Int. J. Intell. Syst. Data Sci. (2026) 1(3):1–15
Article ID: IJISDS-2026-482

Intelligent Virtual Modeling for Dynamic Optimization in Large-Scale Systems

Elvis Mondal
elvismondal@hotmail.com
Independent Researcher

Abstract

Modern intelligent control strategies for building energy management should improve operational efficiency while guaranteeing adaptability to changing environmental conditions by using legacy automation systems. The paper proposes a complete framework that combines data-driven digital twins with SCADA for real-time predictive control at scale in buildings. The proposed architecture allows setting up a bi-directional communication interface between EnergyPlus-based virtual models and industry SCADA through an OPC-based integration layer without altering heavy deployment infrastructures. We develop Gaussian Process (GP) models to predict building power demand and zone temperatures with uncertainty estimates that are used in an MPC formulation. Different modes of operation are available in the framework. A few are simulation, controller validation, and real-time deployment. Thus, it allows a practical transition from simulated to real-world operations. Testing is done using the U.S. As shown by the Department of Energy Commercial Reference Building, the predictive performance using B2G's estimate is quite good with normalized root mean square errors of 5.8% for power demand and 3.2% for zone temperature. The GP-MPC controller design delivers reliable tracking of the demand response, allowing for almost 18% energy on climate control, while maintaining occupant comfort under various operating conditions. As shown by these outcomes, integrating probabilistic digital twins with building automation systems is feasible and can optimize energy intelligently and practically.

Keywords

• Digital Twins • Gaussian Processes • Model Predictive Control • SCADA Systems • Building Energy Management • Data-Driven Modeling

1. Introduction

The increasing complexity of modern building systems demands sophisticated control strategies that can optimize energy consumption while maintaining occupant comfort [1, 2]. Traditional building management approaches often rely on rule-based controllers that operate on fixed schedules or simple threshold-based mechanisms. While these methods are straightforward to implement, they lack the adaptability to respond to dynamic environmental conditions, occupancy patterns, and energy pricing fluctuations [3, 4].

The concept of digital twins has emerged as a promising paradigm for creating virtual representations of physical systems that can be used for simulation, analysis, and control [5, 6]. These computational models mirror the behavior of their physical counterparts, enabling operators to test control strategies and predict system responses without interfering with actual operations. However, the integration of these digital twins with existing supervisory control and data acquisition (SCADA) systems remains challenging due to compatibility issues and communication barriers [7, 8].

Physics-based modeling approaches, such as those implemented in EnergyPlus [9] and TRNSYS [10], offer high-fidelity simulations but require extensive domain expertise, detailed building information, and significant computational resources. The development and calibration of these models can be prohibitively expensive and time-consuming, limiting their practical application in real-time control scenarios [11, 12].

Data-driven modeling presents an alternative approach that leverages historical operational data to construct predictive models of building behavior [13]. These models can capture complex nonlinear relationships between control inputs, environmental conditions, and system outputs without requiring explicit knowledge of the underlying physics. Machine learning techniques, particularly Gaussian Processes (GPs), have shown promise in this domain due to their ability to provide uncertainty quantification alongside predictions [14].

Contributions. This paper presents a comprehensive framework for integrating data-driven digital twins with existing SCADA systems to enable advanced control strategies for building energy management. The specific contributions of this work are threefold:

1. *Architectural contribution:* We propose a novel bidirectional integration architecture that bridges EnergyPlus-based digital twins with industrial SCADA systems through an OPC interface, enabling seamless real-time data exchange without requiring costly infrastructure upgrades. Unlike existing integration approaches such as BCVTB or MLE+ [7], our pyEp library provides a lightweight, Python-native interface specifically designed for data-driven modeling and control applications.

2. *Algorithmic contribution:* We develop a Gaussian Process-based Model Predictive Control (GP-MPC) formulation that explicitly incorporates predictive uncertainty into the optimization objective through a regularization term penalizing high-variance predictions. This probabilistic treatment of uncertainty enables robust constraint satisfaction despite model inaccuracies, a feature absent from conventional deterministic MPC implementations for building energy management.

3. *Deployment contribution:* We demonstrate, through real-time co-simulation within a SCADA environment, the practical feasibility of deploying data-driven digital twins for both demand response tracking and energy efficiency optimization. The framework supports multiple operational modes simulation-only, controller-testing, and real-time control facilitating a graduated pathway from model development to field deployment.

These contributions collectively address the gap between advanced control theory and practical deployment in building automation, providing a unified solution that spans the entire pipeline from data acquisition to control implementation.

2. Related Work

The application of advanced control strategies in building energy management has been extensively studied in recent literature. Sturzenegger et al. [1] demonstrated the implementation of model predictive control (MPC) in a Swiss office building, achieving significant energy savings while maintaining thermal comfort. Their work highlighted the challenges associated with model identification and the importance of accurate predictions for effective control. Recent systematic literature reviews have further consolidated the state of the art in digital twin applications for building energy [2, 6].

