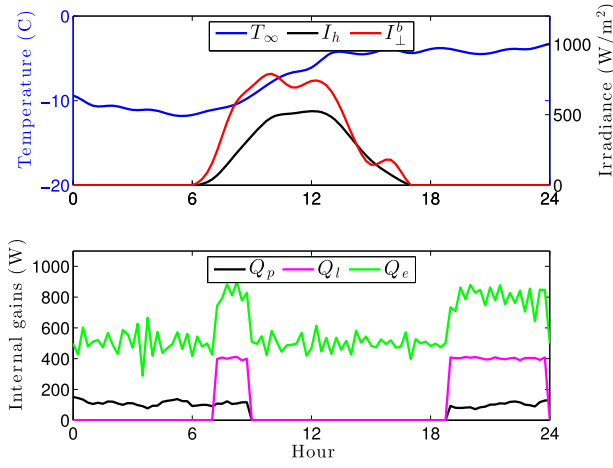


Fig. 4 Input signals from weather (top) and internal heat sources (bottom). The outdoor air temperature (T_∞), total solar irradiance on a horizontal surface (I_h), and beam solar irradiance on a surface normal to the sun ($I_\perp^b$) show that the simulation day is cold and partly cloudy. The internal heat sources model a home on a weekday, with decreased heat flows from people (Q_p), lights (Q_l), and equipment (Q_e) overnight and during work hours.

forcing is time varying and cannot be determined from I_h alone. Nevertheless, the model is sufficiently accurate to give fair controller performance; see Sec. 7.3.

To fit the model, we simulate the (nonlinear, time varying, high dimensional) dynamics in *bsim* for the last 3 weeks of January under a sequence of pseudorandom binary control inputs. We assume perfect knowledge of Q_c , T_∞ , and I_h , but corrupt T_a with zero-mean, white, Gaussian noise with standard deviation $\sigma_v = 1/6$ °C. The model parameters β and σ_w are estimated using linear regression.

Figure 6 shows the one-step prediction errors during the three training weeks and in two subsequent test weeks under thermostatic control with deadband [18 °C, 22 °C]. The maximum absolute error of about 1.5 °C is large. A higher order ARX, ARMAX, or RC network model could achieve better predictions at the cost of introducing more states and parameters. This accuracy/complexity trade-off was explored for a family of RC networks in Ref. [8]. The authors found that a second-order RC network with five parameters

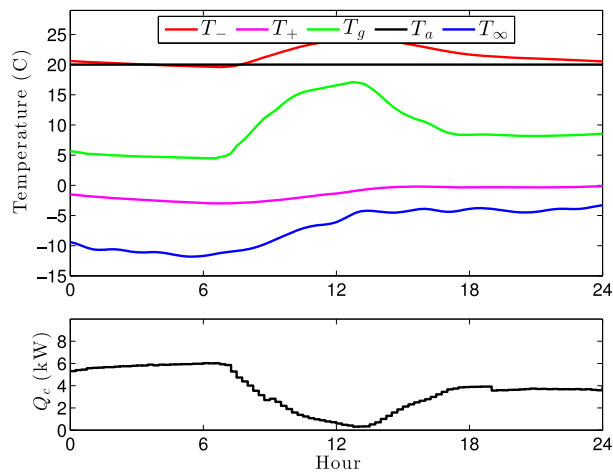


Fig. 5 Temperatures of the air (T_a), glass (T_g), left and right wall surfaces (T_- and T_+), and outdoor environment (T_∞). The bottom plot shows the heat flow required to perfectly regulate T_a at set-point $T_s = 20$ °C under the input signals in Fig. 4.

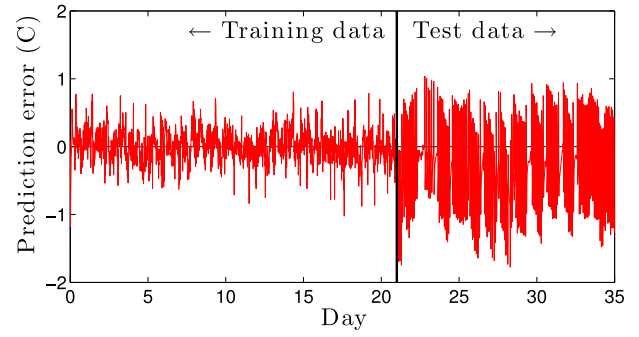


Fig. 6 One-step prediction errors in the training and test data with the static model in Sec. 7.1. The model fit is poor: The errors are autocorrelated and, in the test set, predictions are biased by -0.31 °C.

is able to capture the essential building behavior (confirming a result from the 1980s; see Refs. [53,54]). Less is known about accuracy/complexity trade-offs for buildings with thermally massive construction or strong solar forcing. These questions could be explored in *bsim* by varying physical parameters such as the wall diffusivity α (b.a) and shortwave absorptivity α_s (b.as) and studying the model fit.