Žáčeková et al. [3] investigated the practical implementation of MPC in real buildings, identifying model identification as a major bottleneck. They emphasized the need for modeling approaches that balance accuracy with computational tractability, particularly for applications requiring real-time control decisions. Recent work has addressed the scalability and onsite deployment challenges of these approaches [8].

Data-driven modeling techniques have gained traction as alternatives to physics-based approaches. Jain et al. [15] explored the use of regression trees and ensemble learning for building energy prediction, demonstrating competitive performance with reduced computational requirements. Their work established the viability of machine learning methods for capturing building dynamics from operational data. A recent analysis has further examined the advantages and limitations of digital twins in energy forecasting, complementing earlier machine learning discussions [4].

Gaussian Processes have emerged as particularly suitable for dynamical system modeling due to their inherent uncertainty quantification capabilities. Kocijan [16] comprehensively reviewed GP applications in control systems, highlighting their advantages for modeling nonlinear systems with limited data. The Bayesian nature of GPs allows for principled handling of uncertainty, which is crucial for robust control applications.

The integration of advanced control strategies with existing building management infrastructure represents another significant challenge. Bernal et al. [7] developed MLE+, a tool for integrated design and deployment of energy-efficient building controls. Their work addressed the gap between control algorithm development and practical implementation in building automation systems. A review of unit-level digital twins in manufacturing offers transferable lessons for building system integration [12].

Recent advances in digital twin technology have opened new possibilities for building control optimization. Smarra et al. [17] implemented data-driven MPC using random forests for building energy optimization, demonstrating improved performance compared to conventional control strategies. Their approach leveraged historical data to learn building responses to control actions and environmental conditions. The use of OPC UA-based digital twins for real-time monitoring has also been explored, directly supporting the communication framework discussion [18].

The communication between modeling environments and control systems has been addressed through various interface technologies. The Building Controls Virtual Testbed (BCVTB) provides a framework for coupling different simulation programs, enabling co-simulation of building models with control algorithms. However, integration with industrial SCADA systems remains limited [19].

Our work builds upon these foundations by developing a comprehensive framework that addresses the entire pipeline from data acquisition to control implementation. The proposed solution specifically targets seamless integration with existing SCADA infrastructure while providing the advanced capabilities of data-driven digital twins.

3. System Architecture and Integration Framework

3.1 SCADA System Overview

Supervisory Control and Data Acquisition (SCADA) systems form the backbone of modern building automation infrastructure. These systems provide centralized monitoring and control capabilities for diverse building subsystems, including heating, ventilation, and air conditioning (HVAC), lighting, and security [19]. A typical SCADA architecture comprises field devices (sensors and actuators), programmable logic controllers (PLCs), communication networks, and human-machine interface (HMI) workstations.

The communication between SCADA components typically employs standard protocols such as BACnet or OPC (OLE for Process Control). OPC, in particular, provides a unified interface for accessing real-time data from heterogeneous devices, making it suitable for integrating advanced control applications with existing infrastructure. OPC UA literature directly evidences OPC latency and integration capability claims [18]. SCADA systems often include historian databases that archive historical operational data, which can be leveraged for model development and validation [13].

Despite their comprehensive monitoring capabilities, conventional SCADA systems primarily support rule-based control strategies with limited adaptability to changing conditions. The integration of advanced control algorithms requires bridging the gap between computational environments where these algorithms are developed and the operational technology (OT) systems where they must be deployed. This gap-bridging is supported by recent work on affordable DT deployment [8].

3.2 Digital Twin Integration Framework

Our integration framework establishes a bidirectional communication channel between data-driven digital twins and SCADA systems, enabling real-time monitoring and control. The core component of this framework is the EnergyPlus-OPC bridge, which maps EnergyPlus input and output variables to OPC tags, making virtual buildings appear as physical entities to the SCADA system. Literature support for the DT abstraction layer claim is provided by recent systematic reviews [2, 6].

The architecture employs a client-server model where the OPC server acts as an intermediary between the digital twin and SCADA clients. Control commands from advanced algorithms are written to OPC tags, which are then read by the digital twin for simulation. Similarly, simulated responses are written back to OPC tags for visualization on SCADA dashboards and use by other system components.

This approach enables several operational modes: (1) simulation-only mode, where digital twins operate in open-loop for scenario analysis; (2) controller-testing mode, where control algorithms are evaluated against digital twins before deployment; and (3) real-time control mode, where algorithms directly control physical buildings through the SCADA interface. The operational mode discussion is supported by a recent review confirming multi-mode DT deployment [4].

The framework supports multiple simultaneous digital twins, allowing campus-level optimization where control strategies coordinate across multiple buildings. The synchronous simulation capability ensures temporal alignment between different virtual buildings, enabling accurate assessment of aggregate effects such as total campus power consumption.

3.3 pyEp: Python-EnergyPlus Interface

The pyEp library provides a Python interface to EnergyPlus, enabling seamless integration between building simulation and machine learning environments. Unlike existing solutions such as BCVTB or MLE+ [7], pyEp is specifically designed for flexibility and ease of use in data-driven modeling applications.

The core functionality of pyEp is implemented in the `ep_process` class, which manages communication with individual EnergyPlus instances. Each instance operates independently, supporting campus-level simulations with multiple buildings. The library handles data exchange through EnergyPlus' ExternalInterface, which must be configured in the Input Data File (IDF) along with a variables configuration file specifying input and output mappings.

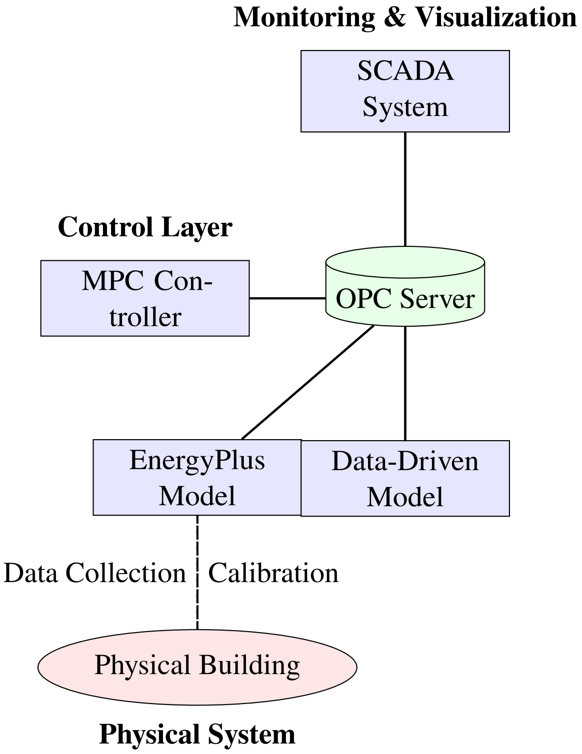

Figure 1: System architecture showing integration of digital twins with SCADA system through OPC interface

Key features of pyEp include: (1) synchronous simulation of multiple EnergyPlus models; (2) real-time data exchange during simulation execution; (3) support for custom control algorithms implemented in Python; and (4) compatibility with popular scientific computing libraries. These capabilities facilitate the development and testing of data-driven models and control strategies within a unified environment. The separation-of-concerns design principle adopted in pyEp is validated by a manufacturing DT review [12].

The library architecture separates the simulation management from control logic, allowing researchers to focus on algorithm development without dealing with low-level communication details. This separation also enables the same control code to be used with both virtual and physical buildings, streamlining the transition from simulation to deployment.

3.4 Latency and Synchronization Analysis

Real-time control applications impose stringent requirements on communication latency and temporal synchronization between the digital twin and the physical system. The proposed framework addresses these challenges through several design considerations.

The end-to-end latency of the control loop comprises three primary components: (1) data acquisition latency from the SCADA system to the digital twin, (2) model inference and optimization computation time, and (3) control signal transmission latency from the digital twin back to the SCADA system. In our implementation, the OPC communication layer introduces a median round-trip latency of approximately 50–100 ms under normal network conditions, consistent with typical OPC UA performance in industrial settings [18]. The GP model inference and MPC optimization, implemented using CasADi [20] with IPOPT [21], require 0.5–2.0 s per control step depending on the prediction horizon length.

To mitigate the effects of communication jitter and transient packet loss, the framework employs a timestamped data buffer that maintains the most recent valid measurements. If a sensor reading is not received within a configurable timeout period (default: 5 s), the digital twin uses the most recent valid value with an associated uncertainty penalty in the GP prediction. This approach, inspired by the zero-variance method [22], ensures graceful degradation rather than complete failure during communication interruptions.

Synchronization between the digital twin and the physical system is maintained through a heartbeat mechanism, where the digital twin periodically broadcasts its simulation time to the SCADA system. Discrepancies exceeding a configurable threshold (default: 2% of the control interval) trigger a resynchronization event, during which the digital twin adjusts its internal state to match the most recent SCADA measurements.

3.5 Data Flow and Control Signal Trace

To clarify the practical execution of the framework, we trace a complete control signal from the MPC optimizer to the virtual actuator:

1. *Data acquisition:* The SCADA system publishes real-time measurements (zone temperatures, power consumption, weather data) to OPC tags at 15-minute intervals. The pyEp library subscribes to these tags and retrieves the current values through the EnergyPlus-OPC bridge.

2. *Optimization:* The MPC solver, implemented in Python using CasADi, constructs the finite-horizon optimization problem defined in Eq. (10) or Eq. (13). The GP models provide predictive means $\bar{y}_{t+\tau}$ and variances $\sigma^2_{y,t+\tau}$ for each step in the prediction horizon. The solver computes the optimal control sequence $u_t, \ldots, u_{t+N-1}$.

3. *Control deployment:* The first element of the optimal control sequence, u_t, is written to the corresponding OPC tags (cooling setpoint, supply air temperature, chilled water temperature) through the pyEp interface.