Another interesting topic is the relationship between gray- and black-box models. The parameters in model (13), for example, can be interpreted in terms of the first-order RC network

$$C_{\text{eff}} \dot{T}_a = \frac{T_\infty - T_a}{R_{\text{eff}}} + Q_c + A_{\text{eff}} I_h$$

where C_{eff} , R_{eff} , and A_{eff} are the effective thermal capacitance, thermal resistance, and solar absorption area. Forward Euler discretization gives

$$T_a^{k+1} = \left(1 - \frac{\Delta t}{R_{\text{eff}} C_{\text{eff}}}\right) T_a^k + \frac{\Delta t}{C_{\text{eff}}} Q_c^k + \frac{\Delta t}{R_{\text{eff}} C_{\text{eff}}} T_\infty^k + \frac{\Delta t A_{\text{eff}}}{C_{\text{eff}}} I_h^k$$

This is consistent with the ARX model (13) only if $\beta_1 + \beta_3 = 1$. In this case, the “cost of grayness” could be defined as the prediction accuracy lost by imposing the constraint $\beta_1 + \beta_3 = 1$ on the least-squares fitting problem, along with any resulting controller performance reduction. While several authors have argued that the RC network structure has benefits—e.g., it can provide initial guesses to estimation algorithms and sanity checks on their output—to our knowledge, the cost of grayness has not been well studied.

7.2 Online Estimation. Building dynamics are naturally time varying: Solar forcing depends on the season and time of day, infiltration rates change as occupants open and close windows and doors, temperature control equipment differs in heating and cooling seasons, and material properties change as materials age. We therefore expect model parameters to change over several time scales. The framework of online estimation allows parameters to be continuously calibrated to measurements, enabling adaptive control. Online estimation of building model parameters has been studied in Refs. [9–13].

In this example, we consider the problem of adapting a subset of the parameters of the ARX model identified in Sec. 7.1. The goal is to reduce the prediction bias apparent in Fig. 6. We simultaneously estimate the indoor air temperature and model parameters using an unscented Kalman filter [55]. We allow the parameters β_1 and β_3 , initialized with the fit from Sec. 7.1, to vary under the random walk model $\beta_i^{k+1} = \beta_i^k + w_{\beta_i}^k$. This gives the augmented system

model

$$\begin{aligned} x^{k+1} &= f(x^k, u^k) + w^k \\ y^k &= x_1^k + v^k \end{aligned} \quad (14)$$

where $x = (T_a, \beta_1, \beta_3)$, $u = (Q_c, T_{\infty}, I_h)$, $w = (w_a, w_{\beta_1}, w_{\beta_3})$, $f_1(x, u) = x_2 x_1 + \beta_2 u_1 + x_3 u_2 + \beta_4 u_3$, and $f_i(x, u) = x_i$ for $i = 2, 3$. We model the disturbance w as zero-mean, white, and Gaussian. The RC network analogy developed in Sec. 7.1 suggests that the parameters β_1 and β_3 should (roughly) sum to one, so we specify a strong negative correlation between w_{β_1} and w_{β_3} .

Figure 7 shows histograms of the one-step prediction errors in the test data with and without parameter adaptation. Adaptation nearly eliminates the prediction bias. The unscented Kalman filter accomplishes this by decreasing β_1 and increasing β_3 over the course of about 12 h, as shown in Fig. 8. The parameters stabilize after the initial adjustment. This is expected since the underlying physical model remains nearly constant over the estimation period. It is less clear how the filter would respond to a large, sudden change in the underlying system, e.g., a window being opened or a mechanical component failing. These questions, which lie in the domain of fault detection, could be explored in *bsim* by perturbing the mass flowrate of infiltration air, $\dot{m}_{\infty}$ (b.mdot), or the fraction of the control heat that convects to the indoor air, ζ_c (b.zc).