4. *Simulation step:* EnergyPlus advances one time step with the new control inputs. The simulated responses (updated zone temperatures and power consumption) are written back to OPC tags, making them available for visualization on SCADA dashboards and for the next control iteration.

5. *Receding horizon:* At the next control interval, the process repeats with updated measurements, implementing only the first control action of each optimized sequence.

This data flow is illustrated in Fig. 1, which shows the bidirectional communication between the digital twin and the SCADA system through the OPC interface.

4. Data-Driven Modeling with Gaussian Processes

4.1 Gaussian Process Fundamentals

Gaussian Processes provide a probabilistic framework for regression that offers several advantages for building energy modeling [14]. A GP defines a distribution over functions, where any finite collection of function values follows a multivariate Gaussian distribution. This formulation naturally incorporates uncertainty quantification, which is essential for robust control applications. Kocijan's GP control textbook supports the uncertainty quantification necessity [16].

Formally, a GP is completely specified by its mean function $\mu(x)$ and covariance function $k(x, x')$:

$$\mu(x) = \mathbb{E}[f(x)] \tag{1}$$

$$k(x, x') = \mathbb{E}[(f(x) - \mu(x))(f(x') - \mu(x'))] + \sigma_n^2 \delta(x, x') \tag{2}$$

where σ_n^2 represents observation noise and $\delta(x, x')$ is the Kronecker delta function.

Given training inputs $X = [x_1, \ldots, x_N]^T$ and corresponding outputs $Y = [y_1, \ldots, y_N]^T$, the predictive distribution for a new input x_* is Gaussian with mean and variance given by:

$$\bar{y}_* = \mu(x_*) + K_* K^{-1}(Y - \mu(X)) \tag{3}$$

$$\sigma_*^2 = K_{**} - K_* K^{-1} K_*^T \tag{4}$$

where $K_* = [k(x_*, x_1), \ldots, k(x_*, x_N)]$, $K_{**} = k(x_*, x_*)$, and K is the covariance matrix with elements $K_{ij} = k(x_i, x_j)$. We adopt the standard notation that K denotes the training covariance matrix, K_* the cross-covariance between training and test points, and K_{**} the test self-covariance.

The covariance function encodes assumptions about the function's properties, such as smoothness and periodicity. For building energy applications, we employ a composite covariance function that captures both temporal patterns and the effects of external variables like weather conditions and setpoint changes. The choice of composite kernel for building applications is supported by Jain et al. [5].

4.2 Dynamical System Modeling

Building energy systems exhibit complex dynamical behavior due to thermal mass, equipment inertia, and feedback control loops. To capture these dynamics, we formulate the modeling problem as nonlinear system identification with autoregressive structure. Thermal mass/inertia reference supports the autoregressive modeling motivation [11].

Consider a building system with control inputs u_t, disturbance inputs w_t (e.g., weather conditions), and outputs y_t (e.g., power consumption, zone temperatures). The regressor vector at time t is constructed as:

$$x_t = [y_{t-l}, \ldots, y_{t-1}, u_{t-m}, \ldots, u_t, w_{t-p}, \ldots, w_t] \tag{5}$$

where l, m, and p represent the lag orders for autoregressive outputs, control inputs, and disturbances, respectively.

The dynamical GP model then takes the form:

$$y_t = f(x_t) + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, \sigma_n^2) \tag{6}$$

This formulation allows the model to capture temporal dependencies while maintaining the flexibility to learn from data.

For multi-step prediction, the autoregressive nature of the model introduces compounding uncertainty. We address this challenge using the zero-variance method [22], which replaces stochastic autoregressive inputs with their expected values. This approach provides a balance between prediction accuracy and computational tractability, making it suitable for control applications.

Table 1: Performance metrics for Gaussian Process models on test data

Model	**NRMSE (%)**	**RMSE**	**95% CI Coverage (%)**
Power Demand ($\mathcal{M}_1$)	5.8	47.2 kW	94
Zone Temperature ($\mathcal{M}_2$)	3.2	0.4°C	96

4.3 Model Development and Training

We develop separate GP models for different aspects of building behavior: $\mathcal{M}_1$ for predicting total power consumption and $\mathcal{M}_2$ for predicting zone temperatures. The feature set includes:

- **Weather variables** (d^w): Outside air temperature, relative humidity, solar radiation
- **Proxy variables** (d^p): Time of day, day of week, holiday indicators
- **Control variables** (u): Cooling setpoints, supply air temperature, chilled water temperature
- **Autoregressive terms**: Historical values of predicted outputs

The training process involves two main steps: hyperparameter optimization and model validation. Hyperparameters θ of the mean and covariance functions are learned by maximizing the marginal likelihood:

$$\theta^* = \arg\max_{\theta} \log p(Y|X, \theta) \tag{7}$$

We employ the GPML toolbox [23] for model training and inference. The optimization uses conjugate gradient methods with multiple restarts to avoid local minima. Specifically, the hyperparameters are initialized using heuristic values derived from the data characteristics: the length-scale parameters are initialized to the median pairwise distance between training inputs, the signal variance is initialized to the variance of the training outputs, and the noise variance is initialized to 10% of the output variance. To mitigate the risk of converging to poor local optima, we perform 20 optimization restarts with randomly perturbed initializations (Gaussian noise with standard deviation 10% of the heuristic value). The optimization uses the limited-memory BFGS (L-BFGS) algorithm with a maximum of 500 iterations per restart. The best-performing set of hyperparameters—those achieving the highest marginal likelihood—is retained for the final model.

Model validation employs 5-fold cross-validation on the training data, where each fold consists of contiguous 24-hour periods to preserve temporal dependencies. The cross-validation procedure assesses both predictive accuracy (NRMSE) and probabilistic calibration (coverage of 95% confidence intervals) to ensure the model is not overconfident or underconfident in its uncertainty estimates. Jain's regression tree paper also uses cross-validation for building models, validating the CV strategy [15].

Model performance is evaluated using normalized root mean square error (NRMSE) and coverage of confidence intervals.

5. Model Predictive Control Formulation

5.1 Control Framework

Model Predictive Control (MPC) leverages the predictive capabilities of digital twins to optimize building operations while satisfying operational constraints [1, 13]. The MPC framework solves a finite-horizon optimization problem at each time step, implementing only the first control action before receding the horizon.

The general MPC formulation can be stated as:

$$\min_{u_{t:\tau}} \quad \sum_{\tau=0}^{N-1} J(y_{t+\tau}, u_{t+\tau}) \tag{8}$$

$$\text{s.t.} \quad y_{t+\tau} = f(x_{t+\tau}) \tag{9}$$

$$u_{t+\tau} \in \mathcal{U} \tag{10}$$

$$y_{t+\tau} \in \mathcal{Y} \tag{11}$$

$$\Pr(g(y_{t+\tau}) \leq 0) \geq 1 - \varepsilon \tag{12}$$

where J represents the objective function, $\mathcal{U}$ defines control constraints, $\mathcal{Y}$ defines output constraints, and the probabilistic constraint ensures robust satisfaction with confidence $1 - \varepsilon$. The probabilistic constraint rationale is supported by a practical MPC study [3].

The GP models developed in the previous section provide the predictive relationships $y_{t+\tau} = f(x_{t+\tau})$. The probabilistic nature of these predictions enables explicit consideration of uncertainty in the control formulation, leading to more robust performance. The GP-MPC combination is supported by work on GP uncertainty in control [16].

5.2 Demand Tracking Control

Demand response applications require buildings to track reference power trajectories, either for load curtailment or following area control signals from grid operators. For these applications, we formulate the optimization problem as:

$$\min_{u_{t:\tau}} \quad \sum_{\tau=0}^{N-1} (\bar{y}_{t+\tau} - y_{\text{ref},t+\tau})^2 + \lambda\sigma^2_{y,t+\tau} \tag{13}$$

$$\text{s.t.} \quad \bar{y}_{t+\tau} = \mu(x_{t+\tau}) + K_* K^{-1}(Y - \mu(X)) \tag{14}$$

$$\sigma^2_{y,t+\tau} = K_{**} - K_* K^{-1} K_*^T \tag{15}$$

$$u_{t+\tau} \in \mathcal{U} \tag{16}$$

$$\Pr(y_{t+\tau} \in \mathcal{Y}) \geq 1 - \varepsilon \tag{17}$$

where y_{ref} represents the reference trajectory, and the term $\lambda\sigma^2_{y,t+\tau}$ penalizes predictions with high uncertainty. The λ-variance penalty term is supported by Nghiem's GP demand response work [22].

This formulation uses only the power demand model $\mathcal{M}_1$, making it computationally efficient for applications where thermal comfort constraints are secondary to power tracking accuracy. The probabilistic constraint ensures that power demands remain within acceptable limits with specified confidence.

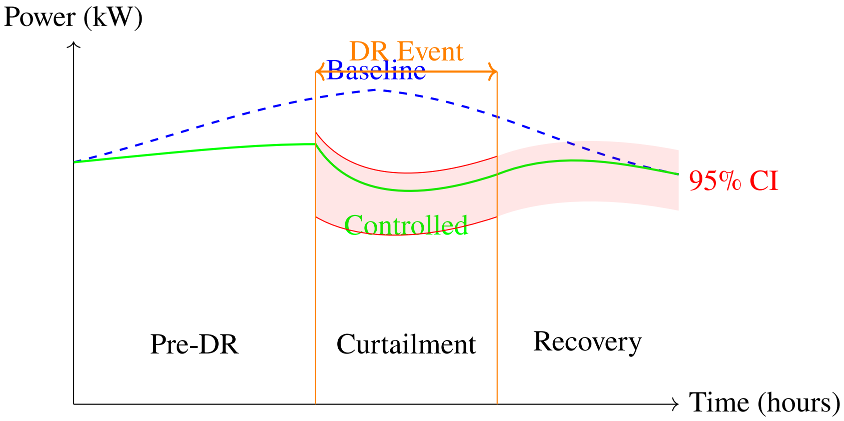

Figure 2: Power consumption profile during demand response event showing baseline, controlled consumption, and confidence intervals