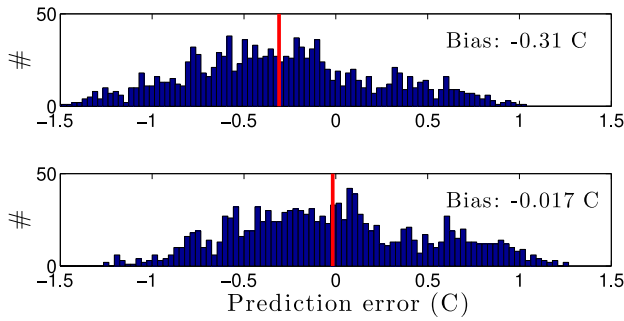


Fig. 7 Histograms of the one-step prediction errors without parameter adaptation (top) and with it (bottom). The online estimation algorithm discussed in Sec. 7.2 nearly eliminates the prediction bias. This is accomplished by adjusting the model parameters β_1 and β_3 , as shown in Fig. 8.

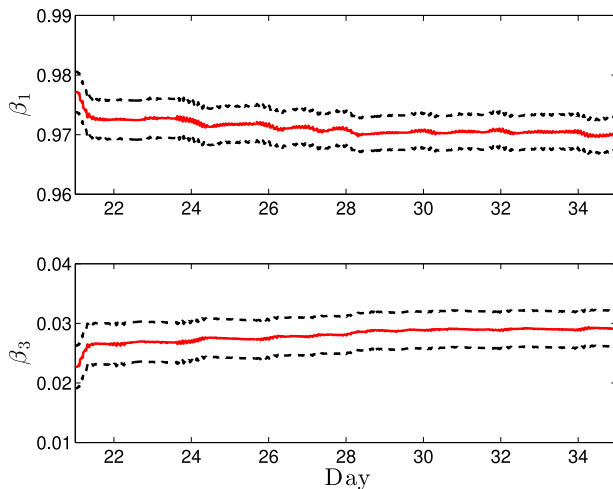


Fig. 8 Unscented Kalman filter parameter estimates and 95% confidence intervals during the two test weeks. The filter remains stable after initially adjusting β_1 and β_3 to reduce prediction bias.

7.3 Model Predictive Control. This example involves efficiently heating a building. We consider the stochastic optimal control problem

$$\begin{aligned} \text{minimize} \quad & \mathbf{E} \Delta t \sum_{k=0}^M Q_c^k \\ \text{subject to} \quad & T_a^{k+1} = (\text{nonlinear bsim dynamics}) \\ & y^k = T_a^k + v^k \\ & T_a^{k+1} \geq T^{\min} \\ & Q_c^k = \mu^k(y^0, \dots, y^k) \in [0, Q_c^{\max}] \end{aligned} \quad (15)$$

where the constraints should hold for each $k = 0, \dots, M$ almost surely. The optimization variable is the control policy $(\mu^0, \dots, \mu^M)$, where $\mu^k: \mathbf{R}^{k+1} \rightarrow \mathbf{R}$ maps observations into controls. The expectation is taken with respect to the joint distribution of the disturbance and noise sequences and the initial state.

Problem (15) is analytically intractable due to the nonlinear dynamics, imperfect state information, and optimization over infinite-dimensional objects (the functions μ^k). While it is not difficult to generate good approximate solutions—a well-tuned thermostat works—we apply two variants of MPC to problem (15) to illustrate the research value of the *bldg* toolbox. The first MPC variant uses the ARX model from Sec. 7.1 with no parameter adaptation. At each time-step, we estimate the temperature using the linear Kalman filter and then solve a truncated, certainty-equivalent version of problem (15) with horizon $H = 6$ h. Each MPC subproblem is a deterministic linear program that generates a planned control trajectory, of which the first control is implemented. We then allow the system to evolve according to the nonlinear, time-varying dynamics in *bsim* and repeat the process.

The second MPC variant is identical, except that state and parameter estimates are simultaneously updated at each time-step using the unscented Kalman filter from Sec. 7.2. Simulating the building under MPC for one day with 15-min time-steps takes about 42 s, with linear programs solved by Gurobi on a 2 GHz Intel Core 2 Duo processor. About 93% of that time is spent in optimization, 6% in *bsim*, and 1% in the unscented Kalman filter.