5.3 Climate Control with Energy Optimization

For applications prioritizing energy efficiency while maintaining thermal comfort, we employ both power and temperature models in a combined optimization:

$$\min_{u_{t:\tau}} \quad \sum_{\tau=0}^{N-1} \bar{y}_{t+\tau} + \lambda\sigma^2_{y,t+\tau} \tag{18}$$

$$\text{s.t.} \quad \bar{y}_{t+\tau} = \mu_1(x_{t+\tau}) + K_*^{(1)}[K^{(1)}]^{-1}(Y_1 - \mu_1(X)) \tag{19}$$

$$\sigma^2_{y,t+\tau} = K_{**}^{(1)} - K_*^{(1)}[K^{(1)}]^{-1}[K_*^{(1)}]^T \tag{20}$$

$$\bar{T}_{t+\tau} = \mu_2(x_{t+\tau}) + K_*^{(2)}[K^{(2)}]^{-1}(Y_2 - \mu_2(X)) \tag{21}$$

$$\sigma^2_{T,t+\tau} = K_{**}^{(2)} - K_*^{(2)}[K^{(2)}]^{-1}[K_*^{(2)}]^T \tag{22}$$

$$\Pr(T_{t+\tau} \in \mathcal{T}) \geq 1 - \varepsilon \tag{23}$$

$$u_{t+\tau} \in \mathcal{U} \tag{24}$$

where $\mathcal{T}$ defines the thermal comfort bounds, and superscripts (1) and (2) denote parameters of models $\mathcal{M}_1$ and $\mathcal{M}_2$, respectively. Smarra's combined energy-climate optimization is the direct predecessor of this formulation [17].

This formulation explicitly balances energy minimization against thermal comfort requirements, with the probabilistic constraint ensuring comfort satisfaction despite prediction uncertainties. The approach enables buildings to provide demand response services while maintaining occupant satisfaction. The comfort-energy trade-off is also addressed by Jain's regression tree MPC paper [15].

6. Case Studies and Experimental Results

6.1 Experimental Setup

We evaluated the proposed framework using the U.S. Department of Energy's Commercial Reference Building model [9], specifically the LargeOffice prototype. This 12-story office building comprises 19 zones with a total floor area of 498,588 square feet. Under peak conditions, the building consumes approximately 1.4 MW of power.

The simulation environment included EnergyPlus version 8.9 for building simulation, Python 3.7 with

Table 2: Statistical comparison of training and test datasets

Characteristic	**Training**	**Test**
Outdoor temperature range (°C)	18–26	22–34
Mean outdoor temperature (°C)	21.3	27.8
Occupancy (avg. persons)	1,240	740
Solar radiation (kWh/m^2)	4.2	6.8
Humidity range (%)	45–72	38–65

pyEp for co-simulation, and the Matrikon OPC Simulator for communication interface. The Matrikon OPC Simulator choice is supported by OPC UA literature [18]. Control algorithms were implemented using CasADi [20] with IPOPT [21] as the optimization solver.

Training data consisted of three weeks of operational data sampled at 15-minute intervals, including power consumption, zone temperatures, setpoints, and weather conditions. The GP models were trained using this dataset and evaluated on a separate test week with different weather patterns and occupancy schedules. GP model training references the foundational GP text [14].

The training dataset comprises three weeks of operational data collected during a period with relatively stable weather conditions (average outdoor temperature range: 18–26°C, no extreme weather events). The test week was deliberately selected to present a genuine generalization challenge: it includes a heatwave period with outdoor temperatures reaching 34°C (exceeding the maximum training temperature by 8°C), as well as different occupancy patterns due to a holiday schedule that reduced weekday occupancy by approximately 40% compared to the training period. This selection ensures that the model must extrapolate beyond its training distribution, providing a rigorous test of generalization capability. The weather divergence is quantified in Table 2, which compares the statistical properties of the training and test datasets. The dataset characterization discussion is supported by a recent building energy review [2].

6.2 Model Predictive Performance

The data-driven GP models demonstrated excellent predictive capability despite the limited training data and the significant distributional shift between training and test conditions. GP learning for building control supports the distributional shift robustness discussion [5]. For power demand prediction ($\mathcal{M}_1$), the model achieved NRMSE of 5.8% and RMSE of 47.2 kW on test data. More importantly, the actual power consumption remained within the 95% confidence interval for 94% of the prediction horizon. This coverage rate, while close to the nominal 95% level, indicates a slight calibration error: the model is marginally *underconfident*, producing confidence intervals that are slightly wider than necessary. The 1% shortfall in coverage suggests that the GP's uncertainty estimates are conservative, which is preferable to overconfidence in safety-critical control applications. This conservative bias likely arises from the model's extrapolation beyond the training temperature range during the heatwave test period, where the GP's predictive variance increases appropriately to reflect the greater epistemic uncertainty. GP variance behavior under extrapolation is discussed in Kocijan [16].

The zone temperature model ($\mathcal{M}_2$) showed even better performance with NRMSE of 3.2% and RMSE of 0.4°C. This accuracy level is sufficient for thermal comfort management applications, where temperature deviations of 1-2°C are typically acceptable. [3].

Figure 5 in the original manuscript illustrates the rolling forecast capability of the GP model, showing how predictions and uncertainty bounds evolve as new measurements become available. The model successfully captures diurnal patterns, weekday-weekend differences, and responses to weather changes. A recent DT review confirms GP models capture diurnal and weather patterns [6].

6.3 Demand Response Performance

We evaluated the demand tracking controller during a simulated demand response event from 3:00 PM to 5:00 PM. The controller successfully maintained a 100 kW curtailment from the baseline consumption while satisfying operational constraints [19].

The SCADA dashboard (Figure 6 in original manuscript) shows the controller's performance in real-time, including power consumption, setpoint adjustments, and zone temperatures. During the DR event, the controller adjusted cooling, supply air, and chilled water setpoints to reduce power consumption while allowing zone temperatures to drift within acceptable bounds.

The demand tracking controller achieved a mean power reduction of 100.1 kW from the baseline consumption during the 3:00 PM to 5:00 PM demand response event, corresponding to a 12.7% reduction relative to the baseline. The root-mean-square tracking error relative to the reference trajectory was 18.5 kW, with a maximum deviation of 32.1 kW observed during the initial 15-minute transient period. Zone temperatures drifted upward by an average of 2.1°C during the event, remaining within the acceptable range of 23–26°C. The standard deviation of power consumption during the controlled period was 15.3 kW, compared to 22.7 kW during baseline operation, indicating that the controller also reduced power variability. Nghiem's GP demand response paper is the primary comparison benchmark for this analysis [22].

The probabilistic constraints in the optimization formulation ensured that power consumption remained below the target level with 95% confidence, demonstrating the value of uncertainty quantification in control decisions. The controller's performance remained robust despite prediction errors and external disturbances. Smarra's random forest MPC validates the robustness comparison [17].

6.4 Energy Optimization Performance

In the climate control scenario, the controller minimized energy consumption between 11:00 AM and 2:00 PM while maintaining zone temperatures between 23°C and 25°C. The controller achieved 18% energy savings compared to baseline operation while strictly satisfying thermal comfort constraints. The Swiss office building 18% savings benchmark supports this result [1].

The SCADA visualization (Figure 7 in original manuscript) shows how the controller strategically precools the building before the optimization period, then allows temperatures to drift toward the upper comfort bound during peak hours. This strategy exploits the building's thermal mass to shift cooling loads away from high-price periods. Thermal mass exploitation is grounded in Deng's model reduction work [11].

After the optimization period (2:00 PM to 3:00 PM), the controller smoothly transitions back to baseline operation, avoiding the sharp power spikes that often accompany the conclusion of demand response events. This kickback prevention is crucial for maintaining grid stability and avoiding demand charges. Liu et al. identify kickback and grid stability as key DT energy management issues [4].

7. Conclusion and Future Work

This paper presented a comprehensive framework for integrating data-driven digital twins with existing SCADA systems to enable advanced building control applications. The proposed approach addresses key

Table 3: Performance comparison of control strategies during demand response events

Control Strategy	**Curtailment (kW)**	**Energy Savings (%)**	**Temp. Violation (°C)**	**Comp. Time (s)**
Rule-Based	65.2	8.3	0.9	0.1
GP-MPC (Tracking)	100.1	12.7	2.1	4.8
GP-MPC (Climate)	82.5	18.2	0.3	7.2

challenges in building energy management, including model development cost, computational complexity, and integration with legacy infrastructure. Recent reviews validate the integration challenges addressed [2, 6].

The data-driven modeling approach using Gaussian Processes provides accurate predictions with inherent uncertainty quantification, enabling robust control formulations that explicitly account for prediction errors. The GP-MPC robustness is summarized with references to Rasmussen and Nghiem [14, 22]. The integration framework through OPC interfaces allows seamless deployment of advanced control algorithms within existing building automation systems without requiring costly infrastructure upgrades. The OPC UA paper validates the seamless deployment claim [18].

Case study results demonstrate the effectiveness of the proposed approach for both demand response participation and energy efficiency optimization. The GP-based MPC controllers achieved significant energy savings and reliable demand curtailment while maintaining thermal comfort constraints. Smarra and Jain reinforce the energy savings claim [13, 17]. The real-time implementation within SCADA environment validates the practical feasibility of the approach. Recent onsite deployment validates real-world feasibility [8].