Figure 9 shows the indoor air temperatures and control heat flows under both MPC variants. The simulation takes place on February 7

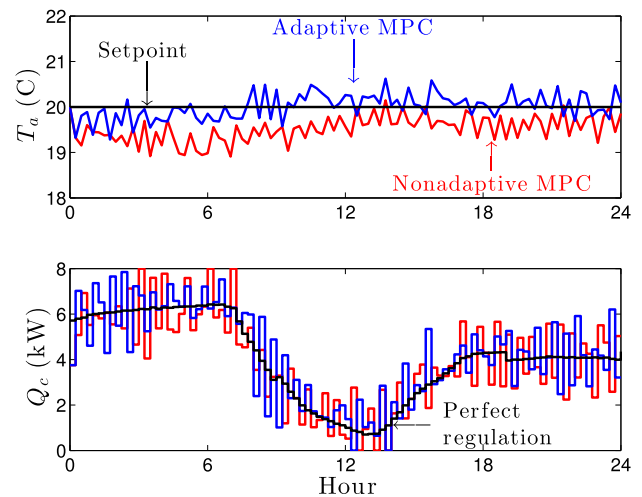


Fig. 9 Indoor air temperature T_a (top) and control heat flow Q_c (bottom) under MPC, with and without parameter adaptation. Both controllers attempt to regulate T_a just above the minimum permissible temperature of 20 °C, giving heat flows that resemble the “perfect regulation” case computed by *bsim*. Both MPC variants frequently underheat due to model error. Parameter adaptation mitigates, but does not eliminate constraint violation.

under the exogenous input signals shown in Fig. 4. Both MPC variants attempt to regulate T_a at the minimum feasible temperature of $T^{\min} = 20^\circ\text{C}$. This generates control trajectories that resemble noisy versions of the “perfect regulation” case discussed in Sec. 5. Due to model error, both MPC variants frequently allow T_a to drop below $T^{\min}$. Adaptive MPC performs somewhat better due to its reduced prediction bias, achieving a time-averaged constraint violation of 0.11°C , compared to 0.48°C for the nonadaptive case.

The constraint violations are an artefact of model mismatch between the controller and the underlying `bsim` dynamics. They could be reduced by using a more accurate, higher-order model, by specifying a safety margin (e.g., by replacing the constraint $T_a^k \geq T^{\min}$ with $T_a^k \geq T^{\min} + \delta$ for some positive δ) or by moving the MPC optimization to a robust or stochastic framework. While several studies have applied robust or stochastic MPC to buildings [19–24], it is not clear how these approaches, which add significant conceptual and computational complexity, compare to using a more accurate model or specifying safety margins.

8 Conclusion

In this article, we developed a computationally efficient, transparent, and high-fidelity testbed for building climate control algorithms. After reviewing the related work, we developed a nonlinear building model and a numerical solution scheme. The model has a small number of governing equations, parameters, and input signals. The numerical solution scheme uses built-in MATLAB differential equation solvers that are familiar to many control engineers. We hope that these properties make the `bldg` toolbox accessible and transparent to building control researchers.

The `bldg` toolbox is also sufficiently high fidelity to be useful for prototyping, analyzing, and comparing building estimation and control algorithms. We supported this claim through empirical validation and the examples of system identification, online state and parameter estimation, and model predictive control. We highlighted several open research questions that the `bldg` toolbox could help investigate.

The scope of the `bldg` toolbox is intentionally narrow. It can simulate one family of single-zone buildings indexed by a small number of parameters. This simplicity has conceptual and computational advantages as evidenced by the large number of high-quality studies done in one- or two-zone testbeds. However, `bldg`’s narrow scope prevents its use for some important research, such as investigating scalability to large multizone buildings. For such research, we recommend the BRCM [27] and OPENBUILD [28] toolboxes.

The `bldg` toolbox is free, open source, and available online [52]. It is intended to eliminate the need to model a building or leave the MATLAB environment before simulating an estimation or control algorithm in a high-fidelity testbed. We hope that this facilitates rapid, reliable research into advanced building controls, accelerating progress toward realizing their environmental and economic benefits at scale.

Acknowledgment

The authors acknowledge the support from the National Science Foundation (NSF) (Grant No. 1711546).

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request. The data and information that support the findings of this article are freely available at: <https://github.com/kevinjkircher/bldg>

References

- [1] Siroky, J., Oldewurtel, F., Cigler, J., and Privara, S., 2011, “Experimental Analysis of Model Predictive Control for an Energy Efficient Building Heating System,” *Appl. Energy*, **88**(9), pp. 3079–3087.
- [2] Schuss, M., Zach, R., Orehounig, K., and Mahdavi, A., 2011, “Empirical Evaluation of a Predictive Simulation-Based Control Method,” Proceedings of the 12th International IBPSA Conference, Sydney, Australia, pp. 918–925.
- [3] Benga, S., Kelman, A., Borrelli, F., Taylor, R., and Narayanan, S., 2012, “Model Predictive Control for Mid-Size Commercial Building HVAC: Implementation, Results and Energy Savings,” International Conference on Building Energy and Environment, Boulder, CO.
- [4] Sturzenegger, D., Gyalistras, D., Morari, M., and Smith, R., 2016, “Model Predictive Climate Control of a Swiss Office Building: Implementation, Results, and Cost-Benefit Analysis,” *IEEE Transactions on Control Systems Technology*, **24**(1), pp. 1–12.
- [5] Ferreira, P. M., Ruano, A. E., Silva, S., and Conceicao, E. Z. E., 2012, “Neural Networks Based Predictive Control for Thermal Comfort and Energy Savings in Public Buildings,” *Energy Build.*, **55**, pp. 238–251.
- [6] Huang, H., Chen, L., and Hu, E., 2015, “A New Model Predictive Control Scheme for Energy and Cost Savings in Commercial Buildings: An Airport Terminal Building Case Study,” *Building Environ.*, **89**, pp. 203–216.
- [7] Privara, S., Cigler, J., Vana, Z., Oldewurtel, F., and Sagerschnig, C., 2013, “Building Modeling as a Crucial Part for Building Predictive Control,” *Energy Build.*, **56**, pp. 8–22.
- [8] Lin, Y., Middelkoop, T., and Barooah, P., 2012, “Issues in Identification of Control-Oriented Thermal Models of Zones in Multi-Zone Buildings,” IEEE Conference on Decision and Control, Maui, HI.
- [9] Radecki, P., and Hencsey, B., 2012, “Online Building Thermal Parameter Estimation Via Unscented Kalman Filtering,” American Control Conference, Montreal, QC, Canada, pp. 3056–3062.
- [10] Radecki, P., and Hencsey, B., 2013, “Online Thermal Estimation, Control, and Self-Excitation of Buildings,” IEEE Conference on Decision and Control, Florence, Italy, pp. 4802–4807.
- [11] Fux, S., Ashouri, A., Benz, M., and Guzzella, L., 2014, “EKF Based Self-Adaptive Thermal Model for a Passive House,” *Energy Build.*, **68**, pp. 811–817.
- [12] Maasoumy, M., Moridian, B., Razmara, M., Shahbakhti, M., and Sangiovanni-Vincentelli, A., 2013, “Online Simultaneous State Estimation and Parameter Adaptation for Building Predictive Control,” ASME 2013 Dynamic Systems and Control Conference, Palo Alto, CA.
- [13] Martincevic, A., Starcic, A., and Vasak, M., 2014, “Parameter Estimation for Low-Order Models of Complex Buildings,” IEEE Innovative Smart Grid Technologies Conference Europe, Istanbul, Turkey, pp. 1–6.