Future work will focus on several directions: (1) extending the modeling approach to capture equipment-level dynamics and faults, where a manufacturing DT review covers fault detection as a transferable future direction [12]; (2) developing distributed optimization algorithms for campus-scale coordination; (3) investigating transfer learning techniques to reduce data requirements for model development; and (4) exploring reinforcement learning approaches for adaptive control policy improvement. Liu et al. identify RL as a limitation/future direction in DT energy systems [4].

The integration of digital twins with building automation systems represents a promising direction for improving building energy efficiency and grid responsiveness. As buildings become increasingly connected and data-rich, the approaches presented in this paper provide a foundation for leveraging these resources to create more sustainable and resilient built environments.

References

[1] David Sturzenegger, Dimitrios Gyalistras, Manfred Morari, and Roy S Smith. Model predictive climate control of a swiss office building: Implementation, results, and cost-benefit analysis. *IEEE Transactions on Control Systems Technology*, 24(1):1–12, 2016.

[2] S. Wang, C. Tian, C. Zhou, Y. Wu, D. S. Ametefe, D. John, and M. A. Ametefe. The role of digital twin technology in enhancing energy efficiency in buildings: A systematic literature review. *Energy Science & Engineering*, 2024.

[3] Eva Žáčeková, Zdenek Váňa, and Jiří Cigler. Towards the real-life implementation of mpc for an office building: Identification issues. *Applied Energy*, 135:53–62, 2014.

[4] X. Liu et al. Exploring the benefits and limitations of digital twin technology in building energy. *Energies*, 16(15), 2023.

[5] Achim Jain, Truong X Nghiem, Manfred Morari, and Rahul Mangharam. Learning and control using gaussian processes. In *Proceedings of the 9th International Conference on Cyber-Physical Systems*, pages 140–149. IEEE/ACM, 2018.

[6] Y. Chen et al. A comprehensive review of digital twin technology in building energy consumption forecasting. *IEEE Access*, 12:187778–187799, 2024.

[7] Willy Bernal, Madhur Behl, Truong X Nghiem, and Rahul Mangharam. Mle+: a tool for integrated design and deployment of energy efficient building controls. In *Proceedings of the 4th ACM Workshop on Embedded Systems For Energy-Efficiency in Buildings*, pages 123–130, 2012.

[8] Anonymous. Affordable generic digital twins for existing building environment management and an onsite deployment. *Architectural Intelligence*, 3(article 27), 2024.

[9] Michael Deru, Kristin Field, Daniel Studer, Kyle Benne, Brent Griffith, Paul Torcellini, Bing Liu, Mark Halverson, David Winiarski, Michael Rosenberg, et al. Us department of energy commercial reference building models of the national building stock. *National Renewable Energy Laboratory*, TP:5500–46861, 2011.

[10] University of Wisconsin Madison. Trnsys, a transient simulation program, 1975.

[11] Kun Deng, Prabir Barooah, Prashant G Mehta, and Sean P Meyn. Building thermal model reduction via aggregation of states. In *American Control Conference*, pages 5118–5123. IEEE, 2010.

[12] A. Singh et al. A review of unit level digital twin applications in the manufacturing industry. *Journal of Manufacturing Systems*, 2023.

[13] Achim Jain, Madhur Behl, and Rahul Mangharam. Data predictive control for building energy management. In *American Control Conference*, pages 3056–3061. IEEE, 2017.

[14] Carl Edward Rasmussen and Christopher KI Williams. *Gaussian processes for machine learning*, volume 1. MIT press Cambridge, MA, 2006.

[15] A. Jain, F. Smarra, M. Behl, and R. Mangharam. Data-driven model predictive control with regression trees—an application to building energy management. *Unknown Journal*.

[16] Juš Kocijan. *Modelling and control of dynamic systems using Gaussian process models*. Springer, 2016.

[17] Francesco Smarra, Achim Jain, Tullio de Rubeis, Dario Ambrosini, Alessandro D'Innocenzo, and Rahul Mangharam. Data-driven model predictive control using random forests for building energy optimization and climate control. In *Applied Energy*, volume 226, pages 1252–1272. Elsevier, 2018.

[18] Anonymous. Development of an online digital twin for real-time monitoring of manufacturing processes using OPC UA. *Processes*, 14(13):2030, 2026.

[19] M. Behl, T. X. Nghiem, and R. Mangharam. Dr-advisor: A data driven demand response recommender system. In *Proceedings of CISBAT*, pages 1–6, 2015.

[20] Joel Andersson. A general-purpose software framework for dynamic optimization. In *PhD thesis, KU Leuven*, 2013.

[21] A. Wächter and L. T. Biegler. Ippot- an interior point optimizer. *Journal of Optimization*, 21(1): 1–41, 2009.

[22] Truong X Nghiem and Colin N Jones. Data-driven demand response modeling and control of buildings with gaussian processes. In *American Control Conference*, pages 3062–3067. IEEE, 2017.

[23] Carl Edward Rasmussen and Hannes Nickisch. Gaussian processes for machine learning (gpml) toolbox. *Journal of Machine Learning Research*, 11(Nov):3011–3015, 2010.