- [14] Liao, Z., and Dexter, A. L., 2010, “An Inferential Model-Based Predictive Control Scheme for Optimizing the Operation of Boilers in Building Space-Heating Systems,” *IEEE Trans. Control Syst. Technol.*, **18**(5), pp. 1092–1102.
- [15] Aswani, A., Master, N., Taneja, J., Culler, D., and Tomlin, C., 2011, “Reducing Transient and Steady State Electricity Consumption in HVAC Using Learning-Based Model-Predictive Control,” *Proc. IEEE*, **100**(1), pp. 240–253.
- [16] Ma, Y., Borrelli, F., Hencsey, B., Coffey, B., Benga, S., and Haves, P., 2012, “Model Predictive Control for the Operation of Building Cooling Systems,” *IEEE Trans. Control Syst. Technol.*, **20**(3), pp. 796–803.
- [17] Castilla, M., Alvarez, J. D., Normey-Rico, J. E., and Rodriguez, F., 2014, “Thermal Comfort Control Using a Non-Linear MPC Strategy: A Real Case of Study in a Bioclimatic Building,” *J. Process. Control*, **24**(6), pp. 703–713.
- [18] Kircher, K. J., and Zhang, K. M., 2015, “Model Predictive Control of Thermal Storage for Demand Response,” American Control Conference (ACC), pp. 956–961.
- [19] Maasoumy, M., Razmara, M., Shahbakhti, M., and Sangiovanni-Vincentelli, A., 2014, “Handling Model Uncertainty in Model Predictive Control for Energy Efficient Buildings,” *Energy Build.*, **77**, pp. 377–392.
- [20] Tanaskovic, M., Sturzenegger, D., Smith, R., and Morari, M., 2017, “Robust Adaptive Model Predictive Building Climate Control,” IFAC-PapersOnLine, **50**(1), pp. 1871–1876.
- [21] Oldewurtel, F., Parisio, A., Jones, C., Gyalistras, D., Gwerder, M., Stauch, V., Lehmann, B., and Morari, M., 2012, “Use of Model Predictive Control and Weather Forecasts for Energy Efficient Building Climate Control,” *Energy Build.*, **45**, pp. 15–27.
- [22] Oldewurtel, F., Jones, C., Parisio, A., and Morari, M., 2014, “Stochastic Model Predictive Control for Building Climate Control,” *IEEE Trans. Control Syst. Technol.*, **22**(3), pp. 1198–1205.
- [23] Ma, Y., Matusko, J., and Borrelli, F., 2015, “Stochastic Model Predictive Control for Building HVAC Systems: Complexity and Conservatism,” *IEEE Trans. Control Syst. Technol.*, **23**(1), pp. 101–116.
- [24] Kircher, K. J., and Zhang, K. M., 2016, “Sample-Average Model Predictive Control of Uncertain Linear Systems,” Conference on Decision and Control, pp. 6234–6239.
- [25] Crawley, D., Lawrie, L., Winkelman, F. C., Buhl, W., Huang, Y. J., Pederson, C. O., Strand, R. K., Liesen, R. J., Fisher, D. E., Witte, M. J., and Glazer, J., 2001, “ENERGYPLUS: Creating a New-Generation Building Energy Simulation Program,” *Energy Build.*, **33**(4), pp. 319–331.
- [26] Klein, S., Beckman, W., Mitchell, J., Duffie, J., Duffie, N., Freeman, T., Mitchell, J., and Kummert, M., 2004, TRNSYS 16—A TRAnSient System Simulation Program, User Manual, Solar Energy Laboratory, University of Wisconsin-Madison, Madison, WI.

- [27] Sturzenegger, D., Gyalistras, D., Semeraro, V., Morari, M., and Smith, R., 2014, "BRCM Matlab Toolbox: Model Generation for Model Predictive Building Control," American Control Conference, Portland, OR, pp. 1063–1069.
- [28] Gorecki, T., Qureshi, F., and Jones, C., 2015, "OpenBuild: An Integrated Simulation Environment for Building Control," IEEE Conference on Control Applications (CCA), Sydney, NSW, Australia, pp. 1522–1527.
- [29] Wetter, M., 2011, "Co-Simulation of Building Energy and Control Systems With the Building Controls Virtual Test Bed," *J. Build. Perform. Simul.*, **4**(3), pp. 185–203.
- [30] Bernal, W., Behl, M., Nghiem, T., and Mangharam, R., 2012, "MLE+: A Tool for Integrated Design and Deployment of Energy Efficient Building Controls," Fourth ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings, Toronto, Canada, pp. 123–130.
- [31] Darivianakis, G., Georgiou, A., Smith, R. S., and Lygeros, J., 2017, "The Power of Diversity: Data-Driven Robust Predictive Control for Energy-Efficient Buildings and Districts," IEEE Transactions on Control Systems Technology, **27**(1), pp. 132–145.
- [32] Gupta, S. K., Kar, K., Mishra, S., and Wen, J. T., 2018, "Singular Perturbation Method for Smart Building Temperature Control Using Occupant Feedback," *Asian J. Contr.*, **20**(1), pp. 386–402.
- [33] Finck, C., Li, R., Kramer, R., and Zeiler, W., 2018, "Quantifying Demand Flexibility of Power-to-Heat and Thermal Energy Storage in the Control of Building Heating Systems," *Appl. Energy.*, **209**, pp. 409–425.
- [34] Qureshi, F., Gorecki, T., and Jones, C., 2014, "Model Predictive Control for Market-Based Demand Response Participation," IFAC Proceedings Volumes, **47**(3), pp. 11153–11158.
- [35] Lymporopoulos, I., Qureshi, F. A., Nghiem, T., Khatir, A. A., and Jones, C. N., 2015, "Providing Ancillary Service With Commercial Buildings: The Swiss Perspective," IFAC-PapersOnLine, **48**(8), pp. 6–13.
- [36] Picard, D., Drgona, J., Kvasnica, M., and Helsen, L., 2017, "Impact of the Controller Model Complexity on Model Predictive Control Performance for Buildings," *Energy Build.*, **152**, pp. 739–751.
- [37] Judkoff, R., and Neymark, J., 1995, "International Energy Agency Building Energy Simulation Test (BESTEST) and Diagnostic Method Building Energy Simulation Test (Bestest) and Diagnostic Method," *National Renewable Energy Laboratory*.
- [38] ASHRAE, 2004, *Standard Method of Test for the Evaluation of Building Energy Analysis Computer Programs*, American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) Standard.
- [39] USDOE, 2019, *Engineering Reference – EnergyPlus*, U.S. Department of Energy.
- [40] USDOE, 2018, *Input Output Reference – EnergyPlus*, U.S. Department of Energy.
- [41] Kircher, K. J., and Zhang, K. M., 2015, "On the Lumped Capacitance Approximation Accuracy in RC Network Building Models," *Energy Build.*, **104**, pp. 454–462.
- [42] Lienhard IV, J. H., and Lienhard V, J. H., 2011, *A Heat Transfer Textbook*, 4th ed., Dover Publications, Mineola, NY.
- [43] Underwood, C. P., and Yik, F. W. H., 2004, *Modeling Methods for Energy in Buildings*, Blackwell Publishing, Malden, MA.
- [44] Khalifa, A., 2001, "Natural Convective Heat Transfer Coefficient—A Review: II. Surfaces in Two- and Three-Dimensional Enclosures," *Energy Convers. Manage.*, **42**(4), pp. 505–517.
- [45] Oppenheim, A. K., 1956, "Radiation Analysis by the Network Method," *Trans. ASME*, **78**, pp. 725–735.
- [46] Holmes, M. H., 2007, *Introduction to Numerical Methods in Differential Equations*, Springer, New York.
- [47] Hamdi, S., Schiesser, W. E., and Griffiths, G. W., 2007, "Method of Lines," *Scholarpedia*, **2**(7), p. 2859.
- [48] Sharaf, A. A., and Bakodah, H. O., 2005, "A Good Spatial Discretisation in the Method of Lines," *Appl. Math. Comput.*, **171**, pp. 1253–1263.
- [49] Crowley, M., and Hashmi, M. S. J., 1998, "Evaluation of Implicit Numerical Methods for Building Energy Simulation," *Proc. Inst. Mech. Eng., Part A: J. Power Energy*, **212**(5), pp. 331–342.
- [50] Wilcox, S., and Marion, W., 2008, "User's Manual for TMY3 Data Sets," National Renewable Energy Laboratory, Tech. Rep. NREL/TP-581-43156.
- [51] Duffie, J., and Beckman, W., 2006, *Solar Engineering of Thermal Processes*, 3rd ed., Wiley, New York.
- [52] Kircher, K. J., 2016, "BLDG, a MATLAB® Building Simulator," Accessed November 7, 2020.
- [53] Laret, L., 1980, "Use of General Models With a Small Number of Parameters," 7th International Congress CLIMA 2000, Budapest, Hungary, pp. 263–275.
- [54] Crabb, J., Murdoch, N., and Penman, J. M., 1987, "A Simplified Thermal Response Model," *Build. Serv. Eng. Res. Technol.*, **8**(1), pp. 13–19.
- [55] Wan, E., and Van Der Merwe, R., 2000, "The Unscented Kalman Filter for Nonlinear Estimation," The IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium, Lake Louise, Alberta, Canada, pp. 153–158.